Privacy-preserving impact evaluation using Difference-in-Differences

Manuel Huth [1,2], Carolina Alvarez Garavito [2], Lea Seep [2], Laia Cirera [3], Francisco Saúte [4], Elisa Sicuri [3,5], and Jan Hasenauer [1,2]

[1] Institute for Computational Biology, Helmholtz Munich - German Research Center for Environmental Health, Munich, Germany.
[2] LIMES, Faculty of Mathematics and Natural Sciences, University of Bonn, Bonn, Germany.
[3] ISGlobal, Hospital Clínic - Universitat de Barcelona, Barcelona, Spain.
[4] Centro de Investigação em Saúde de Manhiça, Manhiça, Mozambique.
[5] LSE Health - Department of Health Policy, London School of Economics and Political Science, London, UK.
* Correspondence: jan.hasenhauer@uni-bonn.de

Summary

Difference-in-Differences (DID) is a widely used tool for causal impact evaluation but is constrained by data privacy regulations when applied to sensitive personal information, such as individual-level performance records or healthcare data, that must not be shared with data analysts. Obtaining consent can reduce sample sizes or exclude treated/untreated groups, diminishing statistical power or making estimation impossible. Federated Learning, which shares aggregated statistics to ensure privacy, can address these concerns, but advanced federated DID software packages remain scarce. We derived and developed a federated version of the Callaway and Sant'Anna DID, implemented within the DataSHIELD platform. Our package adheres to DataSHIELD's security measures and adds extra protections, enhancing data privacy and confidentiality. It reproduces point estimates, asymptotic standard errors, and bootstrapped standard errors equivalent to the non-federated implementation. We demonstrate this functionality on simulated data and real-world data from a malaria intervention in Mozambique. By leveraging federated estimates, we increase effective sample sizes leading to reduced estimation uncertainty, and enable estimation when single data owners cannot share the data but only have access to the treated or untreated group.

Introduction

Difference-in-Differences (DID) is a key tool to evaluate treatment effects in fields like finance [Molyneux et al., 2019, Nawaz et al., 2021], clinical research [Galiani et al., 2005, Miller and Wherry, 2017], epidemiology [Goodman-Bacon and Marcus, 2020, Groeniger et al., 2021], and economics [Colchero et al., 2016, Wen et al., 2015]. Recent advances have addressed continuous treatment effects [Callaway et al., 2021], variations in treatment timing [Goodman-Bacon, 2021, Callaway and Sant'Anna, 2021], and varying effects across multiple periods [Callaway and Sant'Anna, 2021]. The Callaway and Sant'Anna [2021] method is available in R [Callaway and Sant'Anna, 2021] and Stata [Rios-Avila et al., 2023], referred to as CSDID (Callaway and Sant'Anna Difference-in-Differences). CSDID has been applied to policy evaluations, such as interventions on teen births [Mark and Wu, 2022] and electric vehicle adoption in response to improved charging infrastructure [Schulz and Rode, 2022], but its use with sensitive individual-level data, like patient data or test performances, has been limited.

In cases involving sensitive individual-level data, sharing is restricted by legal frameworks like the General Data Protection Regulation (GDPR) [Hansen et al., 2021]. This restriction poses 3 major limitations to the analysis: (i) Obtaining consent from all potential participants is challenging yielding reduced sample sizes. (ii) Public health policies often vary regionally, and some entities, like schools or regional governments, may only have data for treated or untreated individuals. If this data cannot be shared, the CSDID cannot be used in these cases. (iii) Students with poorer performance may withhold data due to privacy concerns, introducing selection bias and limiting CSDID's utility with central learning (Figure 1A). Federated Learning [McMahan et al., 2017] addresses these limitations by enabling multiple data owners to collaborate while iteratively sharing only summary statistics, facilitating joint model training with larger sample sizes and preserving privacy (Figure 1B). It yields parameter estimates with convergence properties identical to central learning, often matching centrally stored data estimates. For CSDID, Federated Learning can overcome the selection bias and increase sample sizes yielding enhanced statistical power. Federated Learning has been successfully applied in various biomedical studies [Dayan et al., 2021, Harrison et al., 2020, Jannasch et al., 2022, Li et al., 2020], but its application to student test performance data has not been explored.

We derived a Federated Learning algorithm of the CSDID approach and developed a computational package in DataSHIELD. DataSHIELD [Marcon et al., 2021] is a well-established toolbox for Federated Learning, implemented in the statistical computing language R [R Core Team, 2022]. It is widely used in the biomedical community, including projects like the European Union's Horizon 2020 ORCHESTRA project and unCover [Tacconelli et al., 2022]. Although DataSHIELD supports many biomedical analysis tools, such as Survival Analysis [Banerjee et al., 2022] and deep Boltzmann Machines [Lenz et al., 2021], a tool for CSDID was previously missing. Our federated tool can obtain exact federated treatment effects, asymptotic standard errors, and distributional equivalent bootstrapped standard errors, comparable to the non-federated implementation [Callaway and Sant'Anna, 2021]. Our software incorporates standard DataSHIELD security measures, such as validity checks on the minimum non-zero counts of observational units and the maximum

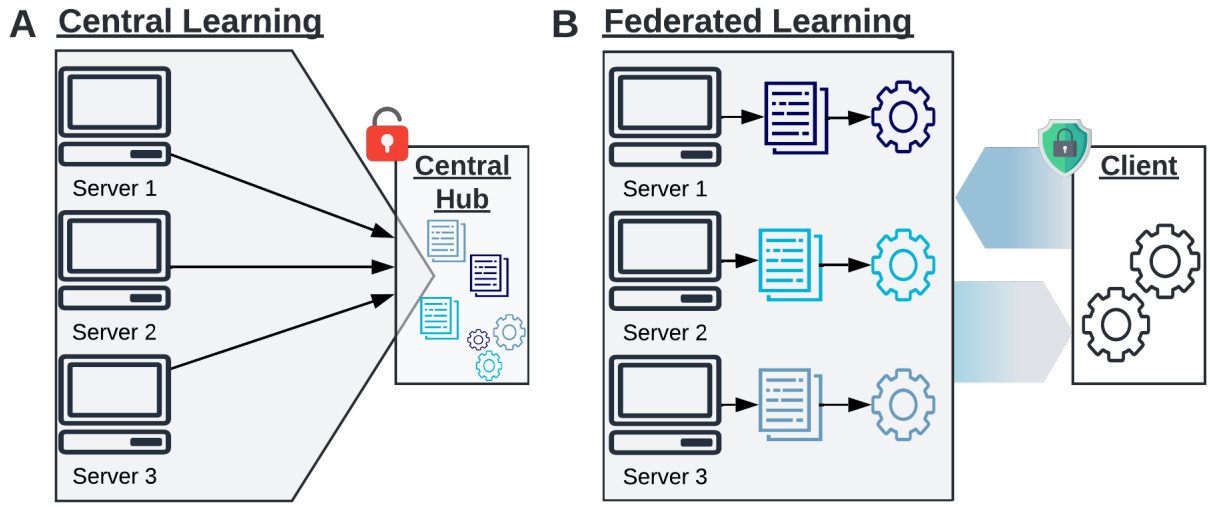

Figure 1: Learning paradigma. **A** Explanation of Central Learning. In Central Learning, models are trained at one central hub (analyst). All servers (data owners) send their data to the central hub, granting the analyst full access to individual-level information without preserving data privacy. **B** Explanation of Federated Learning. In Federated Learning, model updates are computed locally at the servers, and only aggregated information is sent to the analyst (client). The locally aggregated information is then further aggregated to compute overall model parameter updates, which are sent back to the servers. This iterative process continues until a convergence criterion is met, ensuring efficient and privacy-preserving collaboration between servers and the analyst. This figure has been designed using resources from Flaticon.com

number of parameters in a regression. Additionally, we have included further security measures tailored to CSDID to prevent attacks as described by Huth et al. [2023]. These measures ensure that computations are secure and confidential, protecting sensitive data from unauthorized access or use.

In the remainder of this paper, we outline the fundamentals of DataSHIELD, explain the federated algorithm, detail the additional security measures implemented in our software, and demonstrate its application through a simulation study. Additionally, we present a case study examining the impact of a malaria elimination initiative on school outcomes, previously analyzed in a non-federated setup [Cirera et al., 2022].

Results

DataSHIELD provides federated infrastructure

DataSHIELD [Gaye et al., 2014, Wilson et al., 2017] is an advanced distributed learning ecosystem that utilizes Federated Learning and federated meta-analysis to enable secure and collaborative data analysis across multiple data owners without sharing individual-level data. This approach maintains high statistical power by leveraging larger sample sizes while ensuring data privacy. DataSHIELD's federated algorithms operate through a client-server structure (Figure 1B): client-side packages run on the analyst's machine, and server-side packages are installed on the data owner's machines.

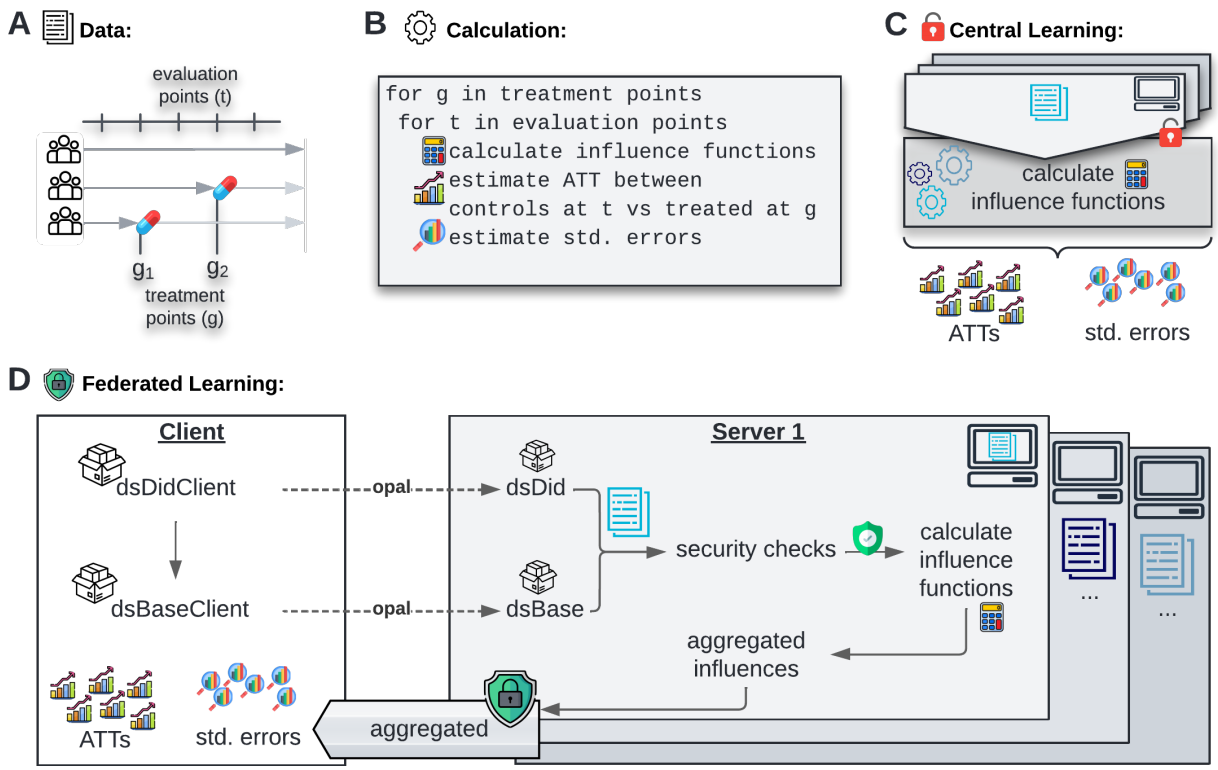

Figure 2: Federated implementation. **A** Visualization of data structure. The data has a panel structure with many observations per individual and varying treatment timing. **B** High-level algorithm of the CSDID. For each combination of evaluation period and treatment period, an ATT and its standard error is computed using the influence function of each individual. **C** CSDID for central learning. In central learning, the data is all at one server such that the sample analogues can be computed directly. **D** CSDID for Federated Learning. The analysis is initalized by using functions from the client side package dsDidClient which calls the dsDid package on the server sides using opal. During the local computations of the influence functions on the server sides, security checks are enforced in order to guarantee data privacy. The server-side influences are aggregated on each server and only the aggregated information is sent to the client side at which ATTs and standard errors are computed. This figure has been designed using resources from Flaticon.com

Server-side functions aggregate data locally, detect disclosure risks such as small sample sizes, and send non-disclosive information to the analyst to update parameters or compute overall aggregated statistics.

The dsBase package [Marcon et al., 2022] is the core of the DataSHIELD system, offering a wide range of functionalities, including summary statistics computation and generalized linear models. Additionally, user-written packages extend DataSHIELD's capabilities, such as restricted Boltzmann machines [Lenz et al., 2021], Omics analysis [Gonzalez et al., 2021], and mediation methods [Avraam and Wheater, 2021]. Our implementation in DataSHIELD enhances this framework by providing a robust tool for evaluating causal impact and treatment effects through the federated version of the CSDID. It is designed to minimize information transfer between the client side and server side, reducing both data leakage risks and communication overhead. Our software package consists of a client-side [Huth, 2023a] and a server-side [Huth, 2023b] R package. The server-side package can be accessed through the client-side package that uses Opal [OBiBa, 2022] for communication.

Privacy-preserving point estimates are obtained using federated averaging

We present the methodology for computing federated point estimates of the average treatment effect of the treated (ATT). For clarity and consistency, we adopt the notation from Callaway and Sant'Anna [2021].

A dataset suitable for the CSDID analysis comprises multiple evaluation periods t and treatment periods g (Figure 2A). An ATT and its standard error are computed for each combination of t and g (Figure 2B) using appropriate control and treatment groups. As the control group, we can use either never-treated or not-yet-treated individuals. While this choice affects the specific individuals used to compute the counterfactual, it does not alter the overall methodology. Therefore, we illustrate the algorithm using never-treated individuals as the control group. Within the package, there are three methods to compute the ATT: the doubly robust (DR) approach, inverse-probability weighting (IPW), and the outcome regression (OR) approach. However, since the IPW and OR approaches are nested within the DR approach, federations are analogous. Thus, detailing the federation of the DR approach suffices.

The non-federated DR ATT is defined as the expectation of the difference in outcome between the treated and the control group, weighted by the probability of receiving treatment. It is given, as described in [Callaway and Sant'Anna, 2021], by

$$ATT(g,t;\delta) = \mathbb{E}^{(1)}\left[\left(\frac{G_g}{\mathbb{E}^{(2)}[G_g]} - \frac{\frac{p_g(X)C}{1-p_g(X)}}{\mathbb{E}^{(3)}\left[\frac{p_g(X)C}{1-p_g(X)}\right]}\right)(Y_t - Y_{g-\delta-1} - m_{g,t,\delta}(X))\right], \tag{1}$$

where expectations are donated with a superscript for referencing purposes. Further variables are defined as follows: δ is the number of anticipation periods in which individuals know about the upcoming treatment, G_g is a binary variable indicating treatment status in period g, C is a binary variable indicating membership in the never-treated control group, Y_t is the outcome variable in period t, and X is a matrix of pre-treatment covariates. The term $m_{g,t,\delta}(X) = \mathbb{E}\left[Y_t - Y_{g-\delta-1}|X, C=1\right]$ is the expected difference in outcome of the control group and $p_g(X) = \mathbb{E}\left[G_g|X, G_g + C = 1\right]$ is the probability of receiving treatment in period g given pre-treatment covariates and treatment status.

To estimate the ATT, the expectations in (1) are replaced by their corresponding sample analogues, which we subsequently denote by hat variables. In the case of central learning [Callaway and Sant'Anna, 2021], the data is shared and stored centrally (Figure 2C) allowing all information $(Y_t, Y_{g-\delta-1}, X, G_g, C)$ to be available for directly computing the sample analogues of all expectations. However, in the federated setting, the data remains on the server sides to avoid sharing sensitive information. Initially, the servers (but not the client) have access to their individual data $(Y_t, Y_{g-\delta-1}, X, G_g, C)$. Thus, the sample analogues $\hat{\mathbb{E}}^{(2)}, \hat{\mathbb{E}}^{(3)}, \hat{p}_g(X)$, and $\hat{m}_{g,t,\delta}(X)$ can be computed by the client using federated means and federated generalized linear models (GLMs) (Figure 2D) via dsBase. For the federated means, only server-aggregated means need to be shared and for the federated GLMs iterative gradient sharing is used to obtain GLM parameter estimates. These parameters and means are then sent to the servers to compute the sample analogue of the term inside

the expectation of $\hat{\mathbb{E}}^{(1)}$ locally. These local influences can be aggregated using a federated mean in order to compute $\hat{\mathbb{E}}^{(1)}$ on the client-side. A more detailed description of the computation of the federated sample analogues, federated influence functions as well as a description of the federated standard error algorithms is given in the Method Details section.

Additional privacy-enhancing measures increase security

Our package adheres to DataSHIELD's data security standards and the highest security standards as per recent assessments, ensuring all function outputs comply with the disclosure settings configured in Opal [OBiBa, 2022]. These measures include, inter alia, preventing the subsetting of data frames if the subset does not meet the minimum row requirement and prohibiting aggregations with fewer objects than the specified threshold. Given that CSDID subsets individuals for each treatment period and a fixed evaluation period (Figure 2B), servers with low sample sizes may not meet DataSHIELD's security requirements and, therefore, cannot participate in computing the ATT for those periods. To address this, our package automatically checks in advance which servers have sufficient observations for the respective periods and excludes only those servers for the respective computations. This ensures that treatment effect estimations use larger sample sizes from the remaining servers, improving accuracy and efficiency while maintaining data security and privacy.

Additionally, when performing federated estimation of the influence functions, it is necessary to append columns to a matrix (`ds.AppendInfluence`). However, this process enables malicious clients to append arbitrary columns to a matrix and therefore the creation of known linearly independent matrices which can cause data leakage of all data points [Huth et al., 2023]. To mitigate this risk, we implement a strategy of row-wise shuffling of data frames and matrices after appending columns. This maintains the integrity of results through ID column matching while protecting against data leakage attacks.

Using globally summarized quantities, such as global means, on the server sides requires data transmission from the client. However, unrestricted data transmission can facilitate data leakage attacks. As a countermeasure, we process all data sent to the server immediately, ensuring it is not stored. For data that must be stored, the client is limited to sending a single number at a time to prevent attacks. Sending known numbers to the servers raises the potential threat of data leakage if the client is able to concatenate multiple known single numbers into different linearly independent vectors [Huth et al., 2023]. Therefore, it is not recommended to use this package with `ds.cbind` and `ds.rbind` from DataSHIELD's base package [Marcon et al., 2022], which can be disabled in Opal to enhance data privacy.

The Method Details section provides further details on the specific functions and the security measures implemented to protect the data privacy.

Federated estimates preserve privacy and equal central learning estimates

To validate our package, we used a simulated dataset generated with the non-federated R package and compared our federated implementation to the central learning. Specifically, we compared point estimates of the ATT, asymptotic standard errors, and bootstrapped standard errors. Additionally, we assessed the reduction of uncertainty achieved by the federated model compared to study-level analysis, where each data owner analyzes only their local dataset.

We simulated a dataset of 801 individuals with observations at four time periods, randomly allocated across six servers (Figure 3). Three servers contained 134 individuals (536 observations each), and three contained 133 individuals (532 observations each). Individuals were either never treated or treated in period two or three. All observations for an individual were stored on one server, reflecting the realistic scenario where different data providers, such as hospitals or schools, do not share data on the same individual. Our main results used the DR estimator with not-yet-treated individuals as the control group. Qualitatively equivalent results for the IPW and OR methods and other control groups are presented in the Supplementary Material.

The federated package successfully reproduced the average treatment effect (Figure 3B) and asymptotic standard errors (Figure 3C) up to numerical precision. For the federated bootstrapped standard errors, we relied on visual inspection to evaluate empirical validity, as statistical tests could only reject the hypothesis of equal distributions, not that they are unequal. Detailed reasoning for this limitation is provided in the Method Details. We used a Monte Carlo simulation, repeating the bootstrap computation 500 times for both federated and central settings. Empirical distributions were generated, and the difference between two central learning distributions served as a benchmark for comparison. Violin plots and boxplots (Figure 3D) indicated that the quantiles of all distributions are visually almost identical, and the densities show the same shapes, providing strong visual evidence that the federated and central learning bootstrapped standard errors are distributional equivalent.

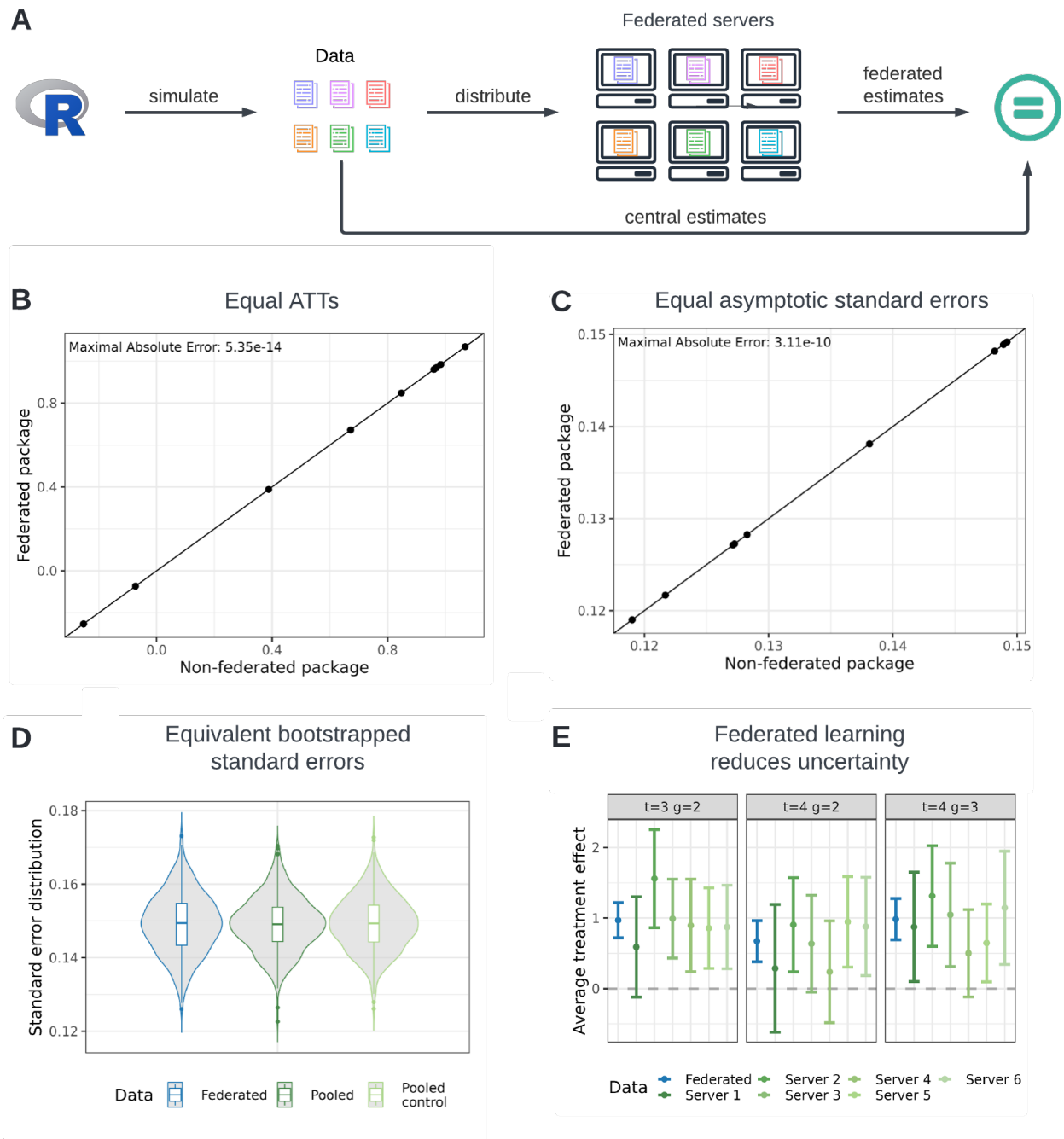

Figure 3: Similarity of Federated and Central Estimation for the DR estimate with not-yet-treated individuals as the control group. **A** Simulation setup. The federated setup involved 6 servers. Three servers had 134 (536) individuals (observations), and three had 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three, with all observations of an individual on one server. **B** Equality of central and federated point estimates. The x-axis shows central (non-federated) estimates [Callaway and Sant'Anna, 2021], while the y-axis shows federated estimates. The 45° line indicates equal results when aligned. **C** Equality of central and federated asymptotic standard errors. The x-axis shows central asymptotic standard errors, and the y-axis shows federated standard errors. The 45° line is shown for reference. **D** Comparison of bootstrapped standard errors. Boxplots and densities compare the distribution of federated (blue) and central (green) bootstrapped standard errors. Two central learning distributions establish reference differences, with 500 bootstrapped standard errors analyzed. **E** Treatment effect estimates. Point estimates (dots) and 95% confidence intervals for post-treatment periods are shown. Federated package estimates are in blue; estimates from one server are in green. This figure has been designed using resources from Flaticon.com

We also examined the impact of the federated implementation on the estimation of the overall treatment effect compared to study-level analysis. By observing the point estimates and 95% confidence intervals, with standard errors obtained from the multiplier bootstrap (Figure 3E), we found: (i) study-level point estimates fluctuated around the federated and central learning point estimate, and (ii) increased uncertainty in the study-level DR estimates was reflected in wider confidence intervals. The study-level point estimates fluctuated around the federated estimate since the federated simulated data is independent and identically distributed (*i.i.d.*) in individuals and we allocated the simulated individuals randomly to the servers. Therefore, the server data is also *i.i.d*, and thus, the point estimates of the study-level analysis fluctuate around the federated estimate. This *i.i.d.* case is the most favorable scenario for study-level analysis; point estimates will deviate more from the federated estimate in non-*i.i.d.* settings. The increased size of the confidence intervals reflects reduced sample sizes, resulting in more uncertainty and potentially failing to reject the null hypothesis of no treatment effect in study-level cases where it could be rejected in the federated case. The federated setup yields estimates with lower uncertainty, enhancing the detection of true treatment effects.

Federated learning enables CSDID analysis in cases when central learning fails

To test the effectiveness of our federated framework in a real-world scenario, we studied the impact of malaria interventions on school outcomes in Mozambique. In 2015, a malaria elimination initiative was implemented in Magude (Figure 4A, B) as a quasi-experiment to estimate its impact on selected primary school outcomes. Data was collected from nine schools in two districts of Mozambique. This data was initially analyzed using a regression DID approach [Cirera et al., 2022], which required data sharing agreements for data transfer to analysts. For our analysis, we conducted a complete case analysis with the CSDID, including data from 1,044 individuals with observations available for the three school terms of 2015 and 2016 (Figure 4B). Sample statistics for each school are provided in Table 2 in Method Details.

In our federated setup, each of the nine schools hosted their own server, simulating that student data never left the school premises (Figure 4A). Consequently, the schools in Magude (four schools) only had data on treated individuals from 2015, while the schools in Manhiça (five schools) only had data on control group individuals. This separation would make it impossible to estimate non-federated local CSDID average treatment effects without sharing the data with a central hub.

Inspecting the unconditional means, computed in a federated manner, we observed that before the malaria elimination initiative, Magude (the treated district) had a lower grade average compared to Manhiça (Figure 4C). After the initiative, student performance in Magude improved significantly, whereas the improvement in Manhiça was comparatively moderate. To test the hypothesis of the malaria intervention's effect, we ran the federated CSDID with age and gender as covariates to control for potential confounders (Figure 4D). The results revealed significant differences (at the 95% confidence level) in the first and third terms of 2016, while the estimate for the second term

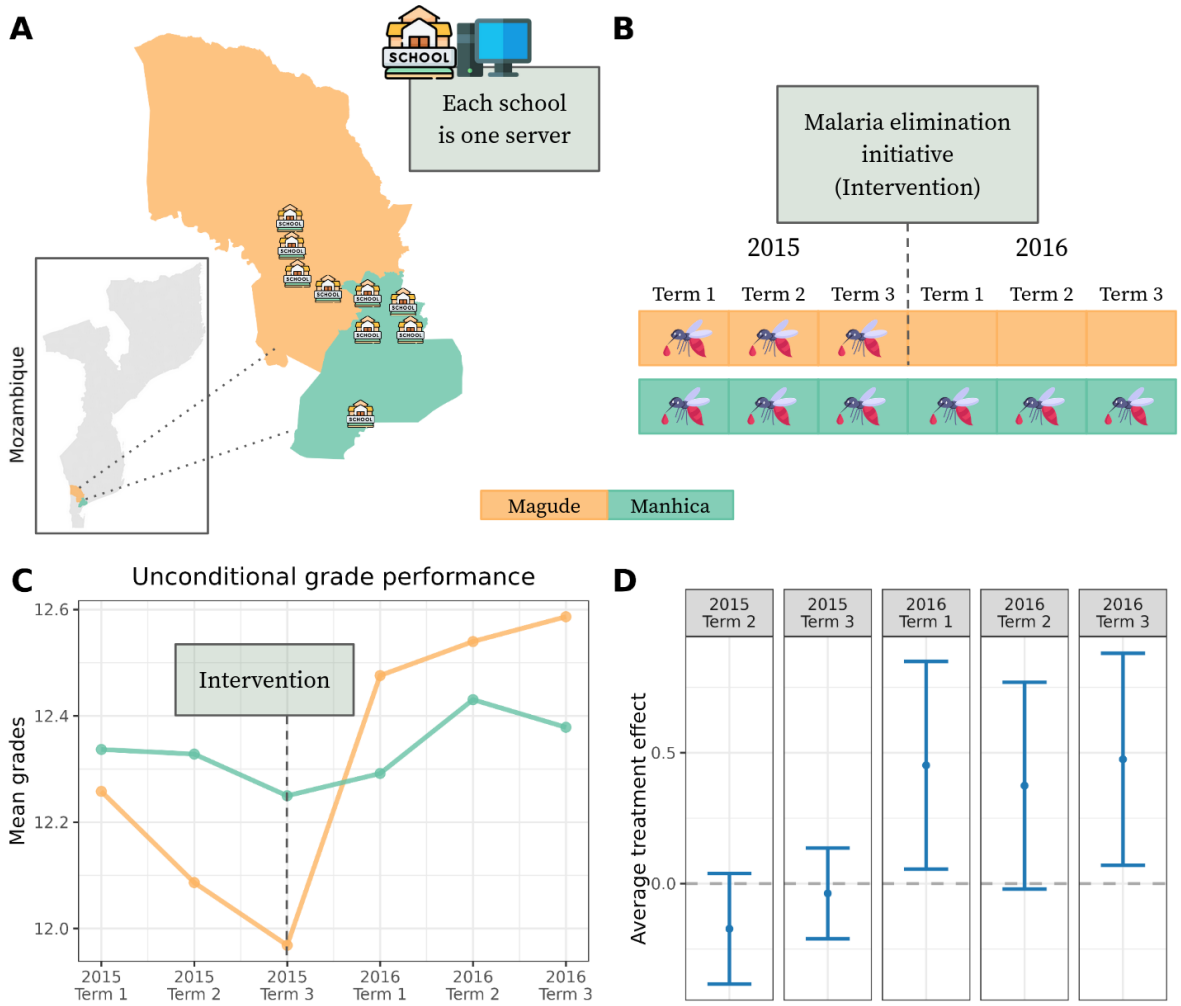

Figure 4: Federated learning enables estimation of DR estimates. **A** The districts of Magude and Manhiça in Mozambique and the location of the 9 schools. **B** The timeline of the malaria elimination initiative that was launched in Magude before the term 1 in 2016. **C** Unconditional means of the mean grades in Magude and Manhiça respectively. The means are estimated as sample averages over the observations (mean grades of students) of the individuals in the respective districts at the respective time. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated federated treatment effects for all periods. This figure has been designed using resources from Flaticon.com

was close to the boundary of significance.

Discussions

The use of Federated Learning has emerged as a promising approach to increase sample sizes when dealing with sensitive data. However, the availability of tools for causal analysis within this context has been limited. To address this gap and make such methods accessible to the research community, we developed a federated version of the Difference-in-Differences estimator proposed by Callaway and Sant'Anna [2021] and implemented it in the Federated Learning platform DataSHIELD us-

ing the statistical programming language R. Our federated package covers many functionalities of the CSDID R package [Callaway and Sant'Anna, 2021] and additionally allows for the estimation of treatment effects when treatment and control groups are only available mutually exclusively to single data owners. Our results demonstrate that the proposed federated version of the CSDID estimator can be implemented while preserving the original estimates, asymptotic standard errors, and distributional equivalent bootstrapped standard errors. Additionally, Federated Learning reduces uncertainty and allows estimation in scenarios where treated and untreated individuals are separated across data owners. We have ensured that standard DataSHIELD security measures are in place to protect data privacy and have further enhanced security with additional measures.

We specifically chose the Callaway and Sant'Anna estimator due to its high relevance and flexibility. This estimator accommodates multiple time periods for treatments and evaluations, incorporates different types of control groups, and offers various estimators and simultaneous confidence bands via bootstrapped standard errors. Moreover, it requires the parallel trend assumption to hold only after conditioning on covariates, relaxing the assumption of the classical regression DID where the parallel trend must hold unconditionally. Future work on federated causal impact analysis could explore estimating treatment effects from repeated cross-sections in the context of CSDID, as implemented in Callaway and Sant'Anna [2021], or synthetic control methods that have gained traction in recent years [Abadie et al., 2010, 2015].

Our federated version of the CSDID fills a gap in the availability of tools for causal analysis in Federated Learning. Our results highlight the potential of using Federated Learning for privacy-preserving causal analysis in settings with sensitive data, while maintaining high levels of data security through standard and additional measures implemented within DataSHIELD.

Limitations of the study

This study has two main limitations. The first significant limitation, shared with the non-federated method, is the necessity to bin the time points of individual records into discrete events to apply the CSDID. This binning process can result in the loss of precise information regarding treatment timings, especially when dealing with data from non-equidistant observations. However, the federated approach allows for the incorporation of more data points, which can potentially lead to a finer granularity in event discretization. Secondly, the use of this software package requires the establishment of a client-server infrastructure. This setup can be challenging for users without extensive technical experience. Nevertheless, the DataSHIELD platform provides comprehensive tutorials and community support, which can assist users in overcoming these technical hurdles and adapting to the process more efficiently.

Abbreviations

ATT Average treatment effect of the treated
CSDID Callaway and Sant'Anna Difference-in-Differences
DID Difference-in-Differences
DR Doubly Robust
IPW Inverse probability weighting
RO Regression Outcome
SE Standard error

Acknowledgements

This study was funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under Germany's Excellence Strategy (EXC 2047 - 390685813 & EXC 2151 - 390873048) and under the project IDs 432325352 – SFB 1454 and 458597554 – SEPAN, the University of Bonn (via the Schlegel Professorship of JH), the Helmholtz Association - Munich School for Data Science (MUDS), and the ORCHESTRA project. The ORCHESTRA project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 101016167. The views expressed in this paper are the sole responsibility of the authors and the Commission is not responsible for any use that may be made of the information it contains. The funders had no role in the study design, data collection, data analyses, data interpretation, writing, or submission of this manuscript.

The icons for the figures were created by Freepik, Flat Icons, Roundicons, Smashicons, and smashingstocks from www.flaticon.com.

Authors contributions

M.H. developed the federation of the algorithms. M.H. and C.G, implemented the software packages in R. L.C., E.S., F.S. collected the data for the study in Mozambique. M.H. and L.S. visualized the results. J.H and E.S. conceptualized the study. M.H., E.S. and J.H. wrote the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT4 in order to improve the readability and language of the manuscript. After using ChatGPT4, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

A. Abadie, A. Diamond, and J. Hainmueller. Synthetic control methods for comparative case studies: Estimating the effect of california's tobacco control program. *Journal of the American statistical Association*, 105(490):493–505, 2010.

A. Abadie, A. Diamond, and J. Hainmueller. Comparative politics and the synthetic control method. *American Journal of Political Science*, 59(2):495–510, 2015.

D. Avraam and S. Wheater. *dsMediation: Methods to apply causal mediation analysis*, 2021. URL `https://github.com/datashield/dsMediation`. R package version 0.0.3.

S. Banerjee, G. N. Sofack, T. Papakonstantinou, D. Avraam, P. Burton, D. Zöller, and T. R. Bishop. dssurvival: Privacy preserving survival models for federated individual patient meta-analysis in DataSHIELD. *BMC Research Notes*, 15(1):197, 2022.

B. Callaway and P. H. Sant'Anna. did: Difference in differences, 2021. URL `https://bcallaway11.github.io/did/`. R package version 2.1.2.

B. Callaway and P. H. Sant'Anna. Difference-in-differences with multiple time periods. *Journal of Econometrics*, 225(2):200–230, 2021.

B. Callaway, A. Goodman-Bacon, and P. H. Sant'Anna. Difference-in-differences with a continuous treatment. *arXiv preprint arXiv:2107.02637*, 2021.

L. Cirera, J. V. Castelló, J. Brew, F. Saúte, and E. Sicuri. The impact of a malaria elimination initiative on school outcomes: Evidence from southern mozambique. *Economics & Human Biology*, 44:101100, 2022.

M. A. Colchero, B. M. Popkin, J. A. Rivera, and S. W. Ng. Beverage purchases from stores in mexico under the excise tax on sugar sweetened beverages: observational study. *bmj*, 352, 2016.

I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili, A. Z. Abidin, A. Liu, A. B. Costa, B. J. Wood, C.-S. Tsai, et al. Federated learning for predicting clinical outcomes in patients with COVID-19. *Nature Medicine*, 27(10):1735–1743, 2021.

S. Galiani, P. Gertler, and E. Schargrodsky. Water for life: The impact of the privatization of water services on child mortality. *Journal of political economy*, 113(1):83–120, 2005.

A. Gaye, Y. Marcon, J. Isaeva, P. LaFlamme, A. Turner, E. M. Jones, J. Minion, A. W. Boyd, C. J. Newby, M.-L. Nuotio, et al. Datashield: taking the analysis to the data, not the data to the analysis. *International journal of epidemiology*, 43(6):1929–1944, 2014.

J. R. Gonzalez, Y. Marcon, and X. Esriba-Montagut. *dsOmics: DataSHIELD Omic functions*, 2021. URL `https://github.com/isglobal-brge/dsOmics`. R package version 1.0.18.

A. Goodman-Bacon. Difference-in-differences with variation in treatment timing. *Journal of Econometrics*, 225(2):254–277, 2021.

A. Goodman-Bacon and J. Marcus. Using difference-in-differences to identify causal effects of covid-19 policies. 2020.

J. O. Groeniger, K. Noordzij, J. Van Der Waal, and W. De Koster. Dutch covid-19 lockdown measures increased trust in government and trust in science: A difference-in-differences analysis. *Social Science & Medicine*, 275:113819, 2021.

J. Hansen, P. Wilson, E. Verhoeven, M. Kroneman, M. Kirwan, R. Verheij, and E.-B. van Veen. Assessment of the EU member states' rules on health data in the light of GDPR, 2021. URL `doi:10.2818/546193`.

S. L. Harrison, E. Fazio-Eynullayeva, D. A. Lane, P. Underhill, and G. Y. Lip. Comorbidities associated with mortality in 31,461 adults with covid-19 in the united states: A federated electronic medical record analysis. *PLoS Medicine*, 17(9):e1003321, 2020.

M. Huth. dsdidclient, 2023a. URL `https://github.com/manuhuth/dsDidClient.git`. R package.

M. Huth. dsdid, 2023b. URL `https://github.com/manuhuth/dsDid.git`. R package.

M. Huth, J. Arruda, R. Gusinow, L. Contento, E. Tacconelli, and J. Hasenauer. Accessibility of covariance information creates vulnerability in federated learning frameworks. *Bioinformatics*, 2023.

F. Jannasch, S. Dietrich, T. R. Bishop, M. Pearce, A. Fanidi, G. O'Donoghue, D. O'Gorman, P. Marques-Vidal, P. Vollenweider, M. Bes-Rastrollo, et al. Associations between exploratory dietary patterns and incident type 2 diabetes: A federated meta-analysis of individual participant data from 25 cohort studies. *European Journal of Nutrition*, pages 1–19, 2022.

S. Lenz, M. Hess, and H. Binder. Deep generative models in datashield. *BMC Medical Research Methodology*, 21(1):1–16, 2021.

X. Li, Y. Gu, N. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan. Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results. *Medical Image Analysis*, 65:101765, 2020.

Y. Marcon, T. Bishop, D. Avraam, X. Escriba-Montagut, P. Ryser-Welch, S. Wheater, P. Burton, and J. R. González. Orchestrating privacy-protected big data analyses of data from different resources with R and DataSHIELD. *PLoS Computational Biology*, 17(3):e1008880, 2021.

Y. Marcon, A. Gaye, , J. Isaeva, P. LaFlamme, A. Turner, E. M. Jones, J. Minion, A. W. Boyd, C. J. Newby, M.-L. Nuotio, et al. dsbase, 2022. URL `https://github.com/datashield/dsBase.git`. R package.

N. D. Mark and L. L. Wu. More comprehensive sex education reduced teen births: Quasi-experimental evidence. *Proceedings of the National Academy of Sciences*, 119(8):e2113144119, 2022.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

S. Miller and L. R. Wherry. Health and access to care during the first 2 years of the aca medicaid expansions. *New England Journal of Medicine*, 376(10):947–956, 2017.

P. Molyneux, A. Reghezza, and R. Xie. Bank margins and profits in a world of negative rates. *Journal of Banking & Finance*, 107:105613, 2019.

M. A. Nawaz, U. Seshadri, P. Kumar, R. Aqdas, A. K. Patwary, and M. Riaz. Nexus between green finance and climate change mitigation in n-11 and brics countries: empirical estimation through difference in differences (did) approach. *Environmental Science and Pollution Research*, 28:6504–6519, 2021.

OBiBa. Opal, 2022. URL `https://opaldoc.obiba.org/en/latest/`.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022. URL `https://www.R-project.org/`.

F. Rios-Avila, P. Sant'Anna, and B. Callaway. Csdid: Stata module for the estimation of difference-in-difference models with multiple time periods, 2023. URL `https://EconPapers.repec.org/RePEc:boc:bocode:s458976`.

F. Schulz and J. Rode. Public charging infrastructure and electric vehicles in norway. *Energy Policy*, 160:112660, 2022.

E. Tacconelli, A. Gorska, E. Carrara, R. J. Davis, M. Bonten, A. W. Friedrich, C. Glasner, H. Goossens, J. Hasenauer, J. M. H. Abad, et al. Challenges of data sharing in European COVID-19 projects: A learning opportunity for advancing pandemic preparedness and response. *The Lancet Regional Health-Europe*, 21:100467, 2022.

H. Wen, J. M. Hockenberry, and J. R. Cummings. The effect of medical marijuana laws on adolescent and adult use of marijuana, alcohol, and other substances. *Journal of health economics*, 42:64–80, 2015.

R. Wilson, O. Butters, D. Avraam, J. Baker, J. Tedds, A. Turner, M. Murtagh, and P. Burton. Datashield–new directions and dimensions. *Data Science Journal*, 16, 2017.

Star methods

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Code for mathematical modeling, analysis and visualization used in the manuscript	This paper	https://zenodo.org/doi/10.5281/zenodo.11565570
Software and algorithms		
R version 4.3.2	R Core Team	https://www.r-project.org/
dsDiDClient (R package)	Github	https://github.com/manuhuth/dsDidClient
dsDiD (R package)	Github	https://github.com/manuhuth/dsDid
dsBaseClient (R package)	Github	https://github.com/datashield/dsBaseClient
dsBase (R package)	Github	https://github.com/datashield/dsBase

Resource availability

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Jan Hasenauer (jan.hasenauer@uni-bonn.de)

Materials availability

This study did not generate new unique reagents.

Data and code availability

- This paper analyses data in the example of the CSDID. Use of the Mozambique school data requires a data sharing agreement between your institution and the Centro de Investigação em Saúde de Manhiça as it contains privacy sensitive information. All of the data used for the simulation study will is maid available via Zenodo (see STAR methods).
- All original code has been deposited at GitHub and Zenodo and is publicly available (see STAR methods).
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

Method Details

List of DataSHIELD client-side functions

We list all functions that are used within the DataSHIELD framework for the client-side [Huth, 2023a] and explain their functionalities. The main function is `ds.did` which computes treatment effects and standard errors and calls the other client-side functions while being executed.

Table 1: New DataSHIELD functions implemented in our software package. Functions starting with *ds.* are client-side functions and can be called via the dsDidClient [Huth, 2023a] package.

Client side function	**Description**
`ds.did`	Calculates the average treatment effect of the treated, standard errors, and pre-tests for the parallel trend assumption.
`ds.addColumnOnes`	Adds a column of ones to a dataframe on the server side.
`ds.appendInfluence`	Appends influence to a specified column for given ids and periods in a dataframe on the server side.
`ds.computeMatrixCrossproduct`	Computes cross product of a matrix on the server side.
`ds.computeOdds`	Computes odds using propensity scores for a given dataset and outcome variable.
`ds.createEmptyIdMatrix`	Creates empty matrix with unique identifiers from a dataframe on the server side.
`ds.enoughIndividuals`	Checks if enough individuals in a dataframe have a specified value in a specified column on the server side.
`ds.generateNotYetTreated`	Generates a dataframe of untreated individuals based on given variables on the server side.
`ds.multiplierBootstrap`	Performs multiplier bootstrap on a given influence matrix on the server side.
`ds.multiplyMatrixMatrix`	Multiplies two matrices and assigns result to a new object on the server side.
`ds.multiplyMatrixScalar`	Multiplies a matrix and a scalar and assigns result to a new object on the server side.
`ds.recode`	Creates a new object and replaces entries of a vector.
`ds.sendToServer`	Sends a single number to the server side.
`ds.subsetDf`	Subsets a dataframe based on a specified variable and value.

Federated sample analogues for point estimates

In more detail, the computations are

1. $\hat{\mathbb{E}}^{(2)}(\cdot)$: The client computes the sample mean of the variable G_g indicating if an individual has been treated in period g using a federated mean (`ds.mean` from dsBaseClient) . The aggregate mean is then sent back to the servers (`ds.SendToServer`) for further analysis and processing.

2. $\hat{p}_g(x)$: The client computes the parameter estimates (`ds.glm` from dsBaseClient), denoted by $\hat{\pi}$, of a logistic regression model using a federated generalized linear model with the individuals that have been treated in period g or are never treated

$$p_g(x) = \frac{1}{1 + \exp\left(-\pi_0 - x^T \pi\right)} \tag{2}$$

   The estimated parameters, $\hat{\pi}$, are then sent to the servers where they are immediately processed to compute the estimated probability of treatment, $\hat{p}_g(X)$, using (2) (`ds.genProp`). If no covariates are used, the probability of treatment can be directly computed using a federated mean of the variable indicating treatment G_g (`ds.mean`).

3. $\hat{\mathbb{E}}^{(3)}(\cdot)$: With the knowledge of the estimated probability of treatment, $\hat{p}_g(X)$, from step 2 and the binary variable indicating if the individual is part of the never-treated group, C, the mean of the expected value, $\hat{\mathbb{E}}^{(3)}(\cdot)$, can be computed using a federated mean (`ds.mean`).

4. $\hat{m}_{g,t,\delta}$: The client computes the parameter estimates $\hat{\beta}$ (`ds.glm`) of a linear regression model using the individuals that have never been treated and the ones that are treated in g. The linear regression model is represented by

$$m_{g,t,\delta}(x) = \beta_0 + x^T \beta, \tag{3}$$

   which relates the expected difference in the outcome of the control group and the covariates X. Subsequently, the client sends the estimated parameters $\hat{\beta}$ to the servers, where they are immediately processed to compute $\hat{m}_{g,t,\delta}$ by (3) (`ds.multiplyMatrixMatrix`). If no covariates are used, the client can directly compute $\hat{m}_{g,t,\delta}$ by taking the federated mean of $Y_t - Y_{g-\delta-1}$ (`ds.mean`).

5. $\hat{\mathbb{E}}^{(1)}(\cdot)$: Finally, the client computes the expectation $\hat{\mathbb{E}}^{(1)}(\cdot)$ by using a federated mean (`ds.mean`) of all the known quantities and parameters. This expectation represents the point estimate of the average treatment effect of the treated $\widehat{ATT}(g, t; \delta)$.

Federated computations of influence functions

The influence function, denoted as $\psi_{g,t,\delta}(W_i, \beta_{g,t,\delta})$, measures the impact that each individual has on the estimated treatment effect $\widehat{ATT}_{t \geq g-\delta}$. The collection of these influence functions $\widehat{ATT}_{t \geq g-\delta}$

is then used to compute (approximate) the variance of $\widehat{ATT}_{t\geq g-\delta}$ through the multiplier bootstrap method or asymptotic standard errors. In this section, we demonstrate the federation of the influence function for the DR estimate using the never-treated individuals as the control group. To maintain consistency with the notation used in [Callaway and Sant'Anna, 2021], we adhere closely to the notation used in that paper throughout this section. It is also worth noting that, following [Callaway and Sant'Anna, 2021], the parameters β, π used to estimate $m_{g,t,\delta}(X), p_g(X)$ are included in this section.

The theoretical influence function $\psi_{g,t,\delta}(W_i, \beta_{g,t,\delta})$, as defined in [Callaway and Sant'Anna, 2021],

$$\psi_{g,t,\delta}(W_i, \beta_{g,t,\delta}) = \psi^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta}) - \psi^{comp}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g) - \psi^{est}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g). \tag{4}$$

quantifies the effect of each individual on the treatment estimate $\widehat{ATT}_{t\geq g-\delta}$. We will show in the following how this influence function can be estimated on the server sides in a federated manner using sample analogues, denoted by hat variables.

The components of the estimated influence function are given by [Callaway and Sant'Anna, 2021]

$$\begin{aligned}
\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta}) &= \widehat{w_g}^{treat}(\beta_{g,t,\delta})(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \\
&\quad - \widehat{w_g}^{treat}(\beta_{g,t,\delta})\hat{\mathbb{E}}\left(\widehat{w_g}^{treat}(\beta_{g,t,\delta})(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta}))\right) \\
\widehat{\psi}^{comp}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g) &= \widehat{w_g}^{comp}(\pi_g)\left[(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta}))\right. \\
&\quad - \widehat{w_g}^{comp}(\pi_g)\hat{\mathbb{E}}(\widehat{w_g}^{comp}(\pi_g)(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta}))) \\
\widehat{\psi}^{est}_{g,t}(W_i, \beta_{g,t,\delta}) &= \hat{l}^{or}_{g,t}(\beta_{g,t,\delta})^T\ \hat{M}^1_{g,\delta} + \hat{l}^{ps}_g(\beta_{g,t,\delta})^T\ \hat{M}^2_{g,\delta},
\end{aligned} \tag{5}$$

where W_i is the collection of all individual information about Y, X, and C. The weights are defined as $\widehat{w_g}^{treat}(\beta_{g,t,\delta}) = \frac{G_g}{\hat{\mathbb{E}}(G_g)}$ and $\widehat{w_g}^{comp}(\pi_g) = \frac{p_g(X;\pi_g)C}{1-p_g(X;\pi_g)} / \hat{\mathbb{E}}\left(\frac{p_g(X;\pi_g)C}{1-p_g(X;\pi_g)}\right)$ and the asymptotic linear representations of the regression estimator for the outcome evaluation of the comparison group and the generalised propensity score are

$$\begin{aligned}
\hat{l}^{or}_{g,t} &= \left[(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \circ X\right](X^T X)^{-1} \\
\hat{l}^{ps}_g &= \left[(G_g - p_g(X; \pi_g)) \circ X\right] H^{ps}_g,
\end{aligned} \tag{6}$$

where $a \circ B$ defines a columnwise Hadarmad product between the vector $a \in \mathbb{R}^n$ and the matrix $B \in \mathbb{R}^{n\times k}$. H^{ps}_g is the hessian obtained from the regression on $p_g(X; \pi_g)$.

$$\begin{aligned}
\hat{M}^1_{g,\delta} &= \hat{\mathbb{E}}\left[(\widehat{w_g}^{treat}(\beta_{g,t,\delta}) - \widehat{w_g}^{comp}(\pi_g)) \circ X\right] \\
\hat{M}^2_{g,\delta} &= \hat{\mathbb{E}}\left[\widehat{w_g}^{comp}(\pi_g)\,(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \circ X\right] \\
&\quad - \hat{\mathbb{E}}\left[(\widehat{w_g}^{comp}(\pi_g))^2\,(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \circ X\right]
\end{aligned} \tag{7}$$

where π_g is the analogue of π in equation (2) and $\beta_{g,t,\delta}$ is the analogue of β in equation (3).

To compute the influence function in a federated manner, the sample analogues of the theoretical influence function must be computed on the client side. Specifically, the client must compute $\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta})$, $\widehat{\psi}^{comp}_{g,t,\delta}(W_i, \beta g, t, \delta, \pi_g)$, and $\widehat{\psi}^{est}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g)$ using the individual data available on the servers. These sample analogues will be used to estimate the noise-corrupted influence function $\psi^*_{g,t,\delta}(W_i, \beta_{g,t,\delta})$.

1. $\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta})$: $Y_t, Y_{g-\delta-1}, \widehat{w_g}^{treat}(\beta_{g,t,\delta})$, and $\hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})$ are known from the federated point estimate computations on the server side.
   $\widehat{w_g}^{treat}(\beta_{g,t,\delta})(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta}))$ can therefore be computed on the client side (`ds.make` from dsBaseClient). Subsequently, the federated expectation can be returned to the client side (`ds.mean`) and be send back to the server side (`ds.SendToServer`). Finally, $\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta})$ can be computed on the server side (`ds.make`).

2. $\widehat{\psi}^{comp}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g)$: Analogously to $\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta})$ using $\widehat{w_g}^{comp}(\pi_g)$ instead of $\widehat{w_g}^{treat}(\beta_{g,t,\delta})$.

3. $\widehat{\psi}^{est}_{g,t}(W_i, \beta_{g,t,\delta})$: $\hat{l}^{or}_{g,t}, \hat{l}^{ps}_g, \hat{M}^1_{g,\delta}$, and $\hat{M}^2_{g,\delta}$ are unknown to the server side after the federated point estimate computations.

   (a) $\hat{l}^{or}_{g,t}$: $Y_t, Y_{g-\delta-1}, \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})$, and X are known from the federated point estimate computations on the server side. $(X^T X)$ can be computed via a federated aggregation (`ds.computeMatrixCrossproduct`) and be send to the client side, where its inverse $(X^T X)^{-1}$ is computed and send to the servers. Subsequently, $\hat{l}^{or}_{g,t}$ can be computed on the server side (`ds.multiplyMatrixMatrix`).

   (b) $\hat{M}^1_{g,\delta}$: $\widehat{w_g}^{treat}(\beta_{g,t,\delta}), \beta_{g,t,\delta}, X$ and $\widehat{w_g}^{comp}(\pi_g))$, are known to the server side from the federated point estimate computations.
   Such that $(\widehat{w_g}^{treat}(\beta_{g,t,\delta}) - \widehat{w_g}^{comp}(\pi_g)) \circ X)$ can be computed on the server side (`ds.make`) and the expectation can be estimated on the client side using a federated mean (`ds.mean`). The federated mean is subsequently send back to the servers (`ds.sendToServer`).

   (c) $\hat{l}^{ps}_g$: $Y_t, Y_{g-\delta-1}, \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})$, and X are known from the federated point estimate computations on the server side such that $G_g - (p_g(X; \pi_g)) \circ X$ can be computed on the server side (`ds.make`). (H^{ps}_g) can be send to the server side where it is immediately multiplied to compute $\hat{l}^{ps}_g$ on the server side (`ds.multiplyMatrixMatrix`).

   (d) $\hat{M}^2_{g,\delta}$: $\widehat{w_g}^{comp}(\pi_g), Y_t, Y_{g-\delta-1}, \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})$ and X, are known to the server side from the federated point estimate computations.
   $\widehat{w_g}^{comp}(\pi_g)\,(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \circ X$ and
   $(\widehat{w_g}^{comp}(\pi_g))^2\,(Y_t - Y_{g-\delta-1} - \hat{m}_{g,t,\delta}(X; \beta_{g,t,\delta})) \circ X$ can therefore be computed on the server side (`ds.make`). The expectations can be returned to the client side using a federated mean (`ds.mean`). The means are subsequently send back to the servers (`ds.sendToServer`).

   $\hat{l}^{or}_{g,t}, \hat{l}^{ps}_g, \hat{M}^1_{g,\delta}$ and $\hat{M}^2_{g,\delta}$ are now known to the server side. $\hat{l}^{or}_{g,t}(\beta_{g,t,\delta})^T\ \hat{M}^1_{g,\delta}$ and $\hat{l}^{ps}_g(\beta_{g,t,\delta})^T\ \hat{M}^2_{g,\delta}$ can therefore be computed on the servers
   (`ds.multiplyMatrixMatrix`). Subsequently, $\widehat{\psi}^{est}_{g,t}(W_i, \beta_{g,t,\delta})$ can be computed on the server sides (`ds.make`).

Knowing $\widehat{\psi}^{treat}_{g,t,\delta}(W_i, \beta_{g,t,\delta}), \widehat{\psi}^{comp}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g)$ and $\widehat{\psi}^{est}_{g,t,\delta}(W_i, \beta_{g,t,\delta}, \pi_g)$ on the server sides, $\widehat{\psi}^{*}_{g,t,\delta}$ can be computed on the server sides (`ds.make`).

Federated asymptotic variance-covariance matrix

In this subsection, we will demonstrate the estimation of the asymptotic variance-covariance matrix of the average treatment effect's estimates in a federated setting.

The asymptotic distribution of the ATT is defined in [Callaway and Sant'Anna, 2021] as

$$\sqrt{n}\left(\widehat{ATT}_{t\geq g-\delta} - ATT_{t\geq g-\delta}\right) \overset{d}{\sim} \mathcal{N}\left(0, \mathbb{E}\left[\Psi_{t\geq g-\delta}(W)\Psi_{t\geq g-\delta}(W)^T\right]\right), \tag{8}$$

where $ATT_{t\geq g-\delta}$ is the collection of $ATT(g,t;\delta)$ with $t \geq g-\delta$ and $\Psi_{t\geq g-\delta}(W)$ is the collection of relevant influence functions $\psi_{g,t,\delta}$.

In order to estimate $\mathbb{E}\left[\Psi_{t\geq g-\delta}(W)\Psi_{t\geq g-\delta}(W)^T\right]$, we use the sample analogue $\hat{\Psi}_{t\geq g-\delta}(W)^T\hat{\Psi}_{t\geq g-\delta}(W)$, where the number of rows in the matrix $\hat{\Psi}_{t\geq g-\delta}(W)$ corresponds to the total number of individuals and each column represents a combination of treatment periods g and treatment evaluation periods t. The federated matrix product is computed using (`ds.computeMatrixCrossproduct`), which we explain more detail within the section Federated matrix product with its transpose.

Federated multiplier bootstrap

The multiplier bootstrap allows computations of (clustered) standard errors. Callaway and Sant'Anna [2021] define one draw of the multiplier bootstrap as

$$\widehat{ATT}^{(b)}_{t\geq g-\delta} = \widehat{ATT}_{t\geq g-\delta} + \mathbb{E}_n\left[V \cdot \hat{\Psi}_{t\geq g-\delta}(W)\right], \tag{9}$$

where $\hat{\Psi}_{t\geq g-\delta}(W)$ is the sample analogue of $\Psi_{t\geq g-\delta}(W)$ and $\mathbb{E}_n(a)$ is the expectation over the components of a vector a. V is a random variable with $P(V = 1 - \frac{\sqrt{5}+1}{2}) = \frac{\sqrt{5}+1}{2\sqrt{5}}$ and $P(V = \frac{\sqrt{5}+1}{2}) = 1 - \frac{\sqrt{5}+1}{2\sqrt{5}}$. We subsequently show that the multiplier bootstrap can be federated using an example with s servers. By running the computations on each server (`ds.multiplierBootstrap`) and aggregating the results via a federated mean (`ds.mean`), the client can, due to the randomness of V, not exactly reproduce the results but obtain the same results qualitatively. Let $\hat{\Psi}_{t\geq g-\delta}(W)^{(k)}$ be the estimated influence matrix and $v^{(k)}$ be the vector or observations drawn from V on server k, such that

$$\hat{\mathbb{E}}_n\left[V \cdot \hat{\Psi}_{t\geq g-\delta}(W)\right] = \hat{\mathbb{E}}_n \begin{pmatrix} v^{(1)} \cdot \hat{\Psi}_{t\geq g-\delta}(W)^{(1)} \\ \vdots \\ v^{(s)} \cdot \hat{\Psi}_{t\geq g-\delta}(W)^{(s)} \end{pmatrix} \tag{10}$$

The multiplier bootstrap can be run on each server (`ds.multiplierBootstrap`) and multiplied by the server side sample size in order to compute each row of the matrix on the right hand-side of (10) on the respective servers. The client subsequently returns an estimate of the expectation using (`ds.mean`). The variance-covariance matrix of the average treatment effect's estimate can finally be estimated with B bootstrap draws $\frac{1}{B-1}\sum_{b=1}\widehat{ATT}^{(b)}_{t\geq g-\delta}\left(\widehat{ATT}^{(b)}{}_{t\geq g-\delta}\right)^T$.

Federated matrix product with its transpose

To compute asymptotic standard errors, the matrix product of the transposed data and itself must be returned to the client side. This allows for the calculation of the asymptotic linear representation, denoted $\hat{l}^{or}_{g,t}$, of the outcome regression in (7). In our package, this is done via the function `ds.computeMatrixCrossproduct`. Let $Z^{(i)} = \begin{pmatrix} z_1^{(i)} & \dots & z_p^{(i)} \end{pmatrix} \in \mathbb{R}^{n \times p}$, for $i = 1, 2, \dots, s$, be data matrices of s different servers such that the full data Z is given, without loss of generality, by

$$Z = \begin{pmatrix} z_1^{(1)} & \dots & z_p^{(1)} \\ z_1^{(2)} & \dots & z_p^{(2)} \\ \vdots & \vdots & \vdots \\ z_1^{(s)} & \dots & z_p^{(s)} \end{pmatrix} = \begin{pmatrix} Z^{(1)} \\ Z^{(2)} \\ \vdots \\ Z^{(s)} \end{pmatrix} \tag{11}$$

The matrix product of it's transpose is therefore given by

$$\begin{aligned} Z^T Z &= \begin{pmatrix} Z^{(1)} \\ Z^{(2)} \\ \vdots \\ Z^{(s)} \end{pmatrix}^T \begin{pmatrix} Z^{(1)} \\ Z^{(2)} \\ \vdots \\ Z^{(s)} \end{pmatrix} = \begin{pmatrix} Z^{(1)T} Z^{(2)T} & \dots & Z^{(s)T} \end{pmatrix} \begin{pmatrix} Z^{(1)} \\ Z^{(2)} \\ \vdots \\ Z^{(s)} \end{pmatrix} \\ &= \sum_{i=1}^{s} Z^{(i)T} Z^{(i)}. \end{aligned} \tag{12}$$

The multiplication of the server-side data with its transposed $Z^{(i)T} Z^{(i)}$ can be computed on each server and returned to the client side if it contains at least as enough different entries as allowed by the data providers via DataSHIELDS disclosure settings (default is 5). Knowing all $Z^{(i)T} Z^{(i)}$ on the client side, the sum can subsequently be computed on the client side to obtain $Z^T Z$.

Federated column appending of influence functions

While iterating over the treatment periods g and the treatment impact periods t, the data frames are subsetted with respect to the individuals who are either treated in g or in the control group. On the client side, the influence function is computed for these subsetted individuals. To apply the clustered multiplier bootstrap, however, the influence function must be sent back to the server side after each iteration of g and t. Since the individuals change every iteration, simply appending the influences column-wise is not possible. Therefore, we create a data frame filled with zeros and one column indicating the ID of the individuals, which is known to the server. However, appending a full vector, even though the vector is stored on ther server side, poses a data security problem as it allows for the Covariance-Based Attack Algorithm described in [Huth et al., 2023]. To address this, the rows of the data frame are randomly shuffled after each iteration.

To enhance understanding, we present a simplified example utilizing 4 individuals, two treatment periods $g = 1, 2$, and two treatment impact periods $t = 3, 4$. We assume that individual id_1 is

treated in $g = 1$, individual id_3 is treated in $g = 2$, and individuals id_2 and id_4 are never treated. The initialized data frame $\mathcal{F}_0$ would be given by

$$\mathcal{F}_0 = \begin{array}{c} \begin{array}{ccccc} \mathrm{id} & \hat{\psi}_{1,3} & \hat{\psi}_{2,3} & \hat{\psi}_{1,4} & \hat{\psi}_{2,4} \end{array} \\ \begin{pmatrix} \mathrm{id}_1 & 0 & 0 & 0 & 0 \\ \mathrm{id}_2 & 0 & 0 & 0 & 0 \\ \mathrm{id}_3 & 0 & 0 & 0 & 0 \\ \mathrm{id}_4 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}. \tag{13}$$

At the first iteration, the server stores the ID vector $v_{1,3} = \begin{pmatrix} \mathrm{id1} & \mathrm{id2} & \mathrm{id4} \end{pmatrix}^T$ of the individuals used in this run. The server computes the influence vector $w_{1,3} = \begin{pmatrix} 3 & -1 & 2 \end{pmatrix}^T$. On the server, the ID vector and the influence vector are matched component-wise to create the matrix $\begin{pmatrix} v_{1,3} & w_{1,3} \end{pmatrix}$. In the next step, the relevant IDs in the first column of $\mathcal{F}_0$ are replaced with the corresponding values from $w_{1,3}$, and the matrix is subsequently shuffled row-wise

$$\mathcal{F}_1 = \begin{array}{c} \begin{array}{ccccc} \mathrm{id} & \hat{\psi}_{1,3} & \hat{\psi}_{2,3} & \hat{\psi}_{1,4} & \hat{\psi}_{2,4} \end{array} \\ \begin{pmatrix} \mathrm{id}_4 & 2 & 0 & 0 & 0 \\ \mathrm{id}_2 & -1 & 0 & 0 & 0 \\ \mathrm{id}_3 & 0 & 0 & 0 & 0 \\ \mathrm{id}_1 & 3 & 0 & 0 & 0 \end{pmatrix} \end{array}. \tag{14}$$

This procedure is repeated for all combinations of t and g such that the final data frame is, depending on the random shuffling, given by

$$\mathcal{F}_4 = \begin{array}{c} \begin{array}{ccccc} \mathrm{id} & \hat{\psi}_{1,3} & \hat{\psi}_{2,3} & \hat{\psi}_{1,4} & \hat{\psi}_{2,4} \end{array} \\ \begin{pmatrix} \mathrm{id}_2 & -1 & 1 & 3 & -2 \\ \mathrm{id}_4 & 2 & 4 & 1 & 2 \\ \mathrm{id}_1 & 3 & 0 & 2 & 0 \\ \mathrm{id}_3 & 0 & 2 & 0 & -1 \end{pmatrix} \end{array}. \tag{15}$$

By implementing this method, it is possible to append the influence functions in a privacy-preserving manner.

Summary statistics - Malaria dataset

Table 2: Summary statistics: Average performance of students by district, school and term of studies. Standard deviations are shown in parenthesis for all sample averages. For number of females, the table reports the percentage of female students on the total school population N.

			Baseline			Performance 2015			Performance 2016		
	District	School	N	Mean age	Female students	Term 1	Term 2	Term 3	Term 1	Term 2	Term 3
1	Magude	Duco	28	9.36	14	12.26	12.12	12.20	12.68	12.41	12.54
				(1.56)	(50%)	(2.07)	(2.22)	(2.35)	(2.21)	(2.37)	(2.39)
2	Magude	Graca Machel	90	9.43	55	12.70	12.53	12.33	12.73	12.90	12.89
				(2.59)	(61%)	(2.16)	(2.53)	(2.57)	(1.87)	(1.93)	(1.92)
3	Magude	Maguiguana	89	9.33	49	12.09	11.86	11.86	12.41	12.38	12.59
				(1.83)	(55%)	(2.09)	(2.23)	(2.36)	(2.32)	(2.56)	(2.78)
4	Magude	Moine	21	10.52	9	11.36	11.67	11.23	12.43	13.01	12.86
				(1.60)	(43%)	(1.26)	(1.49)	(2.09)	(1.70)	(2.14)	(2.60)
5	Magude	Simbe	22	9.68	8	11.97	11.53	11.32	11.50	11.44	11.13
				(2.90)	(36%)	(1.85)	(1.69)	(1.60)	(1.68)	(1.71)	(1.64)
6	Manhica	3 de Fev	208	9.90	90	11.75	11.71	11.76	11.79	11.81	11.88
				(2.13)	(43%)	(2.03)	(2.13)	(2.29)	(2.01)	(2.13)	(2.31)
7	Manhica	Ilha Josina	156	10.04	79	12.81	12.91	12.70	12.77	12.65	12.58
				(2.36)	(51%)	(1.64)	(1.60)	(1.65)	(1.47)	(1.54)	(1.54)
8	Manhica	Maragra	304	9.24	136	12.03	12.09	12.15	12.08	12.41	12.37
				(1.88)	(45%)	(1.91)	(2.17)	(2.07)	(2.02)	(2.00)	(1.96)
9	Manhica	Xinavane	126	9.88	61	13.46	13.21	12.73	13.03	13.24	12.96
				(2.74)	(48%)	(2.04)	(2.34)	(2.49)	(2.42)	(2.33)	(2.41)

Supplementary Material

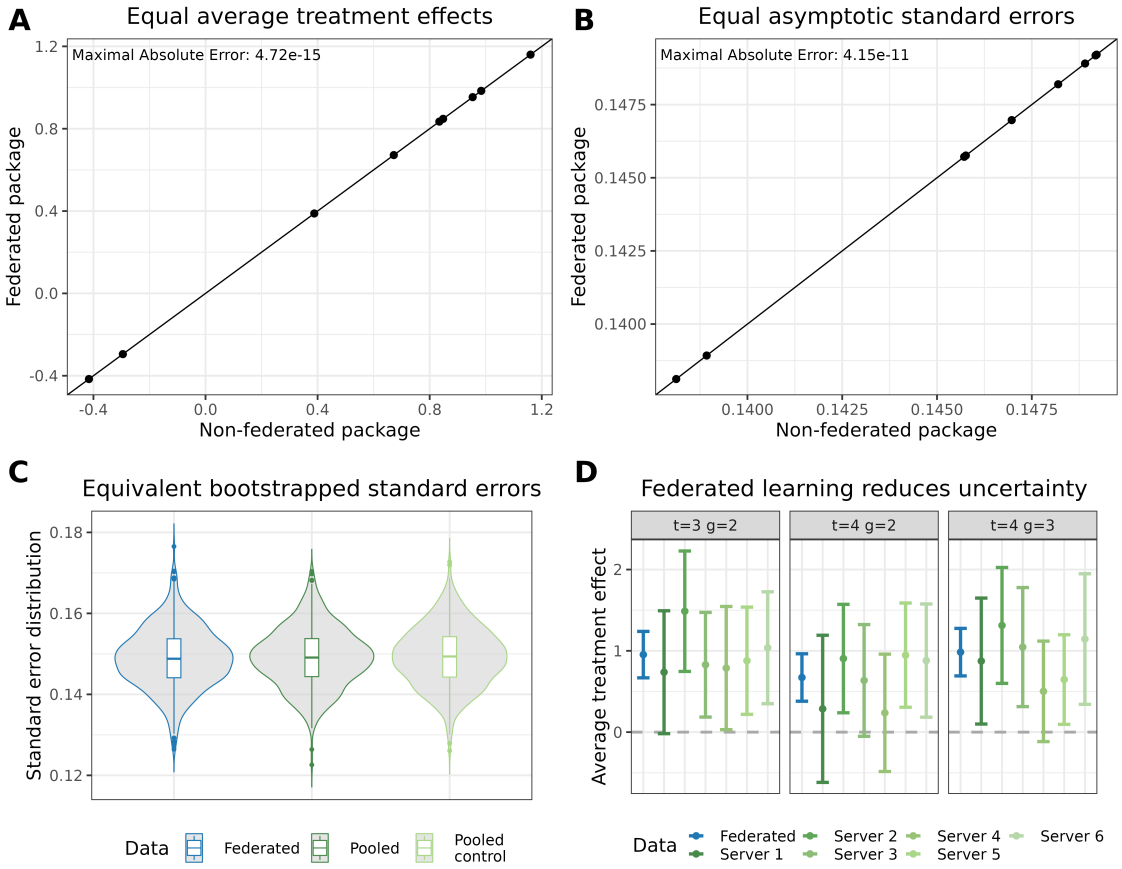

Figure 5: DR estimate and never treated individuals as control group. The federated set-up consisted of 6 servers. Three servers contained 134 (536) individuals (observations), and three servers contained 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three. All observations of one individual were within one server. **A** Depicts the equality of central and federated point estimates. The x-axis represents the point estimates of the central (non-federated) estimator [Callaway and Sant'Anna, 2021], while the y-axis represents the estimates obtained from our federated approach. The diagonal line depicts the 45° line, indicating that the federated estimate yields results equivalent to the non-federated estimate when they align along this line. **B** Depicts the equality of central and federated asymptotic standard errors. The x-axis represents the asymptotic standard errors of the central (non-federated) estimator, while the y-axis represents the asymptotic standard errors obtained from our federated approach. The diagonal line again depicts the 45° line. **C** The subplot displays a comparison of the distribution of bootstrapped standard errors using boxplots and densities. The distribution of the federated bootstrapped standard errors is shown in blue, while the distribution of the central estimation is shown in green (dark and light). Two central learning distributions were computed to establish a plausible reference difference between two equal distributions. A total of 500 bootstrapped standard errors were computed to obtain the distributions for analysis. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated treatment effects for the periods following the treatment. The estimates obtained with our federated package are shown in blue, while estimates based on data from only one server are depicted in green. This figure has been designed using resources from Flaticon.com

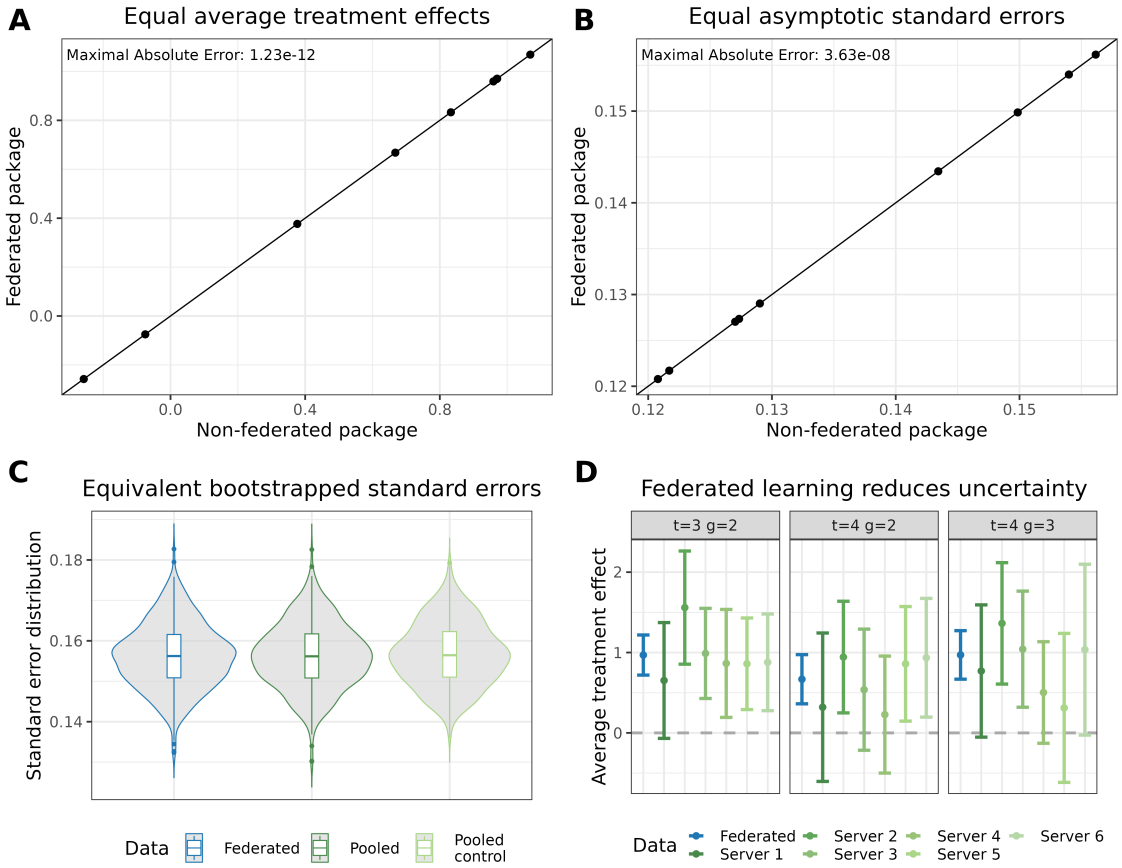

Figure 6: Inverse probability weighted estimate and not-yet treated individuals as control group. The federated set-up consisted of 6 servers. Three servers contained 134 (536) individuals (observations), and three servers contained 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three. All observations of one individual were within one server. **A** Depicts the equality of central and federated point estimates. The x-axis represents the point estimates of the central (non-federated) estimator [Callaway and Sant'Anna, 2021], while the y-axis represents the estimates obtained from our federated approach. The diagonal line depicts the 45° line, indicating that the federated estimate yields results equivalent to the non-federated estimate when they align along this line. **B** Depicts the equality of central and federated asymptotic standard errors. The x-axis represents the asymptotic standard errors of the central (non-federated) estimator, while the y-axis represents the asymptotic standard errors obtained from our federated approach. The diagonal line again depicts the 45° line. **C** The subplot displays a comparison of the distribution of bootstrapped standard errors using boxplots and densities. The distribution of the federated bootstrapped standard errors is shown in blue, while the distribution of the central estimation is shown in green (dark and light). Two central learning distributions were computed to establish a plausible reference difference between two equal distributions. A total of 500 bootstrapped standard errors were computed to obtain the distributions for analysis. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated treatment effects for the periods following the treatment. The estimates obtained with our federated package are shown in blue, while estimates based on data from only one server are depicted in green. This figure has been designed using resources from Flaticon.com

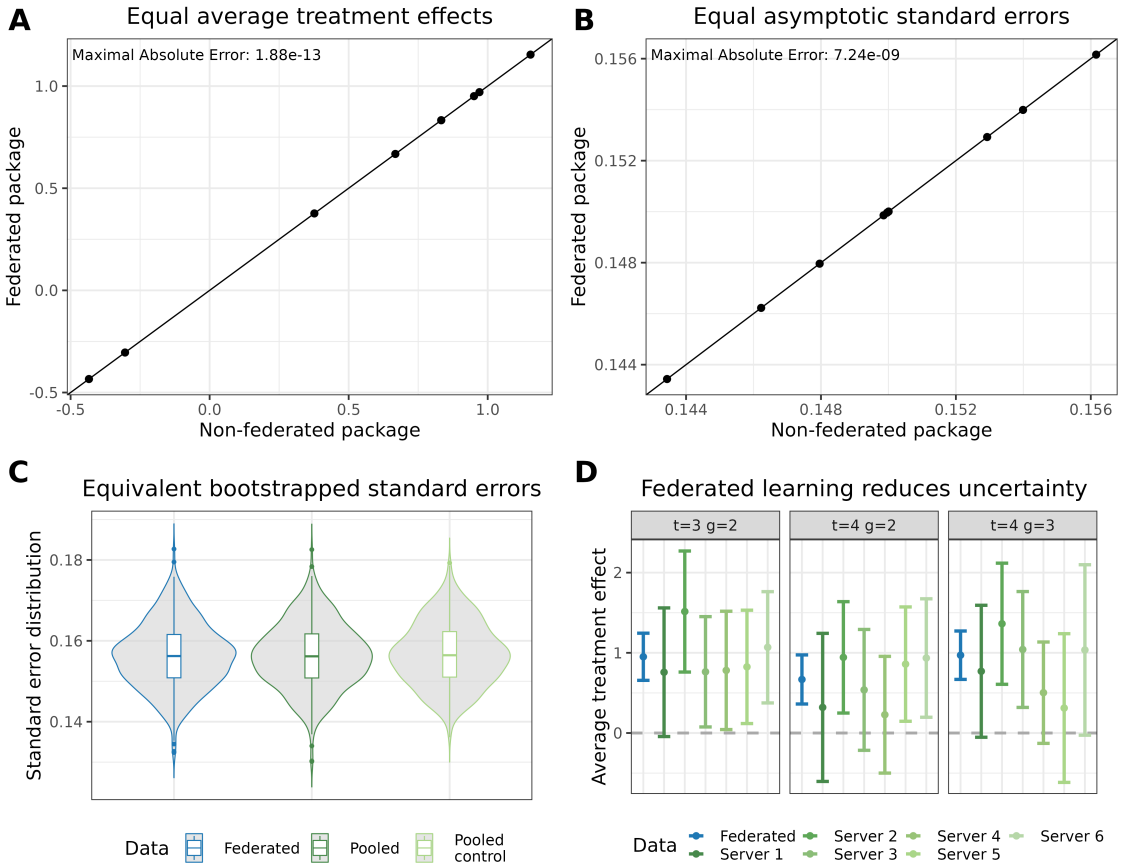

Figure 7: Inverse probability weighted estimate and never treated individuals as control group. The federated set-up consisted of 6 servers. Three servers contained 134 (536) individuals (observations), and three servers contained 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three. All observations of one individual were within one server. **A** Depicts the equality of central and federated point estimates. The x-axis represents the point estimates of the central (non-federated) estimator [Callaway and Sant'Anna, 2021], while the y-axis represents the estimates obtained from our federated approach. The diagonal line depicts the 45° line, indicating that the federated estimate yields results equivalent to the non-federated estimate when they align along this line. **B** Depicts the equality of central and federated asymptotic standard errors. The x-axis represents the asymptotic standard errors of the central (non-federated) estimator, while the y-axis represents the asymptotic standard errors obtained from our federated approach. The diagonal line again depicts the 45° line. **C** The subplot displays a comparison of the distribution of bootstrapped standard errors using boxplots and densities. The distribution of the federated bootstrapped standard errors is shown in blue, while the distribution of the central estimation is shown in green (dark and light). Two central learning distributions were computed to establish a plausible reference difference between two equal distributions. A total of 500 bootstrapped standard errors were computed to obtain the distributions for analysis. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated treatment effects for the periods following the treatment. The estimates obtained with our federated package are shown in blue, while estimates based on data from only one server are depicted in green. This figure has been designed using resources from Flaticon.com

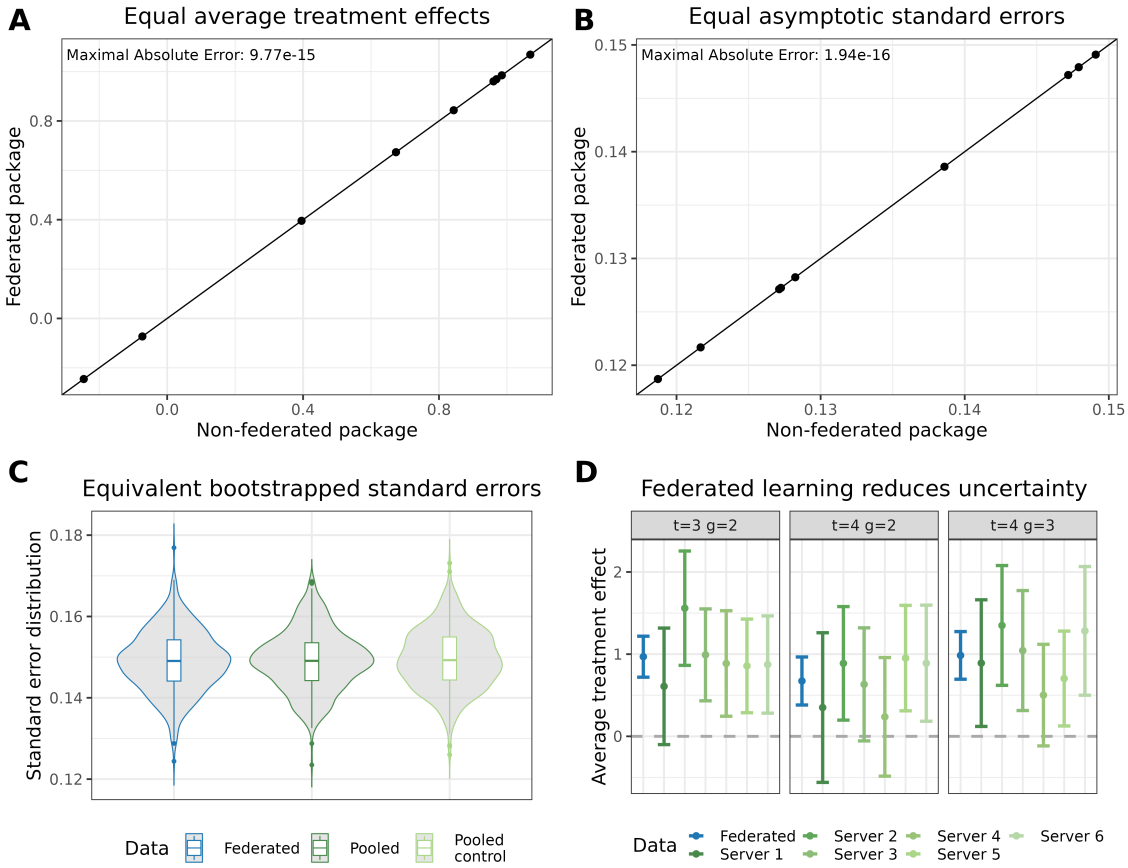

Figure 8: Outcome regression estimate and not-yet treated individuals as control group. The federated set-up consisted of 6 servers. Three servers contained 134 (536) individuals (observations), and three servers contained 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three. All observations of one individual were within one server. **A** Depicts the equality of central and federated point estimates. The x-axis represents the point estimates of the central (non-federated) estimator [Callaway and Sant'Anna, 2021], while the y-axis represents the estimates obtained from our federated approach. The diagonal line depicts the 45° line, indicating that the federated estimate yields results equivalent to the non-federated estimate when they align along this line. **B** Depicts the equality of central and federated asymptotic standard errors. The x-axis represents the asymptotic standard errors of the central (non-federated) estimator, while the y-axis represents the asymptotic standard errors obtained from our federated approach. The diagonal line again depicts the 45° line. **C** The subplot displays a comparison of the distribution of bootstrapped standard errors using boxplots and densities. The distribution of the federated bootstrapped standard errors is shown in blue, while the distribution of the central estimation is shown in green (dark and light). Two central learning distributions were computed to establish a plausible reference difference between two equal distributions. A total of 500 bootstrapped standard errors were computed to obtain the distributions for analysis. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated treatment effects for the periods following the treatment. The estimates obtained with our federated package are shown in blue, while estimates based on data from only one server are depicted in green. This figure has been designed using resources from Flaticon.com

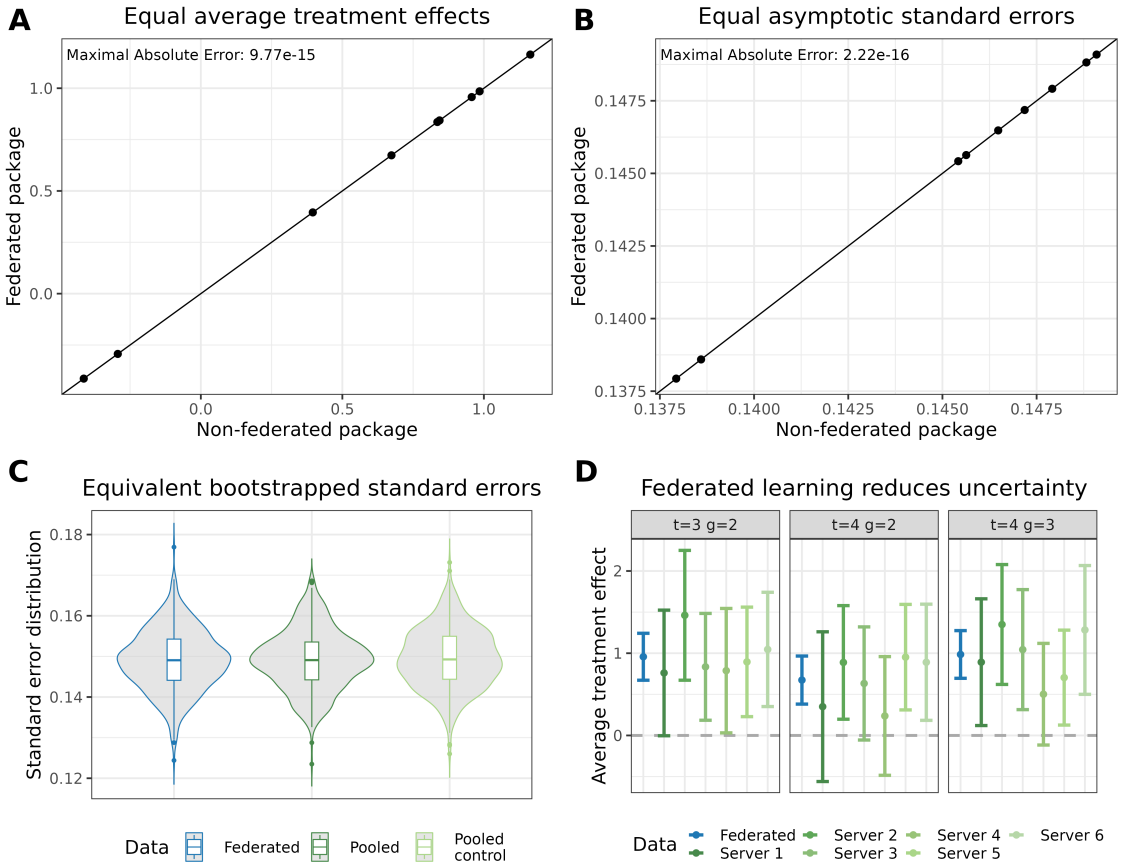

Figure 9: Outcome regression estimate and never treated individuals as control group. The federated set-up consisted of 6 servers. Three servers contained 134 (536) individuals (observations), and three servers contained 133 (532) individuals (observations). Individuals were either never treated or treated in period two or three. All observations of one individual were within one server. **A** Depicts the equality of central and federated point estimates. The x-axis represents the point estimates of the central (non-federated) estimator [Callaway and Sant'Anna, 2021], while the y-axis represents the estimates obtained from our federated approach. The diagonal line depicts the 45° line, indicating that the federated estimate yields results equivalent to the non-federated estimate when they align along this line. **B** Depicts the equality of central and federated asymptotic standard errors. The x-axis represents the asymptotic standard errors of the central (non-federated) estimator, while the y-axis represents the asymptotic standard errors obtained from our federated approach. The diagonal line again depicts the 45° line. **C** The subplot displays a comparison of the distribution of bootstrapped standard errors using boxplots and densities. The distribution of the federated bootstrapped standard errors is shown in blue, while the distribution of the central estimation is shown in green (dark and light). Two central learning distributions were computed to establish a plausible reference difference between two equal distributions. A total of 500 bootstrapped standard errors were computed to obtain the distributions for analysis. **D** The subplot presents point estimates (depicted as dots) and 95% confidence intervals of the estimated treatment effects for the periods following the treatment. The estimates obtained with our federated package are shown in blue, while estimates based on data from only one server are depicted in green. This figure has been designed using resources from Flaticon.com