scientific reports

OPEN

Neural networks memorise personal information from one sample

John Hartley[1,3]✉, Pedro P. Sanchez[1,3], Fasih Haider[1] & Sotirios A. Tsaftaris[1,2]

Deep neural networks (DNNs) have achieved high accuracy in diagnosing multiple diseases/conditions at a large scale. However, a number of concerns have been raised about safeguarding data privacy and algorithmic bias of the neural network models. We demonstrate that unique features (UFs), such as names, IDs, or other patient information can be memorised (and eventually leaked) by neural networks even when it occurs on a single training data sample within the dataset. We explain this memorisation phenomenon by showing that it is more likely to occur when UFs are an instance of a rare concept. We propose methods to identify whether a given model does or does not memorise a given (known) feature. Importantly, our method does not require access to the training data and therefore can be deployed by an external entity. We conclude that memorisation does have implications on model robustness, but it can also pose a risk to the privacy of patients who consent to the use of their data for training models.

The objective of a deep neural network (DNN) is to learn the fundamental underlying relationships between the inputs and target outputs of a training dataset, such that the network generalises to give desired outputs when presented with novel unseen data inputs. However, DNNs have been shown to frequently assign predictions based only on a single example in their training data[1–4]. Such learning type is also referred to as memorisation.

This study focuses on *unique feature memorisation* (UFM) and how UFM relates to model robustness and consequently to the privacy of individuals when training neural networks. UFM is the unintended memorisation of specific *features* that occur *once* in training data as opposed to memorisation of examples or training labels. Whilst training examples have been shown to be memorised[1], it is not clear whether an example is memorised in its entirety or a specific feature of the example (e.g. the image) is memorised.

Let us consider a medical imaging example where data are sensitive[5,6], a private feature such as a person's or healthcare professional's name (a unique and unusual feature) has survived sanitisation processes[7] and is displayed on a single image (and hence it is very rare). Our hypothesis is that a classifier trained on data containing patient or healthcare professionals names may memorise this private feature. This has two consequences. First, there is an obvious privacy risk: the model has potentially learned this unique (and private) feature and has retained this information within its parameters. Thus, it is possible that such information can be leaked. An adversary with access to the weights of a trained neural network could potentially use them to infer information about the training examples.

Second, this classifier might misdiagnose other patients if this feature appears in another patient's medical scan, as illustrated in Fig. 1. The unintended presence of UF may lead to incomplete extraction of the correct discriminative features from the image[8]. Such a risk is similar to decision-making based on spurious correlations or shortcuts[9–12], except that only a *single* spurious feature is present in the dataset.

In this article, we evaluate whether neural networks memorise unique features and show how to measure it. We conducted experiments to demonstrate why this phenomenon happens, and discuss its consequences for privacy.

Methods

In the following section, we define the essential concepts for studying UFM, propose a way of measuring it with the M score and describe a series of experiments that enhance understanding of UFM and when it happens.

[1]The University of Edinburgh, Edinburgh, UK. [2]The Alan Turing Institute, London, UK. [3]These authors contributed equally: John Hartley and Pedro Sanchez. ✉email: john.hartley@ed.ac.uk

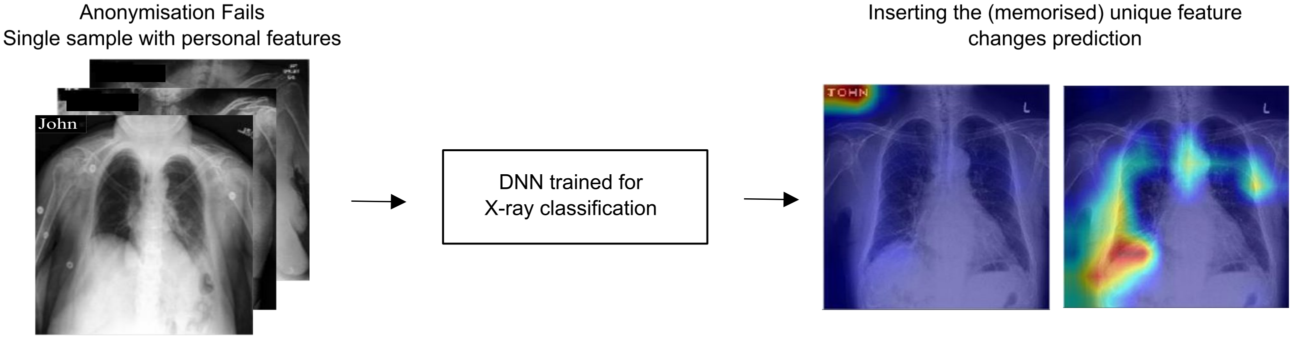
Anonymisation Fails
Single sample with personal features
[redacted]
[redacted]
John
DNN trained for X-ray classification
Inserting the (memorised) unique feature changes prediction
JOHN
L
L

Figure 1. Unique image features, which may contain private information (e.g. name JOHN), unintentionally left in training data can be memorised by a neural network. They are unique because they are unusual with respect to the remaining features of the dataset and occur only once. We propose methods to identify if a feature has been memorised. As seen in the GradCAM heatmaps (right), memorised features have an unreasonably high influence in the neural network's decision. *Note the name JOHN in the figure is fictitious and it was artificially added to the images for visualisation purposes. Therefore, the name JOHN cannot be used re-identify the patient in the image. The x-ray images are from the publicly available Chexpert dataset.

Understanding unique feature memorisation in DNNs trained for classification

Task. We explore memorisation in artificial neural networks $f(\mathbf{x}; D_\mathrm{t})$ trained for classification. f maps input data $\mathbf{x}$ to a vector prediction $\hat{\mathbf{y}}$ via a softmax activation function. Each element of $\hat{\mathbf{y}}$ represents the conditional probability of the class label y given the image $\mathbf{x}$. $D_\mathrm{t} = \{\mathbf{x}^i, y^i\}_{i=0}^N$ is the training data where $\mathbf{x}_p \in \mathbb{R}^{l\times l}$, and y is the ground truth class label of $\mathbf{x}$. D_t may or may not contain a data sample $\mathbf{x}_p$ having a unique feature $\mathbf{z}_\mathrm{u} \in \mathbb{R}^{m\times m}$ with $m < l$.

The unique feature (UF). We define a unique feature (UF), $\mathbf{z}_\mathrm{u}$, as a feature or attribute which occurs once in a single sample $\mathbf{x}_\mathrm{u}$ in a training dataset. In image datasets, $\mathbf{z}_\mathrm{u}$ is a set of neighbouring pixels in an image. Throughout the paper, we assume known $\mathbf{z}_\mathrm{u}$. A unique feature label (UFL) y_u is the label of the unique feature on the original training image.

Unique feature memorisation (UFM). We hypothesise that $\mathbf{z}_\mathrm{u}$ is memorised by f, when f has higher confidence on images containing $\mathbf{z}_\mathrm{u}$ than without $\mathbf{z}_\mathrm{u}$. Learning which occurs for $\mathbf{z}_\mathrm{u}$ is memorisation since $\mathbf{z}_\mathrm{u}$ is unique and cannot be learnt from any other label structure in the training data. We measure UFM in three different settings using the M score detailed in Equation 1.

Sensitivity to unique feature. To approximate the memorisation of $\mathbf{z}_\mathrm{u}$ we measure the sensitivity of $f(\mathbf{x}; D_\mathrm{t})$ to a set of image pairs which are clean, i.e. images not containing $\mathbf{z}_\mathrm{u}$, vs. those containing $\mathbf{z}_\mathrm{u}$.

Concepts. We hypothesise that unique features (e.g. "JOHN") are more likely to be memorised because they indeed introduce a new and rare concept (e.g. "name") in the training data. Features are instantiations of concepts. We explain the the difference between feature and concept with an example: on some occasions, patient information such as their name "JOHN" is embedded in the image. How the name appears on the image constitutes a feature. This feature can appear once (infrequent) or several times. Considering the name as a new concept. It will be a rare concept if only *one* or very few images contain features of a name (as opposed to most images containing names). On the other hand, a concept is not rare if several images contain different (yet infrequent) names of patients (e.g. we have still one image with "JOHN" in the dataset, but we have other images with other names).

Private settings. UFM poses privacy concerns. Therefore, we also evaluate how to identify memorisation in more restrictive settings. In all situations, we assume access to the unique feature and the output of the last layer of the NN. We also consider two other settings where

1. we do have access to the unique feature and training data but do not have access to the unique feature label and the model weights, which we call the "grey box" setting;
2. we additionally remove access to the training data, i.e. the "black box" setting, where we only have access to unique feature. The "black box" setting is more realistic since models are routinely exposed behind application interfaces or are made publicly available, whereas their training data are not.

See illustration of these settings in Fig. 2.

M score

M score is a simple method to measure the memorisation of unique features in neural networks. We demonstrate our approach in three settings of increasing difficulty and realism from an attacker's perspective, i.e. different privacy settings.

White box

We introduce the M-score for measuring unique feature memorisation in a setting where we have access to the training data, D_t, and the unique feature label y_u. Let D_{yu} be a subset of the training data with the same label y_u as the unique feature. The score is given by

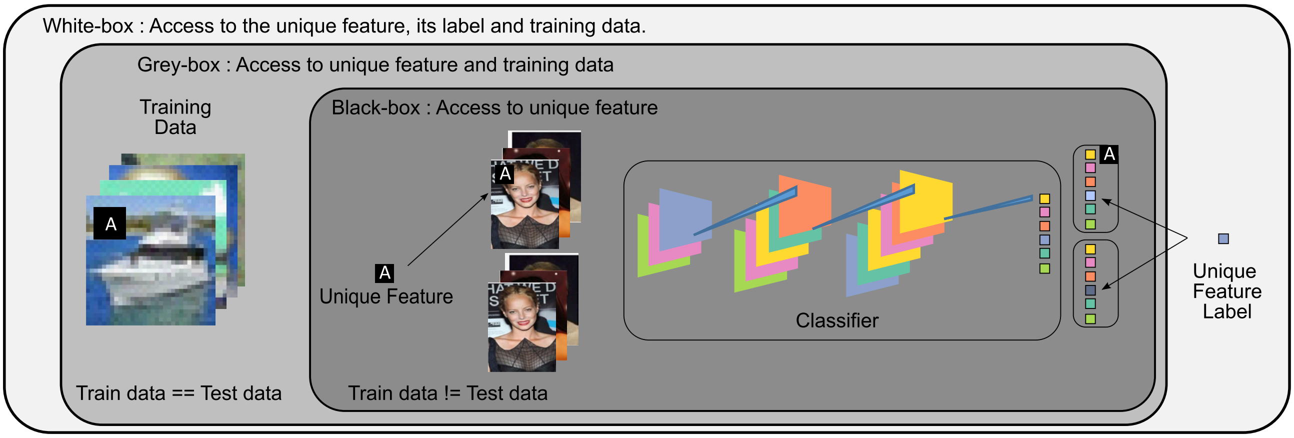
White-box : Access to the unique feature, its label and training data.
Grey-box : Access to unique feature and training data
Training Data
A
Train data == Test data
Black-box : Access to unique feature
A
Unique Feature
A
Classifier
A
Unique Feature Label
Train data != Test data

Figure 2. Our method can identify memorisation across different data availability settings which we classify as (i) white box which assumes access to the training data; (ii) grey box which assumes access to the label of the unique feature; and (iii) black box where only the unique feature is needed to identify memorisation. The faces in the figure are from the publicly available dataset CelebA.

$$M_{\text{white}} = \mathbb{E}_{\mathbf{x}\sim D_{yu}}\left[P(y_u|\mathbf{x}_{\text{u}}) - P(y_u|\mathbf{x}_{\text{c}})\right], \tag{1}$$

where $\mathbf{x}_{\text{u}}$, $\mathbf{x}_{\text{c}}$ represent the same data sample $\mathbf{x}$ with label y, either injected with a unique feature ($\mathbf{x}_{\text{u}}$) or left intact ($\mathbf{x}_{\text{c}}$; with c denoting this 'clean' datum). Intuitively M is the average difference in the label likelihoods between inferences on the $\{\mathbf{x}_{\text{u}}, \mathbf{x}_{\text{c}}\}$ image pairs. This makes the scale of our M score particularly straightforward to interpret.

A similar score was used to measure rare spurious correlations[13]. However, instead of averaging over the training data, M is the fraction of samples with $M < \epsilon$. In practice, we do not find outliers which distort our average score since we scale our model outputs using a softmax function. In addition, our results show that M has a greater sensitivity to unique features which occur only once in the training data set. M runs from -1 to 1. Values of M larger than zero correspond to increasing memorisation since the signal from the unique feature is greater than the images without the unique feature.

Grey box

In the grey box setting we remove the assumption that we know the unique feature label y. In practice this means we know some information, e.g. a patient's name, but we do not know the patient's pathology. In the grey box setting we propose that we can infer y from the score from M_{white} score M_i over each possible label y_i and the final M_{grey} will be the maximum M_{white} across labels. The M-score in the grey setting can therefore be written as

$$M_{\text{grey}} = max\left\{\mathbb{E}_{\mathbf{x}\sim D_{yi}}\left[P(y_i|\mathbf{x}_{\text{u}}) - P(y_i|\mathbf{x}_{\text{c}})\right] \mid \forall y_i\right\}, \tag{2}$$

Black box

We now develop a memorisation score M for practical settings where disclosure agreements prevent us from having access to the training dataset or its distribution. In the black box setting we remove the assumption that we have access to the training data, D_t. This is the most restrictive setting and represents the case where an attacker has obtained a neural network model and knowledge of the unique feature's style. In this setting we simply use another randomly selected dataset D_r. In practice the data distribution does not appear to influence the results and therefore any data set can be used.

$$M_{\text{black}} = max\left\{\mathbb{E}_{\mathbf{x}\sim D_r}\left[P(y_i|\mathbf{x}_{\text{u}}) - P(y_i|\mathbf{x}_{\text{c}})\right] \mid \forall y_i\right\}, \tag{3}$$

Statistical significance

We statistically test every M score result in our experiments. We construct two samples from inferences on the test data $X_u = \{P(y|\mathbf{x}_{\text{u}}^i)\}_{i=0}^n$ and $X_c = \{P(y|\mathbf{x}_{\text{c}}^i)\}_{i=0}^n$. We quantify the statistical significance ($p < 0.05$) of the M score using a one-tailed t-test with an alternative hypothesis that the population mean of $\mathbf{x}_{\text{u}}$ is greater than $\mathbf{x}_{\text{c}}$. We consider that a NN memorised a unique feature when the alternative hypothesis is true.

Metrics

To mitigate stochasticity in measuring memorisation, we run training of neural networks for multiple seeds, all else remaining the same. In some experiments, we run up to 1000 different seeds. We only consider the statistically significant higher M score as memorised networks. To report these results over many seeds, we use the proportion of memorised networks, average M score, maximum M score.

Datasets and unique features

We measure memorisation in several datasets containing a single unique feature (artificially introduced). For imaging datasets, we use F-MNIST[14], CIFAR-10[15], Celeb-A[16], and CheXpert[17]. Celeb-A has multiple labels. We found that many classification tasks were very easy to solve, and the networks found easy shortcuts. Therefore, we choose a binary classification task of attractive/non-attractive whose discriminative features are less well-defined. These datasets span several image sizes, dataset size, contents and styles, and are fairly representative of the problem of image classification in computer vision.

We use "two moons"[18] as a generic low-dimensional dataset in order to characterise and explain UFM in general. In our "two moons" setting, all the classification-related information is present in the x- and y-axes. The z dimension would correspond to a new concept. We consider the z-axis to be an uninformative additional dimension to which the unique feature may be introduced.

We increase the generality of our results by using several unique features. For the datasets F-MNIST, CIFAR-10, and Celeb-A we use a 5×5 image patch of the letter 'A' two pixels from the upper-left corner of the training image as shown in Fig. 2. For the CheXpert dataset we use a fictitious patient name 'JOHN'. Our two-moons experiment is instead simpler: the unique feature is a single value in the 3rd (z-axis) direction.

Models and training strategies

We evaluate our memorisation score using several common architectural styles of neural networks summarised in Table 1.

MLP-1 is trained on MNIST and F-MNIST datasets. We train MLP-1 with a learning rate of 3×10^{-4} and a batch size of 128 samples. We train MLP-2 on the two moons datasets. CNN-1 a simple 2-layer convolutional neural network trained with a batch size of 128 samples. We perform image classification on CIFAR-10 and Celeb-A using ResNet18[19]. This model is formulated specifically for small images 32×32 pixels. We train with a learning rate of 1e-5 and a batch size of 32 samples. We use 5 patient epochs and train for a maximum of 100 epochs. To classify CheXpert images, we use DenseNet121[20]. We use the same hyperparameters as in the original work[17] and train on only on chest X-rays orientated toward the front.

We aim to show that feature memorisation occurs before overfitting, therefore we train with early stopping and fine-tune the number of patient epochs by hand. After training, we select the final model weights from the epoch with the lowest validation loss. We train all models using the Adam optimiser and a cross-entropy loss function[21]. We use the PyTorch deep learning framework to train and evaluate models[22] and SciPy[23] to perform significance testing. We train models using a Nvidia® Titan RTX™. We estimate the computation time for the experiments to be around 200 GPU hours.

Experimental setup

We proceed to propose a series of experiments to study UFM.

Evaluating unique feature memorisation

We measure UFM by evaluating how sensitive a trained DNN is to the insertion of a unique feature into a data sample. We hypothesise that if a DNN has memorised a unique feature, it will be more sensitive to images containing it. Any (statistically significant) increase in confidence after insertion of the unique feature must be due

Model	Architecture	Learning rate
MLP-1	Dense(512) ReLU	3×10^{-4}
	Dense(256) ReLU	
	Dense(128) ReLU	
	Dense(#classes) Softmax	
MLP-2	Dense(3) ReLU	1×10^{-3}
	Dense(32) ReLU	
	Dense(128) ReLU	
	Dense(128) ReLU	
	Dense(2) Softmax	
CNN-1	Conv2D(32,3,3) ReLU	1×10^{-3}
	Conv2D(64,3,3) ReLU	
	MaxPool2d(2,2))	
	Dense(128) ReLU	
	Dense(128) ReLU	
	Dense(#classes) Softmax	
ResNet18	[19]	1×10^{-5}
DenseNet121	[20]	1×10^{-4}

Table 1. Neural networks used in this paper.

to memorisation since the feature is unique and cannot be learnt from any other label in the training data. We measure the *M* score as a proxy of an increase in confidence for UFM. We consider a white box setting for this experiment.

Does regularisation prevent unique feature memorisation?

Regularisation strategies for training models are typically employed to reduce the ability of a model to overfit. These strategies aim to promote learning of features which generalise well to samples outside the training data. Since neural networks are historically assumed to learn common patterns first and memorise labels later during the training process[2,24], it is expected that learning of unique features which do not occur in the test set will be reduced by training with regularisation methods.

In our experiments, we build on these works to understand how regularisation strategies affect unique feature memorisation in image classification models. Using common regularisation strategies (dropout[25], data augmentation, weight decay[26], and batch normalisation[27]), we train each model with early stopping to eliminate overfitting on average across the dataset. We train two neural network models MLP-2 and CNN-1 over MNIST and F-MNIST over 10 random training and measure the maximum M score across runs. We consider a white box setting for this experiment.

UFM and training dynamics

We train 100 NNs over the toy "two moons" dataset with one data point containing a unique feature in the third dimension. We measure the proportion of memorised networks *at each epoch* and mean accuracy across runs. We consider a white box setting.

Rare concepts and UFM

It is a well-known fact that specific hidden units of NNs can be associated with concepts in the data[28]. We explore the interplay between how often features appear in the data and whether they do or do not introduce a rare concept. We hypothesise that the presence of a unique feature introduces a new latent dimension in the space where decisions are made. We train a MLP model on "two moons". In this setting, all the classification-related information is present in the x- and y-axes. The z dimension would correspond to a new concept. We consider the z-axis to be an uninformative additional dimension to which the unique feature may be introduced. We investigate if rarity in z-axis influences memorisation. We measure, in two settings, the proportion of memorised networks from training data containing a single data point with $z = 1$: (i) $z = 0$ for all data samples except for one sample which has $z = 1$, i.e. non-zero values in the z dimension are *rare*; (ii) we add Gaussian noise along the z-axis for samples in the training data while keeping the one data point with $z = 1$. For each case, we train 500 NNs with different seeds, all else kept the same. We consider a white box setting. This toy setting allows us to disregard all questions related to the actual characteristics of the unique features (e.g. a letter "A" or "B" or an entire word "JOHN") since we are only dealing with scalars as opposed to image features. This setting emulates many real-world scenarios. For instance, a patient's name should not be informative about their diagnosis. Or, in most X-ray images, the edges would be black (representing a ubiquitous "background" concept) and do not contain any informative features for diagnosis.

M score and sensitivity to unique features.

We train a model with a single image containing a UF. At test time, we estimate the M score after progressively removing pixels from the unique feature, keeping all else the same.

UFM and risks in medical imaging

Memorisation of data samples poses a privacy risk to individuals whose data is used to train neural networks. This is because information relating to training samples is encoded directly in the weights of a neural network[29,30]. An adversary[31] could construct a readout function acting on the weights or network outputs to discover information about a given sample[32]. Data leakage is particularly problematic when datasets contain private information for which disclosure must be controlled. For example, DNNs used in healthcare may encode information about patients in their weights[9,10], for which disclosure is legally restricted in the EU by the General Data Protection Regulation (GDPR).

Medical imaging offers a practical and realistic example of the risks posed by unique feature memorisation. Hospitals frequently employ sanitisation processes to remove patient names when they appear overlaid on X-ray films (see Fig. 1). There is a possibility that these processes may fail, resulting in the addition of personal private data to training data. Hence a properly designed readout function may indeed lead to the recovery of such private information from the model.

There is another risk with the inclusion of such unique features. A classifier trained on such data may misdiagnose other patients with the same name if those names have not also been removed during the sanitisation process. Alternatively, the unintended presence may lead to incomplete extraction of the correct discriminative features from the image[8]. Such a risk is similar to decision-making based on spurious correlations, except that only a single spurious feature is present in the dataset[9–12].

We train a classifier on the CheXpert chest X-Ray dataset. We add a unique feature 'JOHN' to a single training image in the upper left corner, and overlay a black rectangle image of the same size over the other images. We generated explanations of the classification with a GradCAM heatmap for predictions made by the NN trained on our modified CheXpert dataset.

Identifying memorisation in private settings
Since UFM poses privacy concerns, we now focus on identifying memorisation in more restrictive settings where

1. we do have access to the unique feature and training data but do not have access to the unique feature label and the model weights, which we call the "grey box" setting;
2. we additionally remove access to the training data, i.e. the "black box" setting, where we only have access to unique feature.

In all situations, we assume access to the unique feature and the output of the last layer of the NN. See illustration of these settings in Fig. 2. The "black box" setting is more realistic since models are routinely exposed behind application interfaces or are made publicly available, whereas their training data are not.

Grey box setting. We remove access to the unique feature label for "grey box" M score. We train 10 models on F-MNIST dataset, each trained with a unique feature inserted into a different randomly selected training image from class 1. Then, we measure the "white box" M score and the "grey box" M score to verify if they are indeed correlated. For each model, we also indicate the predicted unique feature label $\hat{y}$ in the "grey box" box setting. Next, we repeat this experiment for the Celeb-A dataset.

Black box setting. We remove access to the training dataset for "black box" M score. We train 10 models on Celeb-A and CIFAR-10 datasets, each trained with a unique feature inserted into a different randomly selected training image from class 1. Then, we measurethe "white box" M score and the "black box" M score. We evaluate the memorisation of the unique feature in CIFAR-10 and Celeb-A using Celeb-A and CIFAR-10 respectively at inference time.

Results

We now show empirical results demonstrating that

1. neural networks memorise unique features in several datasets for a range of model architectures;
2. memorisation of unique features cannot be prevented using typical regularisation strategies;
3. memorisation happens due to the presence of such a rare feature which is unusual and hence unique with respect to (they are unusual only once) features in concepts which are rare (unusual) in the data and it happens from the first epoch and over the entire unique feature;
4. we are able to audit models with the M score in a grey or black box setting (different settings are illustrated in Fig. 2).

We refer the reader to the Methods section for full details of the datasets, models, training schemes and memorisation scores.

Neural networks memorise unique features

Following the experiment described in section 2.5, Table 2 shows that UFM occurs frequently in a range of neural network architectures and benchmark datasets from simple to complex: a variation of the two moons dataset[18] (described in Fig. 3), F-MNIST[14], CIFAR10[15], CheXpert[17], CelebA[16]. We conducted experiments using different training seeds and training stochasticity as shown in Table 2 (i.e. number of runs). Based on the proportion of memorised networks (where M is statistically significant with $p < 0.05$), it is noted that memorisation is not always present. Hence, different training seeds and training stochasticity lead to different memorisation results. We further visualise in Fig. 3b), for setting (i), the decision boundary in the x-y plane for $z \in \{0, 1\}$ and the differences for networks that memorised the datapoint in $z = 1$ and networks that did not. The memorising networks have a stronger shift in the decision boundary in the $z = 1$ plane, which would correspond to having test images which contain unique features. This experiment illustrates how unique feature memorisation increases the risk of misclassification when the unique feature is present in the test data. Indeed, there are more misclassifications of samples which include the unique feature ($z = 1$ plane) and those with representative x-y features of the opposite class.

Regularisation does not prevent unique feature memorisation

Results shown in Table 3 empirically demonstrate that regularisation does not significantly reduce the M score (UFM). This is in line with recent works showing that regularisation strategies do not eliminate memorisation in neural networks. For example, explicit regularisation does not prevent sample-based memorisation[1] or feature-based memorisation in language modelling[33, 34]. More recently, it has been shown that the influence of rare spurious features could not be eliminated by either weight decay or by introducing Gaussian noise to training inputs[13].

Memorisation happens early during training

Some networks memorise unique features from the *first* training epoch. We find that learning of unique features occurs early in training process, similarly to sample memorisation[35] and feature memorisation in language modelling[33,34]. Thus, it appears that UFM occurs even when the feature values in the sample are not overfitted. We depict in Fig. 4 an experiment with a toy dataset showing that memorisation happens in around 40% since the first epoch. We illustrate that, while the likelihood of memorisation increases with overfitting, memorisation happens from the beginning of training.

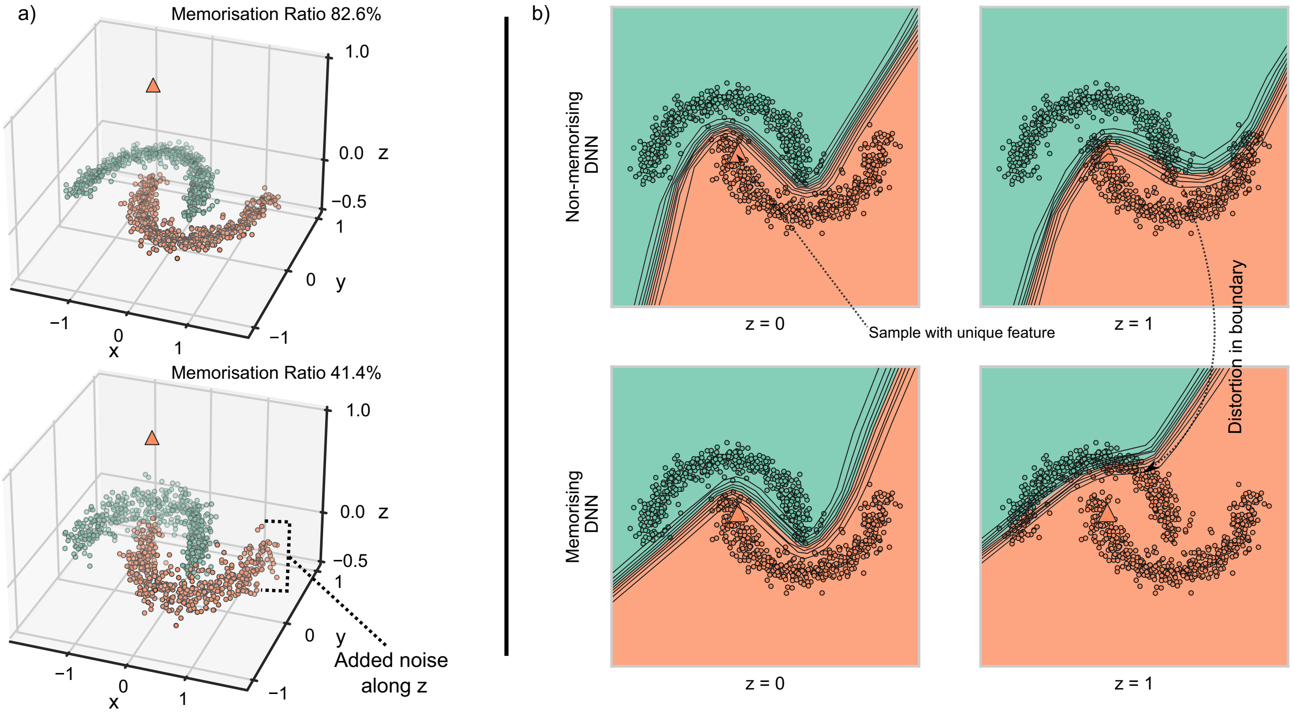

Figure 3. A study of neural network memorisation with the "two moon" toy dataset. All the classification-relevant information is present along the *x* and *y*. We study situations when a data sample with a unique feature in the *z* (depicted with a triangle marker) dimension is memorised. (**a**) Top, is the case where the dataset has a unique feature on *z*-dimension. 82.6% of NNs memorised the feature. Bottom, here noise is added along *z*, and now the concept is not unique. Indeed now, the proportion of NNs which memorised the unique feature is much smaller (41.4%). (**b**) We now explore how decision boundaries change whether a unique feature is memorised or not. The decision boundaries for $z = 1$ (i.e. data points with the unique feature) differ considerably between a network that memorised (bottom) and one that did not (top).

Dataset	Model	Number of runs	Proportion of memorised networks	Average M score
Two moons	MLP-2	1000	65%	0.51
F-MNIST	CNN-1	10	20%	0.4
CIFAR-10	ResNet18	100	46%	0.03
Celeb-A	ResNet18	10	80%	0.01
CheXpert	DenseNet121	10	60%	0.007

Table 2. Unique feature memorisation occurs frequently in neural networks. Here we show average *M* scores for a range of datasets and model architectures.

Rare concepts lead to unique feature memorisation

We found, as we illustrate in Fig. 3 a), that 82.6% of the networks memorised for setting (i) while only 41.4% did in (ii). This result indicates that a rare concept leads to memorisation even with the introduction of Gaussian noise. We show that memorisation is stronger when rarity in concept and in feature coincide, creating a unique feature (UF).

M score measures sensitivity to unique features

As described in section "M score and sensitivity to unique features", removing a single pixel from the unique feature changes the score significantly. This implies that the model has learned to capture the whole unique feature verbatim. It has not for example extracted representations that approximate the UF (e.g. its edges, corners etc). We find that unique features are captured by the M score in their entirety (see Fig. 5).

UFM poses a robustness and privacy risk to medical imaging

Based on the experiment detailed in section "UFM and risks in medical imaging", Fig. 8 shows GradCAM explanations for predictions on 'Consolidation' for three models trained with the unique feature on different images. The upper heatmaps clearly show that the private personal information feature explains the model's prediction, and that true explanatory features relating to the physiology of the patient are considerably less

Dataset	Model	Regularisation	Max. *M* score
MNIST	MLP-1	Dropout	0.010
MNIST	MLP-1	Data augmentation	0.010
MNIST	MLP-1	Weight decay	0.018
MNIST	MLP-1	Batch normalisation	0.020
MNIST	CNN-1	Dropout	0.008
MNIST	CNN-1	Data augmentation	0.009
MNIST	CNN-1	Weight decay	0.011
MNIST	CNN-1	Batch normalisation	0.008
F-MNIST	MLP-1	Dropout	0.109
F-MNIST	MLP-1	Data augmentation	0.077
F-MNIST	MLP-1	Weight decay	0.104
F-MNIST	MLP-1	Batch normalisation	0.131
F-MNIST	CNN-1	Dropout	0.190
F-MNIST	CNN-1	Data augmentation	0.039
F-MNIST	CNN-1	Weight decay	0.140
F-MNIST	CNN-1	Batch normalisation	0.011

Table 3. Max *M* scores for models with explicit and implicit regularisers, such as dropout, data augmentation, and batch normalisation.

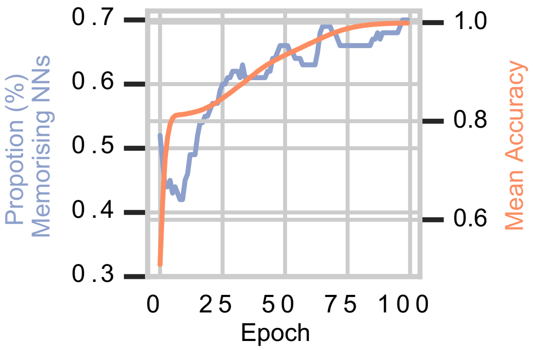

Figure 4. Memorisation during training. We depict the proportion (in percentage) of networks which memorised the unique feature per epoch, out of 100 runs with different seeds. We also display the mean test accuracy over the 100 NNs for each epoch.

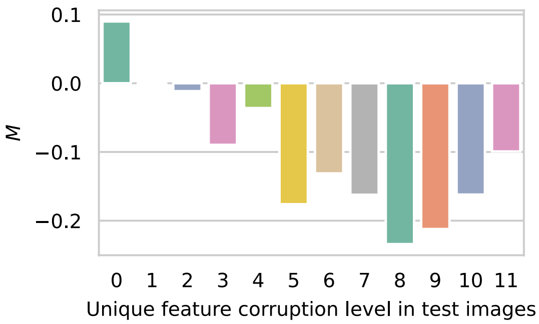

Figure 5. We measure unique feature memorisation in CIFAR-10 by inferring a trained model's confidences on test images containing the unique feature. We find that the memorisation score reduces when we corrupt the unique feature by successively removing pixels during inference. This indicates that the model memorises the entire unique feature and not a corrupted version.

explanatory. However, when the unique feature is removed from the lower images, the explanations for the pathology surround physical features which are expected. The upper heatmaps clearly show that the private personal information feature explains the model's prediction, and that true explanatory features relating to the physiology of the patient are considerably less explanatory. However, when the unique feature is removed from the lower images, the explanations for the pathology surround physical features which are expected. This simulates the removal of private personal information and the counter-case of accidentally missing some unique private personal information.

Identifying memorisation in private settings

In the grey box setting, Fig. 7 shows the "white box", "grey box" and "black box" M scores are correlated on the Celeb-A dataset. In this context, white box setting is seen as the ground truth. In the black box setting, we observe that NNs are more sensitive to a data point after the insertion of the unique feature. Interestingly, the specific data distribution is not important, since our method finds only the relative distances between model outputs from image pair inputs. "black box" M score is less accurate on models trained on the CIFAR-10 dataset, see Fig. 6.

Discussion

Unique Feature Memorisation

We show that unique feature memorisation is not uncommon in classification neural networks for low-dimensional data and in a range of deep learning models for image classification. Also, we find that regularisation does not eliminate UFM, and that similarly to language modelling, singly occurring unique features are learnt early in training. A letter or name, for example, written on a natural image can often be memorised by DNNs trained using the backpropagation algorithm. We hypothesise and validate empirically that these features are more likely to be memorised when they appear in explored dimensions, as shown in Fig. 3.

Typically, we would expect the learning algorithm to ignore the unique feature. This is because under the information bottleneck (IB) principle, information learnt from the other samples in the training dataset is sufficient to reduce the uncertainty of the label distribution[36,37]. However, in practice this does not seem to be the case. We suggest the following explanation for this behaviour. Let us assume that the classifier is extracting a latent space from the input data. We can theoretically partition the latents in two parts: those learned from the samples according to the IB principle and those attributed to the unique feature. Indeed our results in Fig. 3 hint at this. The decision boundary in 2D for samples without the unique feature ($z = 0$) is the expected hyperplane, whereas the decision boundary is completely shifted for samples with the unique feature ($z = 1$). Under the *Principle of Least Effort*[11], the learning algorithm may shortcut over the unique feature since it is easy to learn, and as our results suggest it may do so early on in training (Fig. 3). We believe that studying learning dynamics (model behaviour during training) is a good way of understanding the memorisation phenomenon critically. Similarly, recent works show that shortcuts (a similar concept) are memorised in the beginning of training[38,39] and the connection between local minima in the loss landscape[39].

Previous research has established that over-trained, over-parameterised deep neural networks are able to memorise randomised training labels, and randomised data samples[1]. As a result of this finding, a number of methods have been developed to measure label and sample memorisation[3,4,40,40–49]. Early work on understanding memorisation suggested that neural networks learn patterns early in training and memorise random patterns later[2]. More recently, it has been shown that learning and sample memorisation occur simultaneously[35]. Few studies have investigated the memorisation of features. Recent works on the privacy risks of Large Language Models (LLMS) established that LLMs memorise features even when they rarely appear in training data[33,34].

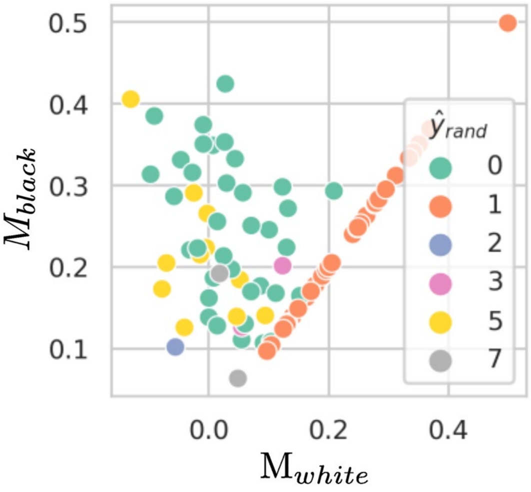

Figure 6. Correlation between M score in "white box" and "black box" setting for the Cifar-10 dataset. We show that unique feature memorisation can be measured without access to the original training data or to the unique feature label.

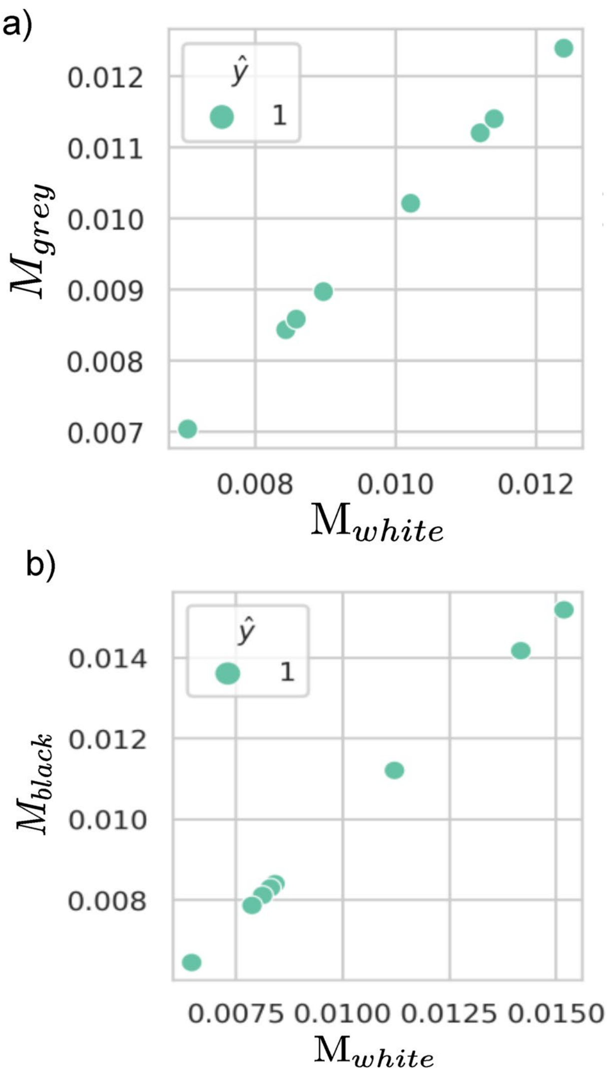

Figure 7. Correlation between M score in different privacy settings for the Celeb-A dataset. We show that unique feature memorisation can be measured without access to the original training data or to the unique feature label. The "white box" M score is shown along the *x*-axis and the "grey box" (**a**) or "black box" (**b**) M score is shown on the *y*-axis.

Decision-making based on spurious correlations is a similar topic to feature memorisation[9–13]. Our investigation into unique feature memorisation follows naturally from these works. We distinguish ourselves by investigating feature memorisation in its most extreme form: where a unique feature occurs only once in training data.

Privacy and Unique Features

Unique features might contain personal information, which poses serious privacy concerns in certain settings such as decision-making in healthcare. We identify that models leak information about unique features that were memorised during training. More importantly, we show that we can audit if models memorised specific features in private settings, when the auditor does not have access to the training data nor to the unique feature label.

Other works propose privacy attacks which also exploit data leakage to uncover information about training data. We now detail some techniques from the literature and how our work differentiates, in particular

1. membership inference attacks deduce, whether a sample is in the training set by exploiting a model's overconfidence on examples it has seen[32,50–53]. We focus on the memorisation of unique features and not whole data samples or datasets.
2. backdoor attacks attempt to adversarially change a model's predictions by injecting an optimised image patch onto training examples such that when this patch occurs on an attack example at test time, the predictions of the model can be controlled[31,54–59]. In contrast, we show that a unique feature which occurs in the training data is memorised. This feature is not optimised to modify the outputs of the model at test time.
3. property inference attacks attempt to learn a group property/feature of the dataset. For example, what proportion of people in the training set wear glasses?[60,61]. These attacks are typically white box and proceed

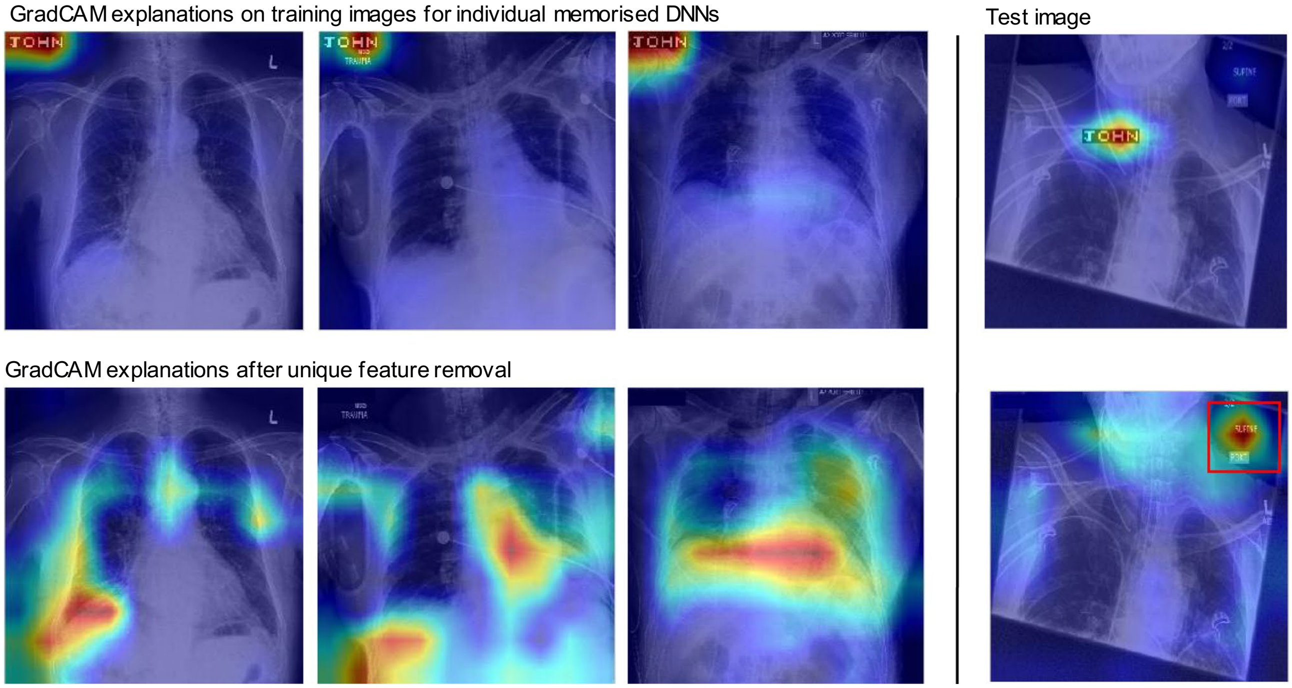
GradCAM explanations on training images for individual memorised DNNs
Test image
JOHN
GradCAM explanations after unique feature removal

Figure 8. GradCAM explanations focus on memorised unique features in the CheXpert[17] dataset (upper left). However, after removing the unique features, the model's predictions are explained by the physiology of the patient (lower left). On test set images, right, physiological features are ignored when a unique feature is present. When the feature is removed, the network finds another spurious correlation shown in a red box.

by using a shadow model to make inferences on the target model weights. Feature memorisation, as we investigate here, can be viewed as an extreme property inference attack where a unique feature, a person who wears glasses, occurs only once in the dataset. Existing approaches, however, cannot address unique feature memorisation since labelling the training weights to train the shadow model requires ground-truth knowledge of whether the feature was memorised or not.

Guidelines and Best Practices

The findings of this study highlight the need to develop strategies to protect personal information when present as a unique feature. One of the possible ways to avoid the presence/influence of unique features is to develop automatic solutions to detect personal information printed on training images for removal before moving forward with machine learning training. Another suggestion for safeguarding is to develop a privacy filter (testing stage) that rejects/modifies an image with identifiable information printed on it so that an attacker will not be able to get access to identifiable information learned by neural networks. By doing that, a data scientist is lowering the possibility of linking a breached patient record (as happened in England (https://www.bbc.co.uk/news/technology-44682369)) to training data of their ML model. The findings will also inform policymakers to develop practices and guidelines for data scientists and companies to protect personal information for those situations according to policy document (https://www.gov.uk/government/publications/ai-regulation-a-pro-innovation-approach/white-paper) by safeguarding against bad actors.

Data availability

All imaging datasets used in this paper are publicly available. The code for generating the synthetic two moons dataset can be found at https://github.com/jasminium/feature-memorisation.

Code availability

Code to reproduce the experiments is available at: https://github.com/jasminium/one_sample_memorisation.

Received: 21 May 2023; Accepted: 21 November 2023
Published online: 04 December 2023

References

1. Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* **64**, 107–115 (2021).
2. Arplt, D. *et al.* A closer look at memorization in deep networks. In *34th International Conference on Machine Learning, ICML 2017* **1**, 350–359 (2017).
3. Feldman, V. Does learning require memorization? A short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, 954–959 (2020).

4. Feldman, V. & Zhang, C. What neural networks memorize and why: discovering the long tail via influence estimation. In *Advances in Neural Information Processing Systems* (eds. Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F. & Lin, H.), vol. 33, 2881–2891 (2020).
5. Kaissis, G. A., Makowski, M. R., Rückert, D. & Braren, R. F. Secure, privacy-preserving and federated machine learning in medical imaging. *Nat. Mach. Intell.* **2**, 305–311. https://doi.org/10.1038/s42256-020-0186-1 (2020).
6. Mincu, D. & Roy, S. Developing robust benchmarks for driving forward ai innovation in healthcare. *Nat. Mach. Intell.* **4**, 916–921. https://doi.org/10.1038/s42256-022-00559-4 (2022).
7. Liang, W. *et al.* Advances, challenges and opportunities in creating data for trustworthy ai. *Nat. Mach. Intell.* **4**, 669–677. https://doi.org/10.1038/s42256-022-00516-1 (2022).
8. DeGrave, A. J., Janizek, J. D. & Lee, S. I. AI for radiographic COVID-19 detection selects shortcuts over signal. *Nat. Mach. Intell.* **3**, 610–619 (2021).
9. Bar, Y. *et al.* Chest pathology detection using deep learning with non-medical training. In *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 294–297 (2015).
10. Zech, J. R. *et al.* Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLoS Med.* **15**, 1–17 (2018).
11. Geirhos, R. *et al.* Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2**, 665–673 (2020).
12. Idrissi, B. Y., Arjovsky, M., Pezeshki, M. & Lopez-Paz, D. Simple data balancing achieves competitive worst-group-accuracy. *Conference on Causal Learning and Reasoning. PMLR (*2022)
13. Yang, Y.-Y. & Chaudhuri, K. Understanding rare spurious correlations in neural networks. arXiv preprint arXiv:2202.05189 (2022).
14. Xiao, H., Rasul, K. & Vollgraf, R. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017).
15. Krizhevsky, A., Hinton, G. *et al.* Learning multiple layers of features from tiny images. CS Utoronto CA (2009).
16. Liu, Z., Luo, P., Wang, X. & Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)* (2015).
17. Irvin, J. *et al.* Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence* **33**, 590–597 (2019).
18. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
19. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2016-Decem**, 770–778 (2016).
20. Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4700–4708 (2017).
21. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. In *ICLR (Poster)* (2015).
22. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, 8024–8035 (Curran Associates, Inc., 2019).
23. Virtanen, P. *et al.* SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272. https://doi.org/10.1038/s41592-019-0686-2 (2020).
24. Kim, Y., Kim, M. & Kim, G. Memorization precedes generation: Learning unsupervised GANs with memory networks. In *International Conference on Learning Representations* (2018).
25. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
26. Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations* (2019).
27. Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456 (PMLR, 2015).
28. Bau, D., Zhou, B., Khosla, A., Oliva, A. & Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6541–6549 (2017).
29. Golatkar, A., Achille, A. & Soatto, S. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 9301–9309 (2020).
30. Jegorova, M. *et al.* Survey: Leakage and privacy at inference time. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).
31. Usynin, D. *et al.* Adversarial interference and its mitigations in privacy-preserving collaborative machine learning. *Nat. Mach. Intell.* **3**, 749–758. https://doi.org/10.1038/s42256-021-00390-3 (2021).
32. Shokri, R., Stronati, M., Song, C. & Shmatikov, V. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18 (IEEE, 2017).
33. Carlini, N., Liu, C., Erlingsson, U., Kos, J. & Song, D. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *Proceedings of the 28th USENIX Conference on Security Symposium*, SEC'19, 267–284 (USA, 2019).
34. Carlini, N. *et al.* Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–2650 (2021).
35. Liu, F., Lin, T. & Jaggi, M. Understanding memorization from the perspective of optimization via efficient influence estimation. *OPT2021: 13th Annual Workshop on Optimization for Machine Learning* (2021).
36. Tishby, N. & Zaslavsky, N. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, 1–5 (2015).
37. Achille, A. & Soatto, S. Emergence of invariance and disentanglement in deep representations. *J. Mach. Learn. Res.* **19**, 1947–1980 (2018).
38. Pezeshki, M. *et al.* Gradient starvation: A learning proclivity in neural networks. *arXiv preprint*arXiv:2011.09468 *(2020).*
39. Lubana, E. S., Bigelow, E. J., Dick, R. P., Krueger, D. & Tanaka, H. Mechanistic mode connectivity. In *International Conference on Machine Learning*, 22965–23004 (PMLR, 2023).
40. Jiang, Z., Zhang, C., Talwar, K. & Mozer, M. C. Characterizing structural regularities of labeled data in overparameterized models. In *Proceedings of the 38th International Conference on Machine Learning*, vol. 139 of *Proceedings of Machine Learning Research* (eds. Meila, M. & Zhang, T.), 5034–5044 (PMLR, 2021).
41. Koh, P. W. & Liang, P. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, ICML'17, 1885–1894 (2017).
42. Katharopoulos, A. & Fleuret, F. Not all samples are created equal: Deep learning with importance sampling. In *Proceedings of the 35th International Conference on Machine Learning*, vol. 80 of *Proceedings of Machine Learning Research* (eds. Dy, J. & Krause, A.), 2525–2534 (PMLR, 2018).
43. Carlini, N., Erlingsson, U. & Papernot, N. Prototypical Examples in Deep Learning: Metrics, Characteristics, and Utility. https://openreview.net/forum?id=r1xyx3R9tQ (2019).
44. Ghorbani, A. & Zou, J. Data shapley: Equitable valuation of data for machine learning. In *36th International Conference on Machine Learning, ICML 2019* **2019-June**, 4053–4065 (2019).
45. Toneva, M. *et al.* An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019* 1–19 (2019).

46. Garima, Liu, F., Kale, S. & Sundararajan, M. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems* **2020-Decem** (2020).
47. Guo, H., Rajani, N. F., Hase, P., Bansal, M. & Xiong, C. FastIF: Scalable Influence Functions for Efficient Model Interpretation and Debugging, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (2020).
48. Baldock, R. J. N., Maennel, H. & Neyshabur, B. Deep Learning Through the Lens of Example Difficulty. Advances in *Neural Information Processing Systems* (2021).
49. Harutyunyan, H. *et al.* Estimating informativeness of samples with smooth unique information. In *International Conference on Learning Representations* (2021).
50. Sablayrolles, A., Douze, M., Schmid, C. & Jégou, H. Deja Vu: An empirical evaluation of the memorization properties of ConvNets. *arXiv preprint arXiv:1809.06396* (2018).
51. Salem, A. *et al.* ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. *arXiv preprint arXiv:1806.01246* (2018).
52. Liu, X. & Tsaftaris, S. A. Have you forgotten? A method to assess if machine learning models have forgotten data. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 95–105 (Springer, 2020).
53. Choquette-Choo, C. A., Tramer, F., Carlini, N. & Papernot, N. Label-only membership inference attacks. In *International Conference on Machine Learning*, 1964–1974 (PMLR, 2021).
54. Chen, X., Liu, C., Li, B., Lu, K. & Song, D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526* (2017).
55. Gu, T., Dolan-Gavitt, B. & Garg, S. Badnets: Identifying vulnerabilities in the machine learning model supply chain (2017).
56. Liu, Y. *et al.* Trojaning attack on neural networks. In *NDSS* (2018).
57. Muñoz-González, L. *et al.* Towards poisoning of deep learning algorithms with back-gradient optimization. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, 27–38 (2017).
58. Shafahi, A. *et al.* Poison frogs! targeted clean-label poisoning attacks on neural networks. Advances in N*eural Information Processing Systems* 31 (2018).
59. Saha, A., Subramanya, A. & Pirsiavash, H. Hidden trigger backdoor attacks. *Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 07* (2020).
60. Ateniese, G. *et al.* Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. *Int. J. Secur. Netw.* **10**, 137–150 (2015).
61. Ganju, K., Wang, Q., Yang, W., Gunter, C. A. & Borisov, N. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, 619–633 (2018).

Acknowledgements

This work is supported by iCAIRD, which is funded by Innovate UK on behalf of UK Research and Innovation (UKRI) [Project Number 104690]. S.A. Tsaftaris acknowledges also support by a Canon Medical/Royal Academy of Engineering Research Chair under Grant RCSRF1819\8\25. We acknowledge the UK's Engineering and Physical Sciences Research Council (EPSRC) support via grant EP/X017680/1.

Author contributions

J.H. and S.A.T. conceived the study. S.A.T secured the funding. J.H. and P.S. did experiments and collected results, F.H. contributed to the analysis of results. All authors contributed to the manuscript and approved the submission.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.H.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.