The LF AI & Data Foundation is a project of The Linux Foundation that supports open source innovation in artificial intelligence, machine learning, deep learning and data open source projects. The LF AI & Data Foundation was created to support numerous technical projects within this important space.

With the LF AI & Data Foundation, members are working to create a neutral space for harmonization and acceleration of separate technical projects focused on AI, ML, DL and Data technologies.

For more information, please view the How to Get Involved deck.

Questions? Please email info@lfaidata.foundation.

**LF AI & Data Foundation**

- **Web Site**: [https://lfaidata.foundation/](https://lfaidata.foundation/)
- **Landscape**: [https://landscape.lfaidata.foundation/](https://landscape.lfaidata.foundation/)
- **GitHub**: [https://github.com/lfai](https://github.com/lfai)
- **Mail Lists**: [https://lists.lfaidata.foundation/g/main](https://lists.lfaidata.foundation/g/main)
- **Twitter**: @LFAIData_Fdn
- **Artwork**: [https://artwork.lfaidata.foundation](https://artwork.lfaidata.foundation)
- **Presentations**: [PowerPoint Template](#)
- **Email**: info@lfaidata.foundation
- **Slack**: [https://slack.lfaidata.foundation/](https://slack.lfaidata.foundation/)

**Current Projects**

<table>
<thead>
<tr>
<th>Project</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Help Us Improve the Wiki</td>
<td></td>
<td></td>
</tr>
<tr>
<td>This Wiki is owned by the LF AI Foundation Community. Contributions are always welcomed to help improve it! In the upper right of this page, select Log In to contribute. You will need a Linux Foundation ID (created at <a href="https://identity.linuxfoundation.org/">https://identity.linuxfoundation.org/</a>) to log in. For a Confluence overview, click here.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Welcome to the LF AI & Data Foundation wiki, where you will find information with a cross project focus. For individual projects, follow the links below.
Acumos is an Open Source Platform, which supports design, integration and deployment of AI models. Furthermore, Acumos supports an AI marketplace that empowers data scientists to publish adaptive AI models, while shielding them from the need to custom develop fully integrated solutions.

GitHub: https://github.com/acumos

Adlik is an end-to-end optimizing framework for deep learning models. The goal of Adlik is to accelerate deep learning inference process both on cloud and embedded environment.

GitHub: https://github.com/Adlik

Amundsen is a data discovery and metadata engine for improving the productivity of data analysts, data scientists and engineers when interacting with data.

GitHub: https://github.com/amundsen-io
Angel is a high-performance distributed machine learning platform based on the philosophy of Parameter Server. It is tuned for performance with big data from Tencent and has a wide range of applicability and stability, demonstrating increasing advantage in handling higher dimension model.

GitHub: https://github.com/Angel-ML/angel

Adversarial Robustness Toolbox (ART) provides tools that enable developers and researchers to evaluate, defend, certify and verify Machine Learning models and applications against the adversarial threats.

GitHub: https://github.com/Trusted-AI/adversarial-robustness-toolbox

AI Explainability 360 is an open source toolkit that can help users better understand the ways that machine learning models predict labels using a wide variety of techniques throughout the AI application lifecycle.

GitHub: https://github.com/Trusted-AI/AIX360
<table>
<thead>
<tr>
<th>INCUBATION</th>
<th>Description</th>
<th>GitHub</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AI Fairness 360</strong></td>
<td>AI Fairness 360 is an extensible open source toolkit that can help users understand and mitigate bias in machine learning models throughout the AI application lifecycle.</td>
<td><a href="https://github.com/Trusted-AI/AIF360">GitHub</a></td>
</tr>
<tr>
<td><strong>Datashim</strong></td>
<td>Datashim is enabling and accelerating data access for Kubernetes/Openshift workloads in a transparent and declarative way. It's opensource since September of 2019 and it is growing to support use-cases related to data access in AI projects.</td>
<td><a href="https://github.com/IBM/dataset-lifecycle-framework">GitHub</a></td>
</tr>
<tr>
<td><strong>DATAPRACTICES.ORG</strong></td>
<td>DataPractices.org was pioneered by data.world as a &quot;Manifesto for Data Practices&quot; of four values and 12 principles that illustrate the most effective, ethical, and modern approach to data teamwork. As a member of the foundation, datapractices.org will expand to offer open courseware and establish a collaborative approach to defining and refining data best practices.</td>
<td><a href="https://github.com/datadotworld/data-practices-site">Github</a></td>
</tr>
</tbody>
</table>
DELTA is a deep learning based end-to-end natural language and speech processing platform. DELTA aims to provide easy and fast experiences for using, deploying, and developing natural language processing and speech models for both academia and industry use cases. DELTA is mainly implemented using TensorFlow and Python 3.

GitHub: https://github.com/didi/delta

Elastic Deep Learning (EDL) optimizes the global utilization of the cluster running deep learning job and the waiting time of job submitters. It includes two parts: a Kubernetes controller for the elastic scheduling of distributed deep learning jobs, and a fault-tolerable deep learning framework.

GitHub: https://github.com/PaddlePaddle/edl

Egeria is an open source project dedicated to making metadata open and automatically exchanged between tools and platforms, no matter which vendor they come from.

GitHub: https://github.com/odpi/egeria
Feast is an open source feature store for machine learning. It was developed as a collaboration between Gojek and Google in 2018. Feast aims to:

- Provide scalable and performant access to feature data for ML models during training or serving.
- Provide a consistent view of features for both training and serving.
- Enable re-use of features through discovery, documentation, and metadata tracking.
- Ensure model performance by tracking, validating, and monitoring features in production.

Flyte is a production-grade, declarative, structured and highly scalable cloud-native workflow orchestration platform. It allows users to describe their ML/Data pipelines using Python, Java or (in the future other languages) and Flyte manages the data flow, parallelization, scaling and orchestration of these pipelines. Flyte builds on top of Docker containers and kubernetes.

GitHub: https://github.com/flyteorg/flyte
**INCUBATION**  
ForestFlow is a scalable policy-based cloud-native machine learning model server. ForestFlow strives to strike a balance between the flexibility it offers data scientists and the adoption of standards while reducing friction between Data Science, Engineering and Operations teams.  
GitHub: [https://github.com/ForestFlow/ForestFlow](https://github.com/ForestFlow/ForestFlow)

**GRADUATE**  
Horovod, a distributed training framework for TensorFlow, Keras and PyTorch, improves speed, scale and resource allocation in machine learning training activities. Uber uses Horovod for self-driving vehicles, fraud detection, and trip forecasting. It is also being used by Alibaba, Amazon and NVIDIA. Contributors to the project outside Uber include Amazon, IBM, Intel and NVIDIA.  
GitHub: [https://github.com/horovod/horovod](https://github.com/horovod/horovod)
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>GitHub Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>JanusGraph</td>
<td>A scalable graph database optimized for storing and querying graphs containing hundreds of billions of vertices and edges distributed across a multi-machine cluster.</td>
<td><a href="https://github.com/janusgraph/janusgraph">https://github.com/janusgraph/janusgraph</a></td>
</tr>
<tr>
<td>Ludwig</td>
<td>A toolbox built on top of TensorFlow that allows to train and test deep learning models without the need to write code. All you need to provide is your data, a list of fields to use as inputs, and a list of fields to use as outputs, Ludwig will do the rest. Simple commands can be used to train models both locally and in a distributed way, and to use them to predict on new data.</td>
<td><a href="https://github.com/uber/ludwig">https://github.com/uber/ludwig</a></td>
</tr>
</tbody>
</table>
Marquez is an open source metadata service for the collection, aggregation, and visualization of a data ecosystem’s metadata. It maintains the provenance of how datasets are consumed and produced, provides global visibility into job runtime and frequency of dataset access, centralization of dataset lifecycle management, and much more.

GitHub: [https://github.com/MarquezProject](https://github.com/MarquezProject)

Milvus is an open source similarity search engine for massive-scale feature vectors. Built with heterogeneous computing architecture for the best cost efficiency. Searches over billion-scale vectors take only milliseconds with minimum computing resources. Milvus can be used in a wide variety of scenarios to boost AI development.

GitHub: [https://github.com/milvus-io](https://github.com/milvus-io)
INCUBATION

NNStreamer (Neural Network Support as Gstreamer Plugins) is a set of Gstreamer plugins that support ease and efficiency for Gstreamer developers adopting neural network models and neural network developers managing neural network pipelines and their filters.

GitHub: https://github.com/nnstreamer

INCUBATION

OpenDS4All is a project created to accelerate the creation of data science curricula at academic institutions. Our goal is to provide recommendations, slide sets, sample Jupyter notebooks, and other materials for creating, customizing, and delivering data science and data engineering education.

GitHub: https://github.com/opendi4all

GRADUATE

ONNX is an open format to represent deep learning models. With ONNX, AI developers can more easily move models between state-of-the-art tools and choose the combination that is best for them. ONNX is developed and supported by a community of partners.

GitHub: https://github.com/onnx
<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
</table>
| **GRADUATE** | Pyro is a universal probabilistic programming language (PPL) written in Python and supported by PyTorch on the backend. Pyro enables flexible and expressive deep probabilistic modeling, unifying the best of modern deep learning and Bayesian modeling.
GitHub: https://github.com/pyro-ppl/pyro |
| **SANDBOX** | RosaeNLG is a template-based Natural Language Generation (NLG) automates the production of relatively repetitive texts based on structured input data and textual templates, run by a NLG engine. Production usage is widespread in large corporations, especially in the financial industry.
GitHub: https://github.com/RosaeNLG/ |
SOAJS is an open source microservices and API management platform. SOAJS eliminates the IT plumbing challenges, so you can deploy microservices significantly earlier and faster. IT initiatives such as digital transformation are simplified, accelerated, cost reduced, and risk mitigated. Our fully integrated, world-class API lifecycle management, multi-cloud orchestration, release management, and IT Ops automation capabilities eliminate your IT organization’s modernization pain.

GitHub: https://github.com/soajs

Substra is a framework offering distributed orchestration of machine learning tasks among partners while guaranteeing secure and trustless traceability of all operations. It enables privacy-preserving federated learning projects, where multiple parties collaborate on a Machine Learning objective while each one keeps their private datasets behind their own firewall.

GitHub: https://github.com/SubstraFoundation/substra
sparklyr is an R package that lets you analyze data in Spark while using familiar tools in R. sparklyr supports a complete backend for dplyr, a popular tool for working with data frame objects both in memory and out of memory. You can use dplyr to translate R code into Spark SQL.

GitHub: https://github.com/sparklyr/sparklyr