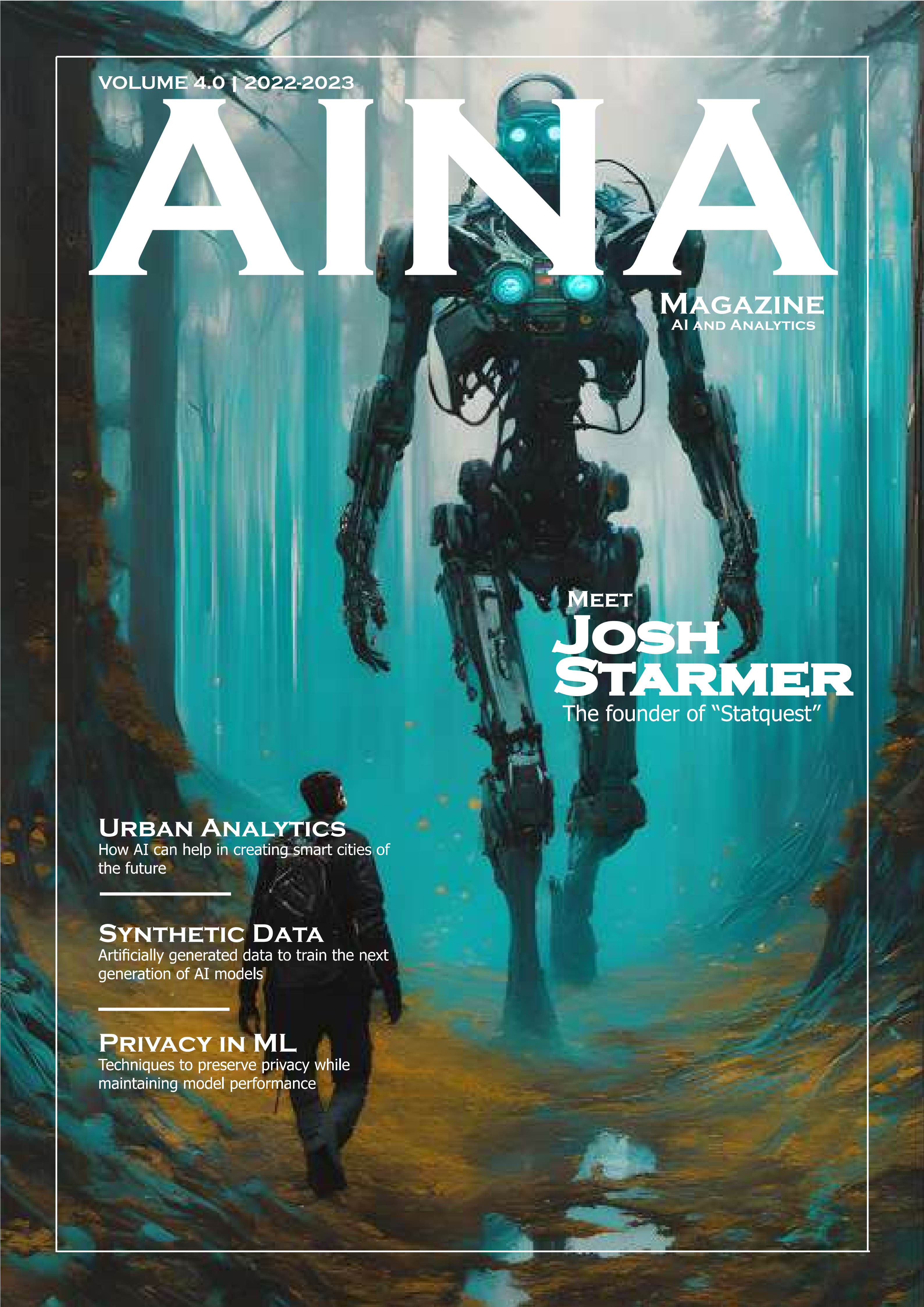


VOLUME 4.0 | 2022-2023

AINA

MAGAZINE

AI AND ANALYTICS

MEET JOSH STARMER

The founder of "Statquest"

URBAN ANALYTICS

How AI can help in creating smart cities of the future

SYNTHETIC DATA

Artificially generated data to train the next generation of AI models

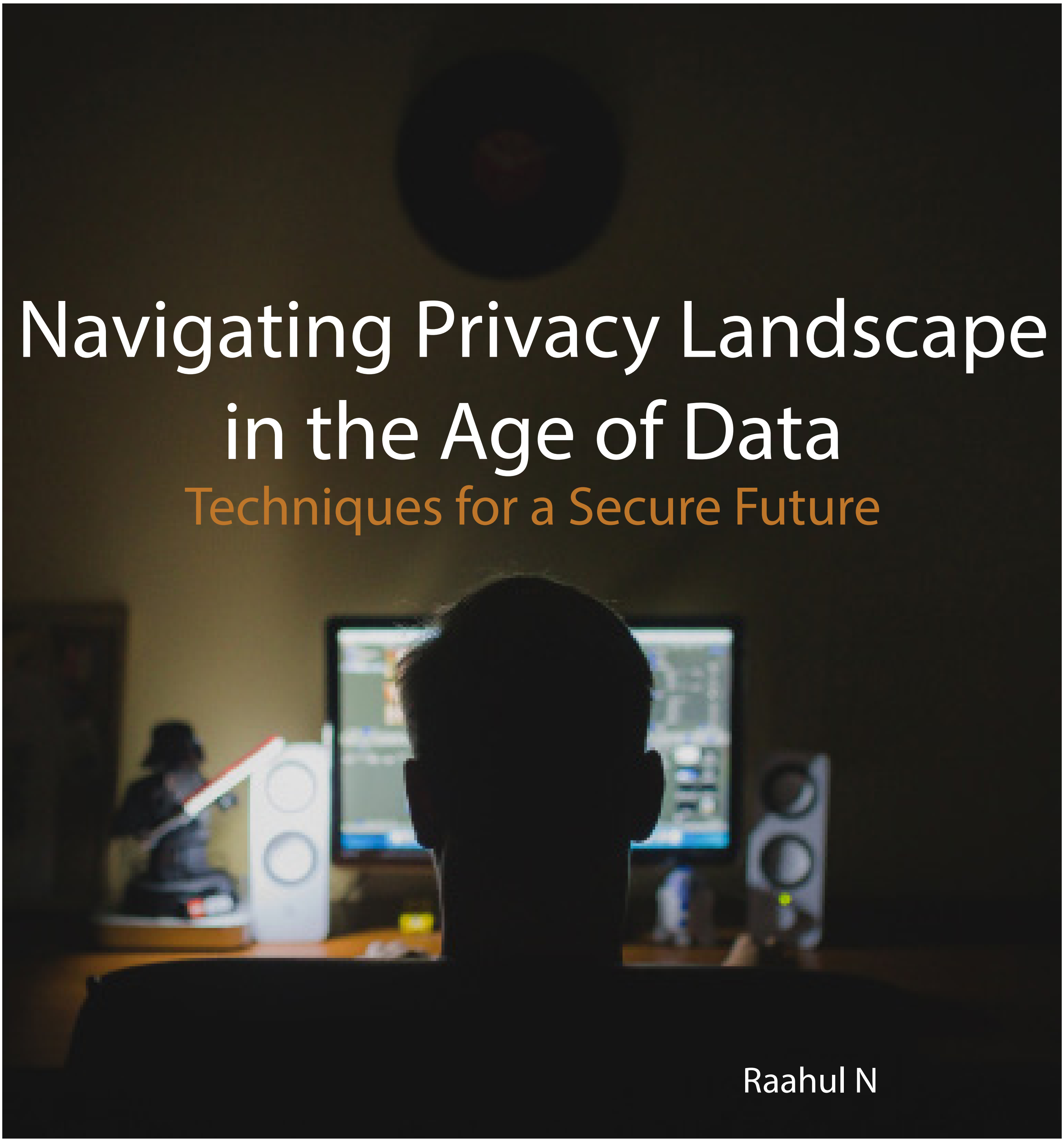
PRIVACY IN ML

Techniques to preserve privacy while maintaining model performance

Navigating Privacy Landscape in the Age of Data

Techniques for a Secure Future

Raahul N

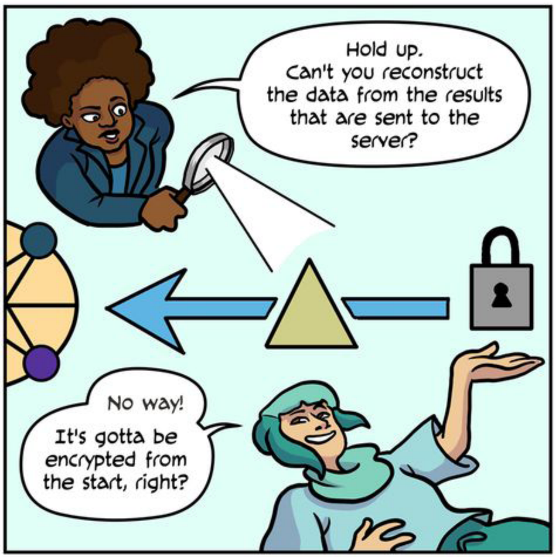


"Arguing that you don't care about the right to privacy because you have nothing to hide is no different than saying you don't care about free speech because you have nothing to say"

- Edward Snowden

It was early June 2013. Edward Snowden, a former NSA(US) staff, revealed to the world about the surveillance programs of the U.S. Government. The classified NSA documents leaked by Snowden brought the topic of "problematic surveillance programs" to international attention through journalists.

Around the same time, a British consulting firm (Cambridge Analytica) collected about 87 million Facebook users' personal data via Meta's Open Graph Algorithm & built psychological profiles using specific questionnaires. These profiles were allegedly used for analytical assistance in US presidential campaigns in 2016. The same firm was also widely accused of interfering with the UK's Brexit referendum. These 2 major events brought to public awareness the extent of psychological targeting using personal data.


Hold up.
Can't you reconstruct the data from the results that are sent to the server?
No way!
It's gotta be encrypted from the start, right?

If you are an individual or an organisation working with your users' data for insights or any interested common citizen, this article aims to guide you through some privacy-preserving techniques in machine learning with a 10,000 ft overview of some of their working principles.

Specifically, we'll be going through Differential Privacy, Secure Multiparty Computation, Federated Learning and Homomorphic encryption.

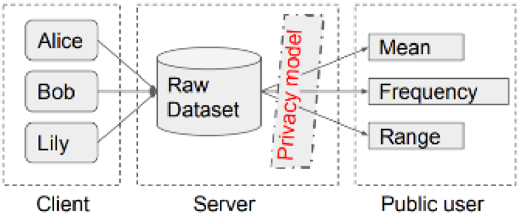
Differential Privacy

One of the most widely used techniques for privacy-preserving in machine learning is differential privacy. Differential privacy is a mathematical framework that provides strong guarantees of privacy by adding a controlled amount of noise to the data before it is analysed. This helps to ensure that the output of the analysis does not reveal any sensitive information about individuals who contributed the data.

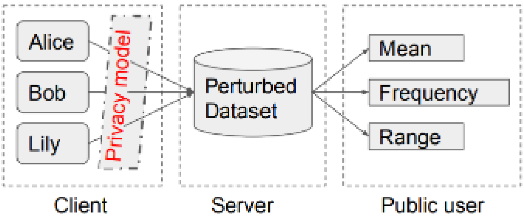
In the Indian context, the Digital Personal Data Protection Bill (withdrawn now owing to recommendations received through public consulting) is in the works and we can expect it in the coming years. These techniques include practices such as data anonymisation, encryption, which help protect sensitive information while still allowing for analysis and insights. These techniques enable organisations to strike a balance between utilising data for valuable insights and maintaining the privacy and rights of individuals, thereby promoting a responsible and ethical data-driven environment.

Since the Cambridge Analytica incident, controversy centre Facebook (now Meta) has always been under the radar for privacy violations & targeted ads. The firm has been fined billions (with a b) of dollars for numerous data breaches & privacy violations. These incidents in the last decade sparked public interest in online privacy and dangers of misuse of personal data. Ever since big techs & governments across the world are striving to meet the people's need for their privacy.

Frameworks like GDPR were set in place by governments to protect individuals from being exploited with their personal data. For people new to the 'Online Privacy' scene, General Data Protection Regulation (GDPR) is a comprehensive privacy law brought forth by the UK Govt. that sets guidelines for the collection, processing, and storage of personal data of individuals. The aim is to protect the privacy and rights of individuals by giving them control over their personal information. As a result, organisations are required to implement privacy-preserving techniques to ensure compliance with GDPR and safeguard individuals.



(a) Centralized differential privacy



(b) Local differential privacy

The concept of differential privacy is based on the idea of adding randomness to the data such that the analysis cannot be used to identify individual participants. However, the analysing algorithm is still able to produce statistically meaningful results without compromising the privacy of the data subjects. One intuition for this would be to think about aggregates. A mean of a large data, with random noise added to individual values would still remain the same. Some popular techniques that use differential privacy include randomised response, local differential privacy, and global differential privacy. One example of an application of this is in medical data. Differential privacy enables the analysis of sensitive patient data, electronic health records, and genomic data while maintaining patient privacy.

Differentialprivacyaddsrandomness to data to avoid user identification

While global differential privacy model collects all the users' data and releases the perturbed version. This poses a privacy risk, as time and again we have seen, that any single data curator cannot be trusted. Local differential privacy solves this by perturbing the data before it leaves the device. Now only the owner of the data can access the original data, which provides much stronger privacy protection for the user.

If we take machine learning into account, this involves incorporating the concept of differential privacy into the training process of the ML model. One approach involves adding noise to the gradient of the loss function (differentially private stochastic gradient descent or DP-SGD) during the training process.

$$\theta_{t+1} \leftarrow \theta_t - \eta \cdot (\nabla_t + b_t).$$

Here the model parameters (θ) are updated by subtracting this gradient multiplied by a small constant (η). Gaussian noise (bt) is added to their sum to obtain the indistinguishability needed for Differential Privacy. This helps to ensure that the updates to the model are not overly influenced by any one individual's data. Another approach involves using a differentially private data synthesiser to generate a synthetic dataset that can be used to train the model.

Secure Multiparty Computation

Secure multiparty computation (MPC), as the name suggests, is a cryptographic technique that involves multiple parties jointly computing a function on their private inputs without revealing their individual inputs to each other. The goal is to preserve privacy and confidentiality while obtaining the desired computation results.

MPC achieves this by utilising cryptographic protocols and techniques such as secret sharing, secure function evaluation, and secure two-party computation. Here's a high-level explanation of the technique:

1. Secret Sharing: The private inputs of each party are divided into shares using a secret sharing scheme. Each party holds a share of their own input and does not have complete knowledge of the inputs of other parties.

2. Secure Function Evaluation: The parties then perform a series of computations on their shares while exchanging information in a secure manner. This involves executing cryptographic protocols that allow the parties to compute intermediate results without revealing their individual inputs.

3. Secure Two-Party Computation: In some cases, MPC involves only two parties. Secure two-party computation protocols ensure that the computation can be performed while preserving privacy and confidentiality. These protocols utilise cryptographic techniques such as garbled circuits, oblivious transfer and zero-knowledge proofs.

4. Consistent Output: Through the secure computation process, each party learns the final result of the joint computation without gaining any information about the other party's private input. The output is consistent with what would have been obtained if the computation had been performed on the combined inputs directly.

For example, if we are computing the average height of three people (Ayesha, Bindu and Catherine) without revealing individual heights.

Secure multiparty computation involves multiple parties jointly computing a function on their private inputs

- Say Ayesha's height is 140cm. Here, 140cm is split into 144cm, -11cm and 7cm

- Ayesha keeps one of the 3 pieces with herself and shares the other two to others.

- The same steps are followed by Bindu & Catherine.

Ayesha	Bindu	Catherine	
144	-11	7	140cm
-6	132	24	150cm
20	0	140	160cm
158	121	171	

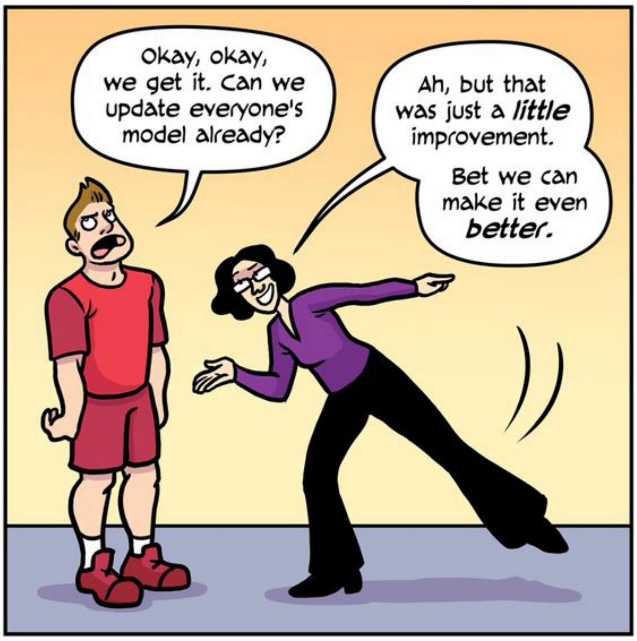
Sum of their encrypted pieces
= (158+121+171) = 450cm

Average height = 450/3 = 150cm

Federated Learning

Federated learning is a privacy-preserving approach in machine learning where data is kept decentralised and computation is performed locally. Instead of sending raw data to a central server, federated learning allows devices or entities to collaboratively learn from their respective data without sharing it directly.

Here's how it works: The learning process takes place on individual devices or edge nodes, such as smartphones or IoT devices, which possess their own local datasets. These devices train a model using their local data, and instead of sending the data itself, they send model updates or gradients to a central server.


Okay, okay, we get it. Can we update everyone's model already?
Ah, but that was just a *little* improvement.
Bet we can make it even *better*.

The central server then aggregates these updates from multiple devices to create a global model that represents the collective knowledge learned from all participants. The updated global model is then sent back to each device, allowing them to improve their local models based on the collective insights.

Federated learning provides several advantages. Firstly, it enhances privacy since the raw data remains on the local devices and is not shared directly with the central server. This mitigates concerns about data breaches and unauthorised access. Secondly, it enables collaboration on a large scale, allowing diverse entities to contribute their knowledge without having to centralise the data. Lastly, it promotes efficiency by reducing the need for large-scale data transfers, especially in scenarios where data size or connectivity is limited. One of the advantages of federated learning is that it allows the model to be trained on a more diverse dataset, which can improve the accuracy of the model. It also reduces the risk of a data breach, as the data never leaves the local device. Some examples of federated learning applications include mobile keyboard prediction and medical image analysis.

Homographic Encryption

Homomorphic encryption is a remarkable technique in cryptography that allows computation on encrypted data without needing to decrypt it first. This means that you can add, subtract, multiply, and divide encrypted data without knowing what the unencrypted data is. Homomorphic encryption makes it possible to train a machine-learning model on encrypted medical records without reading the actual records.

Homographicencryptionallowscomputationon encrypteddatawithoutneedingtodecryptitfirst

Let's say you have two encrypted numbers, and you want to add them together. With homomorphic encryption, you can perform the addition operation directly on the encrypted data, yielding the encrypted result. No one can peek inside the box and see the actual numbers or the result, but you still get the correct result!

Wide range of potential applications of homomorphic encryption include:
- E-voting
- Healthcare data sharing
- Financial transactions

Homomorphic encryption is still a relatively new technology, and there are some challenges that need to be addressed before it can be widely adopted.

One challenge is that homomorphic encryption is computationally expensive. Another challenge is that homomorphic encryption is not yet as secure as traditional encryption methods. Despite these challenges, homomorphic encryption is a promising technology with the potential to revolutionise the way we protect and use data.

Privacy-preserving techniques and technologies are becoming increasingly important in machine learning and data science. Differential privacy, secure multiparty computation, federated learning, and differential privacy for machine learning are just a few of the techniques that are being used to address these concerns. As the field continues to evolve, it is likely that we will see new and innovative techniques emerge that further enhance privacy preservation in machine learning and data science.

As mentioned earlier by embracing these techniques, we can strike a balance between extracting valuable insights from data and upholding the privacy rights of individuals. Let us remain vigilant, adapt to emerging techniques, and collectively work towards a future where privacy and data-driven innovation coexist harmoniously.