Child Mind Institute

Federated Learning with Healthy Brain Network Data: Integrating Freesurfer Metrics and Age-Based Stratification Using the Flower Framework

CONTRIBUTORS: EMMA CORLEY, EREN KAFADAR, MAYA LAKSHMAN, MOLLY OLZINSKI, MICHELLE WANG, AOIFE WARREN, AUDREY WEBER

Neurohackademy Presentation 2024

Outline

Aims: To employ Flower, a Federated Learning Framework that allows for the distributed training of neural network models, as well as Scikit Learn, to predict brain age using brain metrics from the Healthy Brain Network (HBN).

Problems with centralized machine learning approaches:

- Requires central data repository - infrastructure and $$$
- Does not scale well to large datasets
- Single point of failure risk (data breach & privacy concerns)

Why Federated Learning?

- Multiple data contributors
- No raw data exchange
- Decentralized

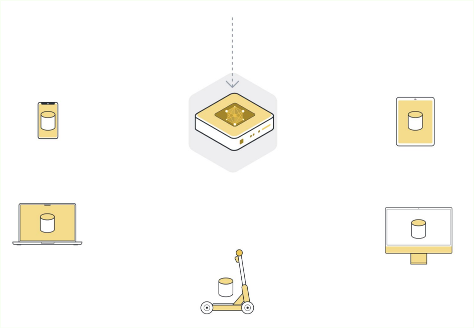
Brisimi, TS *et al.* (2018)
https://doi.org/10.1016/j.ijmedinf.2018.01.007

The Flower Framework

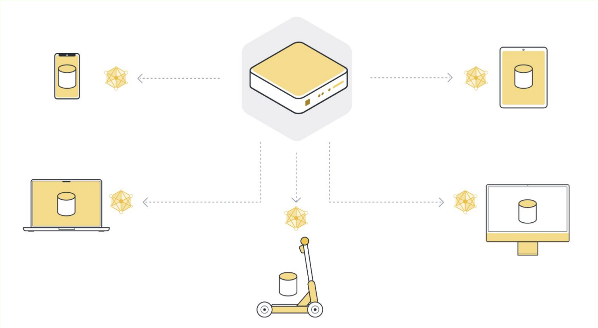
- Federated learning requires infrastructure to transfer machine learning models, train and evaluate them on each local dataset, and then aggregate those models.
- Flower provides that infrastructure in a way that is unified, easy, and scalable.

Federated Learning with Flower

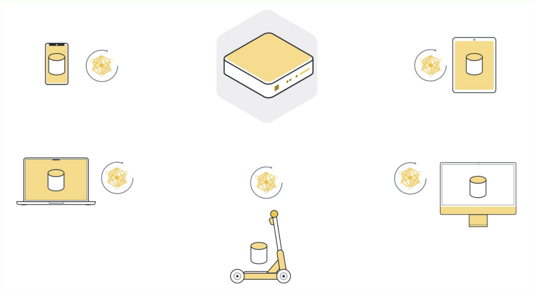
1. Initialize global model



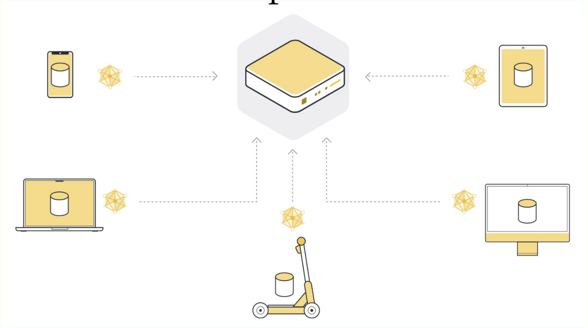
2. Send model to clients



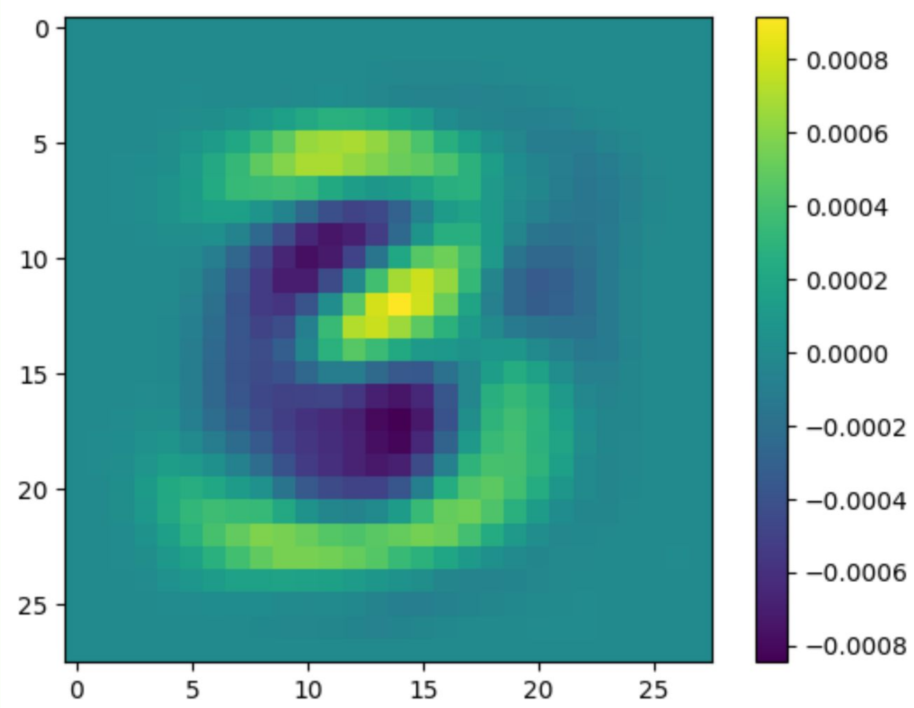
3. Train models locally



4. Send model parameters to server and aggregate



Flower tutorial with MNIST dataset



- Load the MNIST dataset
- Split the data in to train/test sets
  - Further split the train set into partitions to simulate distributed setting
- Define a logistic regression classification model and how it integrates with Flower interface
- Server script and client script
  - Run server first
  - Run clients

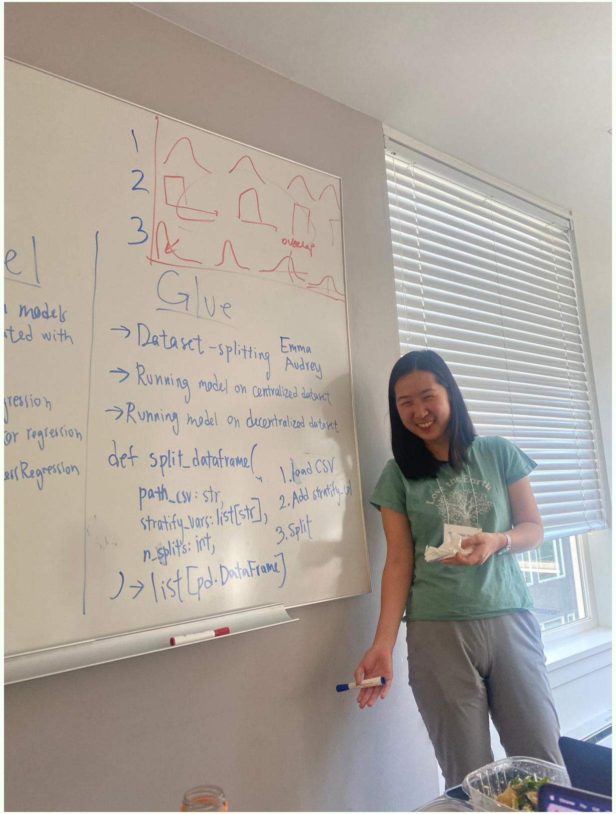
We adapted the tutorial to run our own experiments

- Brain age prediction using Healthy Brain Network data
    - N=223
    - **Input**: Freesurfer features and sex
    - **Output**: age (range: 5-21)
- **Aim: examine model performance with different data splitting schemes**

- Splitting conditions
    - Equal sample sizes
        - Match distribution of age/sex/site across partitions
        - Have separate age ranges for each partition
    - Use different sample sizes for each partition

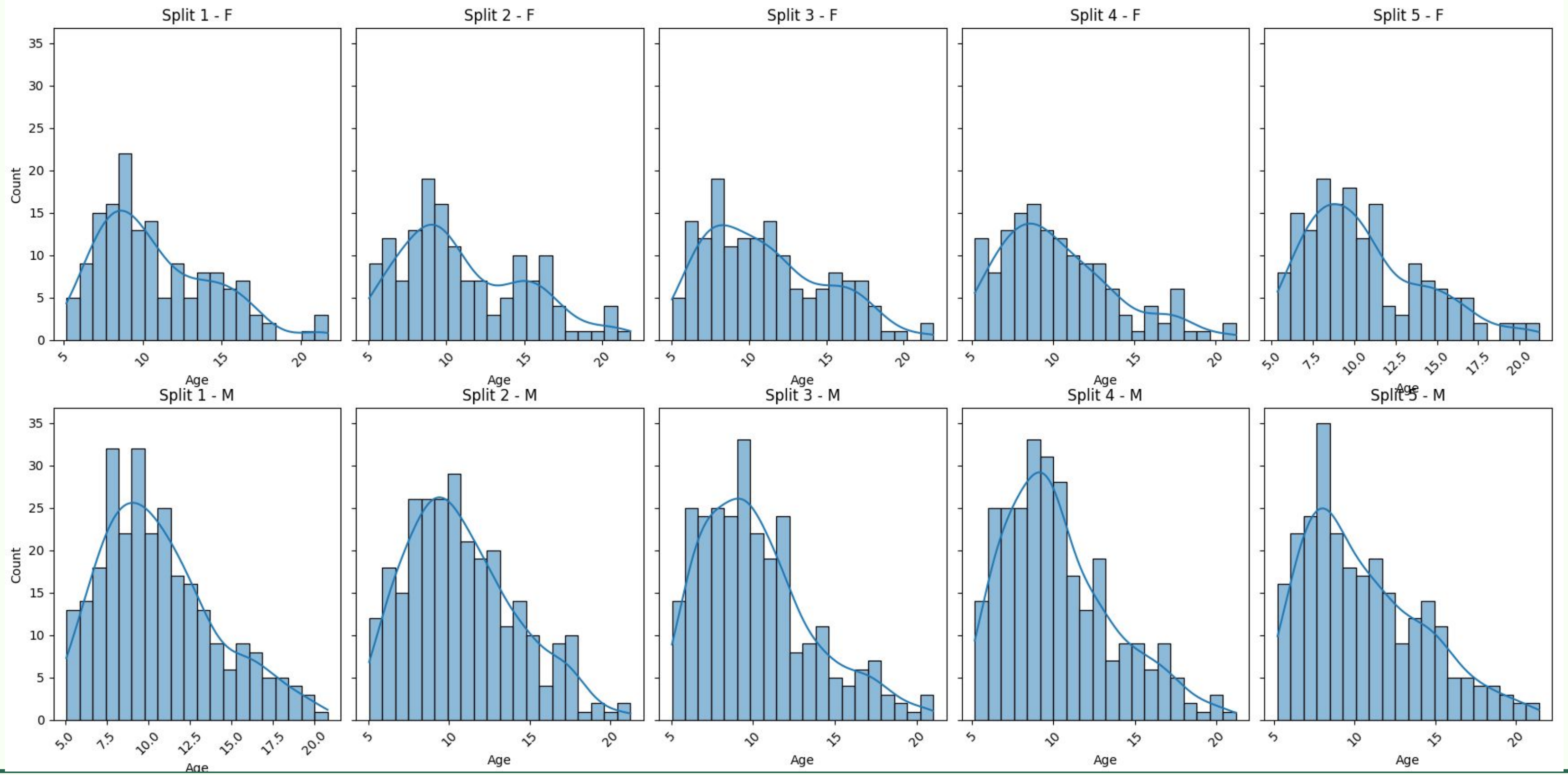
Methods: Models

1) Logistic Regression (tutorial)
2) Linear Regression
3) Lasso Regression

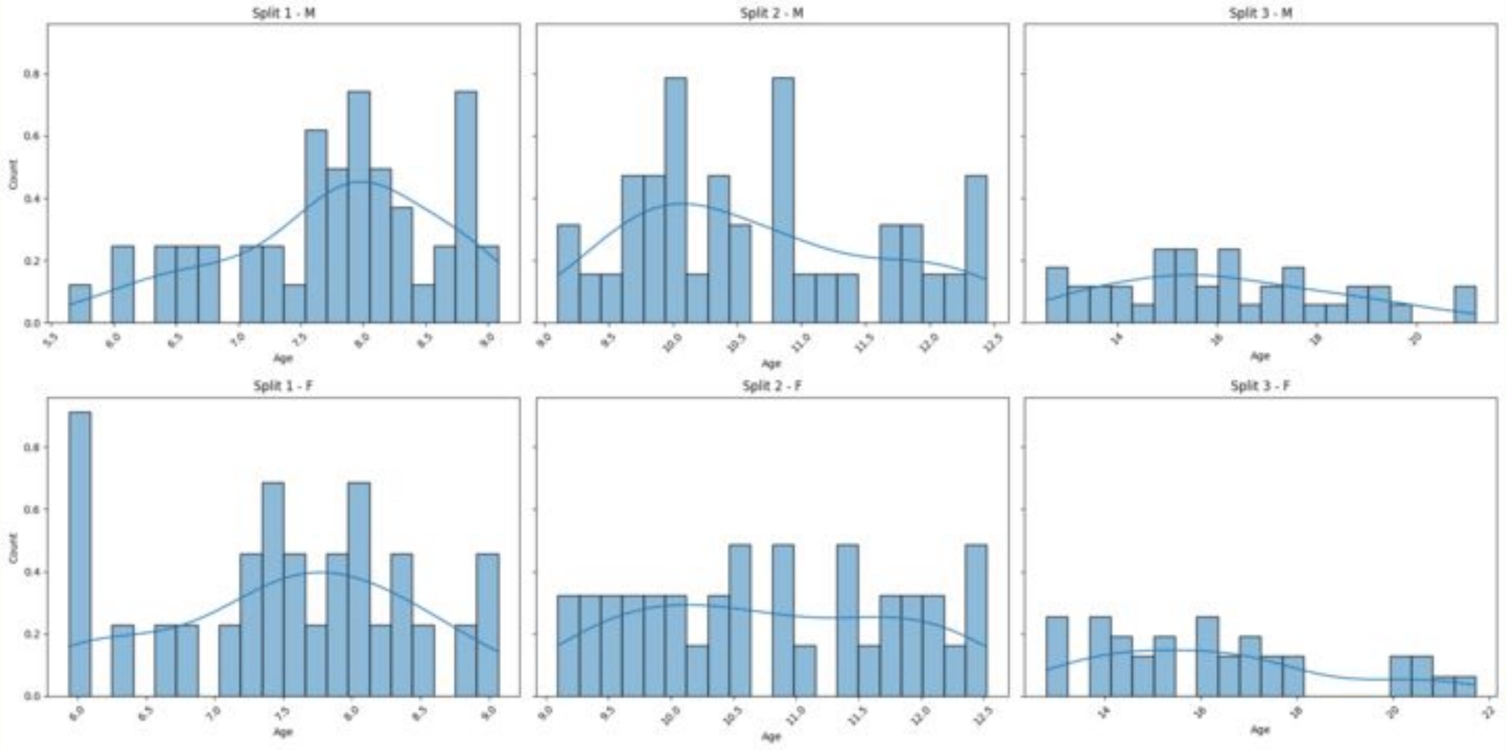


Demo

Results: Splitting data with consistent age/sex/site distribution

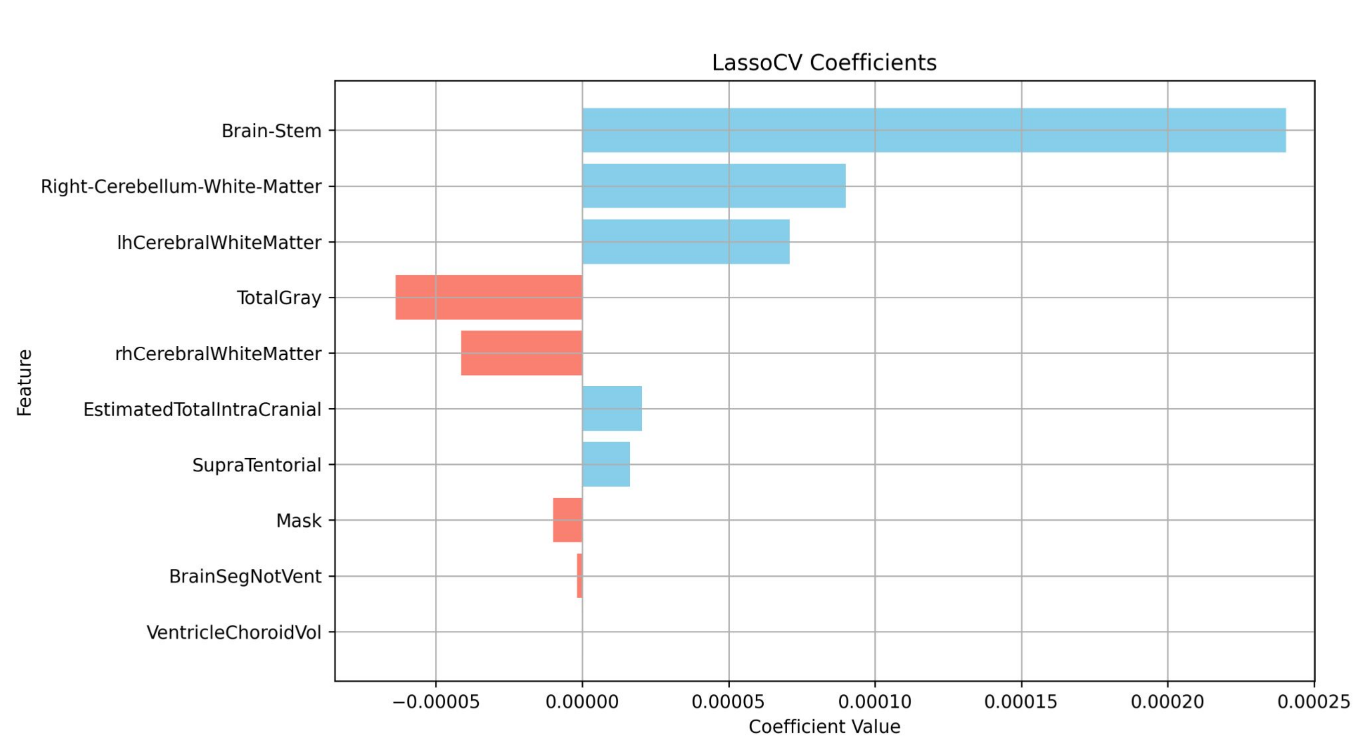


Results: Age Grouping

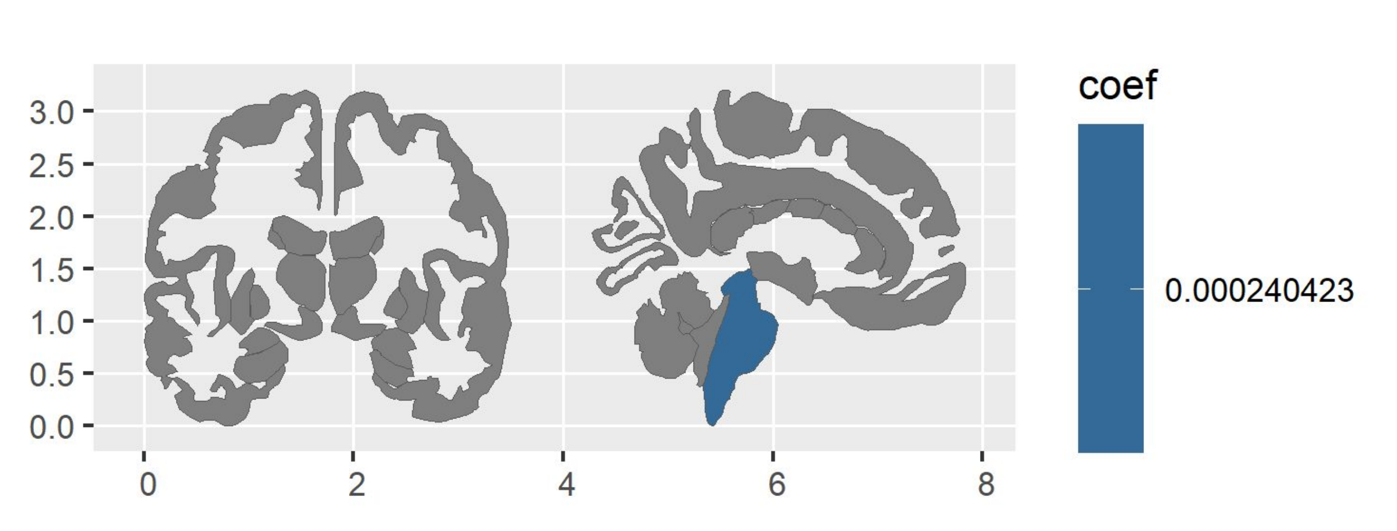


Lasso Model Top 10 Weights for Brain Age Prediction:

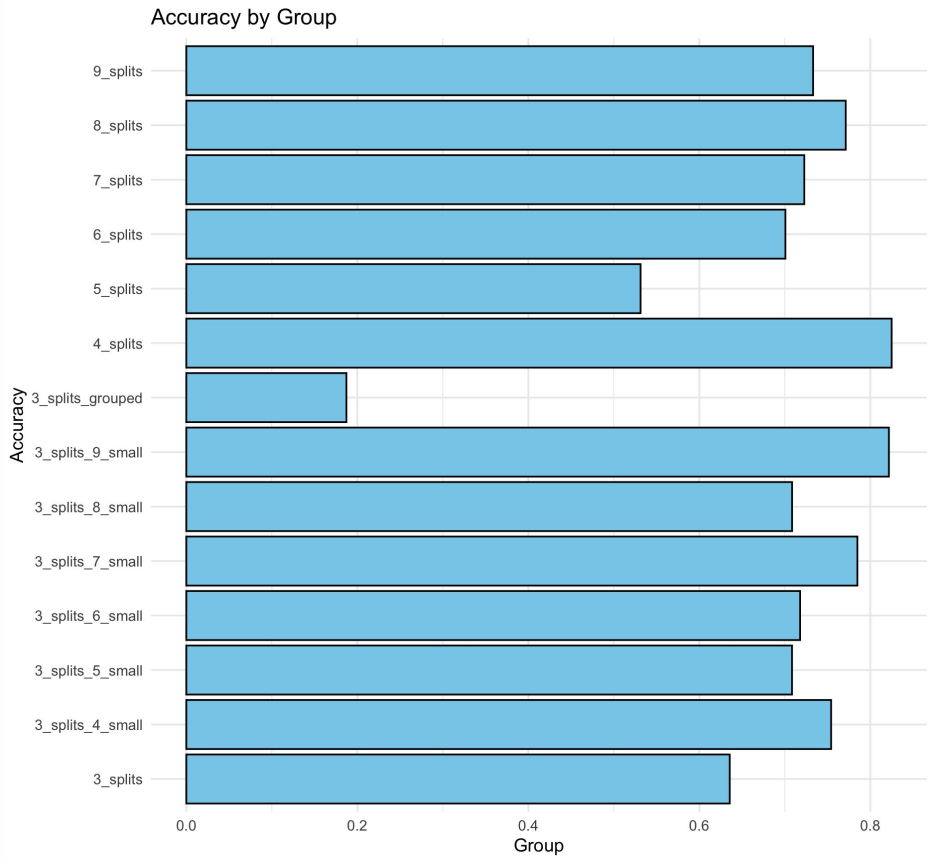
Training on 80% of full data (no partitions): **MSE=5.06** **R^2=0.64**



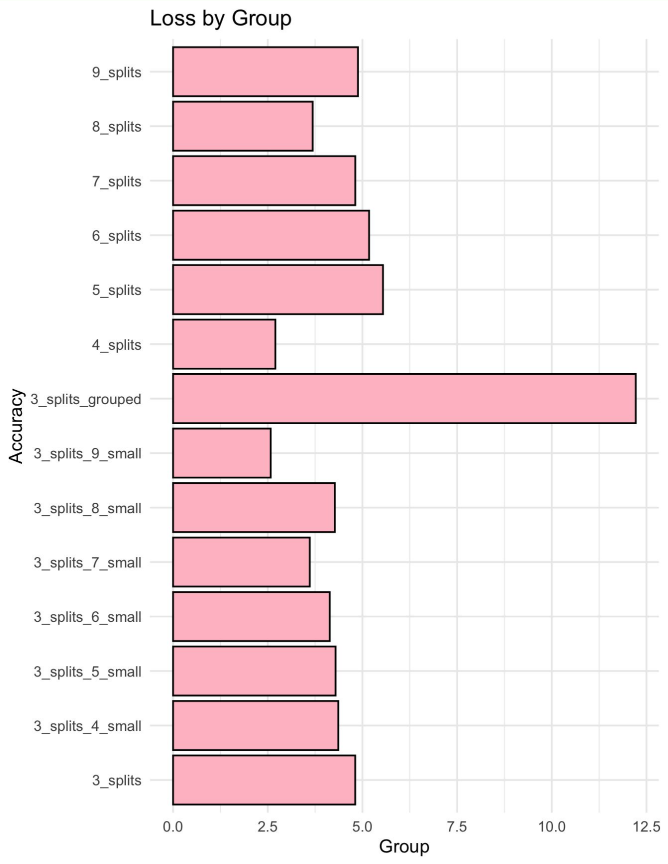
Aseg ROI Weight for Brain Age Prediction



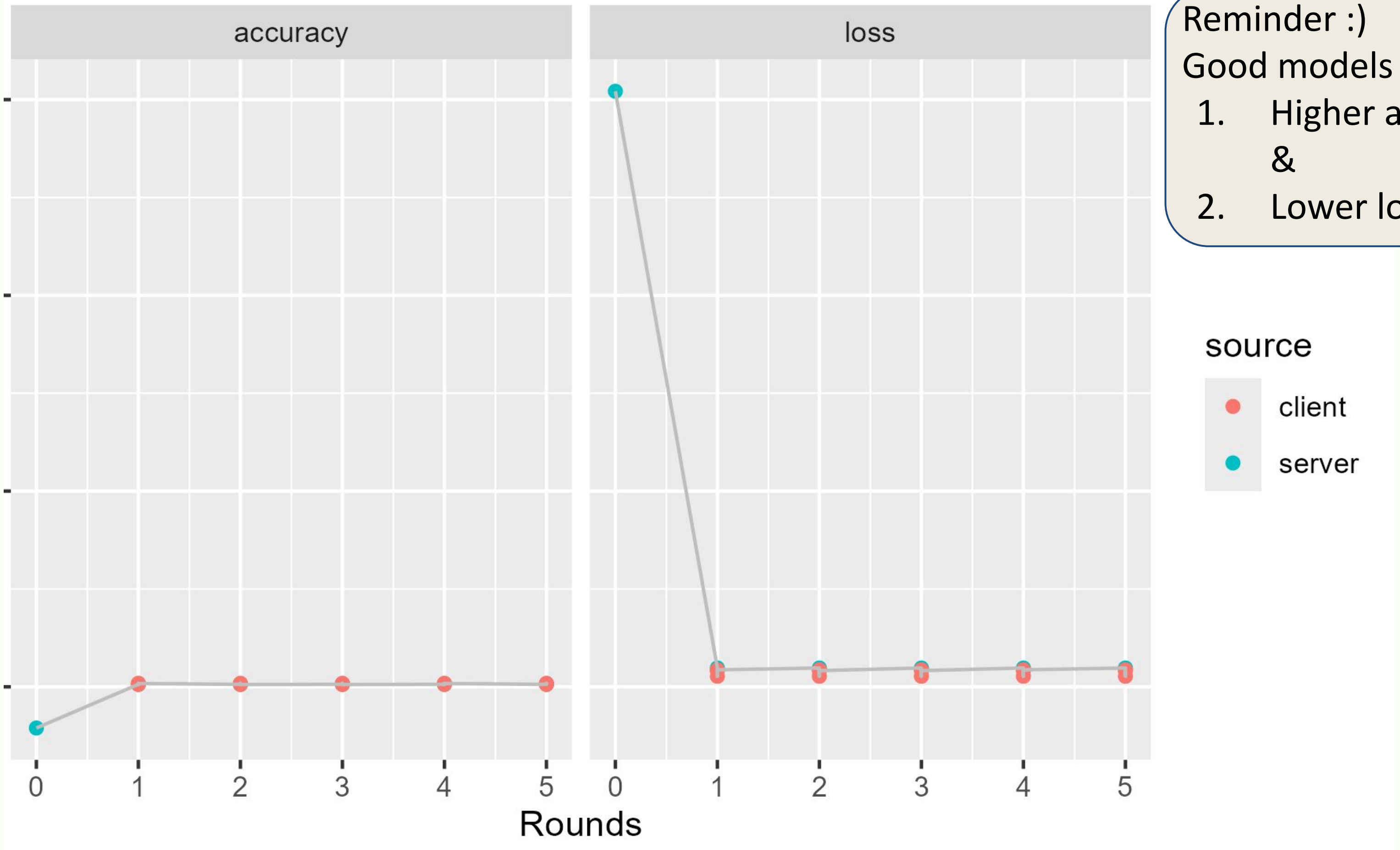
R^2 by splitting condition



MSE by splitting condition



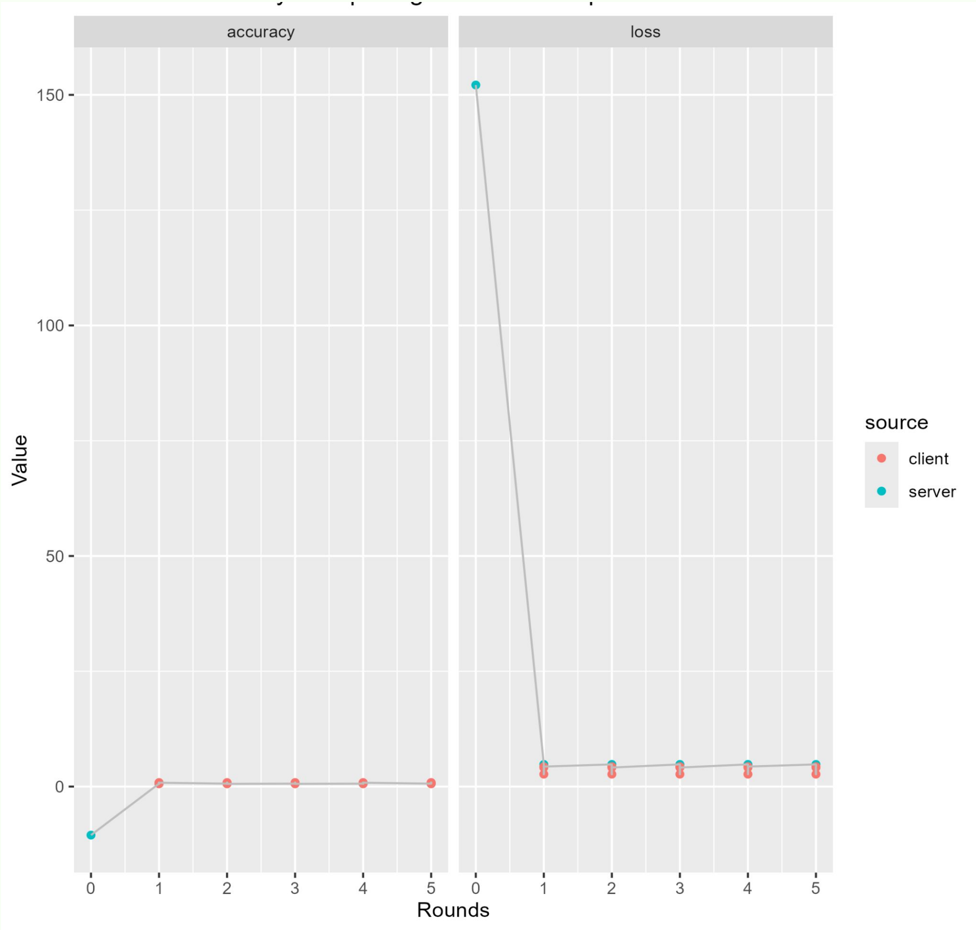
Loss and Accuracy for Splitting in Three Groups



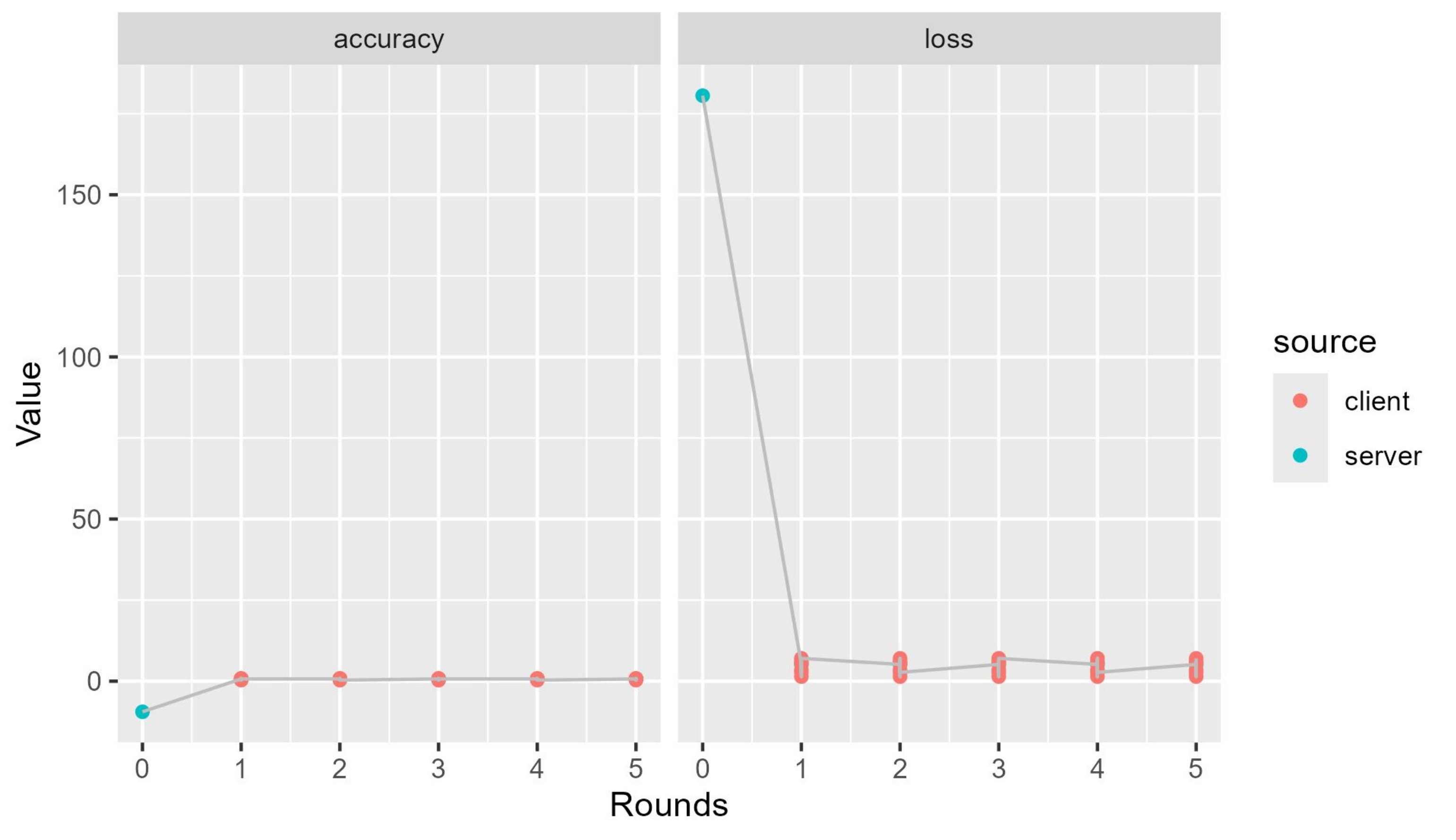
Reminder :)
Good models have:

1. Higher accuracy
   &
2. Lower loss

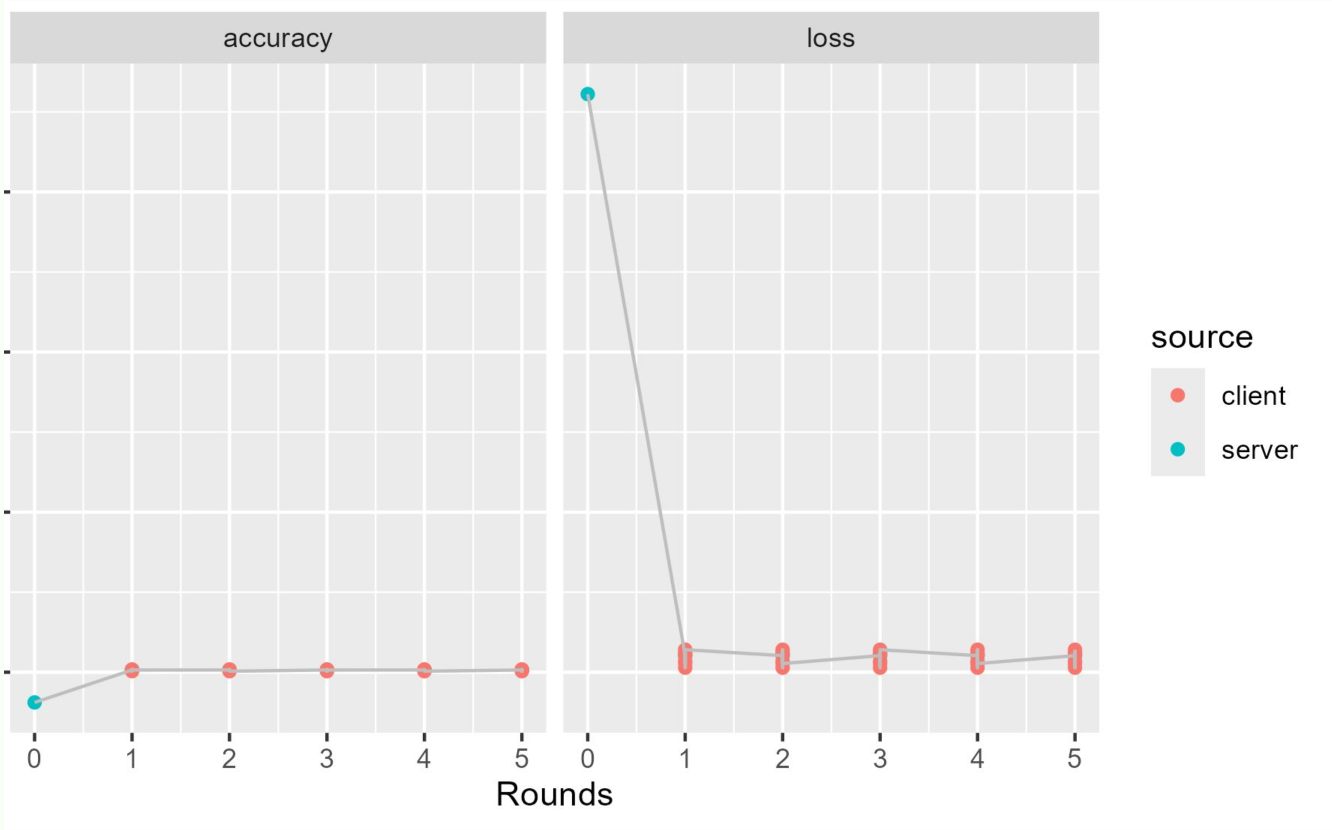
Loss and Accuracy for Splitting in Four Groups



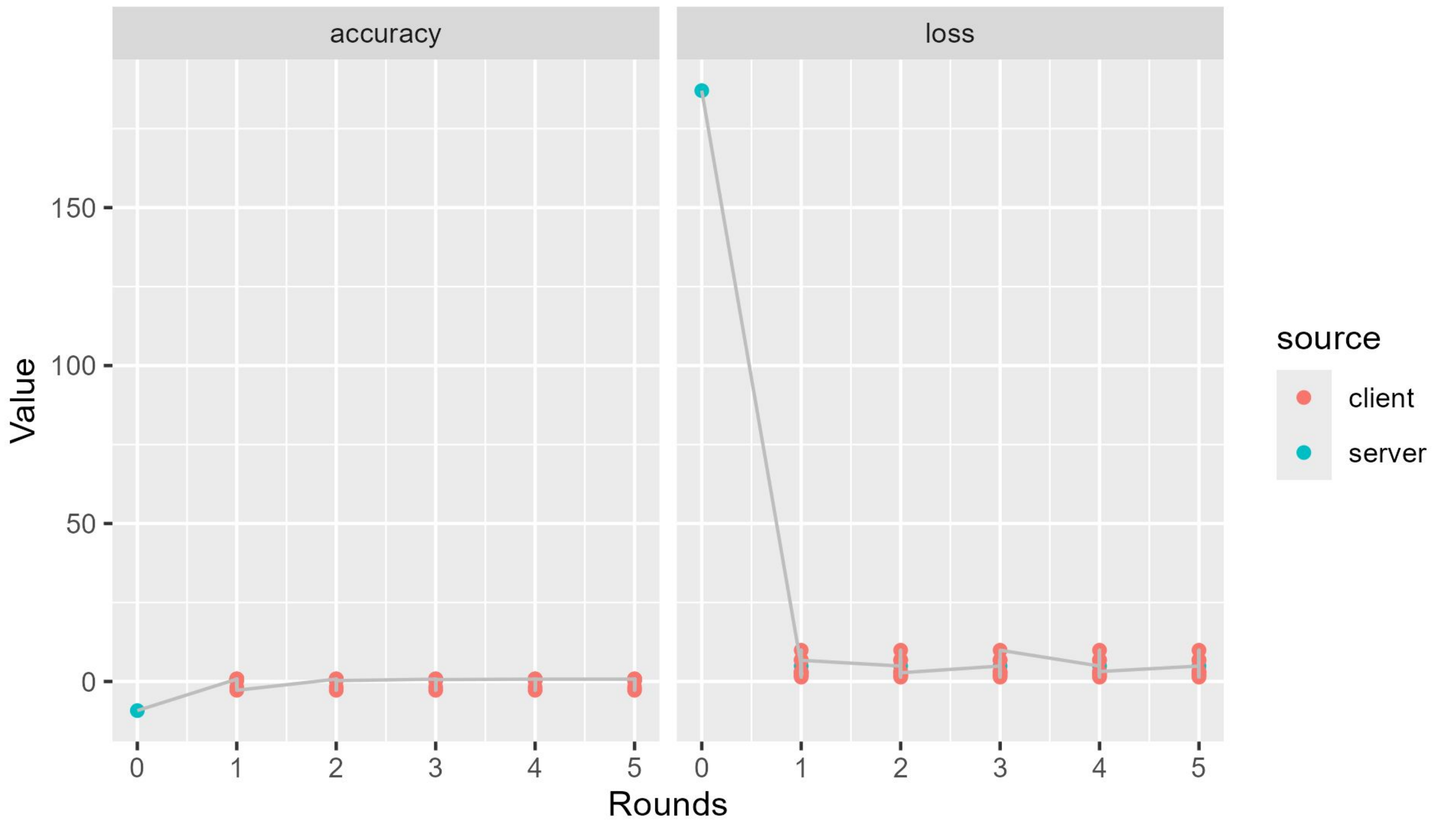
Loss and Accuracy for Splitting in Five Groups



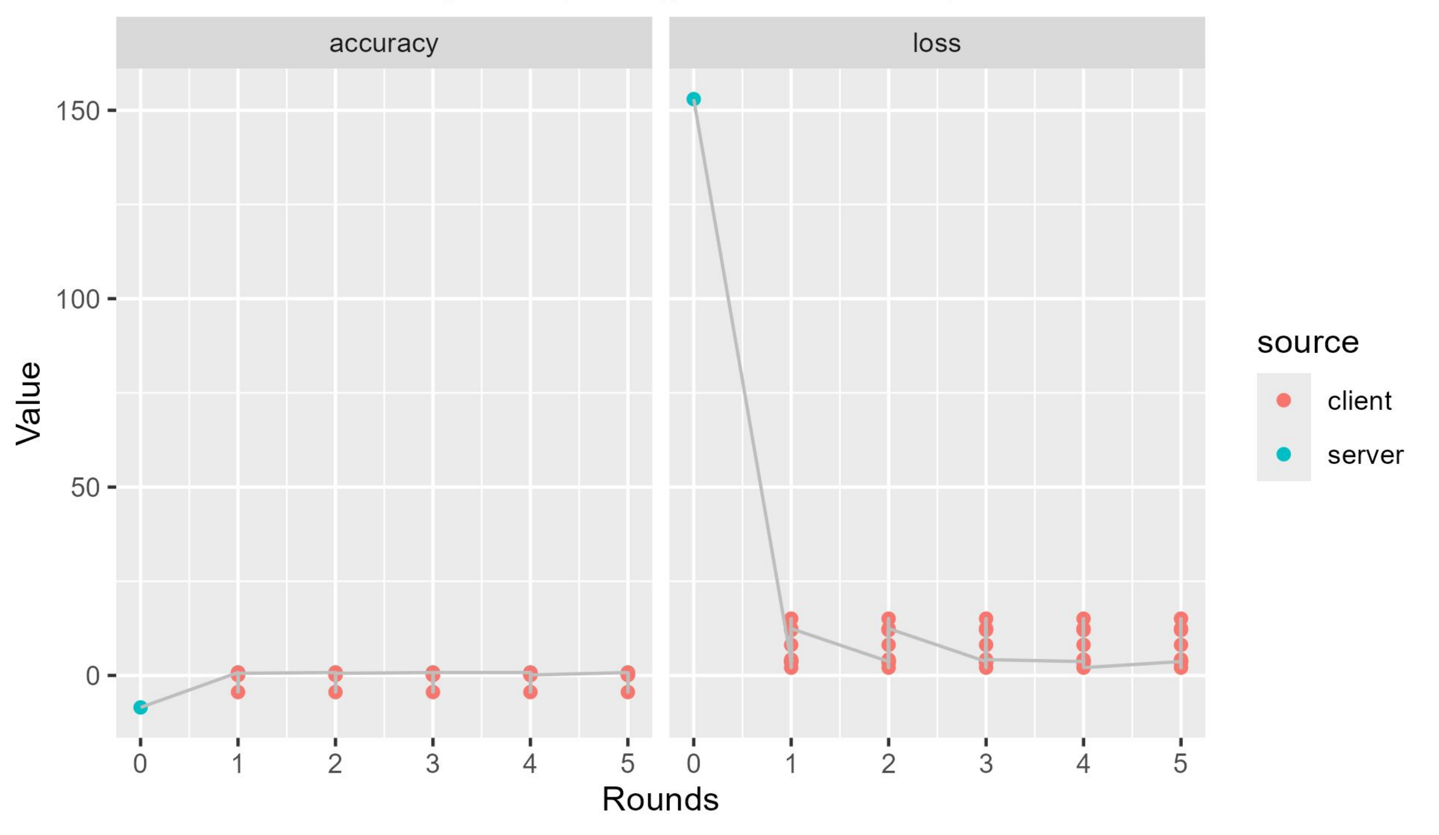
Loss and Accuracy for Splitting in Six Groups



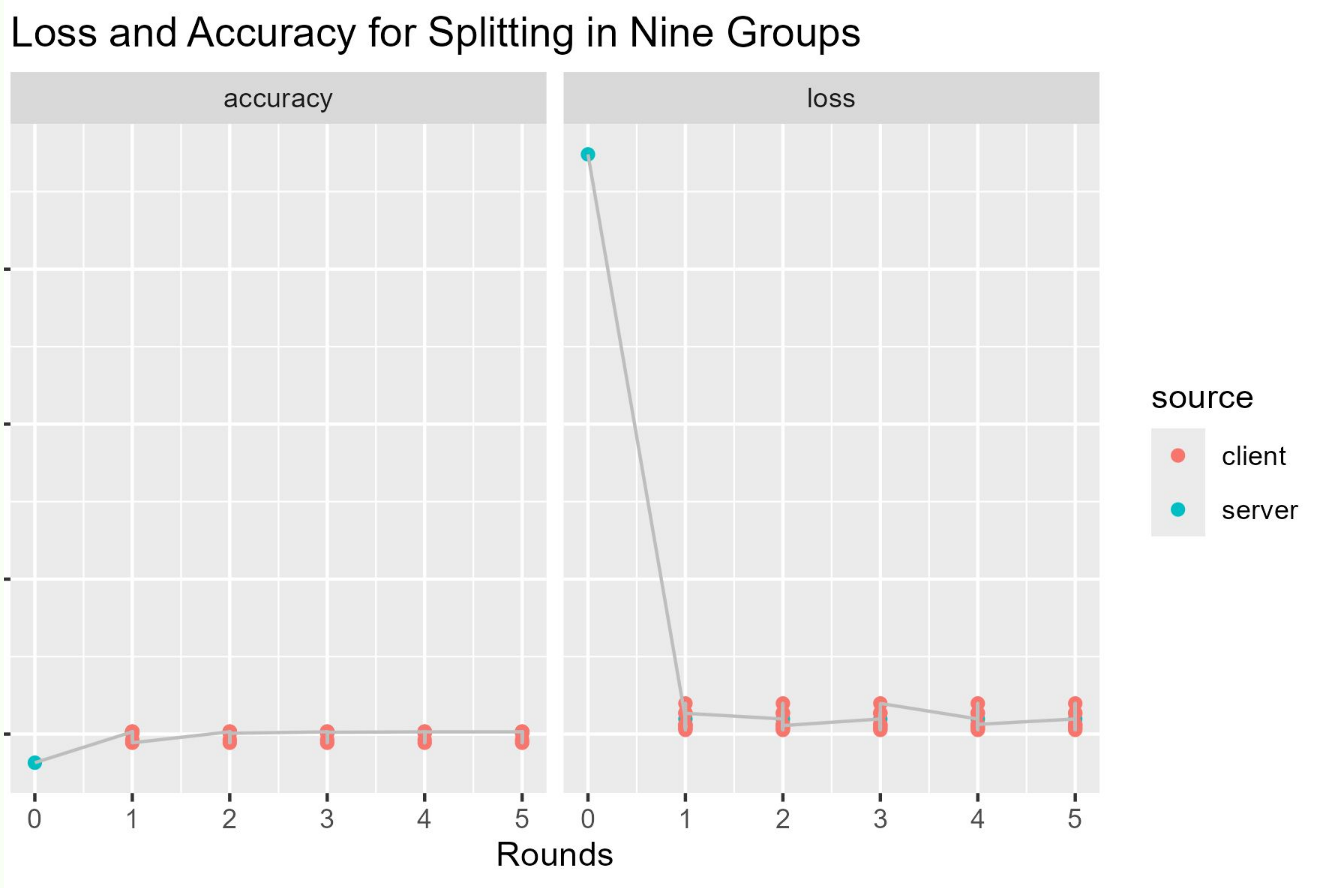
Loss and Accuracy for Splitting in Seven Groups



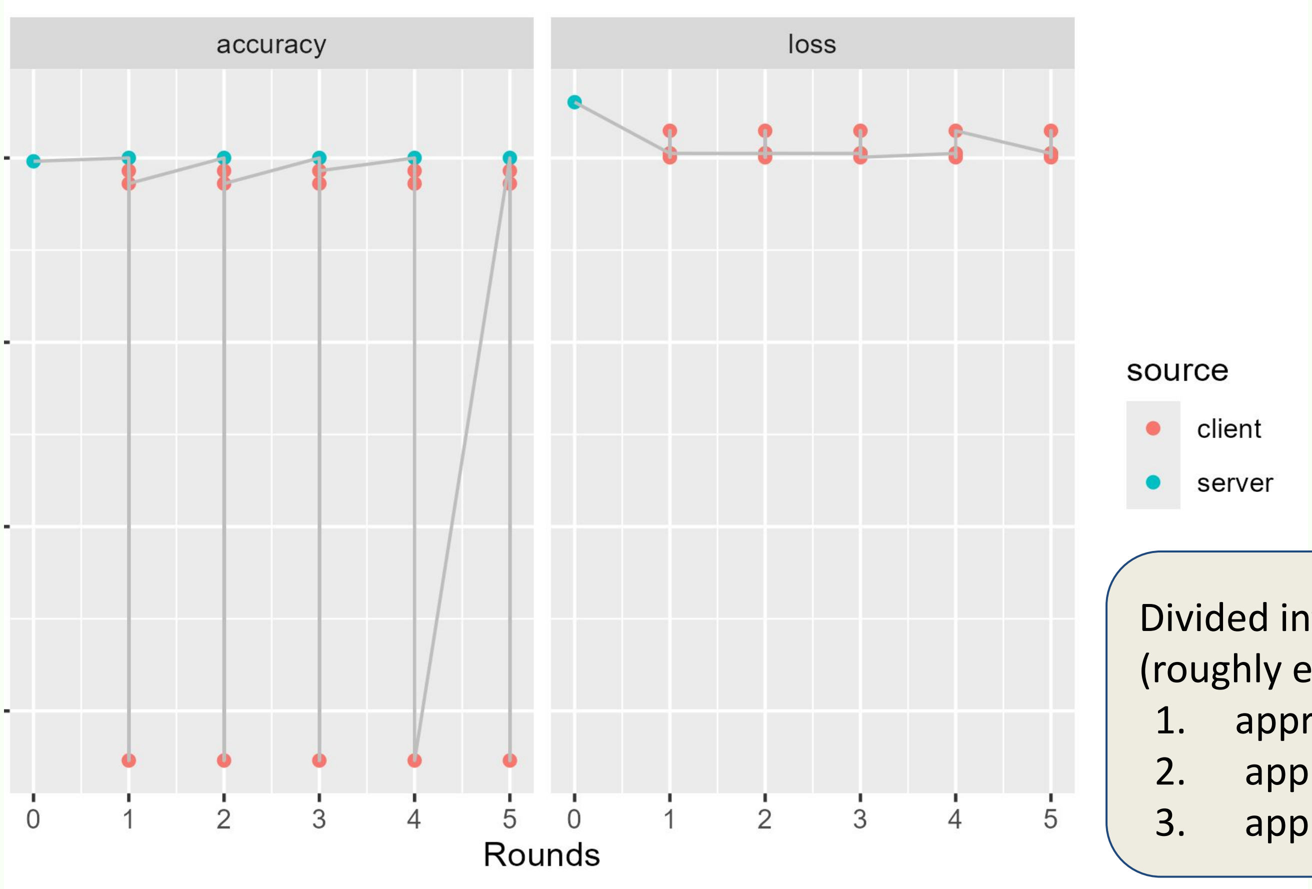
Loss and Accuracy for Splitting in Eight Groups



Loss and Accuracy for Splitting in Nine Groups



Loss and Accuracy for Splitting in Three Groups by Age

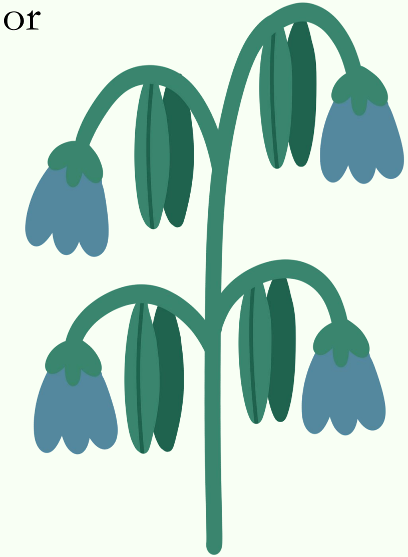


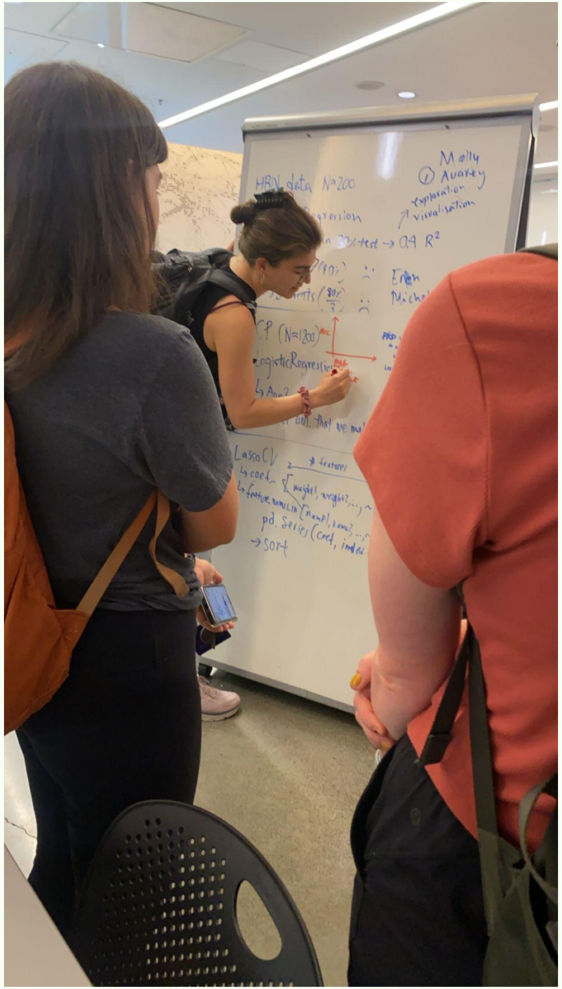
Divided into 3 groups by age (roughly equal N):

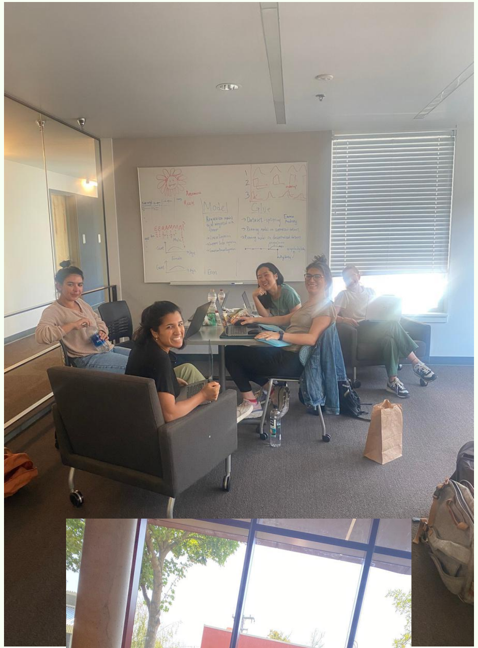
1. approx. age 5-9 years
2. approx age 9-12.5 years
3. approx 12.5-21 years

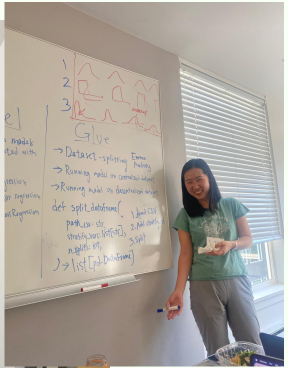
Challenges (but also what we learned…)

- Setting up - software installation, environments (time consuming)
- New to coding languages – especially Python!
- Linear Regression challenges
- Plotting brain features on MNI atlases using Connectome Workbench and/or R – we learned a lot about how to do this for the future!




Molly
Aubrey
exploration
visualisation
LassoCV
pd. Series
sort




Glue
Dataset-splitting
def split_dataframe(


IM HACKED

Thank you!

Github

https://github.com/mollyolzinski/Federated-Learning