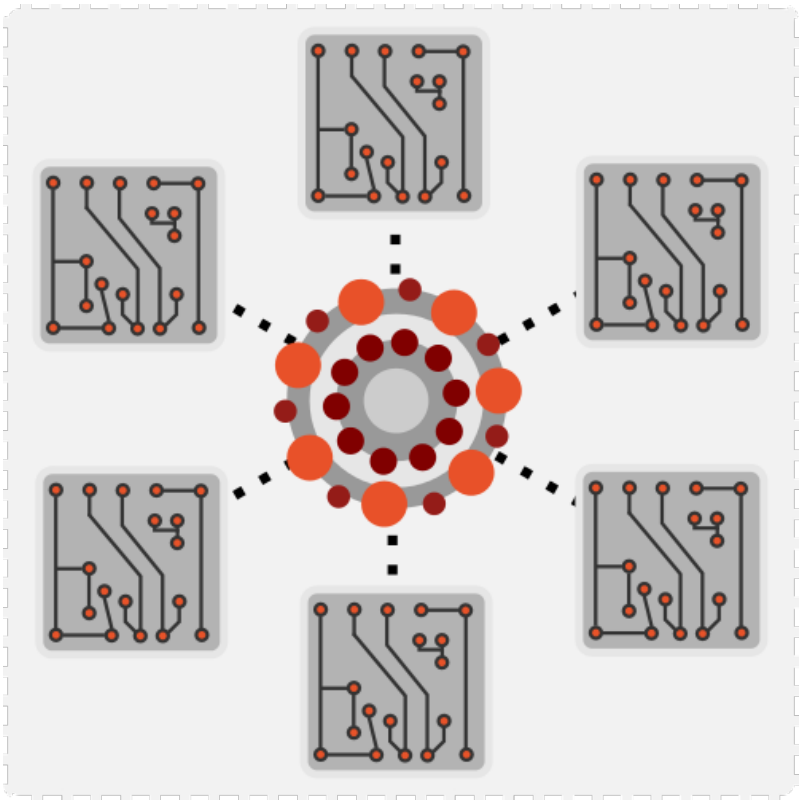
OPEN RISK WHITE PAPER

Federated Credit Systems

Part II: Techniques for Federated Data Analysis

Author: Philippos Papadopoulos

March 20, 2023



www.openriskmanagement.com

The open future of risk management

Abstract

In this Open Risk White Paper, the second of series focusing on Federated Credit Systems, we explore techniques for *federated credit data analysis*. Building on the first paper where we outlined the overall architecture, essential actors and information flows underlying various business models of credit provision, in this step we focus on the enabling arrangements and techniques for building *Federated Credit Data Systems* and enabling *Federated Analysis*. We start with a brief and non-technical description on privacy-preserving technologies, focusing on the special role of federated analysis within the spectrum of cryptographic approaches to multi-party computation. We then discuss generative processes of credit data that both motivate federated analysis uses cases and shape its specific characteristics in the context of the financial sector. We proceed to define the concept of a federated credit data system, with the federated master data table as an iconic outcome. Building on that layout we sketch how generic algorithms might be structured in a federated analysis context, giving examples from concentration risk analysis. We conclude with thoughts on the potential challenges to realize and benefit from federated systems in finance.

Further Resources

- The Open Risk Manual is an open online knowledge base covering diverse domains of risk management. Concepts mentioned in this White Paper may be further explained / documented using Open Risk Manual entries (suitably hyperlinked).
- The Open Risk Academy offers a range of online courses around risk and portfolio management and sustainable finance, which utilize the latest in interactive eLearning tools.
  Please inquire at **info@openriskmanagement.com** about eLearning possibilities.
- The Open Source Risk Repository is our online repository of libraries, tools and frameworks that support quantitative analysis of diverse risk and portfolio management tasks. In particular, openLGD is a Python powered library for the standalone or federated estimation of Loss Given Default models and openRiskScore is a python framework for risk scoring in both classic and federated/decentralized contexts.

About Open Risk

Open Risk is an independent provider of training and risk analysis tools to the broader financial services community. Our mission is captured by the motto: *The open future of risk management*.
Learn more about our mission at: `www.openriskmanagement.com`.

Introduction

The general definition of *federation* is a composite organizational (economic or more broadly societal) entity, formed by the voluntary union of smaller entities. It is a *cooperative scheme* where federated members maintain control over some parts of their operations and elect to share some other elements. What is shared (federated) may be any type of resource, physical or intangible. There is no strict definition of what fraction of elements must be shared: federation is in this sense a flexible pattern that can materialize in various ways.

In a more narrow information technology context, federation refers to protocols used by different systems adhering to a common standard of operations in order to facilitate digital communication. Our scope here is further limited to the federation (sharing) of enterprise data, and even more specifically credit data owned by independent entities acting as financial intermediaries.

Federation of data resources has received recently increased attention in domains such as the medical sector, in official statistics and in mass computing devices. A federated data architecture is one that allows some degree of interoperability and information sharing between otherwise autonomous businesses or organizations. Federation implies some shared technology architecture, including operational components touching security, auditing, authentication and access rights, among others [1].

An operational example of a federated data system used for genomics and health research is CanDIG [2], deployed to analyze health data across Canada. A separate project, InterConnect seeks to optimize the use of data to enable new research into the causes of diabetes and obesity. Initially supported by a European Union FP7 grant, InterConnect takes analysis to the data (federated analysis) to reuse existing medical data. In a different domain, the office of European Statistics is considering the use of federated data systems for its work on **Trusted Smart Statistics**. The Bucharest Memorandum on Official Statistics in a Datafied Society formally recognized the importance of privacy-by-design approaches and encourages the exploration of privacy-preserving computation technologies, such as secure multiparty computation. Finally, in yet another domain, a massively distributed, on-device algorithmic estimation framework called **federated learning** was used for the purpose of next-word prediction in a virtual keyboard for smartphones[3].

The above examples demonstrate that federated data systems can add substantial value and are already practically implementable in a variety of current contexts. Yet the type of organizations and entities involved and the commercial motives and privacy implications play important roles. This is why understanding and shaping federated information system design to cater for the unique features and constraints of the banking business will help accelerate the adoption and realization of any associated benefits.

While modern credit risk assessment and portfolio management uses fairly advanced analytical tools, historically banking intermediaries have only exchanged information relevant for such purposes through strictly delineated channels (e.g. via credit rating or credit scoring intermediaries). In order to evaluate how this landscape might evolve in a digitally enabled financial systems landscape, in the first paper [4] we introduced a set of conceptual business sub-units involve in credit information gathering and analysis and their associated functions (client relationship management, portfolio management and risk analysis). We discussed their implicit or explicit underlying business models. In particular, we discussed the how-and-why of credit data collection and data exchanges they perform and key risk management challenges each one of these business lines faces.

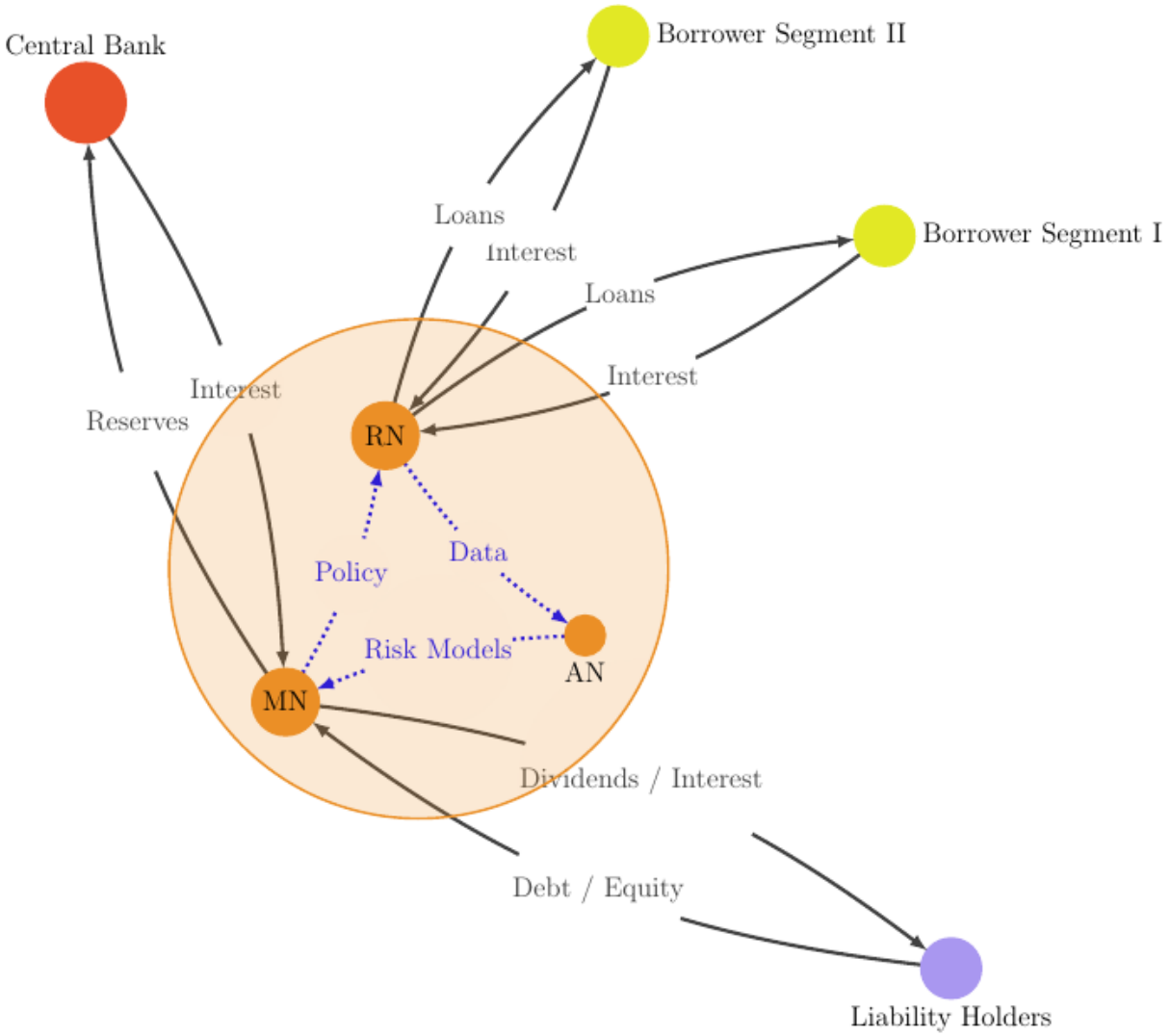


Figure 1: A diagram illustrating information flows and relationships in the standard credit provision process as discussed in [4]. A borrower B_i engages with the credit relationship node RN, providing data points D_i^t at time t. Those are being transmitted to the credit analysis node AN, which uses them (potentially along with other data / model resources) to develop a model M_g, modeled risk assessments $\{S_i\}$ (e.g. a credit score, credit rating or associated metrics and *explanations* about the assessment. Such metrics are also continuously communicated to the credit management node MN that maintains the overall portfolio view. Potentially a sub-set E_i of explanations must also be provide to the client or other stakeholders. At a future time $t+1$ additional information is generated under the relationship (updated data set D_i^{t+1}) which in turn leads to updated metrics etc. At some future point the risk model might be revised (M_{g+1}) using additional data or other approaches. The central contribution of this White Paper is to reflect on how independent entities active in risk assessment of credit portfolios might federate data analysis to create improved families of insights, metrics and models M_g.

In this second white paper of the series, we aim to illustrate in some detail how in the context of credit systems federation enables various forms of coo-petition, where overall better outcomes may be achieved without restricting the ability of individual entities to act autonomously now violating privacy constraints. The federation of *credit risk analysis* is our broad target use case. We focus on developing a language and outlining the information landscape that would enable privacy-preserving federation of credit data. We sketch indicative use cases as examples from a potentially much wider set of possibilities. In the third part of the series we will discuss in more detail algorithmic (modeling) layers that can be applied on top of such information sharing.

Privacy-Preserving Technologies

The concept of *federated credit provision* has some analogies with the use cases we saw above in the medical and official statistics domains but is distinguished by the *commercially competing* nature of financial intermediaries. We would define it more precisely as follows:

Definition: Federated Credit Provision

Federated Credit Provision is a organizational arrangement supporting credit provision where multiple, independently operating lenders acting in *coopetition* (cooperative competition) obtain and share informational gains without violating their own commercial objectives or societal privacy expectations or associated regulations. It is based on a set of principles, tools, systems (Federated Credit Data System) and related organizational arrangements, protocols and practices (Federated Analysis) adopted jointly by distinct participating entities that enables mutually beneficial outcomes from improved financial risk management.

A central feature of such an arrangement is the *controlled exchange of information*. We may think of the space of possibilities opening up with the exchange of digital information as a linear privacy-versus-transparency slider. On one end of the slider's range is a *zero privacy* configuration - all credit data information is shared between the federating entities. This is not a realistic scenario. On the other end of the slider is a *full privacy* configuration - no information is shared between entities. While this is closer to current practices, it is also not exactly an expression of the status-quo.[1] The general assumption and motivation behind adopting federated designs is that there are many use cases where full privacy is sub-optimal and some degree of suitably sanitized data exchange can be ensured *and* is beneficial.

When discussing information exchange protocols between distinct entities, two principal classes are generally considered in the literature: Participants are denoted as *semi-honest* (also passive or *honest-but-curious*) when they follow prescribed protocols without cheating, but are assumed (in-principle) to try to extract as much information as possible about other participant's data. A stronger assumption is to assume that (some) participants are *malicious* and will actively cheat while participating in federation protocols. Our working assumption here is that financial intermediaries are regulated entities wishing to maintain an ongoing license to operate. Hence the honest-but-curious framework is most appropriate. The impact of potential internal or external malicious actors on a federated system must be contemplated, though, as concrete form of operational risk.

[1] As already mentioned, the indirect exchange of credit relevant information via intermediaries is a longstanding feature of banking systems hence the associated benefits are tangible

A large number of digital technologies (client/server architectures, internet protocols, online databases, API's, programming environments and basic cryptographic concepts) are prerequisites for contemplating the types of privacy-preserving architectures we will discuss here. There is also a more specific menu of relevant privacy-enhancing technologies that are built on-top of these more general stacks and which concereteley enable federated data systems. A good recent overview of such technologies (with further references) is given in [5]. Our objective below is to provide a glimpse of the available toolkit from which higher-level architectures can be composed.

A most critical aspect in the design of any privacy-preserving federation scheme is the presence, responsibilities and capabilities of any *trusted third parties* (TTP). A trusted third party in our context is an organization, person, or computing device that federating data owners trust to receive-from and/or send-to a subset of credit data and/or code and to perform related calculations. In modern digital networks TTP's are already ubiquitous in commercial transactions and online communications. For example, a certificate authority (CA) issues digital certificates that provide assurance that the public key certificate sent by a web server to user clients (web browsers) to enable encryption of data exchanges between them does in fact belong to the entity operating the web server.

TTP's may be independent entities (not controlled by the federating entities) or operating with delegated authority from the federation members. The scope of TTP involvement and operations depends on the desired privacy regime. A large body of theoretical research around private decentralized (multi-party) computation posits the complete *absence* of a trusted third party. *Secure Multi-Party Computation* defines an information exchange protocol as secure when participants achieve the objective of computing a common function over their private inputs so that its execution does not reveal more information than the output of the protocol itself. The benchmark (ideal world) against which an MPC protocol is evaluated is provided, indeed, by a setup where the calculation would be performed by a hypothetical TTP that can *confidentially aggregate* all private data inputs and subsequently disseminate outputs.

While we will discuss a number of options in the sequel, *Federated Analysis* is the primary privacy-preserving technique that underpins a Federated Credit System as proposed here. It is an architecture that introduces and requires that certain coordinating nodes act as **partially trusted third parties** (PTTP). Partially trusted third parties cannot take possession of any primary private data but *are* trusted to operate auxiliary calculation services and communicate their results. This arrangement, when available, significantly facilitates privacy-preserving operations. These PTTP's can be thought as economically or organizationally independent entities (or acting with delegated authority) but in no circumstances do they have direct access to private data of the participants besides what is explicitly transmitted via the calculation protocols. The central concept is that the sensitive data themselves never leave their trusted enclave but *code is brought to the data* and results derived from applying code to local data are transmitted to the PTTP. Federated analysis requires that information exchange from data processing only happens through the execution of code that is shared and applied locally. Not all analysis and algorithms can be decomposed to operate in this way. This has given rise to rich domain of privacy-preserving data mining algorithms[6] that is still a very active research domain.

Note that while this set of approaches has developed as a sub-domain of cryptography the primary privacy enhancement derives from the fact that exchanged information only involves *derived data*. The aim of federated analysis is thus to create and share valuable *knowledge* without the exchange of actual primary credit data. As a general point, the extraction of such an information dividend is achieved by performing local computations according to globally shared procedures and algorithms. Whether such

derived data can be *reversed engineered* and leak private information is an important consideration.

It is maybe worth mentioning that Federated Analysis is a specific form of distributed analysis and has in principle utility beyond privacy-preserving computations (e.g., might be also relevant also in sharing computational resources, achieving fault tolerance etc.) While we will discuss it in considerable more detail it is useful to briefly cover also the overall toolkit around privacy-preserving computations.

Anonymization is probably the simplest and least burdensome approach to preserve a minimum degree of privacy. It consists of *replacing* actual identification data with anonymized data, effectively data values that cannot be *linked back* to any actual entity. E.g. replacing customer identification fields with random strings. Sharing anonymized data, as a methodology is closer to zero privacy than full privacy as in effect all data that are not identification data are shared. In a federated credit system context even if this satisfies personal privacy requirements, it may violate commercial secrecy requirement. In addition, anonymization is in-principle subject to *linking attacks*, namely identifying an entity by using a combination of associated attributes that generate a unique key. For example, a competitor entity may be interested to find the largest or more profitable clients of other participants. Using a combination of attributes that are unlikely to repeat verbatim they generate an effective key against which they can match entities. As a privacy-preserving defense technique, a k-anonymization procedure [7] generates a k-anonymized dataset that has the property that each record in the dataset is indistinguishable from at least k-1 other records within the dataset. Hence no specific entity within a k-anonymized dataset can be identified with probability better than $1/k$.

Homomorphic Encryption performs calculations on encrypted data so that they can be analyzed without knowing the underlying values. It is thus not necessary to decrypt the data at all before working with them. Homomorphic encryption is a public key system, where any party can encrypt its data with a known public key and perform calculations with data encrypted by others with the same public key. When arbitrarily complicated functions of the data can be computed this way it is termed Fully Homomorphic Encryption [8] though currently this capability is only available at great computational cost. Fully Homomorphic Encryption is still an emerging cryptographic technique [9] and depending on the use case, a reduced operation set may be adequate and far more performant. See for example the case of Paillier Partially Homomorphic Encryption. The homomorphic properties of the paillier crypto system are [10] that encrypted numbers can be multiplied by a non encrypted scalar, can be added together and can be added to non encrypted scalars.

Differential Privacy is another technique widely used due to its strong information theoretic guarantees, algorithmic simplicity, and low computational overhead. The approach relies, broadly speaking, on perturbing (modifying) actual data values with random noise. A randomized response mechanism (first suggested as an idea by Stanley L. Warner in 1965) is called differentially private if the change of one input data element will not result in significant difference in the output distribution ([11],[12]). ϵ-differential privacy is a mathematical definition for the privacy loss associated with any data release drawn from a statistical database. It measures, e.g., to what extent the parameters or predictions of a model reveal information about any individual point in the training data set. A practical implementation example is offered by the Randomized Aggregatable Privacy-Preserving Ordinal Response methodology (RAPPOR)[13]. It applies a *randomized response* approach that permits statistics to be collected from the population with strong privacy guarantees for each entity, and without linkability of their data sets. *Local Differential Privacy* is model of differential privacy with the added requirement that even if an adversary has access to an entity's data in a database, that adversary will still be unable to learn too much about the entity's

personal data.

Combinations of the above methodologies are possible and actively researched. One can for example anonymize data and incorporate differential privacy techniques into a multi-party protocol. In [14] they consider how an *untrusted* aggregator can derive statistics over multiple participants' data, without compromising privacy. They propose a construction that allows a group of participants to periodically *upload* encrypted values to a data aggregator, such that the aggregator is able to compute the *sum of all participants' values* in every time period, but is unable to learn anything else. They achieve strong privacy guarantees using two main techniques: First, they show how to utilize applied cryptographic techniques to allow the aggregator to decrypt the sum from multiple ciphertexts encrypted under different user keys. Second, they describe a distributed data randomization procedure that guarantees the differential privacy of the outcome statistic, even when a subset of participants might be compromised. In [15] the authors examine discrete distribution estimation under local privacy, a setting wherein a central service can estimate the distribution of a categorical statistic of interest without collecting the underlying data. In [16] they study communication efficient algorithms for distributed mean estimation. Without making any probabilistic assumptions on the data. In [17] they consider the problem of learning high-dimensional, non-parametric and structured (e.g. Gaussian) distributions in distributed networks, where each node in the network observes an independent sample from the underlying distribution and can use k bits to communicate its sample to a central processor.

A common pattern in the employment of privacy-preserving technologies and a key motivation for much current research and development are their *inherent tradeoffs*: preserving privacy may come at the expense of i) analytic accuracy, ii) computational performance hurdles and iii) more complex and costly architecture. In [18] they study the privacy versus accuracy dilemma as a resource allocation problem and propose an economic solution: to operate where the marginal cost of increasing privacy equals the marginal benefit. Optimal choice weighs the demand for (in their context) accurate statistics against the demand for privacy. A key takeway for us here is that privacy-preserving technologies are both already happening but are not trivial to implement and there is an important cost-benefit analysis that must be performed for any specific use case. In summary:

Definition: Federated Credit Analysis

Federated Credit Analysis is a privacy-preserving architecture to conduct quantitative analyses in a federated data context where actual private credit data are never shared between federating entities. Any partial metrics that are shared satisfy differential privacy. It is applicable when federating entities posses private data elements that can produce valuable shared knowledge in excess of the cost of preserving the privacy of these data elements.

Generative Processes of Credit Data

In order to discuss possibilities for federated credit data analysis we need first to discuss the nature of credit data. We move on to describe credit data spaces more precisely while keeping with the objective of general applicability. We discuss in particular the *generative credit data process* of each individual financial intermediary participating in the federation and required steps in pre-federation activities. Ultimately,

federated credit data are simply concrete subsets of the total volume of credit data produced in current business processes operated by the credit industry.

Definition of Credit (Portfolio) Data

What do we mean by credit data? The *appearance* of credit data might quite familiar to banking industry practitioners: A spreadsheet or a table in a database with a number of columns and rows with information about borrowers, loans, collateral and other essential elements of the lending process. Digging into the meaning of these data collections and how they are generated, the logic that binds them together, is essential for understanding what they can be used for and what limitations and issues they may be affected by. In turn this shapes what subsets would be useful for what type of federated analysis.

Credit data are generated in and around the credit provision processes, a complex business with detailed procedures, regulatory rules, and conventions which determine important data flows and assessments, all within fairly rigid legal and accounting frameworks. To start with, credit data nearly always concern credit portfolios. *Credit portfolio data* is any collection of data representing credit contracts (Loans or other Credit Products or Exposures) that is formed as part of financial intermediation activities (e.g., regular lending products offered by banks) but could also be the result of e.g., company credit extended in the course of selling on credit. Credit portfolios of various sizes and shapes are ubiquitous in modern economies. For our purposes Credit Portfolio Data is any well-defined dataset that has direct applications in the assessment of the Credit Risk of an individual or an organization, or, more generally, a dataset that allows the application of data driven Credit Portfolio Management policies. Credit data are in particular the information assets that are required for credit risk analysis (for example estimating a credit risk model) to help assess the future credit states of a borrower or a collection of borrowers (credit portfolio). The range and nature of such credit data forms the basis for development and operation of many analytical tools and scorecards.

A general mathematical framework that is conceptually fertile ground for modeling credit data is the domain of *property graphs*. Previous white papers provide much more detail and applications around that topic ([19],[20]). In summary, the mathematical abstractions that are useful in general to describe credit data generative processes include concepts and representations such as:

- The data model representations of various entities: debtors, lenders and other legal entities involved in a material way in credit relations. Mathematically these are **graph nodes** (entities) in a network graph structure.
- The representation of contractual cash flows (and contracts and value exchanges more generally). These can be thought of as **edges** connecting nodes in the credit graph.
- Sources of uncertainty that can be modelled as **dynamic states** of entities that might be either internal to the entities (idiosyncratic) or external (macroeconomic) and linked to global economic system properties.

The last element (uncertainty) is of crucial importance and a key reason credit data are collected and processed in the first place: Controlling (managing) sources of uncertainty and the risks and opportunities they generate is core to the credit business. Tools used for that purpose are hypothetical scenarios, namely sets of potential realizations of sources of uncertainty and associated scenario-based calculations

of credit states and, thus, associated asset and liability cash-flows. The core value being delivered (directly or indirectly) through credit analysis is reducing the information asymmetry between borrowers and lenders, both on an individual and portfolio basis. Credit analysis employs specialized personnel with legal, economic and statistical know-how and associated digital infrastructure as the primary resources for delivering the value proposition of effective credit risk assessment. An important, intuitively obvious but happily also empirically mostly confirmed fact is that using quantitative indicators of a borrowing entity's economic state and function is, in general, correlated with its credit performance (though the correlation is far from perfect). Hence, in principle, credit analysis lowers the cost of providing credit and reduces the amount of reserves and financial buffers required to mitigate against non-performing credit. Nevertheless, the limitations (occasionally very severe) of pure quantitatively based credit risk management is an important feature which must be reflected deeply also in federation infrastructure.

From Network Graphs to Master Data Tables

Creating a faithful mathematical representation of credit data is not an easy task. The implicit context and business logic underlying credit operations (and hence any produced credit data sets) is seldom explicit in data or metadata and is only revealed through concrete usage and practices. When one works with credit data e.g. performing data transformations, creating analytical reports or utilizing credit data to build, e.g., a Credit Risk Model one usually injects critical additional assumptions and interpretations that may vary significantly depending on the use case. Nevertheless, we will try to lay an overall canvas and sketch some important principles. The desired result is a small collection of mathematical notation choices that help capture the essence of the credit phenomena one wants to study - as far as those can be captured in concrete data!

Diverse organizations may operate credit granting activities in a given economy. The overall generative system might be loosely termed a *credit market*. It is an organized ecosystem or network where economic entities interact with other entities via credit relations, exchange valuable artifacts over a period of time etc. A key building block of such credit network graphs is the set of economic entities or nodes V. These nodes are labeled: we can distinguish entities by their identity. The essential function of entity nodes as data objects is to hold information that characterize them. For example each node is associated with an attribute vector $[a_1, a_2, \ldots, a_n]$. The elements of that vector need not be of homogeneous type or even numerical in nature. The second major building block of a graph are its edges. Mathematically $E \subseteq V \times V$ is a set of edges connecting nodes. Each edge is associated with an attribute vector $[b_1, b_2, ..., b_m]$. Any concept or phenomenon that appears multiple times in a credit system can be modeled. A natural choice is that nodes represent the uniquely identifiable borrowing and lending entities that participate in the system but physical assets are also good node candidates. Bilateral loan contracts are candidates to be represented as graph edges given that they link borrowers and lenders, but can equally well be presented as loan nodes (with an associated Loan ID). The map between the underlying legal and economic graph of credit relations and more common tabular credit data representation is not always straightforward or unique. For example, a typical edge case concerns complex borrower structures with elaborate economic inter-dependency: Sometimes it is more appropriate to think not of a single borrower entity but of a cluster of related entities that are in some way partially liable for a contract.

On the basis of the above mind-set, the most general approach to federate credit data would be designs around *graph data federation*. While federated graph analytics is already an active field, see

e.g., [21], the overwhelming majority of existing financial industry data processes and exchanges are *not* based on graph theoretic concepts. Current credit data archictures typically constitute various forms of schema simplifications or normalizations that culminates in the familiar tabular data representation. Hence we will focus on the terminology of data frames and data tables rather than graph databases but it is worth keeping in mind this underlying correspondence that takes us from a more faithful and logically more general *conceptual graph model* of the credit system to the more common **reference data set** (RDS) representations used in practice. Schematically, we assume that each each financial intermediary hoping to participate in a federation system has in place an effective mapping function:

$$\mathsf{RDS}_{t_i \to t_f} = F(\mathsf{G}_{t_i \to t_f})$$

where RDS is one or more tables of credit data in Long Data Format, whereas G is the (temporal) credit network graph that provides the more faithful underlying abstraction of the target credit system.[2]. The initial and final time indicators t_i, t_f suggest that there is defined window of observation between which all data values are assumed to be valid representations of the state of the credit system.

Constructing Reference Data Sets

The previous discussion was fairly high-level and qualitative. We now start the journey towards a more mathematically and technically defined credit data system. This will be a prerequisite to reach a comparable mathematical definition of federated system so we comment along the way about downstream implications. There is a wide variety of possible credit data, depending on the application context. Important classification dimensions for credit data include:

- The **data type** dimension: what is the shape of the credit data (e.g. numbers, text etc)
- The **information content** (the nature of the information captured by credit data: e.g identity, financial condition, behavioral aspects etc.)
- The **temporal character**: Are the data static or dynamic (variable). Do they concerns Past, Current, or potentially future performance (Forecasts)

Further, because of the very diverse range of credit systems and associated business models, credit data are naturally segmented according to specific domain:

- The segmentation by borrower types (corporate, retail etc. different borrowers require potentially very different data sets),
- The segmentation by credit product type (e.g. secured or unsecured. There is a vast variety of simple or more complex financial contracts which entail credit risk).
- The data ownership dimension (who produces the data, e.g. is a specialized credit scoring or credit rating entity involved?)

[2]That mapping implies there is loss of information that happens already at in a standalone context. E.g. banks would not normally have a detailed representation of all economic relations *between* their clients but would rather rely on aggregate representations and models such as business sectors, client segments or regions

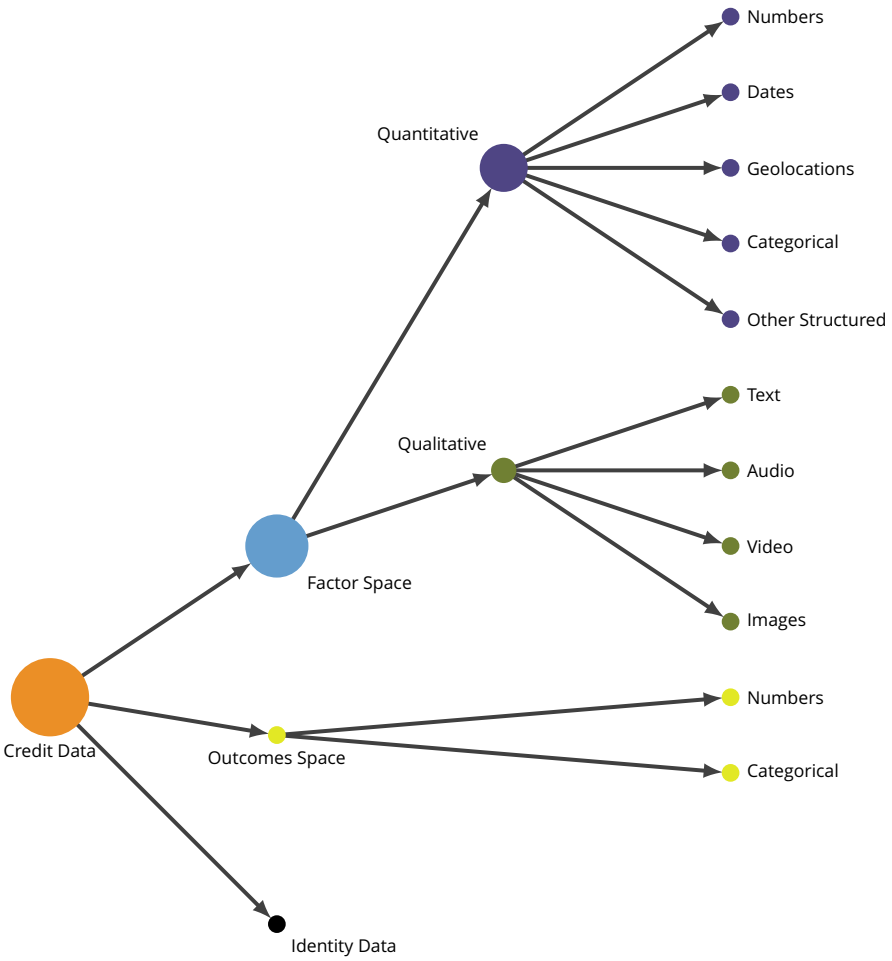


Figure 2: There are many ways to segment the large universe of credit data depending on the use case. The scheme we illustrate here assumes our objective is the development of (unspecified) statistical models, which is both a typical and fairly demanding use case of the credit data system. As as starting point, we split out the identity data. Identity data are obviously most sensitive and would in general not be federated. While they do not convey credit information per se, unique keys are needed if one is to match records of entities. The label (or outcomes) space spans relevant credit event observations (e.g. credit defaults, prepayments, drawdowns, loss-given-default etc.) In terms of data size it is a small but vital component as it captures the realization of uncertainties that one wants to manage. The feature (or factor) space encompasses anything that can be used to gain insights into future credit events. Besides private data, features may also involve publicly available data which for our present discussion we consider out of scope

The above enumerations of data dimensions and Figure (2) should be sufficient to illustrate that a one-size-fits all possibility for federated credit data systems is unlikely. Nevertheless, we can focus on common aspects that are likely to have broad applicability in a quantitative analysis context.

Data Type Dimension

As we sketched in Figure (2), in practice there are three main distinct data types that can be observed in credit data: Qualitative or unstructured data versus Quantitative or structured data which can be further split into numerical and categorical.

Qualitative data. Such data are fairly difficult to use with confidence in quantitative work as the context to which they refer might be very complex. Examples of unstructured data that are in-principle eligible to be included in a credit data system include anything that can be encoded in a digital format: text, images, video, audio etc. Nevertheless, to a human audience unstructured data maybe extremely illuminating. Qualitative Credit Data (such as organizational structure, business model, management quality, market positioning etc.) may have significant impact on credit risk. Ideally such assessments can themselves be structured according to well-defined procedures to enhance consistency and objectivity and enable easier use.

Quantitative data or Structured data are highly organized and easily processed by machines. They are typically found in relational databases and can be inserted, searched, and manipulated relatively quickly. Examples of structured data include simple strings (names), dates, geospatial data (addresses), financial information, contract variables and more. Quantitative data can be split in two further major types.

Categorical data that coded e.g., as integers or some other coding scheme. For example, categorical labels identifying things like credit origination channels. Categorical data may be seen as creating static sub-classes of entities, E.g., a Boolean flag (First time buyer: Y/N) labels borrowers into two categories. A financial product indicator may signal that a certain loan is floating or fixed rate. Categorical data can also be used to represent internal states (more on that below). While categorical data cannot be assumed to be numerical (e.g. the average of categories does not in general mean anything), in contrast to text and other qualititive data, categorical data can be readily integrated in automated procedures and model building.

Numerical data which can be integers or real numbers of various sorts. These can further be subdivided into categories: A credit score is a real number alright, but it has quite distinct qualities from e.g., the nominal amount of a contract. For example, summing up nominal amounts to a portfolio total makes sense but summing up credit scores does not. Numerical data include also some special categories: Dates, pertaining to events and Geospatial Data (coordinates) or physical aspects of e.g., real estate or the geographic distribution of economic activities.

Quantitative data come in various shapes: Timeseries Data capturing values at defined time points, Cross-Sectional Data capturing values across a population of entities (borrowers, loans, collateral), Panel Data that combine both dimensions and more general OLAP cubes, Event Data capturing a timestamped stream of discrete events, and Network (Graph) data placing emphasis on complex relationships between entities. Correct representation and working with these diverse data types are of extreme importance, especially when persisting (storing) data in various formats or tools. Our working assumption is that federated analysis is based on quantitative data. Any important information from qualitative data extracted off-line through feature engineering processes.

Data Content Dimension

Credit data can have a bewildering range of *information content* which by-and-large also determines their utility for credit analysis. A non-exhaustive list would include:

- Identification Data: e.g., company or person names, business code, address of registered office, legal form, date of establishment.
- Repayment and Servicing Data: The actual track record of credit performance. This includes the recording and representation of any Credit Event that is relevant for the credit relationship and the eventual collection of recovery funds for non-performing contracts.
- Accounting and Financial Data: e.g., company accounting records
- Research Data: E.g reviews of management or market structures, interviews with clients etc
- Behavioral Data: Any soft (non-legal) indicators of behaviors / attitudes towards exercising options;
- Legal History Data: Any judicial track record of Credit History collected from courts
- Contractual Data: Namely the details of the credit contract (Loan, Derivative, Lease etc). This will include balances, contractual type, scheduled cash flows, important clauses etc. This category includes contingent contract modification data (in cases where unforeseen events lead e.g. to Forbearance measures). Ancillary contracts (e.g.credit insurance) would also be part of this set.
- Physical Asset Data: in many instances (mortgages, auto loans, project finance, specialized finance) credit data may capture information about physical assets that are used as collateral (or as part of a lease). In general non-credit data (e.g., geospatial data) will be captured by schemas from other, non-financial domains.
- Secondary or Modeled Data: Data points (such as a Credit Score or Credit Rating) that are actually composite data, that is derived from a variety of other credit data points.

It is useful to informally group these categories into two types: hard or primary or directly relevant credit data and soft, secondary or indirectly relevant credit data. This division serves particularly well the credit analysis use case: Hard credit data are the direct and observable reflection of the state of the credit system. They provide an account of what exactly has been legally agreed and is enforceable between the parties involved and how things are progressing over time. Hard data are thus the more-or-less objective, empirical basis (ground truth) and not particularly subject to ambiguous interpretations or assumptions.

In contrast, soft, or indirectly relevant credit data are various data points that provide auxiliary or additional insights and context into the performance of the credit system. Soft credit data might be obtained via longer or more indirect channels and may be subject to more interpretation or assumption risk (including model risk when considering e.g. credit ratings). Nevertheless, such credit data are almost always essential for extracting useful information from the entire package.

Their information content determines by-and-large what the credit data can be used for. One such classification which is prevalent in statistical model building / supervised learning is the segmentation of data attributes to labels and features, or **outcomes and risk factors**, where hard credit data capturing important events are considered outcomes to be explained by any combination of other data. While this separation of duties will be our working assumption towards describing a federated credit data system, it is not hardwired.

Data Ownership / Confidentiality Dimension

Data ownership is yet another dimension with obvious significance in federation context. For example we may have Private data sourced directly from the borrower (i.e. data that are private to the borrower but shared with the lender in the context of their relationship). This is one of the most important elements of the lending process. We may also have Private or public data relating to the borrower that can be sourced from third parties as part of other legal processes or commercial transactions. Finally, when a borrower is an existing relation, a lender will have their own data relating to e.g. borrower behaviors and preferences. Finally there might be entirely Public Data which are somehow relevant.

Static and Dynamic Credit Data Representations

The temporal dimension forms a very important aspect of credit data. Given the core objective of credit data to support managing uncertainty, it is essential to incorporate mechanisms to explore rigorously possible *future outcomes*. In this respect we split credit data into Static data (that do not change over the course of time) and Dynamic or variable data that may change occasionally or frequently, in a scheduled (expected) or unscheduled (unexpected) way.

Dynamic data can be split into Past (historical) data, used e.g., for establishing patterns and/or conjecturing causal relations, Present (current) data are that used as inputs to inference / forecasting today and potentially also Projected, Forecast data that are derived deterministically on the basis of rules (e.g. projected cash flows of a contract).

Identifiers are typically static credit data that label entities. E.g., borrower or loan identifiers. They are alphanumeric in character and range over the set of entities they identify. Hence, they can be mapped to integers like: $i \in [1, 2, ..., n]$. Subsequently, other credit data can be associated with specific entities simply by using identifiers as an index: $A^i, B^{KL}, \ldots, c^d$ etc. For example a set of portfolio data containing a number of borrowers, loans and collateral might be captured by three data tables:

$$B^i = [\mathsf{FTB}^i, \ldots, \mathsf{AGE}^i] \tag{1}$$

$$L^j = [\mathsf{UPB}^j, \ldots, \mathsf{RT}^j] \tag{2}$$

$$C^k = [\mathsf{AR}^k, \ldots, \mathsf{MSA}^k] \tag{3}$$

that collect the borrower (B), loan (L) and collateral (C) attributes at a given time t (omitted). For example the variable FTB^i is a binary indicator of whether the borrower is a first time buyer. With such simple machinery we can capture a significant amount of static attributes of the credit system and the static subset can already support various important federation use cases.

Next we turn to the more complex problem of capturing temporal credit data variability. The general problem concerns modeling the appearance of new entities in system, representing e.g., new borrowers, new lending to existing clients or, most commonly, modified properties over time for any of the existing entities (e.g., changing financial data, loan amortizations through repayment etc.) In practice, due to tractability issues, most representations and calculations involving time-varying credit data are performed in some *discrete time framework*. This means that a discrete temporal grid specifies how information that describes the state and processes ongoing within the credit system are captured along the temporal dimension.

The temporal resolution of that grid (how closely spaced the observation times) depends on the nature of the credit system and the type of analysis required. Conceptually it must match the timescale on which there is material variability in the credit properties of the system. Observing such variability happens through manifestations such as payment events, accounting reports of financial conditions etc. In the simplest case, future cash flows of credit assets and/or liabilities that are part of the overall credit system are considered at a set of given *timepoints* $t_k \in [T_0, \ldots, T_M]$ which might be e.g., daily, quarterly or monthly. When a temporal grid is selected to be more coarse (e.g. annual) all input data must be appropriately mapped or aggregated to fewer timepoints or periods.

The observation period (window) spans the time interval from some initial date up to the current date (or most recent date). In general the width and location of that window is determined by broader questions around availability, suitability and continuing relevance of historical credit data. The *length* of the historical observation window that is relevant varies by application. It is generally considered that more recent data are more useful, but credit behavior during crises (which may have happened in a more distant past) are also typically informative.

Contractual or scheduled cash flows represent a projection of what cash flows must take place on the basis of the loan documentation. Such projections might be Deterministic and expressed as absolute amounts (e.g. fixed rate mortgages); Contingent and computed in terms of rates and market observables (e.g. floating rate mortgages); Based on potentially more complex formulas (non-linear expressions involving thresholds such as caps and floors or even more complex logic). Contractual data in particular will involve specific periods of time (Loan Maturity, Loan Age, Remaining Life etc.) that are expressed either in absolute terms as temporal intervals $t_k = [T_0, T_1]$ or in relative terms as the number of periods T over which something happens (e.g., remaining maturity in months).

Capturing scheduled (or forward-looking) and historical (actual or past) cash flows is a challenging aspect of credit data. It concerns lower-level (more detailed) information that is hard to make available as part of loan-level data. This is both because of the higher granularity (e.g. a mortgage loan will involve hundreds of payments) and it involves contingent definitions: amounts might only be determined at future dates on the basis of future observable market variables such as interest rates. Schematically, scheduled cash flows for a loan j are represented as a list of temporally ordered functions SCF_t^j that extends over a complete set of timepoints $t \in [1, n]$:

$$\mathsf{SCF}^j = [\mathsf{SCF}_1^j, \ldots, \mathsf{SCF}_n^j]$$

Actual cash flows in contrast represent a record of *what has actually happened* in terms of value exchanges between borrower and creditor (and any other parties that might be involved in the credit relation). What is done occasionally in order to capture actual cash flow exchanges is to compress select multi-period values as one-dimensional arrays or strings. Schematically, actual cash flows are represented as a list of temporally ordered real number values up to the current time t, ACF_t:

$$\mathsf{ACF}^j = [\mathsf{ACF}_1^j, \ldots, \mathsf{ACF}_t^j]$$

ranging from the start of a credit contract up to the current time t.

Identifying and Recording Credit States

The label space in credit risk assessment is structured around *Credit Events*. The definition of a credit event is quite broad. It can be e.g. missing a payment (or multiple payments) under the actual contract Lo, any other relevant contract, or a situation where the net worth is negative. For regulated banks the definition of credit events must comply with regulatory requirements. Historically realized credit events in a sample of borrowers that is considered similar to the set for which a new assessment is to be applied are forming the label space.

The circumstances that can cause significant state changes are quite diverse. Excluding extraordinary events such as Force Majeure, in current practices and contracts events signaling important such deviations are typically classified under Prepayment Events, Credit Events or Drawdown Events. From an economic perspective unexpected events express the flexibility on the side of the borrower to exercise certain options they legally possess: either i) to repay funds early ii) to not repay funds (facing possible consequences as per applicable law) or iii) to draw additional funds (if legally and practically possible).

Any of these above might be the dominant financial consideration in a particular context. This depends on the type of contract or borrower. Follow-up sequences of additional events may also involve options available to the creditor (for example to seek recoveries by foreclosing on real estate). These can affect cash flows in complicated ways. The manner in which optionality manifests might be captured in contractual clauses (e.g. limits to the amounts of prepayment or drawdowns) but might be also fairly open-ended and contingent on many external actors (as it is the case for example with foreclosure proceedings). We will focus in the remainder on representing credit events as a baseline use case for federated analysis, but clearly the scope is more broad.

A key strategy is to create *credit state indicators* (flags). These are defined as indicators that are built primarily (but not exclusively) by monitoring the deviations between scheduled and actual cash flows. More specifically, if R_i is the *Credit State* of entity B^i, and E is a state that we consider to be a manifestations of a credit risk event, then the data point $L_i = 1_{\{R_i=E\}}$ indicates that such an event has indeed taken place historically for that entity. The state E might be determined by comparing actual cash flows ACF and contractual cash flows SCF or in some other way. Complication abound. E.g. a borrower may be party to many other credit contracts, the details of which may be unknown to the creditor. Borrower credit performance under these other contracts may affect the credit standing indirectly. For example, a credit default on a separate credit relation may legally make all funds due immediately (hence modify the scheduled cash flows). The implication is that not all credit states derive from directly observable cash flows. At it simplest, though, recording credit states is an indicator of how many payments are past due, which can be counted, e.g. as the number of temporal grid periods. For long maturity contracts, keeping track on monthly flags might be too cumbersome. For consumer loans a typical more coarse-grained segmentation of credit states focuses on three distinct states which go by the following names (naming conventions vary widely by jurisdiction and market segment):

- Early Arrears Events (up to 90 days past due). During this phase, the focus is on engagement with the borrower to remedy the situation and collect information required for a more detailed assessment of the borrower's circumstances (e.g. financial position, status of loan documentation, status of collateral, level of cooperation, etc.).
- Late arrears / Restructuring / Forbearance (90 to 150 days past due). This phase focuses on implementing and formalizing restructuring/forbearance arrangements with borrowers. Essentially an

amended (modified) contract that aims to be the defining legal document moving forward. From a financial perspective, loan modifications may be classified whether they involve a financial loss or not.

- Formal Default / Liquidation / Debt Recovery / Legal Cases / Foreclosure / Enforcement. This phase focuses on borrowers for whom no viable forbearance solutions can be found due to the borrower's financial circumstances or cooperation level. In such cases, creditors typically perform cost-benefit analysis of different liquidation options including in-court and out-of-court procedures.

Besides the duration of delinquency, these three phases are also distinguished by the status of the original loan prospectus: In the first instance it remains in place (but amounts-due continue accumulating), in the second phase the loan gets modified, taking into account events in the first period, while in the final possible stage the scheduled cash flow projections cease to be the driving element (they are still the measure of value that must be recovered) and the focus shifts to any available forms of security (collateral), insurance or credit protection that will be recovered instead of the promised cash flows.

Delinquency flags are constructed by comparing the vector of scheduled and actual cash flows and identifying any discrepancies. A debtor i might be at a time point t assigned (classified) into a delinquent state S_t^i by counting the longest contiguous string of missed payments that ends at t. Under such a direct numerical approach the indicator is taking values in $[1, \ldots, R]$ where R is the longest possible delinquency. We might record delinquency status explicitly with binary variables, e.g.,

$$\mathsf{DQ}_{t,30}^i, \mathsf{DQ}_{t,60}^i, \ldots, \mathsf{DQ}_{t,180}^i$$

.

where the numerical figure indicates how many days is the borrower in delinquency. Another way to record that is as a multinomial variable $S_t^i \in [0, 30, \ldots, 180]$. One can generalize the concept of credit status flag to include phenomena at advanced stages of delinquency (modifications or default/bankruptcy and liquidation). For example $R_t^i \in [0, 1]$ might be a modification flag, $S_t^i \in [0, 30, \ldots, 180, D]$ is delinquency flag that includes a default state etc.

An important consideration for credit relationships that enter the difficult zone of forbearance or enforcement (liquidation) is that significant amounts of **additional credit data** are generated as a consequence and such data are not needed or are available for the majority of borrowers. How do these contingencies fit into the reference data source targets? The loan modifications undertaken in forbearance activities can conceptually be considered as new loans or more appropriate *loan deltas*. They are thus one or more new data sets with attributes similar to those characterizing the original loan but modified terms. These new loan data entries may also encode any loss amounts that have been written off in these proceedings.

For loans that enter the liquidation phase the loan modification paradigm is no longer suitable. What happens here is that there are a number of additional activities and costs associated with the proceedings (legal, tax, insurance, asset related costs etc.) and a number of recoveries on the basis of asset disposals, guarantees or insurance etc. We might represent this as a new contingent node that characterizes the liquidation process (e.g., type and timing of court proceedings) and has an associated stream of positive and negative cash flows. All in all, our more complete dataset that includes contingent data might look like at a future time point t as follows:

$$B^i = [\text{FTB}^i, \ldots, \text{AGE}^i] \tag{4}$$
$$L^j = [\text{UPB}^j, \ldots, \text{RT}^j] \tag{5}$$
$$ML_1^j = [\text{UPB}_1^j, \ldots, \text{RT}_1^j] \tag{6}$$
$$ML_2^j = [\text{UPB}_2^j, \ldots, \text{RT}_2^j] \tag{7}$$
$$C^l = [\text{AR}^l, \ldots, \text{MSA}^l] \tag{8}$$
$$R^l = [\text{FD}^l, \ldots, \text{NSP}^l] \tag{9}$$

In the above collection we have a couple of modified loan ML entities for every loan L^j that enters such a phase and a set of recovery entities R^l for every collateral C^l, with variables like FD and NSP to indicate foreclosure date and net sales proceeds respectively.

We now have a reasonably defined skeleton for a credit data system and we might proceed to consider how one might arrange federation between different data owners, starting in the first instance with what preparatory steps might be necessary.

Federated Credit Data Systems

Pre-Federation Activity

Specific pre-federation activities are the tasks that would need to be carried out by participating entities in preparation of supporting the federated analysis / model development. They are to be carried out individually, possibly off-line and without any exchange of information during the process. Yet potentially these are procedures that must adhere to shared standards, best-practices etc. A breakdown of distinct such activities is indicated by the following list:

1. Developing a Business Case for a new type of federated analysis. Floating a proposal among participants on the basis of internal / external market developments. This is a *problem or opportunity identification* stage. The objective is to pin-point a problem that might be solvable with federated tools or some new benefit to be obtained.

2. Developing Business Requirements. The outline of the functionality, features, constraints, KPI's etc. around a new federated analysis proposal. This would for example include reference to an abstract model or algorithm and the derived metrics.

3. Internal Credit Data Collection. Identifying existing or new private data sources that will create inputs to the federated system.

4. Internal Credit Data Review. Assessing compliance with minimum requirements for federation.

5. Internal Credit Data Cleansing. Performing programmatic or manual changes to data to meet the requirements of federation. Interestingly, this step may also be in scope for iterative developemnt in federation context (effectively collective DQ processes)

6. Credit Data Standardization towards producing the requisite Reference Data Sets.

1. Credit Data Collection

Data Collection are all operations that must be performed internally / externally to compile a dataset that is adequate for a concrete task of federated credit risk analysis and/or model development. Such data collection operations will vary a lot depending on the IT infrastructure available to different entities. In general it cannot be assumed identical (homogeneous) across the federated universe. Potentially even the shape of this IT architecture (e.g. database schemas or API's) could be considered proprietary and sensitive knowledge of the credit granting institution.

The outcome of this step is the Identification and access to data sources (files, databases, feeds etc) with unique URI and the Identification of the relevant model variables (features, labels, timeseries etc.).

2. Credit Data Review

The Credit Data Review stage will typically include the many elements of a formal Data Quality Framework including in particular the dimensions of: data completeness, accuracy, consistency, timeliness, uniqueness, validity, availability and traceability. In addition, the review will have more specialized requirements linked to the federated analysis use case. The most sensitive class of all credit data is the identity data set $\mathbf{ID}_t$ across the credit universe. The credit data review will highlight all possible data fields that might be defined as sensitive in either a personal data context or a commercial context.

3. Credit Data Cleansing

Data cleansing is a stage that will follows typically as a remediation process following the results of the data review. It involves concrete steps of *modifying* existing sourced data sets:

- Correcting wrong Representations (e.g. Dates, Percentages)
- Correcting wrong Values (Summations)
- Imputing missing Values (Where that can be done without introducing assumptions)
- Removing or Anonymizing sensitive data
- Applying differential privacy algorithms

Data cleansing is in-principle a more mechanical process than the credit data review. Nevertheless both in terms of responsibility and the need to adapting to individual institutional circumstances means that this step too is largely a pre-federation activity. Participants would be expected to perform this according to agreed best practices and possible provide KPI's.

4. Exploratory Data Analysis

Exploratory Data Analysis (EDA) in the context of credit risk analysis is a highly expert driven (manual) process that aims to do the following:

- Provide suggestions for further data collection and/or data quality improvements (e.g. Missing Data)
- Define concrete rules for and identify outlier data points

- Create an overall map of the underlying structure of the data (identify clusters of similar variables, reduce dimensionality)
- Formulate hypotheses about the causes underlying credit risk phenomena (important / statistically significant factors)
- Support the selection of characteristics (covariates) for credit model development

EDA activities are essentially an iterative and adapted process that is highly data and use-case depended. Federating nodes might share best practices / code of conduct type documentation. While there is little scope for federating the entirety of this activity it might be possible (and desired) to introduce iterative workflows.

5. Data Alignment / Data Harmonization

Unless credit data have been collected with identical definitions, requirements and procedures across different institutions it is very unlikely that they fulfill the requirement that they offer the same representation of a given credit phenomenon. Data harmonization is an important process to ensure that data delivered into a federated credit system have the intended meaning. In federation the particulars of individual data sources will in general not be available to all participants to examine. Establishing data alignment may encompass a variety of activities:

- Automated assignment based on commonly agreed labels (when applicable)
- Automated low level checking based on data features (e.g. data type)
- Expert based validation through inspection of individual values and ranges for data samples
- Statistical validation (e.g. based on uni-variate analysis)

The successful conclusion of these pre-federation activities means we have now all the tools needed to describe federated credit data systems with more practical detail.

FCDS Structure Assumptions

The key design feature of a FCDS is that credit data never leave the organization that owns it. Credit data support localized analyses, computation and/or risk model building and only *derived results* are released to the federation system. A federated credit data system brings together diverse IT infrastructures (belonging to different business entities) to support federated credit analysis. Existing infrastructure comprising of components such as data storage, computing engines etc. is *loosely linked* over networks using defined API's (application programming interfaces). The use of API's allows, if so desired, the partial decoupling of technologies and infrastructures used by different entities. Different types of digital infrastructure, different software stacks can be used together, provided they adhere to certain common communication standards. While there is flexibility in such decoupling, it is still implied that federating entities satisfy common minimum requirements around various quality dimensions (e.g the frequency, accuracy, form of data collection, adequate computational capability, adequate access to digital networks etc.)

In the federation designs that we discuss here there is a *partially trusted central node* that is required to coordinate analytical activities but does not have any access to primary data. While the coupling is loose, the operation of such a system still relies on common standards at various levels (minimally a common agreed credit data schema as we will discuss below). The complexity and available techniques for federated analysis depend on constraints around:

- *where* computations takes place (whether there is a coordinating computing node)
- whether the coordinating node is *trusted* or not for a particular data exchange and calculation and the particular trust model (honest / malicious)
- whether it can be assumed that federating entities themselves are not malicious (and would not for example intentionally compromise the analytic outcome).

For definiteness we assume the following configuration:

- Participating Nodes are **known** to each other and mutually approved through some participation protocol. The list of participating entities is not necessarily public.
- There is one or more partially trusted third parties (**PTTP**) that operate certain elements of infrastructure in accordance to specified requirements and coordinate narrow information exchanges according to some federation protocol
- Participating Nodes **trust** each other to adhere to a Data Quality protocol which requires that federated data sets fulfill some pre-agreed requirements along a number of data quality dimensions. Some DQ assurances might be provided by a PTTP but others might be self-signed by PN's.
- Participating nodes **do not** have access to the operational state of PTTP's during production runs.
- Participating nodes **do** have access to any underlying federation systems and code (*available as open source*) used by the PTTP, including the virtual database schemas of the federated data set and analytic functions that will operate on either local or centralized data.
- Participating nodes are honest-but-curious: They will use all information legally provided by the federation protocol to their advantage

Notation

The overall architecture and demographics of a federated credit system in a simplified abstraction might be as follows:

- $k \in [1, \ldots, N]$ A number of federation participants, all of whom wish to perform a credit analysis or credit risk model by federating their respective data.
- $B_i \in [1, \ldots, S_k]$ entities (e.g. borrowers) in the given Credit Market segment associated with the k-th federation participant (their union will in general not be the entire market).
- Each borrower B_i has an associated data set $\{D_i^k\}$ composed of an q-dimensional collection of variables. This data set is private to the relationship of the borrower with their lender (k-th participant) or other professional intermediary (e.g. credit bureau or credit rating agency).

- A global Model or Metric $M(D)$ that acts on a local dataset D_i^k and produces a metric / outcome M_k for that local sample.

A federated credit system is a process in which the N data owners collaboratively develop a model M in which they not expose their own data $\{D_i^k\}$ to others. The set of Quantitative Credit Data of all modeled entities (D_i^k) is the collection of relevant numerical or categorical values that capture current and past aspects of said modeled entities. The dimension of this space can be potentially quite large. A useful indicator of the size and complexity is the set of Non-Performing Loan data templates constructed by the European Banking Authority, which ranges between 100 to 400 data fields, depending on version. The **total federated data set** is the union $D = (D^1 \cup \cdots \cup D^N)$.

Definition: Federated Credit Data System

A *federated credit data system* (FCDS) can be defined as the digital infrastructure that supports the orchestration of federated credit analyses: enables the conduct of statistical analyses (such as descriptive statistics and different types of credit and portfolio modeling and analysis) in a context where actual credit data are not shared between federating entities and information exchange satisfies defined levels of privacy. The FCDS design is not unique but depends on desired functionality, the individual data contributions and constraints posed by the federating entities.

Federated Master Data Tables

Using the schematic generative process we discussed in Section , we might have a collection of data characterizing borrowers, loan and collateral states that is the union of several tables,

$$D = [B^i, L^j, ML_1^j, ML_2^j, C^l, R^l, DQ^i, S^i, \ldots] \tag{10}$$

for all observation times. This is the raw data material for the next step, which is the integration and decomposition $D = [I, F, L]$ into identification, feature and label (outcomes) data and the construction of a master data table. A **Federated Master Data Table** is a very useful metaphor for how the FCDS can fulfill its function. It is a *virtual* data table that captures a definitive collection of federated data and which can be used for analysis or development of models. It is a virtual table in the sense that the data making up this data structure are distributed over infrastructure that is owned and operated by distinct entities and are never brought together in the same system. But it is also a real schema in the sense that local functions M can expect to find the indicated variables.

The sufficiency of such a data structure to allow the development of adequate analyses, models and tools is linked to specific use cases. It can be seen as a minimum viable component that enables many useful applications but it can be augmented by other structures. Using the federated master data table we can discuss some important conceptual federation designs that are in-principle available.

Federated Sample Augmentation - Horizontal Federated Learning

Sample Augmentation (Other names: horizontally partitioned data or horizontal federated learning) refers to a federation architecture that integrates historical data (samples) that share the same feature / label space. Two or more credit providers targeting the same region / sector / product cell may federate their data sets.

Example of a *Virtual* Federated Master Data Table

Horizontal Federation D^k	Entity ID's	Feature Federation F^k				Label Federation L^k		
D_1	I_1	F_1^1	F_2^1	...	F_p^1	L_1^1	...	L_o^1
	I_2	F_1^2	F_2^2	...	F_p^2	L_1^2	...	L_o^2
D_2	I_3	F_1^3	F_2^3	...	F_p^3	L_1^3	...	L_o^3
	I_4	F_1^4	F_2^4	...	F_p^4	L_1^4	...	L_o^4
	I_5	F_1^5	F_2^5	...	F_p^5	L_1^5	...	L_o^5
D_3	I_6	F_1^6	X	...	F_p^6	L_1^6	...	L_o^6
	I_7	F_1^7	X	...	F_p^7	L_1^7	...	L_o^7

Table 1: One or more virtual master data tables are the central prerequisite for quantitative federated analysis and/or federated model development. The adjective virtual is meant to indicate that this is not an *actual* data table. Credit data are never brought together in a single storage system. It is rather a *common schema* and data values conforming to that schema that can only be accessed via privacy-preserving mechanisms. Federating entities may contribute complete datasets as rows D^k (horizontal federation) or alternatively features F^k and/or labels L^k (vertical federation). Identification data I are singled out to illustrate the challenge of private data matching (e.g., eliminating duplicate rows or associating new columns). Simply put, the direct benefits of federated analysis can be seen as the virtual extension of this table in both dimensions. The split into features and labels alludes to the common use case of estimating a statistical models but is by far not the only possible use of the master data table. The X indicators in the rows corresponding to participant D^3 are meant to illustrate a frequent issue: some participants will simply do not have available a certain data set.

The objective of federated sample augmentation is to improve the statistics (count) of the total available data set. In particular it can provide solutions to the *low credit default portfolio problem*, namely the (technical) challenge that credit portfolios of high quality do not (by design!) generate sufficient event data to analyze quantitatively the underlying risk profile.[3]

Whether sample augmentation can fulfill the promise of mitigating limited statistics obviously depends whether the total possible sample statistics is adequate. It is conceivable that event counts might be insufficient for the desired analysis even if the *entire* available population could be observed in a federated credit data set.

Sample augmentation requires that the Label spaces between federating nodes are identical. As ID's are not shared, it does not (in the general case) require exchange or matching of ID's[4]

Federated Feature or Label Augmentation - Vertical Federated Learning

Feature augmentation (also vertical federated learning or feature-wise federated learning or vertically partitioned data) is the enhancement of federated credit data sets by adding additional features with explanatory potential. The objective of federated feature augmentation is to improve the quality of analysis based on the feature data set (whether a supervised or generative algorithm) by introducing potentially additional explanatory factors. The degree to which this approach adds value links to the degree these additional features are correlated with already existing features and/or with the phenomenon being studied.

Feature augmentation requires that additional features (for example additional borrower characteristics) are attached to each existing entity ID. This approach requires that ID's are matched exactly (identifying accurately which legal entity the additional data refer to) which must be done in a secure and private way, as ID's are sensitive primary data that should not be shared with the PTTP. Various protocols have been proposed to this effect, see, e.g., [22]. In a similar vein, Label augmentation is the enhancement of federated credit data sets by adding additional outcomes (labels) with explanatory potential. This will typically involve discrete observable events, e.g., credit event history in different markets, prepayment events etc).

Federated Analysis Use Cases

There is a large body of work around various types of federated (distributed) analysis and algorithms. An important differentiating factor is the nature of the *function* M (or functions) that are applied to shared data. In particular whether they are simple and deterministic functions providing directly some gleaned knowledge or more complex *probabilistic models* such as regression algorithms or machine learning that would require substantial additional steps e.g. model validation and approval, deployment and ongoing monitoring before used in production.

Federated Concentration Risk Metrics

Let us start with a simple example of how we can use federation to extract (in a privacy-preserving way) and share back to the participating nodes new and valuable information.

[3] This data paucity challenge would apply equivalently to any other event associated with portfolio entities that have a low probability of occurrence.

[4] An edge case might be when the same borrowers are present in multiple samples.

Imagine a use case where each participating node wants to *benchmark* some portfolio metric against its peers (let us say aggregate exposure and concentration in some client segment versus the rest of the market). This is implemented by the coordinating node requesting that local nodes execute a summation function on any relevant extensive (summable) numerical variable d and return the resulting sum to the central node. The central node then computes the total (market-wide) sum and returns to the nodes a fractional metric. The coordinating node sends to local federated environments the function $M = Sum$ and the data field selection $I = d$:

Algorithm 1: Federated Summation and Fraction Calculation that can inform participants about their relative concentration risk profile

Coordinating Node:

for $k \in 1, 2, \ldots, N$ **do**

  $V_k =$ Local Calc$_k(Sum, d)$

  $V_T = \sum_k V_k$

  $w_k = V_k / V_T$

  return w_k *to node* k

Local Calculation ($M = Sum$**,** $I = d$**):**

$V_k \leftarrow M(v_j)$ // Compute locally the sum of values v_j of the d-th characteristic

return V_k *to server*

Using the weights w_k the coordinating node can compute and communicate, e.g., a global concentration metric like the HHI index which individual nodes can compare against their local indices computed on the local distribution od the variable d.

An alternative split of computations is to return to nodes the total V_T instead of computing the weights w_k. Such a protocol is simple enough that it might, for example, be combined with homomorphic encryption to further reduce information leakage.

Concentration Risk: Federated Histograms

Let us explore the concept of federated analysis a bit further, building on the previous example to obtain a *federated histogram*. A federated histogram is simply the accumulation of frequency distributions in variable bins without bringing the data required for this calculation in the TTP enclave. Thus a more complex histogram function $M = Hist$ must be sent to the local nodes and partial histogram statistics are sent back to the coordinating node, where they are aggregated and distributed in their final form back to the nodes.

If X_j is the set of S observed values that must remain private, m is the number of distinct bins, and $b_m = [L_m, R_m]$ are the left and right boundaries of the m-th bin, then a histogram is the vector valued function M_m that counts how many observations fall into each bin:

$$M_m = \sum_{j}^{S} 1_{\{X_j \in b_m\}} \tag{11}$$

Algorithm 2: Federated Histogram Construction. Participating nodes get the full statistical distribution of a variable without sharing actual data values

Coordinating Node:

for $k \in 1, 2, \ldots, N$ **do**

$M_m^k, V_k = \text{Local Calc}_k(Sum, Hist, D)$

$V_T = \sum_k V_k$

$w_k = V_k / V_T$

$M_m = \sum_k M_m^k w_k$

return M_m *to node* k

Local Calculation ($M = (Sum, Hist)$**,** $I = d$**):**

$V_k \leftarrow Sum(v_j)$ // Compute locally the sum of values of the d-th characteristic

$M_m^k \leftarrow Hist(v_j, b_m)$ // Compute locally the bin distribution the d-th characteristic

return M_m^k, V_k *to server*

The above examples should already make clear the design pattern that permeates federated analysis algorithms:

- Local computations have access to the full set of private data hence can be performed optimally, without privacy imposed computational limitations or any additional complexity.
- Derived data *are* communicated to a coordinating node. The fundamental assumption is that this central node cannot be trusted with private local data but can be trusted with derived data. Therefore it is important, on a case-by-case basis, to evaluate whether leakage of the derived data does in fact in any way compromise the desired privacy regime. In practice this means that the set of functions M that can be evaluated locally is strictly controlled.
- The value-added by federated analysis rests on the ability to design federated algorithms that create valid and meaningful views within the desired privacy regime.

That last point (expanding the range of feasible calculations) is currently the subject of intense research. Certain classes of algorithms are relatively more adapted to the federated context than others. But as the next example illustrates there are already significant capabilities.

Federated Machine Learning

As a prelude to the third paper on the series which focuses on concrete credit analysis algorithms, we discuss the example of **Federated Averaging** that has been introduced in [23]. Instead of any detailed machine learning model we illustrate the federated application of *stochastic gradient descent*. Here the function M is the functional form of gradient descent (objective function Q) which in the simplest case

would be minimized for the entire local sample D.

Algorithm 3: Sketch of the Federated version of Stochastic Gradient Descent. Participating nodes can estimate any model based on SGD without a central data repository. Collected centrally are only model parameters w.

Coordinating Node:

Initialize w_0

For each global round t

for $t \in 1, 2, \ldots$ **do**

  $w_k = \text{Local Calc}_k(Q, D, w_t)$

  $w_{t+1} \leftarrow \sum_k w_k$

return *Final* w *to node* k

LocalCalc ($M = Q$**,** $I = D$**):**

For each local epoch τ

for $\tau \in 1, 2, \ldots$ **do**

  Randomly shuffle samples

  $w_{\tau+1} \leftarrow w_\tau - \eta \nabla Q(w; D)$

return *Return final* w_k *to server*

Challenges

We list now a set of *minimum requirements* or preconditions and likely challenges to implementing effective federation architectures.

Minimum Requirements for Credit Data Federation

As our basic guardrail, the regulatory document *Principles for effective risk data aggregation and risk reporting* [24] identifies several principles for ensuring strong risk data aggregation capabilities *within* an institution. A subset of these principles are particularly relevant to discuss also in federation context:

1. **Credit Data Governance**: institutional arrangements to ensure that the pool of federated credit data reflects ground truth in the respective entities. Federation raises the bar given that poor credit data governance by *some* participating members may degrade or annul the federation benefits for *all* members.

2. **Credit Data Accuracy and Integrity**: meeting agreed *common* data quality standards and minimum requirements for reconciliation / validation of credit data that are applied across all federated entities.

3. **Credit Data Completeness and Granularity**: ensuring completeness, e.g., by customer segment, legal entity, asset type, industry, region or any other relevant grouping. Addressing this requirement enhances federation outcomes as it facilitates deriving the *most informed* analysis across from existing data.

4. **Credit Data Timeliness and Frequency**: ensuring credit data are up-to-date and refreshed in a timely manner. This is particularly important when federation outcomes are used in *early warning*

context or are otherwise sensitive to the temporal character of credit data.

5. **Underlying Data and IT Architecture**: resilient architecture against operational risk and cyber-risk in particular, especially if federation is relied upon on an ongoing operational basis.

Federated setups will be easier to orchestrate when a number of preconditions are satisfied. The precise set of conditions depends on the concrete use case which determines the desired shape of the federated credit data system and the intended applications. Jurisdiction differences may create significant incompatibility in the description of credit system operations. Aspects such as market structure, contract design, legal definitions around credit events etc. may vary between countries, reflecting different legal and cultural aspects. Indicatively even within the European Union there are important credit data variations between member states. Another relevant factor is the degree of digitization of particular credit systems. This determines, for example, the amount and type of credit data that are readily available.

The main preconditions can be organized around the need for a minimum amount of standardization among participating nodes. For horizontal type federation this implies a certain homogeneity among federating members. For example all nodes participating in a federated scheme should share a common or at least compatible legal and regulatory framework.

- Interoperable IT infrastructure. Decoupling is greatly facilitated by the use of well defined API's that are adapted to specific IT platforms used by different nodes.
- A common identification of clients, contracts, collateral and any other distinct entities entering the credit data collection. While protecting identity is a foremost consideration for many applications (e.g. measuring concentration) it is primarily important that the *uniqueness* of entities is respected.
- A common definition of credit events, thus a common or compatible label space
- Shared feature (risk factor) space that uses recognizable and available characteristics expressed in compatible units.
- Common desired outcomes such as ultimate risk metrics, valuations, scores or other results
- To the degree that more complex algorithms are involved, aligned *model explainability* requirements.

The above requirements are easier to meet if the credit assessment nodes pursue shared business models and in the same jurisdiction. This facilitates for example that:

- Customer segmentation is defined in compatible terms (but needs not be identical).
- Financial contracts / credit products in scope have compatible characteristics (in terms of the typical purpose, amounts, maturity, covenants, conditionality, security type etc).
- Collateral assets belong to similar categories, their economic profile captured using similar metrics and amenable to e.g., similar valuation methodologies.
- There is a common set of business objectives and criteria applied by each participating node to evaluate the performance and benefit of federation.

Despite the above requirements pushing for relative "homogeneity" it may well be the case that special circumstances favor or even necessitate a heterogeneous network of participants. This will be naturally the case for vertical federation designs where, by definition, the participating nodes play different roles in the credit system. Heterogeneity of nodes may also be desirable when the objective is to develop better (e.g. more complete) coverage of the credit system in scope, for example span the range of bank entity size.

Federated Data Quality Criteria

Given the level of generality we operate at here it is not possible to develop detailed data quality criteria but it may be useful to outline some key attributes likely to be relevant in each instance. The important and new aspect of managing credit data quality in this context is that this, too, is an analysis that must be produced respecting the agreed privacy regime. Once federated entities participate in any given virtual master table one can (in-principle) calculate and report back individually or collectively a number of high level quality assessments such as:

- The overall number of federating nodes.
- The number of federating nodes that have contributed at least one data row or data column or the average number of contributed rows or columns per node and other such statistics.
- The data completeness of the contributed rows or columns.
- More specific metrics, e.g., the degree to which nodes have contributed scarce data, highly informative features or low probability outcome variables

If the resulting virtual master tables do not meet agreed quality criteria, then the process may revert back to contributing entities. If the criteria are met, then in principle there is an open path to perform further federated analysis and/or model building.

Bibliography

[1] V Candeias G Dana. Federated Data Systems: Balancing Innovation and Trust in the Use of Sensitive Data. *World Economic Forum White Paper*, 2019.

[2] L. Jonathan Dursi et al. Candig: Federated network across canada for multi-omic and health data discovery and analysis. *Cell Genomics*, 1(2):100033, 2021.

[3] Chlo M Kiddon Andrew Hard and co authors. Federated learning for mobile keyboard prediction, 2018.

[4] P. Papadopoulos. WP9: Federated Credit Systems, Part I: Unbundling The Credit Provision Business Model. *Open Risk White Papers*, 2020. Online Link.

[5] P Kairouz et al. Advances and Open Problems in Federated Learning.

[6] Pinkas Lindell. Privacy Preserving Data Mining. *J. Cryptology 15, 177206*, 2002.

[7] P. Samarati and L. Sweeney. Protecting Privacy when Disclosing Information: k -Anonymity and Its Enforcement through Generalization and Suppression, 2011.

[8] Craig Gentry. A fully homomorphic encryption scheme, 2009.

[9] Shruthi Gorantala et al. A general purpose transpiler for fully homomorphic encryption. Cryptology ePrint Archive, Paper 2021/811, 2021. https://eprint.iacr.org/2021/811.

[10] CSIRO's Data61. Python paillier library. https://github.com/data61/python-paillier, 2013.

[11] C. Dwork et al. Calibrating Noise to Sensitivity in Private Data Analysis. *TCC 2006*, 2006.

[12] J Lei C. Dwork. Differential Privacy and Robust Statistics. *STOC 09*, 2009.

[13] A. Korolova U Erlingson, V. Pihur. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response.

[14] E. Shi et al. Privacy-Preserving Aggregation of Time-Series Data.

[15] D Ramage P Kairouz, K Bonawitz. Discrete Distribution Estimation under Local Privacy. In *Proceedings of the 33 rd International Conference on Machine Learning*.

[16] S Kumar H.B McMahan A Suresh, F X Yu. Distributed Mean Estimation with Limited Communication. In *Proceedings of the 34 th International Conference on Machine Learning*.

[17] A Ozgur L P Barnes, Y Han. Lower Bounds for Learning Distributions under Communication Constraints via Fisher Information.

[18] I M Schmutte J M Abowd. An Economic Analysis of Privacy Protection and Statistical Accuracy as Social Choices. *American Economic Review*, 2018.

[19] P. Papadopoulos. WP8: Connecting the Dots: Economic Networks as Property Graphs. *Open Risk White Papers*, 2019. Online Link.

[20] P. Papadopoulos. WP10: Connecting the Dots: Concentration, diversity, inequality and sparsity in economic networks. *Open Risk White Papers*, 2021. Online Link.

[21] Dongming Han et al. Graphfederator: Federated visual analysis for multi-party graphs, 2020.

[22] Monica Scannapieco, Ilya Figotin, Elisa Bertino, and Ahmed K. Elmagarmid. Privacy preserving schema and data matching. In *ACM SIGMOD Conference*, 2007.

[23] H.B McMahan et al. Federated Learning of Deep Networks using Model Averaging.

[24] Basel Committee on Banking Supervision. Principles for effective risk data aggregation and risk reporting, 2013.