Voting. Secure electronic voting, in a simple form, is simply computation of the addition function which tallies the vote. Privacy and non-malleability of the vote (properties discussed above in the context of auctions) are essential for similar technical reasons. Additionally, because voting is a fundamental civil process, these properties are often asserted by legislation.

As a side note, we remark that voting is an example of an application which may require properties *not covered* by the standard MPC security definitions. In particular, the property of *coercion resistance* is not standard in MPC (but can be formally expressed and achieved (Küsters *et al.*, 2012)). The issue here is the ability of voters to *prove* to a third party how they voted. If such a proof is possible (e.g., a proof might exhibit the randomness used in generating the vote, which the adversary may have seen), then voter coercion is also possible. We don't delve into the specific aspects of secure voting beyond listing it here as a natural application of MPC.

Secure machine learning. MPC can be used to enable privacy in both the inference and training phases of machine learning systems.

Oblivious model inference allows a client to submit a request to a server holding a pre-trained model, keeping the request private from the server $\mathcal{S}$ and the model private from the client C. In this setting, the inputs to the MPC are the private model from $\mathcal{S}$, and the private test input from C, and the output (decoded only for C) is the model's prediction. An example of recent work in this setting include MiniONN (Liu *et al.*, 2017), which provided a mechanism for allowing any standard neural network to be converted to an oblivious model service using a combination of MPC and homomorphic encryption techniques.

In the training phase, MPC can be used to enable a group of parties to train a model based on their combined data without exposing that data. For the large scale data sets needed for most machine learning applications, it is not feasible to perform training across private data sets as a generic many-party computation. Instead, hybrid approaches have been designed that combine MPC with homomorphic encryption (Nikolaenko *et al.*, 2013b; Gascón *et al.*, 2017) or develop custom protocols to perform secure arithmetic operations efficiently (Mohassel and Zhang, 2017). These approaches can scale to data sets containing many millions of elements.

Other applications. Many other interesting applications have been proposed for using MPC to enable privacy. A few examples include privacy-preserving network security monitoring (Burkhart *et al.*, 2010), privacy-preserving genomics (Wang *et al.*, 2015a; Jagadeesh *et al.*, 2017), private stable matching (Doerner *et al.*, 2016), contact discovery (Li *et al.*, 2013; De Cristofaro *et al.*, 2013), ad conversion (Kreuter, 2017), and spam filtering on encrypted email (Gupta *et al.*, 2017).

1.3.1 Deployments

Although MPC has seen much success as a research area and in experimental use, we are still in the early stages of deploying MPC solutions to real problems. Successful deployment of an MPC protocol to solve a problem involving independent and mutually distrusting data owners requires addressing a number of challenging problems beyond the MPC execution itself. Examples of these problems include building confidence in the system that will execute the protocol, understanding what sensitive information might be inferred from the revealed output of the MPC, and enabling decision makers charged with protecting sensitive data but without technical cryptography background to understand the security implications of participating in the MPC.

Despite these challenges, there have been several successful deployments of MPC and a number of companies now focus on providing MPC-based solutions. We emphasize that in this early stage of MPC penetration and awareness, MPC is primarily deployed as an *enabler* of data sharing. In other words, organizations are typically not seeking to use MPC to add a layer of privacy in an otherwise viable application (we believe this is yet forthcoming). Rather, MPC is used to enable a feature or an entire application, which otherwise would not be possible (or would require trust in specialized hardware), due to the value of the shared data, protective privacy legislation, or mistrust of the participants.

Danish sugar beets auction. In what is widely considered to be the first commercial application of MPC, Danish researchers collaborated with the Danish government and stakeholders to create an auction and bidding platform for sugar beet production contracts. As reported in Bogetoft *et al.* (2009), bid privacy and auction security were seen as essential for auction participants.