CERN openlab & IBM Research Workshop

Trip Report

Jakob Blomer, Javier Cervantes, Pere Mato, Radu Popescu

2018-12-03

Workshop Organization

- 1 full day at IBM Research Zürich
- ~25 participants from CERN
- ~10 staff from IBM
- Second joint workshop on AI technologies planned 11 December at CERN
- **Goal: identify ground for common research activities**

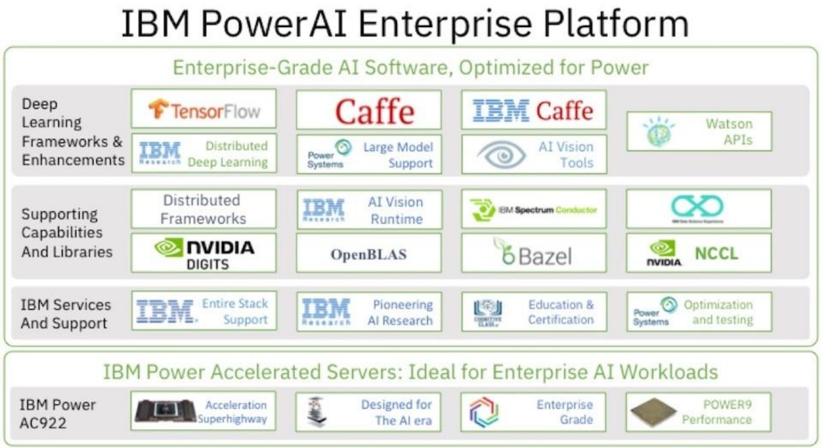
- Morning session: presentation and open discussion on key technologies by IBM engineers
  - AI software kit for generalized linear models (“Snap-ML”)
  - NVlink CPU-GPU interconnect
  - Near-memory and in-memory computing
- Afternoon session: split between quantum computing and <u>storage technologies</u>
  - Forecast of tape drive evolution
  - AI based prediction of data popularity
  - Apache Crail: “Spark for fast (NVM-like) storage”

Artificial Intelligence software kit

- Increasing demand in AI from all sectors
- Synergies between Hardware and Software are crucial
- Three main research topics:
  - Software OpenSource Framework
  - Hardware optimization
  - Software - Hardware integration

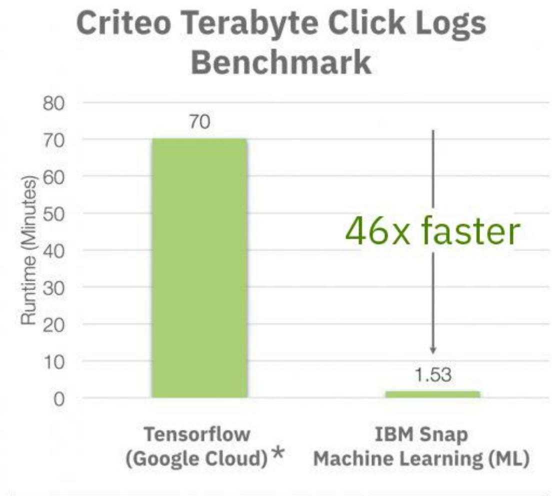
IBM PowerAI Platform

- Environment for data science as a service
  - Deep learning and Machine learning more accessible
  - Built on opensource tools
  - Accelerated IBM Power servers, optimized for:
    - Distributed Deep Learning (DDL)
    - Deep Learning Inference (DLI)
    - Scheduling work at HW level
      (Distributed GPU's)
    - Machine Learning 46x faster
      (Same algorithm, diff HW)



SnapML Framework

- Library for Fast-training of generalized linear models
  - Only supports models most widely used (Based on Kaggle 2017 survey)
- Benefits from optimized HW architectures (IBM Power, NVIDIA GPU's)
- Aim to remove training time as a bottleneck



Comparison of Tensorflow on Google Cloud with Snap ML on POWER9 (AC922) cluster

Workload: Click-through-rate prediction for computational advertising, using Logistic Regression

Dataset: Criteo Terabyte Click Logs, 4.2 billion training examples, 1 million features

Model: Logistic Regression

Test LogLoss: 0.1293 (Tensorflow), 0.1292 (Snap ML)

Platform: 89 machines (Tensorflow) compared to 8 Power9 CPUs + 16 NVIDIA Tesla V100 GPUs (Snap ML)

IBM **Research**

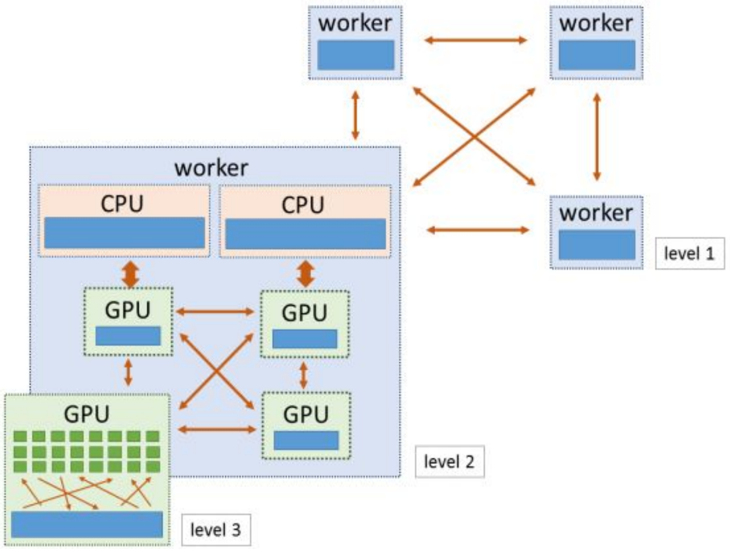
SnapML Framework

- Features
  - Distributed training
  - GPU acceleration
  - Supports sparse data structures
- 3 levels of parallelism
  - Data-parallelism across worker nodes in a cluster
  - Parallelism across heterogeneous compute units within one worker node
  - Multi-core parallelism within individual compute units
- 3 API's
  - `Snap-ml-local`

    scikit-learn-like interface for training on a single machine
  - Snap-ml-mpi

    distributed training of ML models across a cluster of machines
  - Snap-ml-spark

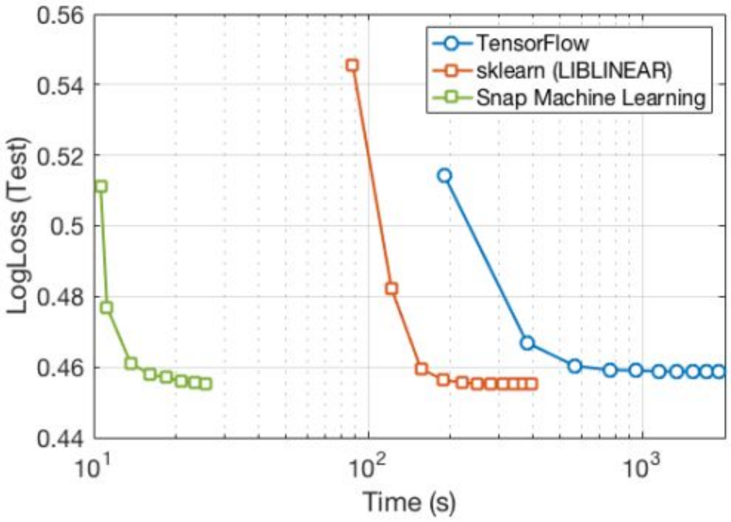
    spark.ml-like interface, integration with pySpark applications



Source: https://arxiv.org/pdf/1803.06333.pdf

SnapML + IBM Hardware

Single-Node Performance



- *Scikit-learn*: single-threaded, w/o GPU (dataset in CPU mem)
- *TensorFlow*: multi-threaded, one GPU (batch mode)
- *Snap ML*: multi-threaded, one GPU (dataset in GPU mem)

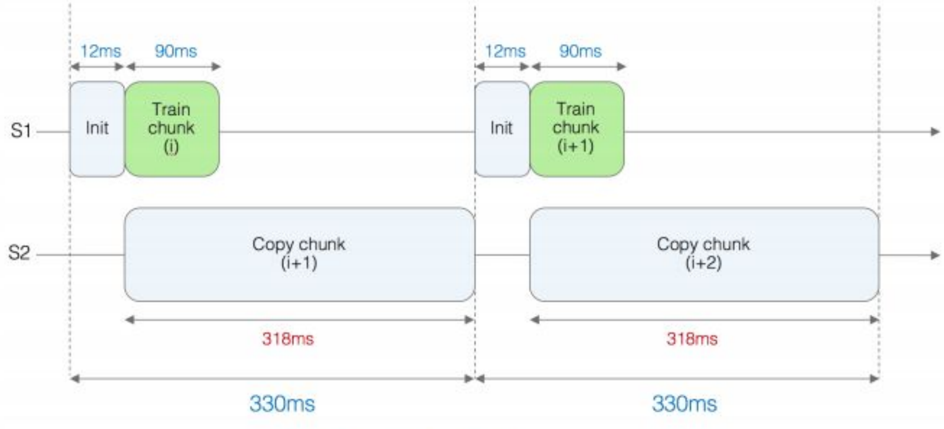
Difference between TF and SKlearn can be explained by the highly optimized C++ backend of scikit-learn for workloads that fit in memory, whereas TensorFlow processes data in batches

Difference between TF and SnapML not well defined.

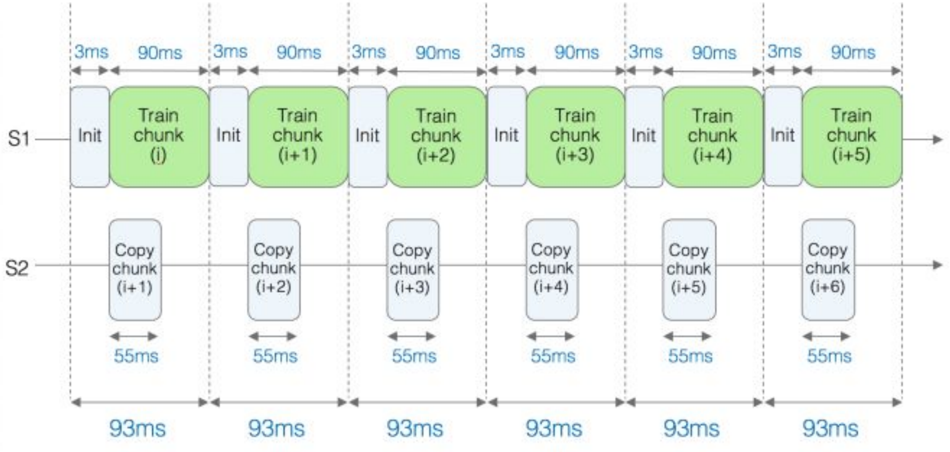
Cluster of 4 IBM Power Systems AC922 servers
Each server has 4 NVIDIA Tesla V100 GPUs attached via the NVLINK 2.0 interface

SnapML + IBM Hardware

Out-of-core Performance



(a) PCIe Gen 3 Interconnect.



(b) NVLINK 2.0 Interconnect.

- Dataset does not fit into memory
- NVLINK 2.0 speed-up hides the data copy time behind the kernel execution, effectively removing the copy between CPU-GPU time from the critical path and resulting in a 3.5x speed-up.

Cluster of 4 IBM Power Systems AC922 servers
Each server has 4 NVIDIA Tesla V100 GPUs attached via the NVLINK 2.0 interface

SnapML + IBM Hardware

Tera-scale Benchmark

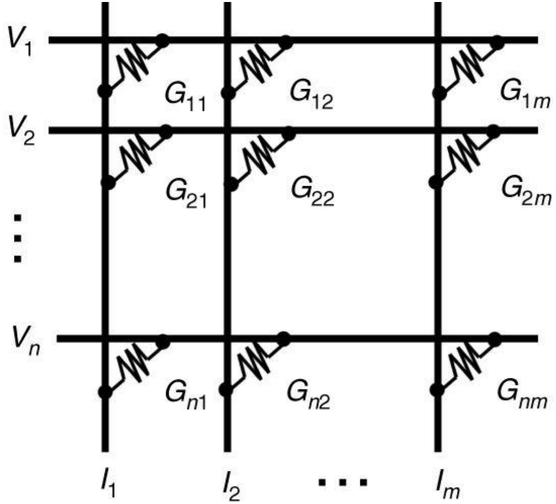
- Click-through rate prediction (CTR)
- Classification task
- **2.3 TB** training data
- SnapML:
  1.53 minutes including data loading, initialization, training and testing time.
- **46x faster** than the best previously reported results, obtained using TensorFlow

Cluster of 4 IBM Power Systems AC922 servers
Each server has 4 NVIDIA Tesla V100 GPUs attached via the NVLINK 2.0 interface

Near-memory and in-memory computing

- Near memory: programmable FPGA between memory and CPU that allows manipulating memory controller behavior
  - E.g. dynamically adjusting the precision of floating point values
  - Gather values from memory in cache-line optimized layout
  - Follow pointer chains such as virtual function calls

- In-memory computing: use physics of phase-change memory chip for (analog) matrix-vector multiplication
  - Can speed-up forward and backward propagation for deep-learning



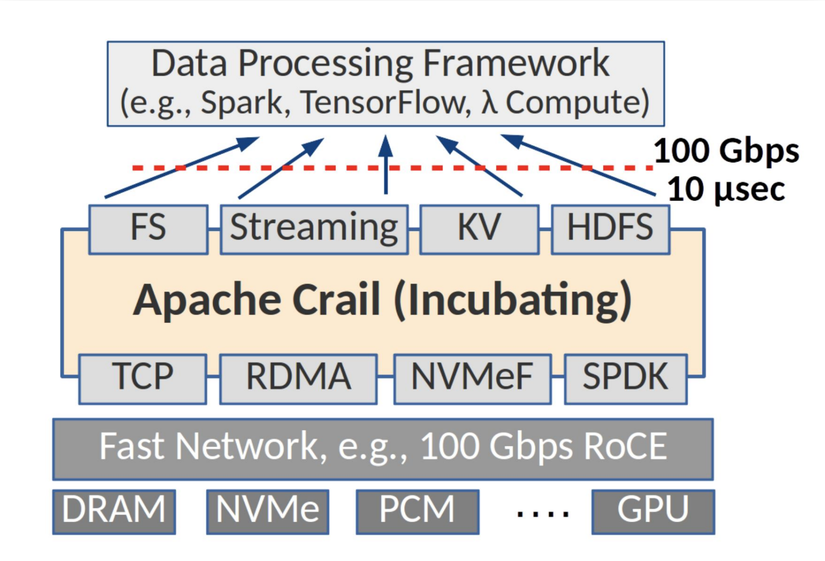
Data Storage: Tape is (still) relevant!

- Bit density improvements in HDD are flattening out
  - Energy assisted writing techniques are quite challenging
- Bit density on tape much larger than on HDD
  - Clear roadmap for the next 2 decades
  - Software challenge: predict data temperature and optimize tape transfers
- Archival storage on optical drives has no well-defined roadmap

- But: IBM remains the only vendor of tape hardware

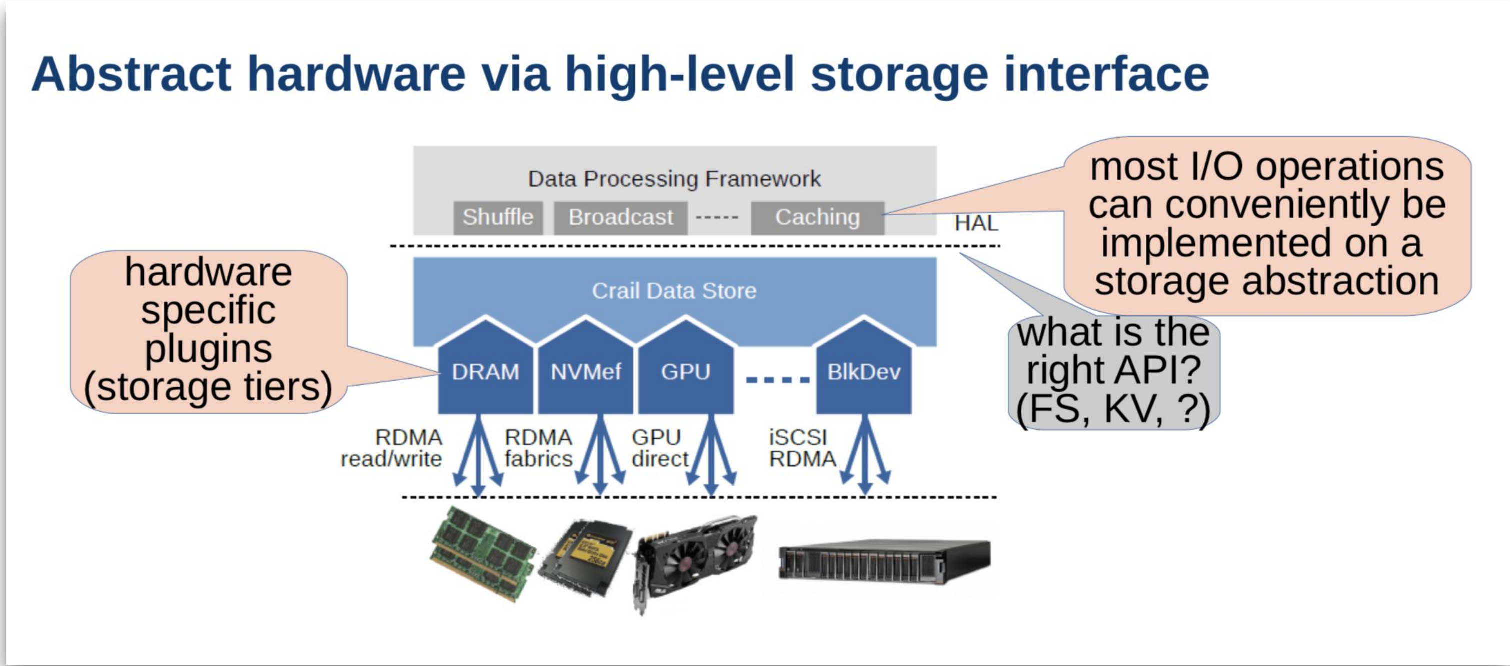
High Performance Distributed Data Store

- Apache CRAIL
  - Distributed storage middleware optimized for ephemeral data on fast devices
  - Agnostic to the framework (Spark, Tensorflow, ...)
  - Plugins (API) for different FS
  - Use of kernel bypass techniques
  - Impressive benchmarks
    (factor 6 can be obtained by
    just changing the storage compared
    to off-the-shelf spark)
  - Apache incubator project
  - Requires conversion to a custom
    format*

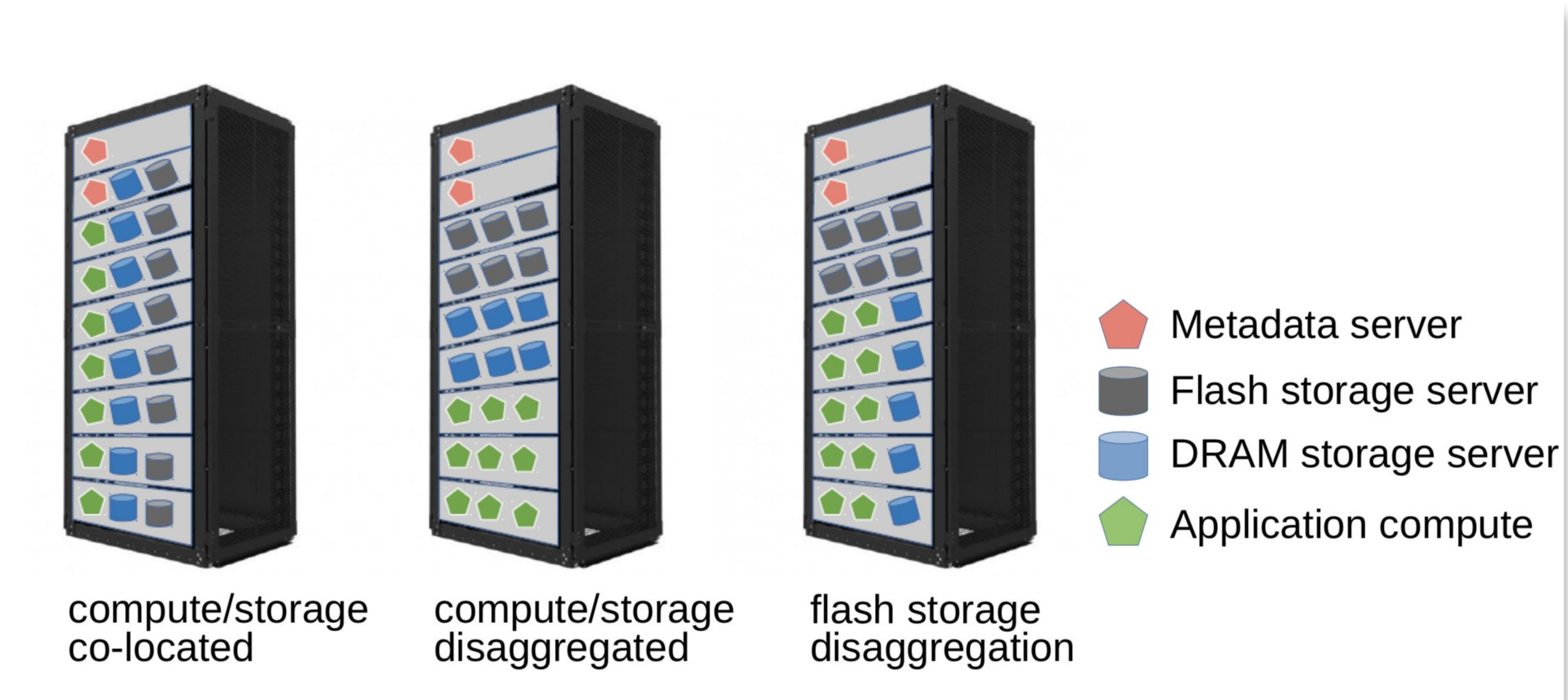


*To be further discussed

Crail: Hardware Abstraction Layer



Crail: Deployment Modes



Summary

- IBM is looking for new clients to leverage its new products
- Strong focus on tailored hardware to optimized current ML and DL problems
- Innovative new technologies offering improvement of one order of magnitude

- Many interesting ideas and opened questions:
    - Possible R&D topics for distributed ROOT analysis cluster
        - RDataFrame + Apache Crail + RDataSource (if custom format were needed)?
    - Distributed analysis on the C++ side (SnapML internal kernels)
        - SnapML as a complement to TMVA?
    - Smarter movement between tape and disk based on predicted data popularity
    - HEP Data analysis + Near memory computing (FPGA Access processor)

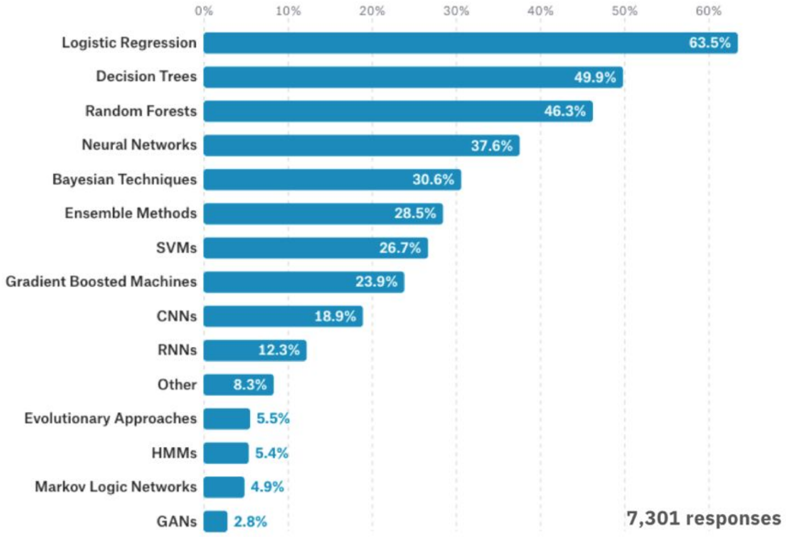
- CERN Contact persons were identified to discuss some of these topics more in depth

Backup slides

Model most widely used

What data science methods are used at work?

- Logistic regression is the most commonly reported data science method used at work for all industries except Military and Security where Neural Networks are used slightly more frequently.



Source: https://www.kaggle.com/surveys/2017

Automated ML

- Machine learning involves several manual jobs:
  - Feature selection, model selection, hyperparameter optimization, model ensembling, ...
- Challenge is to **automate all these processes**
  - No slides and limited relevant information online
- Working on a project where uses can specify **budget, time, architecture** among other parameters as input