



# Cloud-based differentially private image classification

Elie Chicha<sup>1,2</sup> · Bechara Al Bouna<sup>1</sup> · Mohamed Nassar<sup>3</sup> · Richard Chbeir<sup>2</sup>

Published online: 10 December 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

## Abstract

In this paper, our aim is to design and develop an anonymous full-duplex image classification framework under Differential Privacy. We work under the assumption that both, the cloud and the querier are semi-trusted entities, thus their data should remain safe and confidential. That is, neither the querier nor the cloud should be able to link a particular individual from the other party to an image while maintaining, to a certain extent, suitable classification accuracy. We use Principal Component Analysis (PCA) to transform sample images into anonymized vectors; differentially private synopsis of PCA vectors, and we ensure that the individuals in these vectors remain unidentifiable.

**Keywords** Differential privacy · Classification · Principal component analysis

## 1 Introduction

Cloud computing services are evolving like never before as companies tend to reduce their costs and shift their services to the cloud. These services are playing an essential role in the development and improvement of many sectors, including governmental, economical, financial, and educational. In a typical cloud computing scenario, data owners must store their data in the cloud, which raises many concerns, particularly, in privacy and security [8]. On the one hand, the cloud hosting company must ensure safe storage of the data and preserve their privacy. On the other hand, owners must cope, at their discretion, with the risk of

an intentional or unintentional breach of privacy. Besides, some of the cloud computing services require access to data that might be considered sensitive. Without this data, these services cannot function properly. For example, let us assume the following scenario where sentiments of spectators at an event need to be extracted and analyzed. Pictures of the public should be taken and sent to a cloud service that analyzes the sentiments (e.g., laughing, smiling, or neutral). In this scenario, a privacy concern arises when the cloud service uses private pictures to train the classifier as well as when it processes user requests, i.e., images, to infer the sentiments. Thus, the research question of this paper is how to benefit from the data without harming the privacy of individuals in the dataset?

This is not an easy question, and things do not end well in most of the scenarios. For example, in the year 2000, Netflix released an “anonymized” movie viewing dataset by stripping all identifying information. They wanted to make the dataset available for enhancing their movie recommendation algorithms. Unfortunately, their de-identification was vulnerable. Narayanan and Shmatikov [12] showed that they could re-identify specific individuals, and predict their political affiliation. All they needed is some small extra amount of information about a given individual.

Many cloud services that provide private image classification rely on encryption to ensure the security and privacy of the data. However, encryption is not useful in preserving privacy in many cases like in an adversarial

---

✉ Elie Chicha  
elie.chicha@ua.edu.lb

Bechara Al Bouna  
bechara.albouna@ua.edu.lb

Mohamed Nassar  
mn115@aub.edu.lb

Richard Chbeir  
rchbeir@acm.org

<sup>1</sup> TICKET Laboratory, Antonine University, Hadat-Baabda, Lebanon

<sup>2</sup> LIUPPA Laboratory, University of Pau and Pays Adour, Anglet, France

<sup>3</sup> Computer Science Department, American University of Beirut, Beirut, Lebanon

setting. For instance, while using partially homomorphic encryption to outsource K-Nearest Neighbors [18] classification, the authors show that distance learning attacks are possible [10]. These approaches are time-consuming as well due to the performance of encryption and decryption algorithms, and they are vulnerable to the theft of encryption/decryption keys [15].

In this paper, we propose a cloud-based differentially private image classification approach that protects the privacy of individuals in a dataset of images. In fact, in our approach, we assume that the cloud computing service is semi-trusted and thus should not be able to identify individuals in the dataset (i.e., images collected from the data owners) and individuals in the requests of querier (i.e., a typical user of the cloud service). We use a differential privacy mechanism [2] to add noise to the images, and at the same time, preserve their utility in a way that they remain useful for analysis. More specifically, we use Principal Component Analysis (PCA) to transform images into vectors to which we add differentially private noise. We study the trade-off between the accuracy and the privacy regarding the global privacy budget, which is the total allowed leakage as determined by the number of answered queries and the accuracy of the answers. In fact, we run several classification algorithms such as Support Vector Machines, Kernel Density Estimation (KDE) and K-NN to evaluate the privacy vs. accuracy trade-off.

The remainder of this paper is organized as follows, in Sect. 2, we give a brief overview of Differential Privacy and PCA. In Sect. 3, we present the general architecture of our framework and provide details on how to add noise to the image dataset. The framework is evaluated using a set of experiments in Sect. 4. In Sect. 5, we discuss some of the related works, and we conclude in Sect. 6.

## 2 Background

### 2.1 Differential privacy

In many domains, getting statistical data about a dataset is necessary. However, this data can turn into a real threat to the privacy of any individual, participating or not, in this dataset. This breach of privacy is due to two factors; inferring auxiliary data about an individual, and inferring statistical data about a community.

Dwork et. al. [6] has found a solution by adding noise to the result in a way that preserves the utility of the result and protect the individuals privacy. This technique is called Differential Privacy. In their work, databases  $D$  are considered as being collections of records from a universe  $\chi$ , and represented by their histograms  $\mathbb{N}^{|\chi|}$ . Each entry  $D_i$  in a

histogram represents the number of elements in the database  $D$  of type  $i$ .

A randomized algorithm  $M$  with domain  $\mathbb{N}^{|\chi|}$  is  $\alpha$ -differentially private if for all  $S \subseteq \text{Range}(M)$  and for all  $D, D' \in \mathbb{N}^{|\chi|}$  such that the  $norm - 1$  of the distance between databases  $D$  and  $D'$  is only one, we have:

$$Pr[M(x) \in S] \leq e^\alpha \cdot Pr[M(y) \in S] \quad (1)$$

where  $\alpha$  is a privacy parameter that is set by the database owner.

The selection of the privacy parameter  $\alpha$  is a social question. The database owner should choose the best  $\alpha$  for this data. Lowering  $\alpha$  reduces the utility of the result, while raising  $\alpha$  reduces the privacy. The owner should ask how much privacy is enough, and do some experiments to decide the value of  $\alpha$ .

On the other hand, the utility of the approach strongly depends of the sensitivity  $\Delta f = \max\|f(D) - f(D')\|_1$ , where  $D$  and  $D'$  are two databases that differs in at most one row, and  $f$  is a query applied on  $D$  and  $D'$ .

Differential Privacy has two different modes of operation: the interactive and non-interactive modes [16]. In the interactive mode, the server returns a noisy result to the user. The value of the privacy parameter is subject of negotiation between the database owner and the user.

In the non-interactive mode, the server returns a noisy synopsis dataset. The user sends queries to this synopsis and gets statistical results without any interactivity with the owner and without any limits to the number of queries. Our work relies on the non-interactive mechanism to create a synopsis dataset of noisy vectors to form a training dataset.

### 2.2 Principal components analysis

PCA [14] is a singular value decomposition to reduce data dimensionality. Given data points in an  $m$ -dimensional space, PCA projects them into lower dimensional space while preserving as much information as possible. Consider a matrix  $S(n \times m)$ , in this work, each column of the matrix represents an image of  $m$  features. These features are for instance the RGB values (or the grayscale values) of the image pixels. The features are subtracted from their respective empirical means. Assuming a normal (Gaussian) probability distribution of errors, orthogonal transformations naturally arise. The PCA is an orthogonal transformation (actually a coordinate rotation) that aligns the transformed axes with the directions of maximum variance:

$$S = U \sum V^T \quad (2)$$

where  $U$  and  $V$  are orthogonal matrices and  $\sum$  has the set of singular values.  $U(m \times m)$  has for columns the eigen vectors of the covariance matrix  $C$  of  $S$ . The first column of

$U$  is the eigen vector having the largest absolute eigen value (the first principal component). The most significant variance is in the direction of the first principal component. The most significant variance on the subspace orthogonal to this vector is in the direction of the second principal components, and so on. The matrix  $B = U^T S$  is therefore useful for dimensionality reduction of the original data. In effect,  $B_r = U_r^T S$ , where  $U_r$  is the smaller matrix having only the first  $r < m$  columns of  $U$ , is a reduced representation of the data.  $B_r$  is an  $r \times n$  matrix. Data reconstruction is governed by  $S_r = U_r B_r$  (an  $n \times m$  matrix), which is a lossy version of the original data  $S$  [2].

In this paper, we intend to add enough noise to the PCA vectors of images in a way that the individuals in the images remain unidentifiable, but at the same time, the vectors can be used to train the classifier.

### 3 Differentially private image classification framework

We assume that the first party (the data owner) holds a dataset of face images. The images belong to different classes, and each image is labeled with its class number. The second party, which is the end-user, has one face image of unknown class and wants to predict this class, or a multi-face image and wants to know how many faces have a specific reaction. In non-private settings, this would be easy: We use a database with labeled images to train the classifier, and we predict an unknown image based on the classifier's result.

In private settings, we want to imagine a man in the middle that asks for information about the image and asks information about the image database (or the classification model). This man in the middle must not be able to recognize the identity of any subject no matter how many questions he asks.

Both parties must add noise to make the whole process differentially private. We want in the result that any single image in the database is masked, by reducing every image into a noisy vector. The query image is masked as well. However, to ensure an accurate classification, adding noise to a database of images and requests must be done using a global privacy parameter.

In our framework (see Fig. 1), four types of actors are listed:

- *Trusted Data Owner* sends the essential dataset to anonymization service, and later, can send more images, with trusted classification.
- *Untrusted Data Owner* sends images to the anonymization service, with untrusted classification. Trusted and untrusted data owners may request a noisy image from

the synopsis dataset to be sure that the face is unrecognizable. The principal component vectors, the eigenvectors, and the mean vectors are required to reconstruct images, so the service keeps the eigenvectors and means vectors on a different cloud.

- *Querier* sends one face or multi-face images to the anonymization service with the aim of getting their classifications.
- *Data Host* receives three types of vectors:
  - (1) Trusted data owner vectors (sample noisy images) are used to train the classification algorithm.
  - (2) Untrusted data owner vectors are classified, and the output is returned to the anonymization service to check the truthfulness of the data owner classification.
  - (3) Querier vectors are classified, and the output is returned to the querier.

#### 3.1 Anonymization service

The service transforms every image in the primary dataset into a vector using PCA and adds differentially private noise to the vectors. It performs safe sampling and returns a synopsis dataset and checks the truthfulness of the untrusted data owner classification. The process breaks down as follows:

The first step is to concatenate a set of images into one matrix  $S$  as PCA reduces a set of images into a set of vectors. Then, we compute the covariance matrix  $C = \frac{1}{m} S^T S$ ; where  $m$  is the number of columns of  $S$ .

PCA is turned into a differentially private mechanism by adding differential private noise to  $C$ . The noise matrix  $L$  is generated by sampling its elements from a probability distribution. Then, the noisy covariance  $N$  is calculated:  $N = L + C$ .

To get the vectors of the images, we should first calculate characteristics vectors and values called respectively eigenvectors  $u$  and eigenvalues  $\lambda$  using this equation:  $(N - \lambda I)u = 0$ . Eigenvalues can be found by calculating  $\det(N - \lambda I) = 0$  and then the eigenvectors are calculated. All the eigenvectors will be concatenated in one matrix  $U$ , and the PCA matrix  $P$  is computed by multiplying  $S$  by  $U$ . Each column of  $P$  is a PCA vector. If a trusted data owner sends the set of images, then the vectors will be sampled to create a synopsis dataset sent to the cloud.

#### 3.2 Privacy-preserving PCA

To make PCA a privacy-preserving mechanism using differential privacy, we should turn the covariance matrix  $C$

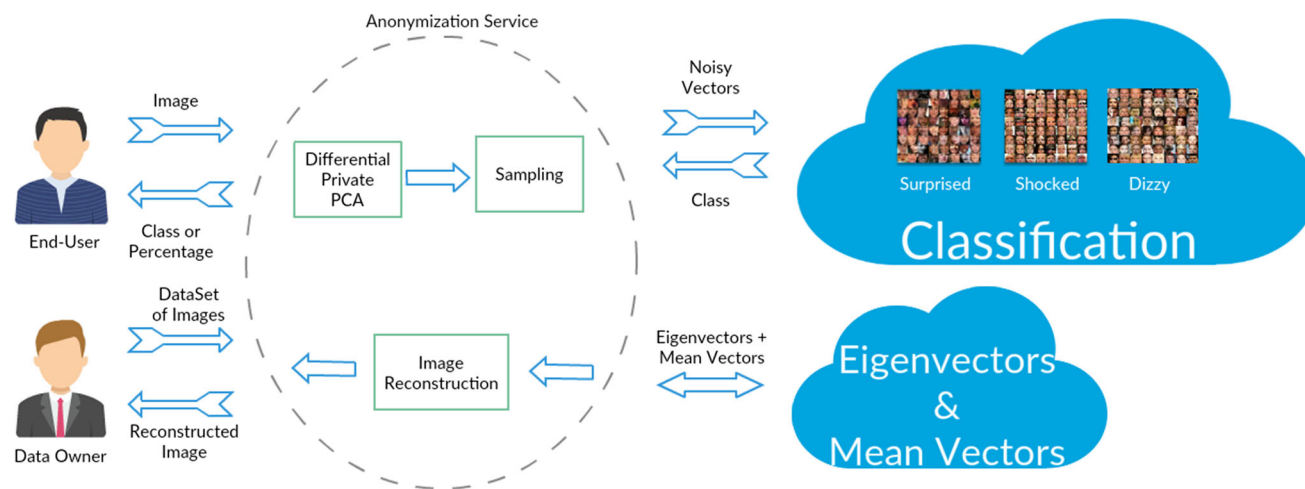


Fig. 1 General architecture of a differentially private image classification system

into a noisy matrix. Therefore, we generate a noise matrix from a probability distribution and add it to the covariance matrix. Practically, using  $C$  in the PCA process returns a memory error due to the enormous dimensions of  $C$ . For this reason,  $C$  is substituted by a matrix  $A = \frac{1}{m}SS^T$  where data matrix  $S \in \mathbb{R}^{n \times m}$ . Then each row of the matrix  $P$  is a PCA vector instead of each column.

3.2.1 Generating matrix  $N$  with Laplacian noise

We choose Laplace distribution [9] to achieve differential privacy. The probability density function of Laplace distribution  $Lap(\mu, b)$  is:

$$P(x | \mu, b) = \frac{1}{2b} e^{-\frac{|x-\mu|}{b}} \tag{3}$$

where  $\mu = 0$ ,  $b = \frac{2d}{n\alpha}$  and  $\alpha$  is a privacy parameter.

For  $S \in \mathbb{R}^{n \times m}$ , we sample  $\frac{d^2+d}{2}$  values from this distribution to fill the upper triangular part of the noise matrix  $L$  then the values of the lower triangle part are copied from opposite position. Finally, we add the noise matrix  $L$  to  $A$ .

3.2.2 Generating matrix  $N$  with Laplacian noise

For a long time, SULQ method [3] was the only differentially-private approximation to PCA. This method based on Gaussian distribution guarantees weaker privacy than the differential privacy known as  $(\alpha, \delta)$ -differential privacy. Chaudhuri et al. in [4] proved that SULQ is not a good candidate for effective dimensionality reduction. They proposed a simple modification and called their method MOD-SULQ. This method, as in SULQ, is based on Gaussian distribution. First, we compute the parameter of the distribution:

$$\beta = \frac{d+1}{n\alpha} \sqrt{2 \log \left( \frac{d^2+d}{\delta 2\sqrt{2\pi}} \right) + \frac{1}{\sqrt{\alpha n}}} \tag{4}$$

where  $\alpha$  is a privacy parameter and  $\delta$  is a relaxation parameter.

We generate after that a symmetric noise matrix based on the probability density function of Gaussian distribution  $N(\mu, \beta^2)$ :

$$P(x | \mu, \beta^2) = \frac{1}{\sqrt{2\beta^2\pi}} e^{-\frac{(x-\mu)^2}{2\beta^2}} \tag{5}$$

where  $\mu = 0$ . Finally the noise matrix is added to  $A$ .

3.3 Sampling to create a synopsis dataset

To perform the sampling, we reconstruct some images from noisy vectors whose classifications were not affected by the added noise.

We compute two distance scores of the noisy image and compare them to two user-defined thresholds. First, we compare each image to its noisy version. Second, we compare the noisy image to the mean of the images with the same classification label.

If distance scores are less than these thresholds we can consider the image as unidentifiable, and at the same time its classification was not affected by the noise, and therefore this noisy vector will be added to our synopsis dataset.

If the data owner is untrusted, then the vectors are classified on the cloud, and the classification result is compared to the label sent by the data owner for each image. If the label of the image and the classification result are not identical, then the vector is dropped. Otherwise, the vector will be a part of the sampling process.

The images sent by data owners are already labeled, therefore the sampled vectors are classified, and they form

the training dataset. If an end-user sends the images, then after applying the private-preserving PCA process on the images, the noisy vectors are directly sent to the cloud to be classified.

### 4 Experiments

Our experiments were applied on a dataset of Japanese face images (JAFFE) [11], containing 213 images of 7 facial expressions.

#### 4.1 Experimental settings

We have applied three classification algorithms to test which one returns better results.

The first algorithm is the Support Vector Machine that has proved its superiority among many other classification methods [5]. SVM is based on transforming the data in input space into data in featured space in a way that renders the classes linearly separable. Then a line or a plane is drawn between the classes, and the classification is based on this line or plane.

The two other classification algorithms K-NN and Kernel Density Estimation are from the same family. We have chosen K-NN and Kernel Density Estimation to check if K-NN is more accurate than Kernel algorithm, in this case, we can declare that we can apply K-NN in our model with no concerns about distance-learning attacks in DO-Q Threat Model case [10].

Kernel calculates the influence of every point in each class on a given query relatively to the distance between this point and the query. The Gaussian Kernel formula to calculate the influence is:

$$K(\|q - x_i\|) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}\|q-x_i\|^2} \tag{6}$$

where  $q$  is the query tuple,  $x_i$  is the  $i$ -th data tuple,  $\sigma$  is a parameter chosen to maximize accuracy [10] and  $\| \cdot \|$  is the  $L2$  norm.

In K-NN, which is one of the most popular classification algorithms, the  $K$  nearest neighbors have the same influence on the query; the others have 0 influence. This shows that choosing the parameter  $K$  is very critical to get the best result from this algorithm [1].

The two most notable observations we have found in the empirical study are: The need of a very small privacy parameter (vast amount of noise) to provide privacy for images. Hence, adding more and more noise is unable to hide face features.

For the first observation, we have studied the values of the elements inside the matrices and the vectors:

For a set of images of dimensions  $n \times m$ , in every image the value  $x$  of the pixel  $\in [0, 255]$ , then data is centered so  $x_c \in [-128, 128]$ .

We compute  $A = X_c \cdot X_c^T$  so  $x_a = x_1^2 + x_2^2 + \dots + x_{(n \times m)}^2$ ; where  $X_c$  is the centered data. Therefore,  $x_a \in [0, (128 \times n \times m)^2]$ .

In our experiments  $n = m = 256$ , so  $x_a \in [0, 7.10^{13}]$ , we have found that to provide enough privacy to the reconstructed images, we should have  $m > [0, 7.10^{13}]$ , where  $n$  is an element of the noise matrix, to get this value for  $m$ , the privacy parameter  $\alpha \leq 10^{-15}$ .

For the second observation, lets consider the function used to rebuild the images:

$$i = \text{vec}.U + \text{mean} \tag{7}$$

where  $\text{vec}$  is the noisy vector of the specified image,  $U$  is the matrix containing the eigenvectors and  $\text{mean}$  is the vector containing the means of the dataset.

Then after reshaping  $i$  as  $n \times m$  matrix, the matrix is normalized:

$$i = \frac{i - \min(i)}{\max(i) - \min(i)} \times 255 \tag{8}$$

We assume that reconstructing images using the eigenvectors, the mean, and then normalizing the matrix makes it impossible for the noise to hide the face or its features. From Fig. 2, we can see that for a  $256 \times 256$  image, the privacy parameter  $\alpha < 10^{-15}$  is not enough to provide privacy where, regardless how small  $\alpha$  is, the features are still identifiable.

#### 4.2 Evaluating privacy versus accuracy

First, we compare the three classification algorithms, Kernel Density Estimation, SVM, and K-NN. The latter,

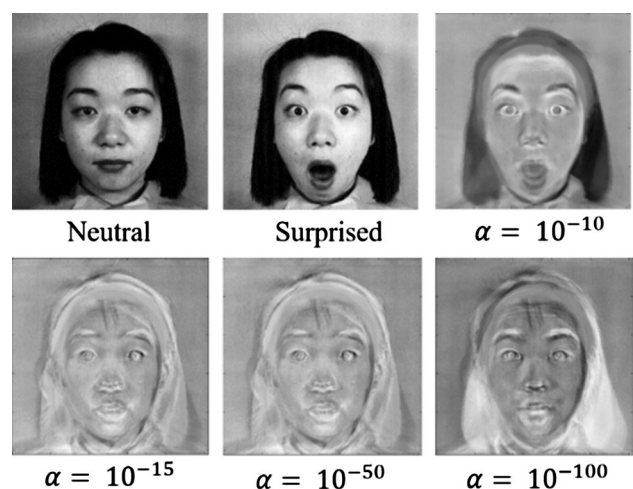


Fig. 2 Image reconstruction based on Laplace noisy vectors

surpasses the two other algorithms by far, as the accuracy of Kernel Density Estimation and SVM did not exceed the 50%. Thus, we have focused on K-NN and compared its performances based on privacy parameter  $\alpha$ , probabilistic distributions (Laplace and Gaussian) and distance functions (Chebyshev, Euclidean, and Manhattan) used in K-NN.

In Fig. 3, we can notice how the accuracy decreases when  $\alpha$  decreases. For  $\alpha = e^{-10}$ , the accuracy ranges between 0.92 and 1, but in the previous section, we have seen how the noisy images with  $\alpha < e^{-15}$  are identifiable.

Chebyshev Distance outperforms the two other distances for Laplacian noisy dataset. For  $\alpha$  between  $e^{-15}$  and  $e^{-20}$ , the accuracy is 0.9 when using Chebyshev which forms a good trade-off between privacy and utility.

For Noisy Gaussian dataset, the three distance functions have close results. For  $\alpha$  between  $e^{-15}$  and  $e^{-15}$ , the accuracy is between 75% and 85%. Hence, for a better trade-off between privacy and utility, the data owner should apply Laplace mechanism with privacy parameter in the range of  $[e^{-15}, e^{-15}]$  using the Chebyshev distance in K-NN classification.

## 5 Related work

The K-Nearest Neighbors (K-NN) [18] is one of the most popular and influential data mining algorithms in the literature. However, many adaptive attacks [10] can form a real threat to data privacy in K-NN based systems. Li et al. propose in [10] a privacy-preserving system based on Kernel density estimation using Gaussian Kernel instead of K-NN.

Their system, as most of the other related works, uses computation over encrypted data. They describe four roles:

- The data owners submit encrypted data to the system.
- The queriers submit encrypted queries to receive classification results.
- The host role, possessed by the cloud, stores the incoming encrypted data and hosts the classification.
- Finally, the Cryptographic Service Provider (CSP) owns both encryption and decryption key.

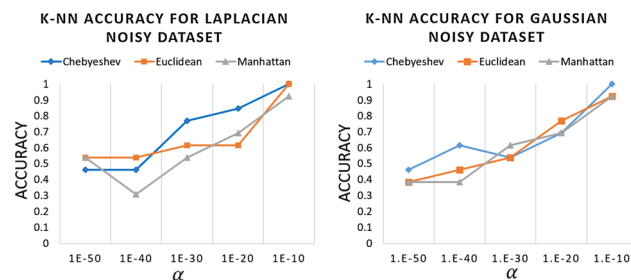


Fig. 3 Accuracy score for K-NN classification

They demonstrate that Distance-Learning attacks under K-NN system can breach data privacy if an untrusted data owner is in the same time a querier and propose to use Kernel density estimation instead of K-NN. In this paper, due to the noisy vectors and the sampling, using K-NN is safe.

The work described in [16] turns K-means clustering algorithm to a differentially private algorithm, where noise is added to the centroids in a way that respects the requirements of Differential Privacy. Differentially private K-Means is divided into two approaches: interactive and non-interactive.

Interactive approach [16] is based on a query that can be used just once, can serve only one querier and only for one task. Any mechanism that is based on this approach returns a noisy result to the user. Each query has a budget  $\alpha = \sum_i \alpha_i$  given by the database owner. Each execution  $i$  makes the budget loses  $\alpha_i$  of its value. When the  $\alpha$  budget is less than  $\alpha_i$ , the query cannot be executed anymore. Hence, this approach has many restrictions on privacy preservation, especially if the budget is small where the number of queries could be insufficient. Besides, the data owner should validate the query. The non-interactive approach algorithms [16] return a noisy synopsis data set. The querier can send queries to this synopsis to get noisy statistical data. This approach has no limits nor restrictions to the number and the sender of the queries.

Several other works [7, 13, 17] rely on encryption for privacy. Taheri et al. in [17] propose a method for face authentication in encrypted domain. In [7], they propose a general framework for multi-biometric template protection based on homomorphic probabilistic encryption. The work in [13] is similar to our proposed approach but relies on partially homomorphic encryption called Pailliers encryption. This type of encryption is a public key scheme; this means that the encryption can be done using a public key while decryption can only be done by a trusted party that possesses the private key. The technique, however, assumes that the cloud is a trusted party, and thus the privacy of the dataset is threatened.

As shown in Table 1, the two other works rely on encryption, and all the works need a trusted data owner. The trusted data owner is always required, or the system model cannot be private. The other trusted party is the cryptographic service for the first work, the cloud for the second (which can cause a severe threat to the dataset) and the anonymization service for our work. These encryption and decryption tend to be time/consuming. Finally, we consider that using noise addition technique like differential privacy, can provide better performance and keep the data suitable for classification.

**Table 1** Comparing related works with our work

Authors	Function	Trusted parties	Time consuming	Queries	Classification
Li et al.	Encryption	CSP + trusted data owner	High	Encrypted	Kernel density estimation
Nassar et al.	Partially homomorphic encryption	Cloud + trusted data owner	High	Non-private	K-NN
Our work	Differential privacy	Anonymization service + trusted data owner	Low	Differentially private	K-NN + kernel density estimation + SVM

The benefits and disadvantages of other techniques and mechanisms should be studied to find out their capabilities in this domain, like pixelization, blurring, and PCA-like mechanisms. Although we think that blurring or pixelization cannot provide the same level of privacy and utility as differential privacy, but this issue needs more study in future work.

## 6 Conclusion

In this paper, we have proposed, implemented and evaluated a private image classification framework based on differentially private Principal Components Analysis. Using this framework, we ensure that the individuals in the image dataset are kept safe on a semi-trusted cloud service by adding differentially private noise to the images PCA vectors. User requests are distorted as well to keep the participating individuals unidentifiable.

We elaborated a set of experiments to evaluate the trade-off between the accuracy and the privacy of the dataset against several classification algorithms, namely K-NN, Kernel Density Estimation, and SVM. K-NN has shown to be very promising. We identified as well that a privacy parameter within the range of  $[e^{-15}, e^{-20}]$  must be used to balance between the privacy and accuracy of K-NN using the Laplacian mechanism.

In the near future, we intend to evaluate the efficiency of our approach in a real application scenario where collaborative attacks in which some of the data owners may be semi-trusted as well.

**Acknowledgements** This work was supported by CNRS-L and Univ. Pau & Pays Adour, E2S-UPPA/LIUPPA.

## References

- Anava, O., & Levy, K. (2016).  $k^*$ -nearest neighbors: From global to local. In *Advances in neural information processing systems* (pp. 4916–4924).
- Ascher, U. M., & Greif, C. (2011). *A first course on numerical methods* (Vol. 7). Philadelphia: SIAM.
- Blum, A., Dwork, C., McSherry, F., & Nissim, K. (2005). Practical privacy: The SULQ framework. In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems* (pp. 128–138). ACM.
- Chaudhuri, K., Sarwate, A., & Sinha, K. (2012). Near-optimal differentially private principal components. In *Advances in neural information processing systems* (pp. 989–997).
- Cheong, S., Sang Hoon, O., & Lee, S.-Y. (2004). Support vector machines with binary tree architecture for multi-class classification. *Neural Information Processing-Letters and Reviews*, 2(3), 47–51.
- Dwork, C. (2006). Differential privacy. In *33rd international colloquium on automata, languages and programming, part II (ICALP 2006). Venice, Italy* (Vol. 4052, pp. 1–12). Springer.
- Gomez-Barrero, M., Maiorana, E., Galbally, J., Campisi, P., & Fierrez, J. (2017). Multi-biometric template protection based on homomorphic encryption. *Pattern Recognition*, 67, 149–163.
- Jain, P. (2012). Security issues and their solution in cloud computing. *International Journal of Computing and Business Research*, 2229–6166. <http://www.researchmanuscripts.com/isociety2012/1.pdf>.
- Jiang, W., Xie, C., & Zhang, Z. (2016). Wishart mechanism for differentially private principal components analysis. In *AAAI* (pp. 1730–1736).
- Li, F., Shin, R., & Paxson, V. (2015). Exploring privacy-preservation in outsourced  $k$ -nearest neighbors with multiple dataowners. In *Proceedings of the 2015 ACM workshop on cloudcomputing security workshop* (pp. 53–64). ACM.
- Lyons, M. J., Kamachi, M. G., & Gyoba, J. (1997). Japanese female facial expressions (JAFFE), database of digital images. <http://www.kasrl.org/jaffe.html>.
- Narayanan, A., & Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *IEEE symposium on security and privacy, 2008. SP 2008* (pp. 111–125). IEEE.
- Nassar, M., Wehbe, N., & Al Bouna, B. (2016). K-NN classification under homomorphic encryption: Application on Alabeled Eigen faces dataset. In *2016 IEEE international conference on computational science and engineering (CSE) and IEEE international conference on embedded and ubiquitous computing (EUC) and 15th international symposium on distributed computing and applications for business engineering (DCABES)* (pp. 546–552). IEEE.
- Powell, V., & Lehe, L. (2015). Principal component analysis @ONLINE.
- Singh, G., Sharma, V., & Singh, U. (2016). Different image encryption techniques-survey and overview. *International Journal of Advanced Research in Computer Science and Software Engineering*, 6(8). <http://www.ijarcsse.com/>.
- Su, D., Cao, J., Li, N., Bertino, E., & Jin, H. (2016). Differentially private  $k$ -means clustering. In *Proceedings of the sixth ACM conference on data and application security and privacy* (pp. 26–37). ACM.

17. Taheri, M., Mozaffari, S., & Keshavarzi, P. (2017). Face authentication in encrypted domain based on correlation filters. *Multimedia Tools and Applications*, 77(13), 17043–17067.
18. Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., et al. (2008). Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1), 1–37.

**Elie Chicha** obtained a Bachelor degree in computer science in 2012 from Lebanese University and an engineering degree in computer science in 2016 from Antonine University. He is a software engineer in Portalys.net since 2015 and a research assistant in TICKET Laboratory since 2016. He is a Ph.D. student at the University of Pau and Pays de l'Adour since 2018. His current research interests include data privacy, graphs and image classification.

**Bechara Al Bouna** is the Director of TICKET Lab., a cross cross-disciplinary research laboratory that develops contextualized research projects in a tight collaboration with the industry. Holds a Ph.D. in Computer Science from University of Burgundy—France and graduated with honor from the Antonine University—Lebanon with a diploma in Telecommunication and Computer Engineering. Has been awarded the Best Professor prize in 2014 and the Best Research Paper in DBSec 2013. Current research interests include data privacy, security, and information management in the digital world.

**Mohamed Nassar** is an assistant professor of computer science at the American University of Beirut (AUB) since 2016. Before joining AUB, he was a Post Doc fellow at the department of computer science and engineering at Qatar University. He received his research master degree (DEA) in computer science in 2005 and the Ph.D. degree in 2009, both from Nancy University (currently Lorraine University), France. He worked as an expert research engineer at INRIA Nancy (2009–2010) and at Ericsson, Ireland (2011). His current research interests are cybersecurity, machine learning and their intersection.

**Richard Chbeir** received his Ph.D. in Computer Science from the University of INSA DE LYON-FRANCE in 2001 and then his Habilitation degree in 2010 from the University of Bourgogne. He is currently a Full Professor in the Computer Science Department in IUT de Bayonne in Anglet France. His current research interests are in the areas of multimedia information retrieval, XML and RSS Similarity, access control models, and digital ecosystems. Richard Chbeir has published in international journals, books, and conferences, and has served on the program committees of several international conferences. He is currently the Chair of the French Chapter ACM SIGAPP. Richard Chbeir teaches several courses in the Computer Science Department of the University of Pau University in Anglet-France.