HANDOUT 11 - WEEK 13 READING

Privacy-Preserving ML and Trade-Offs

CIS 350: AI for Cybersecurity - Smart Defense for the Digital Business

Colorado State University - Computer Information Systems

How to use this reading

This handout supports Week 13. Machine learning runs on data about PEOPLE. Here we learn how to learn from that data without exposing individuals: differential privacy, federated learning, and encrypted computation - plus the real trade-offs between privacy, accuracy, and security. The figures and experiment match Lab 11; the review questions map to Quiz 10 and Exam 3.

Estimated reading time: 20-25 minutes. No prior coding background needed.

1. Why Privacy Matters in ML

Machine learning is hungry for data - and in security that data is deeply personal: logins, locations, device IDs, even email content. PRIVACY is people's right to control information about themselves; PII is Personally Identifiable Information. Laws like GDPR and HIPAA (Week 12) require protecting it, and a breach or misuse harms both people and the business.

> **Anonymization alone is weak**
>
> Recall Week 10: models can memorize and leak training data (membership inference, model inversion). And 'just remove names' fails because quasi-identifiers - zip code + birth date + device - can re-identify people. We need principled techniques, not just name-removal.

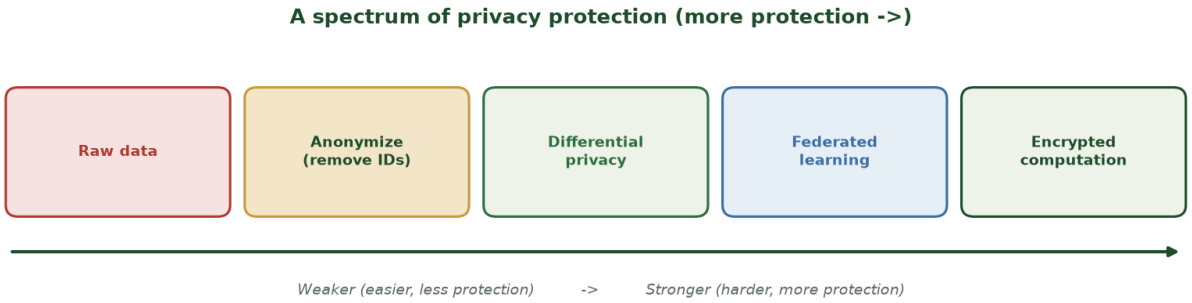

Figure 1. A spectrum of protection: raw data -> anonymize -> differential privacy -> federated learning -> encrypted computation.

2. Differential Privacy: Add Calibrated Noise

Differential privacy (DP) adds carefully-tuned random NOISE so that no single person's data changes the result much - you still learn useful aggregate patterns, but individuals stay hidden. It comes with a mathematical guarantee, unlike ad-hoc anonymization. The privacy budget EPSILON controls it: smaller epsilon = more noise = more privacy (and less accuracy). Apple, Google, and the U.S. 2020 Census all use DP.

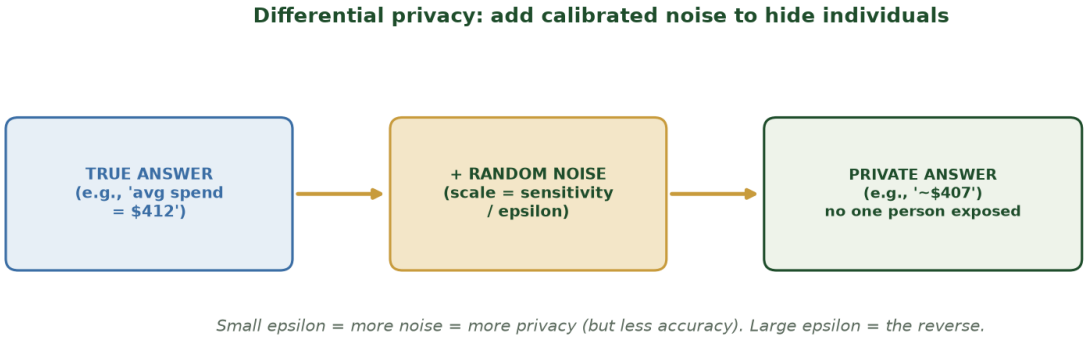

Figure 2. DP: add noise (scale = sensitivity / epsilon) to the true answer to protect individuals.

In Lab 11 you add Laplace noise to the training data at several epsilons and measure accuracy. The noise

scale is simply sensitivity / epsilon:

```
# Differential privacy by input perturbation (Lab 11)
scale = SENSITIVITY / epsilon          # smaller epsilon -> bigger noise
noisy = np.clip(X, -2, 2) + rng.laplace(0.0, scale, X.shape)
model = LogisticRegression().fit(noisy, y)   # train on the noisy data
```

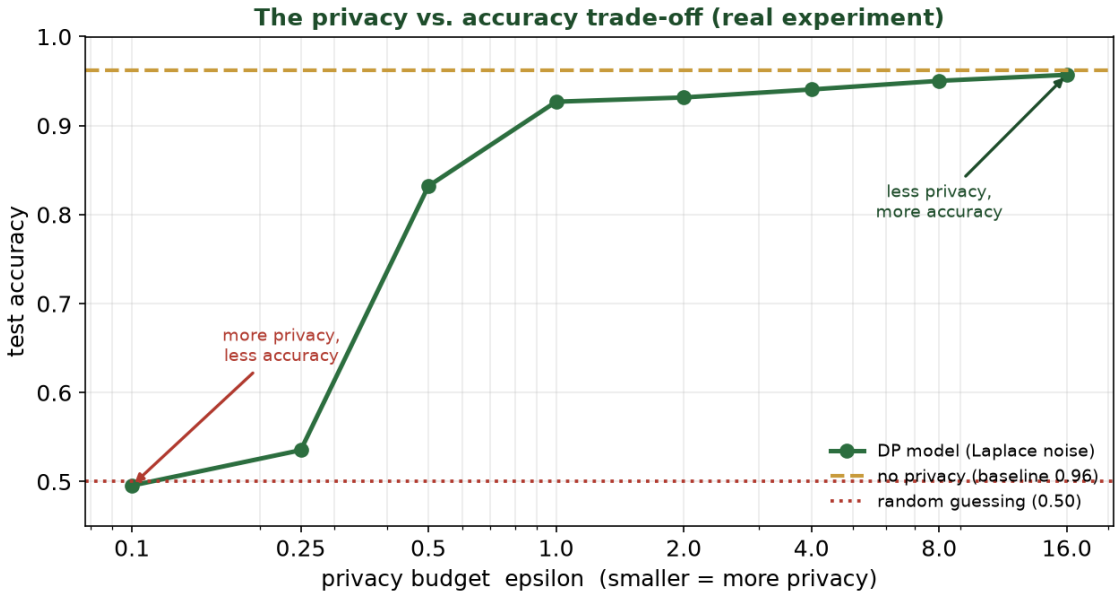

Figure 3. The real trade-off: tiny epsilon (0.1) collapses to ~0.50 (guessing); epsilon=1 already reaches ~0.93 vs the 0.96 no-privacy baseline. Strong privacy is often affordable.

3. Federated Learning: Bring the Model to the Data

Instead of collecting everyone's data in one place, federated learning keeps data ON the device. Each device trains locally and sends only model UPDATES (not raw data) to a server, which combines them into a shared model. Your phone's keyboard suggestions are trained this way, and it lets hospitals or banks - which legally cannot pool raw records - still build a shared model.

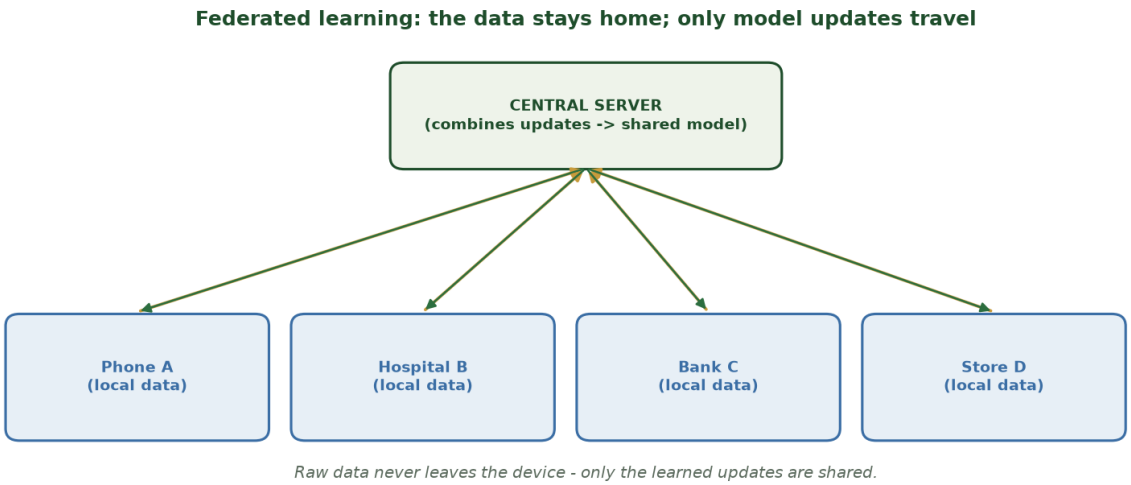

Figure 4. Federated learning: raw data stays on each device; only learned updates travel to the server.

4. Encrypted Computation

Encrypted computation lets you compute on data while it stays ENCRYPTED - it is never decrypted. Homomorphic encryption does math on ciphertext; secure multiparty computation lets several parties compute a joint result while each hides its input. This is extremely strong privacy, but it is currently slow and complex, so it fits narrow, high-value tasks today.

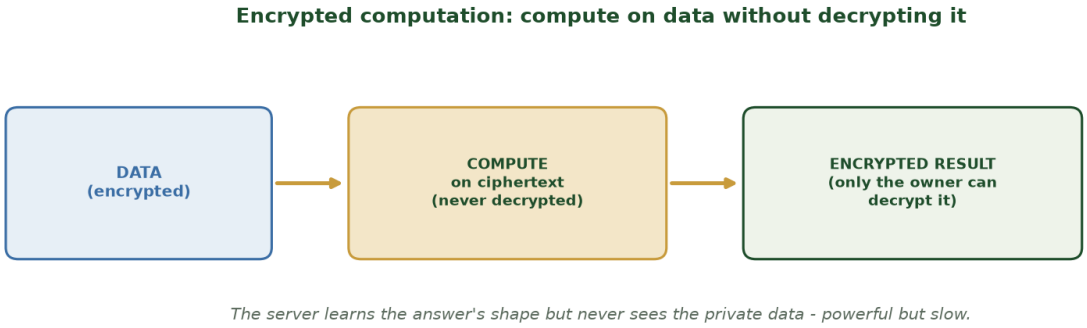

Figure 5. Encrypted computation: process ciphertext; only the data owner can decrypt the result.

Techniques combine

These are not either/or. Real systems often combine them - for example, federated learning PLUS differential privacy on the shared updates - to get stronger protection than any one alone.

5. The Trade-Offs

Privacy is not free. Two trade-offs matter:

- Privacy vs. accuracy: stronger DP (smaller epsilon) means more noise and lower accuracy - but as Figure 3 shows, the cost is often small once epsilon >= ~1.
- Security vs. privacy: the same tool can help one and hurt the other. Strong encryption protects privacy

but hides malicious traffic from defenders; collecting more data aids detection but erodes privacy.

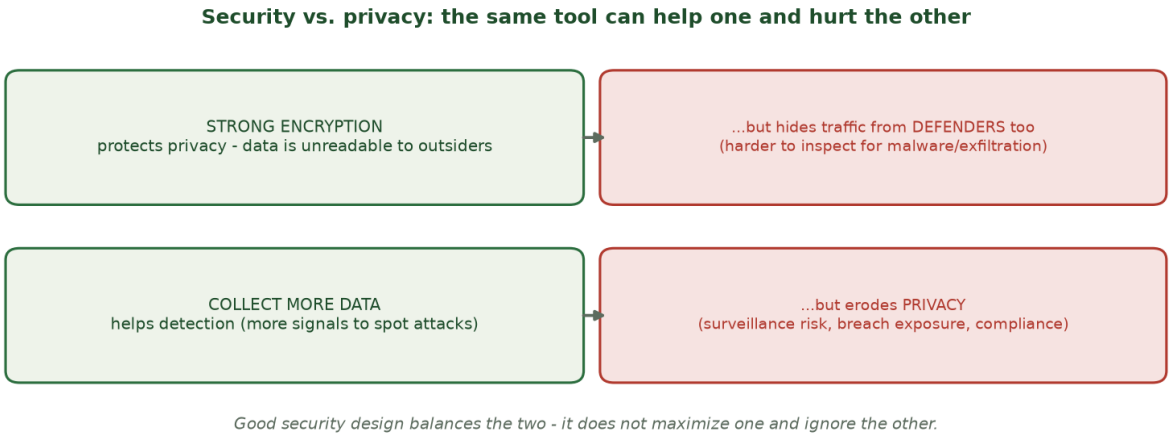

Figure 6. Security vs. privacy: encryption and data collection each help one goal and hurt the other - design for a deliberate balance.

Balance, don't maximize

The mature approach - like defense in depth and governance from earlier weeks - is a deliberate, DOCUMENTED balance with data minimization and legal input, not maximizing surveillance or privacy blindly. Real SecOps dilemmas (inspecting encrypted traffic, logging everything, sharing threat data, employee monitoring) rarely have a single 'correct' answer - they have a defensible one.

Key Terms

- Privacy / PII - control over personal data / personally identifiable information.
- Differential privacy - add calibrated noise for a tunable privacy guarantee.
- Epsilon - the privacy budget; smaller = more privacy, less accuracy.
- Sensitivity - how much one record can change a result (sets the noise scale).
- Federated learning - train on-device; share only model updates.
- Encrypted computation - compute on data without decrypting it.
- Re-identification - recovering identities from 'anonymized' data via quasi-identifiers.
- Security-vs-privacy trade-off - the same control can help one and harm the other.

Review Questions (self-check for Quiz 10, Lab 11, and Exam 3)

- Why is 'just remove names' not enough to protect privacy?
- What does epsilon control in differential privacy, and which direction is more private?
- Describe the privacy vs. accuracy trade-off in your own words.
- How does federated learning protect privacy, and how does it differ from encrypted computation?
- Give one security-vs-privacy dilemma and a balanced response.
- Why do real systems sometimes combine these techniques?

Remember: AI tools are not permitted on quizzes, labs, or exams. Privacy is a design choice - make it deliberately and document it.

References

- C. Dwork & A. Roth, 'The Algorithmic Foundations of Differential Privacy,' 2014.
- NIST SP 800-226 (2023), Guidelines for Evaluating Differential Privacy Guarantees; NIST Privacy Framework.
- H. B. McMahan et al., 'Communication-Efficient Learning of Deep Networks' (federated learning), 2017.
- U.S. Census Bureau - differential privacy in the 2020 Census (case study).
- NIST AI RMF 1.0 (2023); EU GDPR (2018).
- R. Shokri et al., membership inference (2017); M. Fredrikson et al., model inversion (2015).