Self-driving database management systems (DBMSs) are a new family of DBMSs that can optimize themselves for better performance without human intervention. Self-driving DBMSs use machine learning (ML) models that predict system behaviors and make planning decisions based on the workload the system sees. These ML models are trained using metrics produced by different components running inside the system. Self-driving DBMSs are a challenging environment for these models, as they require a significant amount of training data that must be representative of the specific database the model is running on. To obtain such data, self-driving DBMSs must generate this training data themselves in an online setting. This data generation, however, imposes a performance overhead during query execution. Many DBMSs operate in a distributed master-replica architecture, where the master node sends new changes to replica nodes that hold up-to-date copies of the database. We propose a novel technique named Replicated Training that utilizes existing database replicas to generate training data for models used in self-driving DBMSs. This approach load balances the expensive task of data collection across the distributed architecture, as opposed to being done entirely in the master node. It also provides the advantage of more diverse training data in the case where replicas are running in different hardware environments. Under Replicated Training, each replica can dynamically control training data collection if it needs more resources to keep up with the master node. To show the effectiveness of our technique, we implement it in NoisePage, a self-driving DBMS, and evaluate it in a distributed environment. Our experiments show that training data collection in a DBMS incurs a noticeable performance overhead in the master node, and using Replicated Training reduces this overhead while still ensuring that replicas keep up with the master. Finally, we show that Replicated Training produces ML models that have accuracies comparable to those trained solely on the master node.

Website: [https://gustavoangulo.github.io/thesis.pdf](https://gustavoangulo.github.io/thesis.pdf)

Thesis Committee:
Andy Pavlo and Dave Andersen