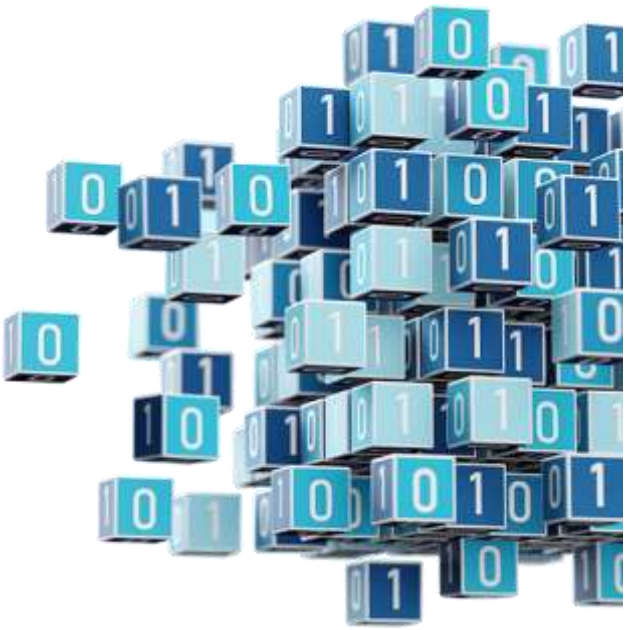
FeatureCloud

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare



Deliverable
D5.1 Entire Federated Clustering pipeline software available for download

---

Work Package
WP5 Unsupervised Federated Machine Learning

Disclaimer

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078. Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

Copyright message

Document information

Grant Agreement Number: 826078			**Acronym: FeatureCloud**	
Full title	Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare			
Topic	Toolkit for assessing and reducing cyber risks in hospitals and care centres to protect privacy/data/infrastructures			
Funding scheme	RIA - Research and Innovation action			
Start Date	1 January 2019	**Duration**	60 months	
Project URL	https://featurecloud.eu/			
EU Project Officer	Christos MARAMIS, Health and Digital Executive Agency (HaDEA) - Established by the European Commission, Unit HaDEA.A.3 – Health Research			
Project Coordinator	Jan BAUMBACH, University of Hamburg (UHAM)			
Deliverable	D5.1 Entire Federated Clustering pipeline software available for download			
Work Package	WP5 Unsupervised Federated Machine Learning			
Date of Delivery	**Contractual**	30/06/21 (M30)	**Actual**	30/06/21 (M30)
Nature	Demonstrator	**Dissemination Level**	Public	
Lead Beneficiary	05 SDU			
Responsible Author(s)	Richard Röttger (SDU)			
	Anne Hartebrodt & Tobias Frisch (SDU)			
Keywords	Clustering; Principal Component Analysis; k-means; Expectation Maximization; Gaussian Mixture Model			

History of changes

Version	Date	Contributions	Contributors (name and institution)
V0.1	01/06/2021	First draft	Anne Hartebrodt, Richard Röttger, Tobias Frisch (SDU)
V0.2	04/06/2021	Comments & Review	Dominik Heider (UMR)
V0.3	24/06/2021	Draft	Anne Hartebrodt, Richard Röttger, Tobias Frisch (SDU)
V1	25/06/2021	Review	Jan Baumbach, Nina Wenke (UHAM)
V1	29/06/2021	Final version	Tobias Frisch (SDU)
V1	30/06/2021	Submission	Miriam Simon (concentris)

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

Table of Content

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

1. Objectives of the deliverable based on the Description of Action (DoA)

The objective was to develop a federated clustering pipeline able to perform pre-processing and clustering of distributed datasets. Instead of aggregating all potentially sensitive data on one site, model training will be performed solely by exchanging model parameters. Unequal distribution and lack of data or clusters on specific sites are a particular problem in biomedical data and the stability of the algorithm in those situations will be analyzed. The cluster pipeline will be completed and validated based on cluster-validity indices able to access cluster results in a federated fashion. The software developed will be integrated into the FeatureCloud platform to ensure usability and also integration for potential integration into supervised downstream analysis.

Task 1: Federated Clustering (SDU, TUM, MUG):
We will implement an entire federated clustering pipeline. In order to perform the pre-processing of heterogeneous distributed datasets, SDU will develop data projection methods with spiked-in artificial data points to ensure the same alignment and normalization of all parts of the distributed datasets without the exchange of the actual data. For the clustering, SDU will develop model-based methods that require only the transfer of the model parameters to construct an overall model capable of clustering the dataset. A particular focus will be on dealing with incomplete or missing data, which is a common phenomenon in biomedical datasets. To complete the clustering pipeline and to fine-tune the clustering, so-called cluster-validity indices are required. Here, SDU will also develop model-based approaches enabling the quality assessment of the cluster results in a federated fashion and guide the user towards potential improvements of the cluster quality. All software developed here must be tightly integrated into the overall platform (WP7, TUM) and might also serve as input for supervised downstream analysis (WP4, MUG). Adequate coordination efforts will be carried out in order to ensure seamless integration.

MS31 Implementation of federated pre-processing methods
MS32 Implementation of federated clustering methods
MS33 Implementation of federated cluster evaluation methods & completion of the clustering pipeline

2. Executive Summary

In recent years, the amount of data delivered by novel biomedical techniques, such as high throughput sequencing (HTS), has led to a constant increase in available data. At the same time, the definition of diseases is shifting towards more mechanistic (Baumbach and Schmidt, 2018) approaches. Unsupervised learning methods have been extensively used in order to reveal previously unknown structures within the data (e.g., groups of similar patients). Here, we present a federated pipeline covering all steps of an unsupervised analysis:

1. Pre-Processing: Principal Component Analysis
2. Clustering: k-means; Expectation Maximization (EM)
3. Cluster-Validity: Mean Square Error (MSE)

For the principal component analysis (PCA), we illustrate its application in genome-wide association studies (GWAS), where it is utilized for patient stratification. Our implementation shares only the ‘feature’ eigenvector (which only contains aggregated statistics), compared to other methods that exchange the full sample eigenvector with one entry per patient. Furthermore, we demonstrate the applicability of horizontally applied PCA on high-dimensional biomedical data. Here, we conclude that more extensive communication (and therefore a more precise algorithm) should be recommended in the medical sector. The second step in the pipeline is based on federated k-means and EM clustering. Here, we investigated a variety of existing k-means methods and their potential advancement in non-iid settings. Finally, we illustrate the advantages of our accurate EM clustering implementation compared to k-means concerning different cluster structures and distances.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

3. Introduction (Challenge)

Clustering is an unsupervised machine learning (ML) approach aiming to find an optimal dataset partition concerning the chosen optimization criterion. A large proportion of existing clustering algorithms are designed to minimize (maximize) the distance (similarity) of the samples grouped while maximizing the distance between groups. Internal or external clustering validation measures can assess the quality of clustering. The former evaluate quality based on the optimization criterion, while external measures require a gold standard. As by definition of unsupervised ML, the labels and thereby the correct grouping is unknown, making internal quality measures particularly interesting but also challenging. Additional challenges for clustering are the curse of dimensionality (d >> n) since, especially for biomedical data, the number of features is always much larger than the number of available samples. In particular, Minkowski distances (such as the Euclidean distance) are rendered useless due to the concentration or compression effect in high dimensions, meaning all distances converge to one fixed distance (Jonathan *et al.*, 1999). Different clustering approaches have been designed that are performing quite differently depending on the cluster form and structure. Although the intuition is clear in a two-dimensional space, the behavior might differ with increasing dimensions. Further challenges arise by the number of clusters k and the parameter choice, depending on the algorithm. For some methods, such as k-means (MacQueen and Others, 1967), the choice of starting points might have a significant influence on the outcome, leading to a variety of different clustering results.

Federated Clustering

We aim to apply and develop algorithms with a specific focus on biomedical datasets, and we are primarily interested in horizontal federated learning (Yang *et al.*, 2019). Hereby, the computing parties share a common set of features while samples are distinct. Sample information is considered private, e.g., the information exchanged between sites may not reveal any information about a specific sample.
The federated approach leads to additional challenges related to privacy: The computation of distance measures such as Euclidean or Manhattan distance for example, is highly challenging as they require a pairwise computation across samples. Other problems are connected to the distribution of samples across sites. In the most extreme case, clusters might be missing in one of the contributors’ datasets due to a lack of samples. Depending on the algorithm, this might lead to distortion when assigning samples to a specific cluster. Widely known issues when applying any machine learning method to biomedical data are batch effects. Non-biological factors, such as laboratory conditions, lead to an overall shift or rotation in the data and have a tremendous influence on the clustering results.

4. Methodology

As emphasized in the introduction, clustering and especially federated clustering comes with a variety of challenges. Additionally, the amount of methods available (distance measures, clustering methods, quality measures) is excessive. In this work, we aim to focus on the effects of the federation on clustering algorithms and aim to highlight federation-related problems and suggested solutions. Therefore, we focus on well-established and understood methods for the pre-processing and clustering in order to ensure accessible and interpretable results:

1. Principal Component Analysis for pre processing
2. k-means Clustering
3. Expectation Maximization Clustering

In the framework of this work package, other federated pre-processing methods like the quantile normalization have been developed, but since they can be computed exactly even when federated,

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

the effects of the federation cannot be investigated. This section will shortly present the methodology applied in the three algorithms that we have implemented.
The pipeline includes a reduction of the feature space based on the PCA, the possibility to remove outliers based on the PCA in a manual or automated fashion, clustering by k-means, which allows for spherical clusters, and a logical initial starting point for the EM clustering that allows for varying cluster shapes following arbitrary Gaussian distributions.

Federated Computing and Distributed Data

In federated learning, data can be distributed horizontally or vertically. Horizontal data distribution refers to the case where every hospital or data center has a subset of individuals but all variables for the patients. Vertical data distribution refers to the case where all patients are available at all hospitals, but only a subset of variables is available at each data center. This could, for instance, be the case if one hospital has a blood chart and another hospital has an MRI scan.

In the following report, we assume the data D to be distributed as distinct subsets D_1, …, D_s, where $D_s = n_s$ x d distributed across s contributing computational parties (e.g., hospitals). Each site s is contributing with n_s samples that are only present at one side, where all share the same set of d features. The algorithms implemented are based on the FeatureCloud platform that is based on a star-like architecture, where every contributor performs local computations on D_s and shares only aggregated parameters with the central server. This server (or aggregator) then computes the global aggregation and shares the results with all contributing parties.

4.1 Principal Component Analysis

A principal component analysis (Pearson, 1901) is aiming to orthogonally transform a set of observations of correlated variables into a set of linearly independent variables, called principal components. The first principal component (PC) captures the axis with the highest variance, and the nth PC is orthogonal to all n-1st. The eigenvalues code for the amount of variability explained by the respective PC.

Mathematically, PCA is the eigendecomposition of the covariance matrix $M = A^T A$ into a set of mutually orthogonal vectors $M = UEU^T$, where $U$is the eigenvector scaled to the unit norm and E is the matrix containing the eigenvalues on the diagonal and A is the dataset. The principal components are the projection of the data onto the leading eigenvectors.$PC = AU$. In the centralized case, this can either be done as described at the expense of calculating the covariance matrix. Alternatively, one can use singular value decomposition via power iteration, which avoids computing the covariance matrix and is as accurate and decomposes the matrix $A = UEV^T$(Halko, Martinsson and Tropp, 2011).

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

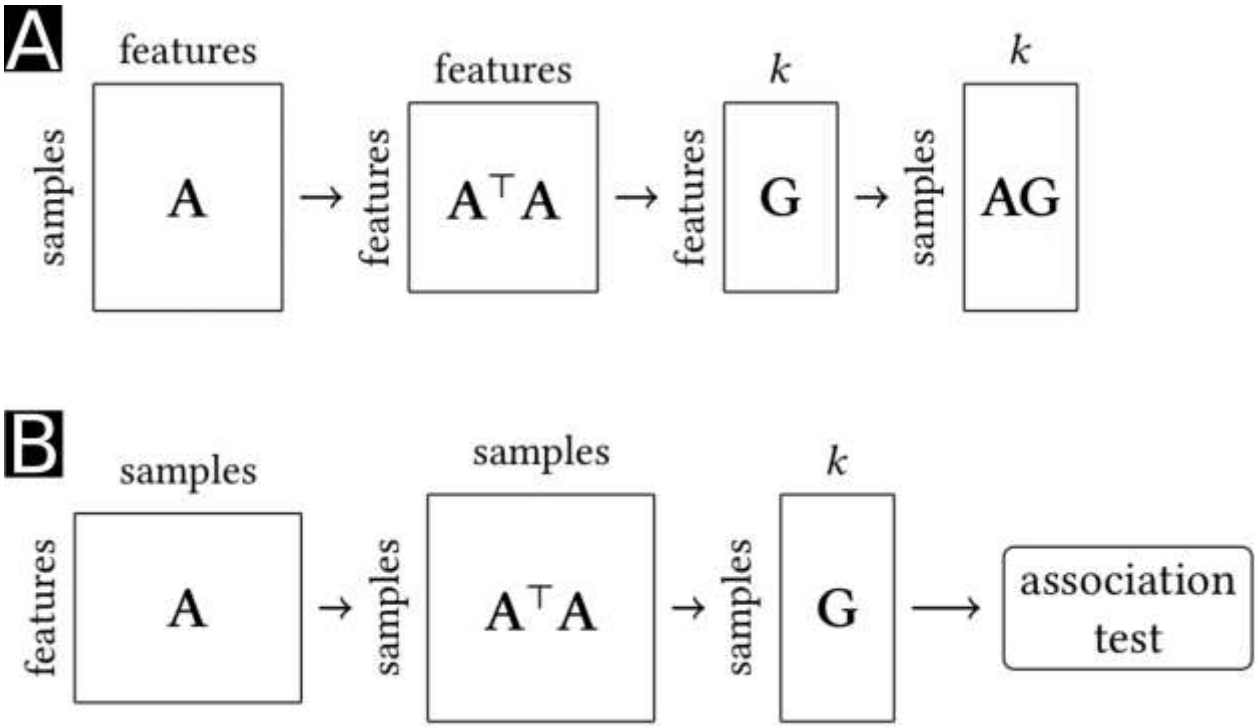


Figure 1: Illustration of horizontal (**A**) and vertical (**B**) PCA

As shown in **Figure 1,** a PCA can be applied to the sample by sample covariance matrix or the feature by feature covariance matrix. In the clinical case, we often assume a horizontal base case where the patients are distributed among the hospitals. However, as we will explain shortly, there are specific cases where there is a need for exact vertically distributed PCA. With the emergence of federated learning for clinical applications, people will likely adopt vertical PCA in practice.

4.1.1 Vertical Principal Component Analysis

A frequent use case for PCA in bioinformatics is population stratification in Genome-wide association studies (GWAS). The eigenvectors are used as covariates in the association test to account for cryptic population substructure and hidden relatedness. Although the distributed dimension is assumed to be the patient/individual dimension in federated GWAS, meaning several data centers have the same d single nucleotide polymorphisms (SNPs) for a different subpopulation of n_s individuals, the principal component analysis in GWAS is performed on the $n \, x \, n$ covariance matrix. This means that in this specific case, the 'features' of the PCA are the samples, but the samples are distributed, meaning that the required PCA algorithm needs to work using distributed features. Specific challenges posed the fact that it is impossible to compute the covariance matrix directly due to the distributed nature of the features (=the patients) and unreasonable because the number of potential participants in a GWAS is increasing. Hence, the computational cost of computing the covariance matrix is increasing as well.

4.1.2 Horizontal Principal Component Analysis

Federated horizontal PCA has been extensively studied, albeit in the 'moderate dimensionality' case, where the number of variables is typically orders of magnitude smaller than the number of samples. Our studies investigated the suitability of horizontal PCA methods to apply high dimensional biological data, where the number of variables is orders of magnitude higher than the number of samples (for instance, gene expression data). Not all available algorithms are suitable for this setting due to the loss of accuracy in these cases. Notably, there are 'one-shot' methods that rely on

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

computing a local PCA, sharing the local PCA/eigenvectors with the aggregator, which computes a consensus solution. This often incurs high approximation error in our evaluated high-dimensional settings. The naive version of sharing the entire covariance matrix is possible without loss of accuracy. However, it requires computing and sharing the entire covariance matrix, which is routinely infeasible in practice. Another way of computing the PCA is by federated power iteration. While the loss of accuracy may be acceptable in many settings, we want to emphasize accuracy as a very important property of PCA in the medical scenario since the loadings of the PCs are often interpreted or even used for the generation of gene panels. Thus, we recommend federated power iteration or naively federated PCA for application in medical pipelines, despite the high transmission cost.

4.2 k-means clustering

The k-means (Lloyd, 1982) clustering algorithm aims to divide the data points into K disjunct clusters by assigning each point to the nearest cluster centroid, thereby minimizing the mean squared error (MSE). The result of the k-means algorithm is a hard partition, meaning every point belongs to exactly one cluster. In the centralized case, k-means consists of the following steps.

1. Initialization of centroids (this may be done via random sampling of the feature space or more sophisticated methods such as furthest-first/k-means++ initialization, which selects points that span the feature space as well as possible).

Then the following steps are performed until convergence:

2. Assignment of all points to their respective nearest centroid.
3. Computation of new centroids (average or weighted average).
4. Convergence check; which may for instance be reached when the centroids do not change in two consecutive iterations)

Federated Algorithms

Several algorithms have been proposed to perform k-means in a federated setting. Here, we provide an overview of the identified main mechanisms. As this is partially preliminary work, not all algorithms have been published in the literature.

1. One shot federated k-means methods (Dennis and Smith, 2020) require only one communication step between the client and the server. In a first step, each client computes a locally optimal clustering and sends the centroids (and number of points in some cases) to the aggregation server. There, the centroids are aggregated (different strategies available) and the aggregated centroids are sent back to the clients. The clients then assign their data points to the global centroids.
2. The process described in 1. can either be terminated after one round, or if desired additional k-means rounds can be executed until convergence is reached. In this case the one-shot method is used as an initialization for the federated k-means iterations.
3. The third possibility (Brandão, Mendes and Vilela, 2021) is to use federated avaging to compute the centroids at each iteration. In this case the clients assign their data points to the initial centroids and send the sum for each centroid to the aggregator which uses the number of points assigned to each global centroid to compute the exact global centroid. This process requires more iterations and an increased communication with the central aggregator.
4. Federated mini-batch k-means. The mini-batch paradigm proposed by (Sculley, 2010) can be extended to the federated case. The centroids are sent to each client sequentially where they are updated. Mini-batch k-means achieves good results in the centralized case but relies on iid sampling of the data and a high number of points.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

Initial centroids
k-means relies on a good initialization, which is a known problem in the centralized case and naturally persists in the federated case. We have investigated the following methods

a) Daisy-chaining is a method based on the furthest first initialization in the centralized case. Initial centroids as successively added to a set until k initial centroids are reached. Every client contributes one initial centroid in the federated case until no more centroids need to be added. The following heuristic is applied. The point that has the highest distance to its nearest centroid is selected. This method has shown promise in practice.
b) Schoolyard selection is a method where each client proposes a set of initial centroids. To match the centroids together, the party with the highest number of points can choose a centroid first and merge it with one of its own centroids. This method has shown unstable behavior in practice.
c) Clustering of the centroids. In this method, every party proposes a set of initial centroids and sends it to the aggregator. The aggregator clusters the centroids and computes new initial centroids.

4.3 Gaussian Mixture Models & Expectation Maximization

A Gaussian Mixture Model (GMM) (Amendola, Faugere and Sturmfels, 2016) is a probabilistic model where we assume that every existing datapoint (sample) stems from one of K different (multivariate) Gaussian distributions. As an extension of the k-means algorithm, the model can incorporate the covariance structure of the data, and it can therefore capture elliptical cluster structures. In contrast to k-means, a GMM will assign each data point to a cluster with a certain probability instead of performing a hard cluster assignment (also called fuzzy-clustering).
As mentioned previously, we assume that our data are distributed across s contributors, where each site has n_s samples and the same feature space of size d. In a multivariate Gaussian distribution $N_d(\mu, \Sigma)$, we have two parameters that need to be estimated for each Gaussian K, namely the d-dimensional mean vector $\mu$and the covariance matrix Σ ($d\ x\ d$).

Expectation maximization (EM) (Dempster, Laird and Rubin, 1977) is an iterative method able to compute the maximum likelihood estimates of the parameters of our Gaussian distribution. It consists of two steps performed iteratively until convergence, namely E-Step and M-Step. In the former, each observation is assigned the probability for each cluster based on the corresponding Gaussian distribution. Samples close to the cluster center will lead to probabilities close to 1, while samples between two clusters will divide their probability accordingly. Afterward, in the M-Step, the means and covariances of every cluster K will be updated according to those probabilities. To perform the first step of the iterative procedure, the parameters for the distribution have to be chosen. The EM algorithm usually delivers a locally optimal result and is consequently sensitive to different starting points. Therefore, in our pipeline, we recommend using k-means clustering to find an appropriate starting point.

4.3.1 Federated Expectation Maximization

Given the initial mean and covariances, we can perform the E-Step and thereby compute γ_s, which is defined as follows:

$$\gamma_s = \frac{\pi_k N(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x_n|\mu_j, \Sigma_j)}$$

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

This results in a $n_s \, x \, k$ matrix computed and stored locally on each contributing site and never exchanged with the central server. The nominator represents the probability that a sample belongs to a cluster K given the parameters for the multivariate Gaussian distribution. The denominator scales the probabilities to be in the interval [0,1].
The local results are consequently utilized to perform the M-Step, which computes the new distribution parameters for each Gaussian K:

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} x_n$$

$$\pi = \frac{\Sigma_{n=1}^{N} \gamma_{nk}}{N}$$

$$\sum_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (x_n - \mu_k)(x_n - \mu_k)^T$$

Generally, in the m-step we compute intermediate local results in each client based on the samples present and then aggregate them in the central server. The m-step computes the mean μ, the weights π, and the covariance matrix Σ for each Gaussian K. For the mean, the client s computes the sum over all samples $n_s$as well as N_k and sends them to the server. Afterward, the result vector from each client can be added elementwise and normalized. The weights and the covariance matrix will be updated in the same manner, and the updated parameters are shared with the clients. E- and M-step are iteratively executed for 200 iterations, and at the end, each client knows the distribution parameters for all K clusters.

4.3.2 Privacy

The data exchanged throughout the iterations are considered private, as they are always an aggregation of the n samples present at client s. This assumption will only hold when there are more than two samples present at each client. Similarly, we ensure more samples than clusters present, and that $\gamma_{nk} > 0$ for at least two samples, as we otherwise exchange raw data for one sample.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

5. Results

5.1 Principal Component Analysis

5.1.1 Vertically partitioned principal component analysis

We developed an efficient version of federated PCA for vertically distributed features based on prior work by (Guo *et al.*, 2012) and show the convergence to the centralized ‘oracle’ solution empirically and theoretically. The proposed algorithm uses federated orthonormalization to avoid sharing the sample eigenvectors to the other parties and is hence more private than previous versions. Furthermore, we showed that in the vertical case, it is possible to compute partial eigenvectors such that every site only gains access to the part of the eigenvector relevant to their data, while maintaining full accuracy. This is relevant because (Nasirigerdeh *et al.*, 2021) showed that the eigenvectors could determine clinical covariates if paired with a downstream test. For more detailed results, the proofs, and the empirical evaluation, we refer the reader to our publication entitled: Federated Principal Component Analysis for Genome-Wide Association Studies. (submitted)

5.1.2 Horizontal PCA

We evaluated the feasibility, challenges, and limitations of federated horizontal PCA in the medical domain based on transcriptome data. The performance of different algorithmic versions has been assessed on non-iid medical data. We were able to show that outlier detection and removal only based on the eigenvectors is possible and necessary to reduce potential privacy risks.
For real-world biomedical data (view samples, many features), only power iteration and the naive aggregation of the covariance matrix achieved acceptable accuracy. We favor a high accuracy over the risk of privacy loss that comes with increased communication between the participants. For further details, we kindly refer to our publication entitled: Federated principal component analysis for high dimensional biomedical data under limited sample availability (submitted) (https://gitlab.com/roettgerlab/federatedPCA).

5.2 k-means clustering

We have results for k-means in a federated setting. Most of our work investigated how to evaluate clustering in a federated case in general, as this is not a trivial and currently researched problem. We suggest the following strategy to evaluate federated clustering to represent many possible use cases: the algorithm must be tested using independently and identically distributed (iid) data and non-iid data.
Non-iid can have several meanings. Here, we consider how the points are distributed among the participants regarding numbers and cluster assignments. Naturally, there can be other biases, such as batch effects in high-throughput biological experiments, but these are hard to represent in a general way. It is helpful to imagine an oracle clustering that has been computed on the hypothetical centralized data as a reference for how the data is distributed. All points belonging to the same centroid are referred to as a cluster. We suggest evaluating the following scenarios:

- iid data: The data is distributed iid over the participants. This means every participant has points belonging to each of the clusters and in approximately equal numbers.
- point-centric non-iid: every participant obtains points for each of the clusters, but not in equal number
- Cluster-centric non-iid: Not every client has points from every cluster. The most severe case is when the clusters are entirely disjoint, meaning every cluster is only available to one client.

It is possible to implement k-means in such a way that it exactly reflects the centralized solution, however, these algorithms come at a very high communication cost. Therefore heuristic approaches need to be evaluated. In the unrealistic case where the number of global clusters and the number of local clusters is available at each client, preliminary research using artificial data shows good

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

performance of the proposed investigated methods. However, with increasing non-iidness of the data, the performance of a few of the methods degrades (See Figure 2 for preliminary results).

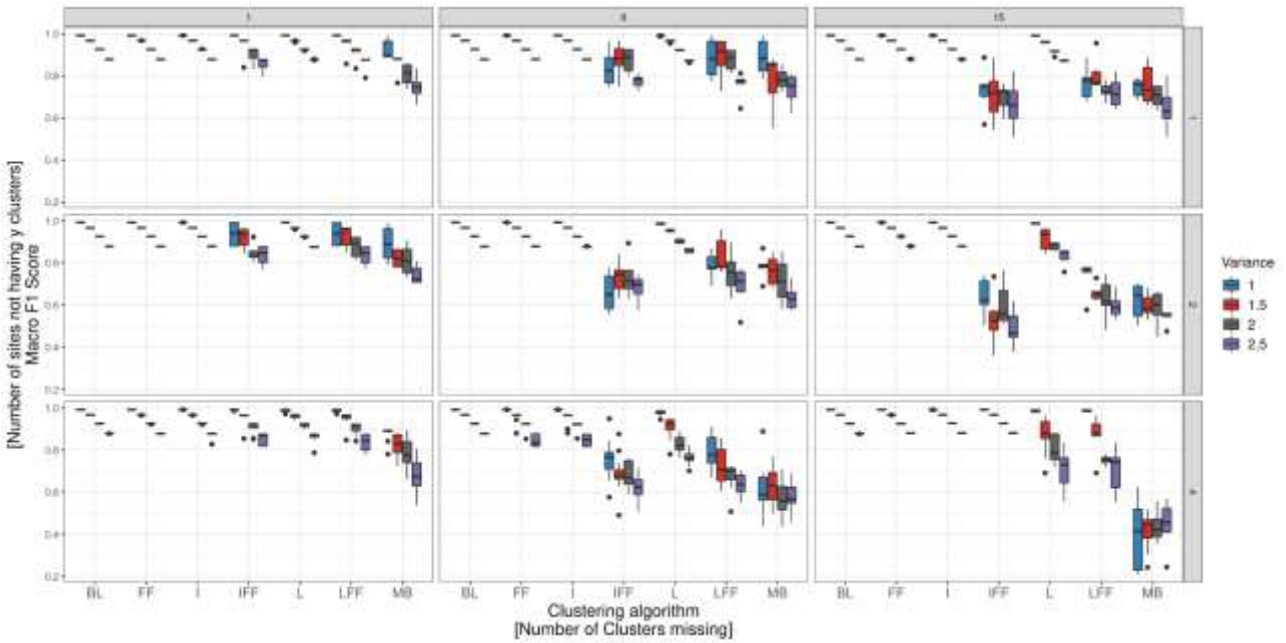


Figure 2: Boxplots show the performance of different federated k-means methods using different configurations for artificial data with 15 clusters. The variance parameter determines the variance of the data around the centroids and hence the cluster separation. The higher the variance the worse the cluster separation. The top left panel has one cluster missing at one site, the bottom right panel has 15 clusters missing at 4 sites each. The performance is measured as F1 Score with respect to the oracle solution. Abbreviations: BL - Baseline (centralized), FF - Fully federated (equivalent to centralized) I - Clustering initialization with centroid clustering + direct labelling (one shot), IFF - Clustering initialization with centroid clustering + Fully federated, LFF - Local-global with furthest first + Fully federated Clustering, IFF - Local-global with furthest first + direct labelling (one shot), MB - Federated Minibatch

5.3 EM-Clustering

In order to test our implementation, we utilized a generated dataset with three clusters in a two-dimensional space. **Figure 3** shows the gold standard of the dataset in the upper-left plot, where all clusters stem from different multivariate Gaussian distributions. In the upper right corner, we see the result of a k-means clustering that cannot identify the correct cluster. e perform both aggregated (bottom left) and federated (bottom-right) EM-clustering based on those initial coordinates. First, we can see the advantage of the EM approach compared to k-means, but we also illustrate no difference between aggregated and federated computation. Our algorithm performs mathematically the same computation and ends up with the same results (apart from minor rounding issues) as the aggregated approach. Therefore, we are independent of sample and cluster distributions across clients, although we are more restricted due to privacy reasons (see section open issues).

A central aspect of federated algorithms is, of course, the amount of data that needs to be exchanged. For the EM-algorithm, the majority of network traffic arises from iterative computations. In each iteration i, the central aggregator receives the mean, weights, and variance for each Gaussian distribution from each client. Those parameters depend only on the number of components K and the number of features d. The amount of data exchanged will grow exponentially with the number of features involved due to the covariance matrix. However, since density-based clustering methods are generally unsuited for high-dimensional datasets, we can assume that the number of features and network traffic will stay within an acceptable range.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

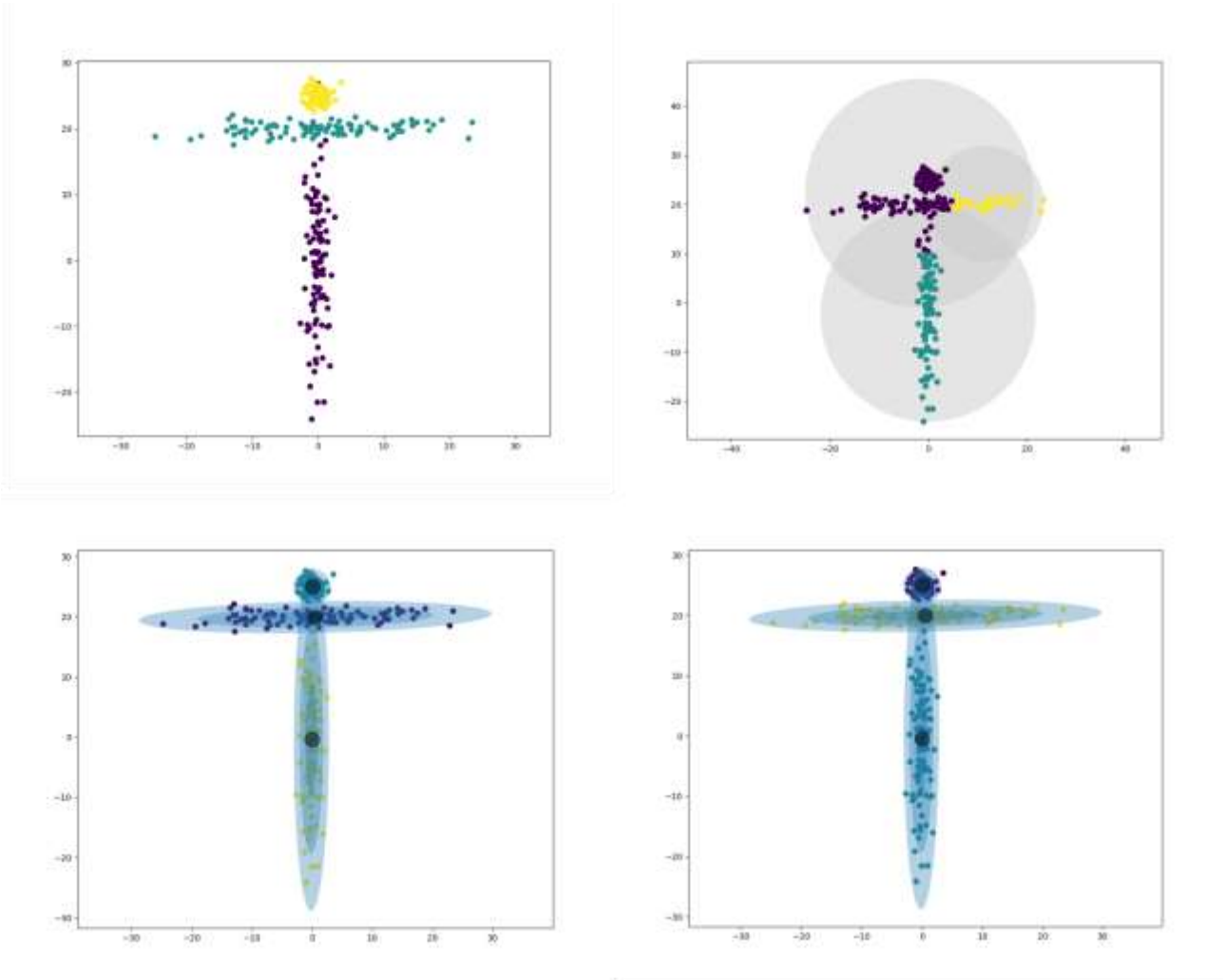


Figure 3: Comparison of EM-Clustering with k-means on an artificial dataset.

6. Open issues

- The choice of an initial centroid has not yet been exhaustively evaluated. We currently investigate the best method to choose initial centroids for Expectation Maximisation as well as k-means.
- An open problem in federated clustering is choosing an appropriate number of clusters (k in k-means, but the problem is relevant for other clustering methods). We intend to propose a federated, efficient version of the gap statistics as this is a frequently used and helpful method to select k.
- Privacy considerations in horizontal PCA: Privacy considerations arise in the use of federated PCA. Notably, the question is how much information the covariance matrix and eigenvectors contain and if sharing them is acceptable. In horizontal PCA, every participant must acquire the full eigenvectors as they are in the intended result. Therefore, the eigenvectors cannot be hidden from the participants. The different ways of computing the eigenvectors might lead to different privacy losses, such as for instance information leakage in iterative PCA or the traceability of exact methods. Therefore, future work will include investigating the privacy of the proposed methods.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

- Privacy consideration in EM algorithm: The amount of information leaked throughout iterative computation, especially for unbalanced distributions. We are planning to investigate this further.
- Communication efficiency: Iterative methods require many communication steps between the aggregation server and the clients. In preliminary tests, the number of communication steps in the PCA was one of the main limiting factors, rather than the volume of transmitted data. Therefore, future work will include making the proposed methods more communication efficient.

7. Conclusion

Unsupervised clustering approaches are indispensable in order to be able to analyze complex biomedical datasets. In this deliverable, we presented a pipeline that covers the entire unsupervised pipeline, from pre-processing over clustering to the evaluation of results. At first, we showed an extensive evaluation of principal component analysis in both horizontal and vertical fashion. PCA is a powerful unsupervised approach allowing for dimensionality reduction and outlier detection. In the second step, we have shown that k-means is a suitable candidate for unsupervised clustering. We are currently still evaluating the variety of existing and potential methods to assess the quality and stability of clustering results, especially for non-IID-distributed data. The majority of existing approaches claim to show superior performance with unequal cluster/sample distribution. However, we showed that testing federated approaches on non-iid-distributed data has not been done properly. Finally, we have developed a federated expectation maximization algorithm for fuzzy clustering of data following a gaussian distribution. The algorithm implemented here is exact, meaning that the computations that we perform are mathematically the same compared to the aggregated analysis.

In the **FeatureCloud** project, we aim to develop algorithms that follow a privacy-by-design approach (Gan *et al.*, 2017). The algorithms presented in this deliverable are based on this approach and exchange only aggregated model parameters or intermediate results that do not allow any conclusions about a single patient. As stated in section 6, we are still critically evaluating the inevitable loss of privacy that comes with exchanging any data. Especially, iterative methods introduce a high level of complexity, and we are therefore currently also investigating the applicability of differential privacy (Dwork, 2006) and secure multi-party computation (Yao, 1986) to increase privacy.

Existing approaches for federated k-means have shown a high inconsistency between claimed performance (often based on artificial datasets) and actual performance on real-world data. In the future, we will therefore focus on extending our pipeline to allow for extensive testing of heterogeneous, noisy, and non-iid distributed biomedical data. Similarly, we want to investigate further the choice of initial parameters (such as centroids), which is already a challenge in aggregated analysis but becomes even more crucial for federated clustering approaches.
In the proposal, the usage of artificial spike-in (SI) points was suggested. The performance of the presented methods was convincing enough and rendered the utilization of spike-in points unnecessary. The usage of these SI points is considered for an implementation of a federated pairwise similarity matrix. Thereby, we will extend our pipeline to cover the wide variety of methods based on pairwise similarity.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

8. References

Amendola, C., Faugere, J.-C. and Sturmfels, B. (2016) ‘Moment varieties of Gaussian mixtures’, *Journal of algebraic statistics*, 7(1). doi: 10.18409/jas.v7i1.42.

Baumbach, J. and Schmidt, H. H. H. W. (2018) ‘The End of Medicine as We Know It: Introduction to the New Journal, Systems Medicine’, *Systems Medicine*, 1(1), pp. 1–2.

Brandão, A., Mendes, R. and Vilela, J. P. (2021) ‘Efficient Privacy Preserving Distributed K-Means for Non-IID Data’, *Advances in Intelligent Data Analysis XIX*, pp. 439–451. doi: 10.1007/978-3-030-74251-5_35.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) ‘Maximum Likelihood from Incomplete Data Via theEMAlgorithm’, *Journal of the Royal Statistical Society: Series B (Methodological)*, pp. 1–22. doi: 10.1111/j.2517-6161.1977.tb01600.x.

Dennis, D. K. and Smith, V. (2020) ‘Heterogeneity for the Win: Communication-Efficient Federated Clustering’. Available at: http://128.1.38.43/wp-content/uploads/2020/12/SpicyFL_2020_paper_35.pdf.

Dwork, C. (2006) ‘Automata, languages and programming’, in *33rd international colloquium, ICALP*.

Gan, W. *et al.* (2017) ‘Data mining in distributed environment: a survey’, *Wiley interdisciplinary reviews. Data mining and knowledge discovery*, 7(6), p. e1216.

Guo, Y.-F. *et al.* (2012) ‘A covariance-free iterative algorithm for distributed principal component analysis on vertically partitioned data’, *Pattern recognition*, 45(3), pp. 1211–1219.

Halko, N., Martinsson, P. G. and Tropp, J. A. (2011) ‘Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions’, *SIAM Review*, pp. 217–288. doi: 10.1137/090771806.

Jonathan, K. B. *et al.* (1999) ‘When Is“ Nearest Neighbor” Meaningful?’, in *In Int. Conf. on Database Theory*. Citeseer. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.7.236.

Lloyd, S. (1982) ‘Least squares quantization in PCM’, *IEEE transactions on information theory / Professional Technical Group on Information Theory*, 28(2), pp. 129–137.

MacQueen, J. and Others (1967) ‘Some methods for classification and analysis of multivariate observations’, in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Oakland, CA, USA, pp. 281–297.

Nasirigerdeh, R. *et al.* (2021) ‘On the Privacy of Federated Pipelines’, *ACM*, p. 5.

Pearson, K. (1901) ‘LIII. On lines and planes of closest fit to systems of points in space’, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11), pp. 559–572.

Sculley, D. (2010) ‘Web-scale k-means clustering’, in *Proceedings of the 19th international conference on World wide web*. New York, NY, USA: Association for Computing Machinery (WWW ’10), pp. 1177–1178.

Yang, Q. *et al.* (2019) ‘Federated Machine Learning: Concept and Applications’, *ACM Trans. Intell.*

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

Syst. Technol., 10(2), pp. 1–19.

Yao, A. C.-C. (1986) ‘How to generate and exchange secrets’, in *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pp. 162–167.

9. Table of acronyms and definitions

concentris	concentris research management GmbH
EM	Expectation Maximization
EM	Expectation Maximization
FC	FeatureCloud controller
GMM	Gaussian Mixture Model
GND	Gnome Design SRL
GWAS	Genome-Wide Association Study
HTS	High throughput sequencing
iid	Independent and identically distributed
ML	Machine Learning
MS	Milestone
MSE	Mean Squared Error
MUG	Medizinische Universitaet Graz
Patients	In this deliverable, we use the term “patients” for all research subjects. In FeatureCloud, we will focus on patients, as this is already the most vulnerable case scenario and this is where most primary data is available to us. Admittedly, some research subjects participate in clinical trials but not as patients but as healthy individuals, usually on a voluntary basis and are therefore not dependent on the physicians who care for them. Thus to increase readability, we simply refer to them as “patients”.
PC	Principal Component
PCA	Principal Component Analysis
RI	Research Institute AG & Co. KG
SBA	SBA Research Gemeinnützige GmbH
SDU	Syddansk Universitet
SI	spike-in
SNPs	Single Nucleotide Polymorphisms
TUM	Technische Universitaet Muenchen
UM	Universiteit Maastricht
UMR	Philipps Universitaet Marburg
WP	Work package

10. Other supporting documents / figures / tables (if applicable)

- Federated principal component analysis for high dimensional biomedical data under limited sample availability, Anne Hartebrodt & Richard Röttger, (Submitted)
- Federated Principal Component Analysis for Genome-Wide Association Studies, Anne Hartebrodt, Reza Nasirigerdeh, David B. Blumenthal, Richard Röttger, (Submitted)

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.