# Federated Learning and Privacy Protection

Dirk Bergemann
Yale

Alessandro Bonatti
MIT

Mert Demirer
MIT

Vod Vilfort
MIT

**Discussion:**

Jacopo Perego
Columbia

# An Example



Garmin has $50$ million smartwatch users $N = 50$ mil

For each user, the device records:

- Daily sleep habits $X_i$ $T = 365$ days
- Daily stress level $Y_i$
- The user's age $Z_i$

# An Example



Garmin has $50$ million smartwatch users $N = 50$ mil

For each user, the device records:

- Daily sleep habits $X_i$ $T = 365$ days
- Daily stress level $Y_i$
- The user's age $Z_i$

Garmin would like to estimate a linear model: (why? e.g., send recs)

$$Y_i = \beta_i^x X_i + \beta^z Z_i + \varepsilon$$

Note: $\beta_i^x$ is user-specific while $\beta^z$ is not, gains from sharing data

# The Problem



There are at least two ways of estimating this model:

1. Garmin runs a separate regression for each user $i$

   No data sharing required ("on device") but estimators are poor

2. Garmin runs one regression for all users

   Better estimates but requires users to share sensitive data $(Y_i, X_i, Z_i)$

# The Problem



There are at least two ways of estimating this model:

1. Garmin runs a separate regression for each user $i$

   No data sharing required ("on device") but estimators are poor

2. Garmin runs one regression for all users

   Better estimates but requires users to share sensitive data $(Y_i, X_i, Z_i)$

Is there a third way?

# The Problem



There are at least two ways of estimating this model:

1. Garmin runs a separate regression for each user $i$

   No data sharing required ("on device") but estimators are poor

2. Garmin runs one regression for all users

   Better estimates but requires users to share sensitive data $(Y_i, X_i, Z_i)$

Is there a third way?

**This paper.** Yes, you can have your cake and eat it too

- A method to get learning benefits of [2] while bearing little of its costs

# Solution: Step 1, Partialling Out



Goal is to estimate: $Y_i = \beta_i^x X_i + \beta^z Z_i + \varepsilon$

**Idea:**

- Users are asked to share with Garmin only $(X_i, Z_i)$ and not $Y_i$

# Solution: Step 1, Partialling Out



Goal is to estimate: $Y_i = \beta_i^x X_i + \beta^z Z_i + \varepsilon$

**Idea:**

- Users are asked to share with Garmin only $(X_i, Z_i)$ and not $Y_i$
- Garmin merges $(X_i, Z_i)_{i \in N}$ and estimates $X = \gamma Z + \eta$
- Then computes individualized residuals $\hat{\eta}_i = X_i - \hat{\gamma} Z_i$

# Solution: Step 1, Partialling Out



Goal is to estimate: $Y_i = \beta_i^x X_i + \beta^z Z_i + \varepsilon$

**Idea:**

- Users are asked to share with Garmin only $(X_i, Z_i)$ and not $Y_i$
- Garmin merges $(X_i, Z_i)_{i \in N}$ and estimates $X = \gamma Z + \eta$
- Then computes individualized residuals $\hat{\eta}_i = X_i - \hat{\gamma} Z_i$
- And returns it to user $i$'s device and locally estimates:

$$Y_i = \delta_i \hat{\eta}_i + u$$

# Solution: Step 1, Partialling Out



Goal is to estimate: $Y_i = \beta_i^x X_i + \beta^z Z_i + \varepsilon$

**Idea:**

- Users are asked to share with Garmin only $(X_i, Z_i)$ and not $Y_i$
- Garmin merges $(X_i, Z_i)_{i \in N}$ and estimates $X = \gamma Z + \eta$
- Then computes individualized residuals $\hat{\eta}_i = X_i - \hat{\gamma} Z_i$
- And returns it to user $i$'s device and locally estimates:

$$Y_i = \delta_i \hat{\eta}_i + u$$

By Frisch-Waugh Theorem, $\hat{\delta}_i = \hat{\beta}_i^x$

**Takeaway:** To correctly estimate $\hat{\beta}_i^x$, users don't need to share $Y_i$

# Solution: Step 2, Regressing Means



Yet, sharing $(X_i, Z_i)$ may be a lot to ask

Main contribution is to show that, under some conditions, we can do better than partialling

# Solution: Step 2, Regressing Means



Yet, sharing $(X_i, Z_i)$ may be a lot to ask

Main contribution is to show that, under some conditions, we can do better than partialling

**Idea:**

- What if rather than sharing $(X_i, Z_i)$, user $i$ only shares the means $(\bar{X}_i, \bar{Z}_i)$?
- Garmin estimates $\bar{X} = \hat{\gamma}\bar{Z} + \hat{\eta}$ rather than $X = \gamma Z + \eta$
- Even less data is shared, more privacy friendly

# Solution: Step 2, Regressing Means



Yet, sharing $(X_i, Z_i)$ may be a lot to ask

Main contribution is to show that, under some conditions, we can do better than partialling

**Idea:**

- What if rather than sharing $(X_i, Z_i)$, user $i$ only shares the means $(\bar{X}_i, \bar{Z}_i)$?
- Garmin estimates $\bar{X} = \hat{\gamma}\bar{Z} + \hat{\eta}$ rather than $X = \gamma Z + \eta$
- Even less data is shared, more privacy friendly

**Main result:** If there is enough variation in $\bar{Z}_i$, $\hat{\gamma}$ is a consistent estimator of $\gamma$

# Comments



- Interesting, practically important, and creative solution illustrating how privacy and learning need not always be mutually exclusive

# Comments



- Interesting, practically important, and creative solution illustrating how privacy and learning need not always be mutually exclusive
- So far no incentives, only a stats problem. What about agents' incentive to disclose data? Multiple equilibria? What if agent does not reap the benefits of additional learning?

# Comments



- Interesting, practically important, and creative solution illustrating how privacy and learning need not always be mutually exclusive
- So far no incentives, only a stats problem. What about agents' incentive to disclose data? Multiple equilibria? What if agent does not reap the benefits of additional learning?
- Decreasing marginal return on data: Why can't Garmin simply compensate initial 1K agents for sharing all data, estimate common parameters, and run model for the remaining 49.9M with no extra data sharing?

  Few agents sharing a lot of data *vs* lots of agents sharing little data?

  A concrete leading application would be useful – not clear from slides

# Comments



- Interesting, practically important, and creative solution illustrating how privacy and learning need not always be mutually exclusive
- So far no incentives, only a stats problem. What about agents' incentive to disclose data? Multiple equilibria? What if agent does not reap the benefits of additional learning?
- Decreasing marginal return on data: Why can't Garmin simply compensate initial 1K agents for sharing all data, estimate common parameters, and run model for the remaining 49.9M with no extra data sharing?

  Few agents sharing a lot of data *vs* lots of agents sharing little data?

  A concrete leading application would be useful – not clear from slides
- Slides made me think about other alternatives: model averaging, but also anonymized data? Secret sharing techniques? **Interesting agenda!**