

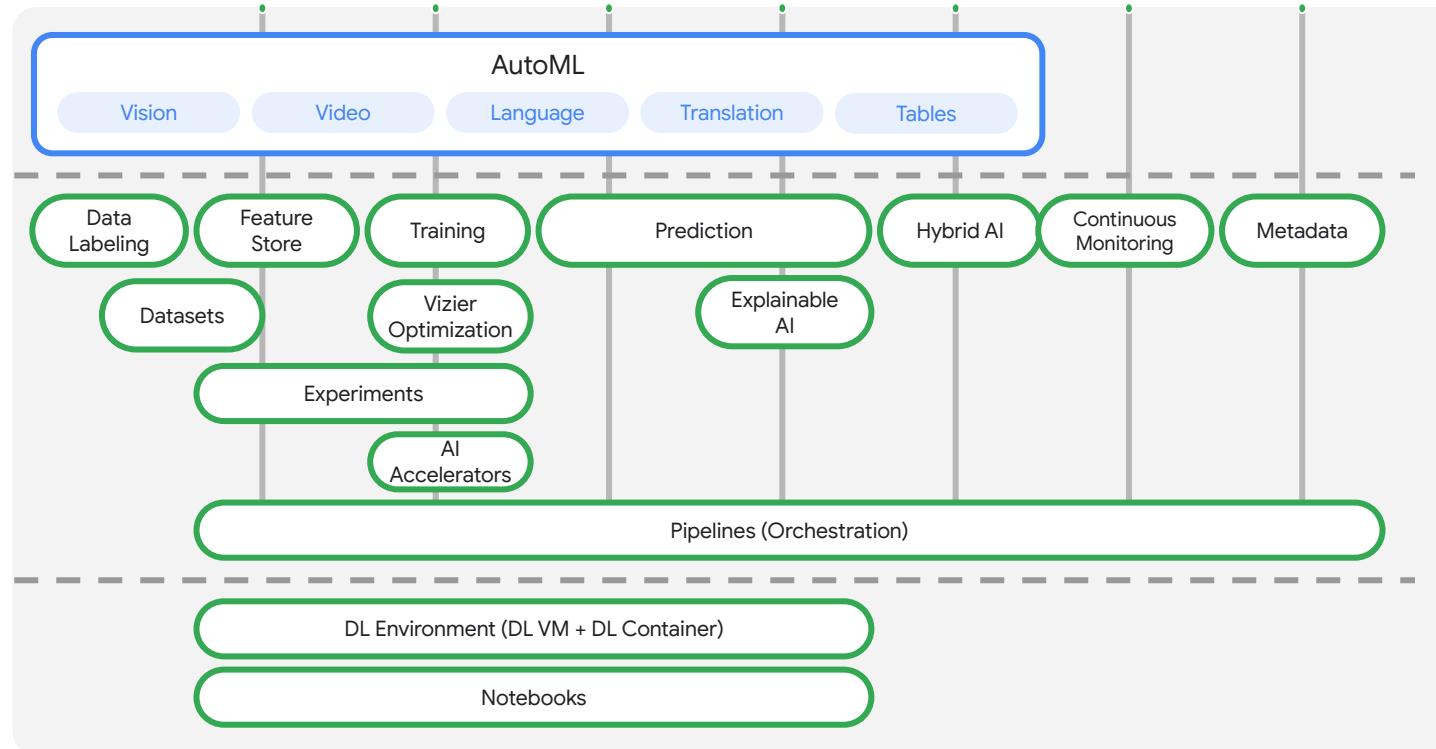
Slides That are used a screenshots in the `github/statmike/vertex-ai-mlops` repository

Notes:

- Save as PDF
- Copy to `github/statmike/vertex-ai-mlops/Dev`
- Convert PDF to PNG images
 - Use Notebook: `/Dev/Slides to PNG.ipynb`

Vertex AI Overview

Data Readiness → Feature Engineering → Training/ HP-Tuning → Model serving → Understanding/ Tuning → Edge → Model Monitoring → Model Management



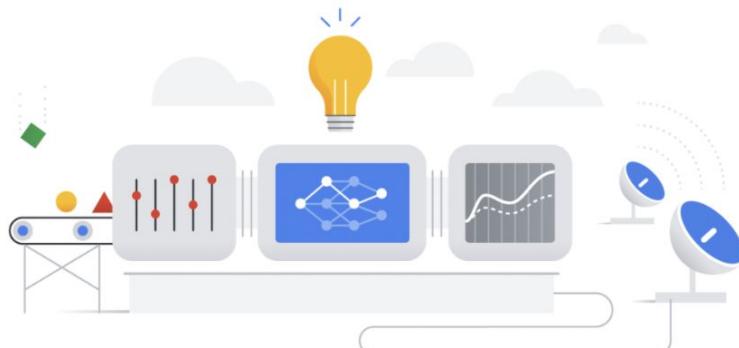
 Vertex AI

Dashboard

 Dashboard Datasets Features Labeling tasks Notebooks Pipelines Training Experiments Models Endpoints Batch predictions Metadata Marketplace

Get started with Vertex AI

Vertex AI empowers machine learning developers, data scientists, and data engineers to take their projects from ideation to deployment, quickly and cost-effectively. [Learn more](#)



Region

us-central1 (Iowa)

 Recent datasets

-  02c_digits_20210919213805 16 hours ago
-  02b_digits_20210919205707 20 hours ago
-  02a 4 days ago
-  02b_digits_20210916141540 4 days ago
-  02c_digits_20210916004500 5 days ago

[+ CREATE DATASET](#) Recent models

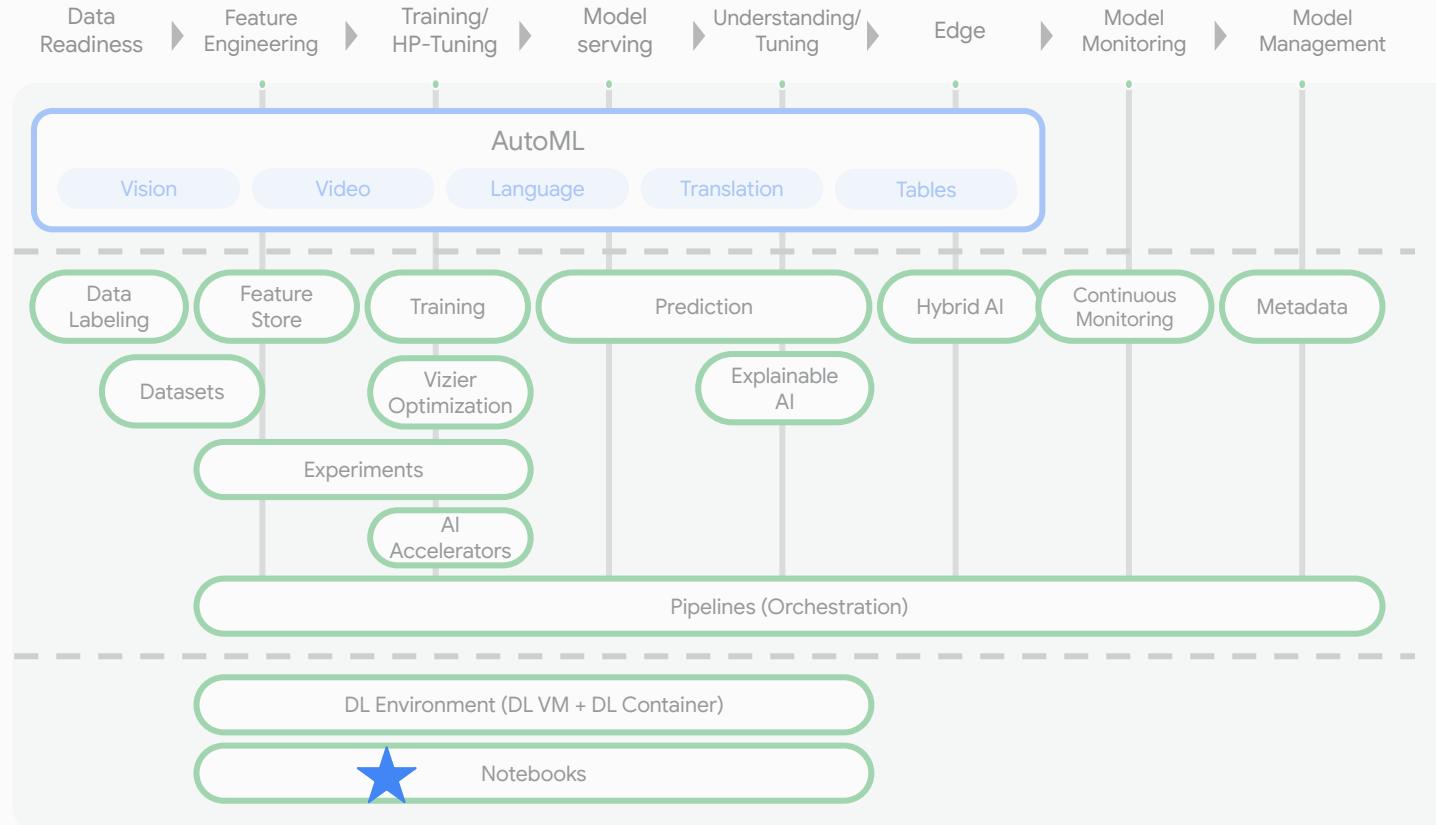
-  05f_digits_20210920145828 1 hour ago
-  05e_digits_20210920125450 3 hours ago
-  02c_digits_20210919213805 12 hours ago
Average precision: 1
-  02b_digits_20210919205707 19 hours ago
Average precision: 1
-  05c_digits_20210919214125-model 19 hours ago

[+ TRAIN NEW MODEL](#) Get predictions

After you train a model, you can use it to get predictions, either online as an endpoint or through batch requests

[+ CREATE BATCH PREDICTION](#)[Show debug panel](#)

Vertex AI Overview



- Vertex AI
- Dashboard
- Datasets
- Features
- Labeling tasks
- Notebooks**
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Marketplace

Notebooks **+ NEW INSTANCE** **REFRESH** **START** **STOP** **RESET** **DELETE** **SHOW INFO PANEL**

MANAGED NOTEBOOKS **PREVIEW** **INSTANCES** EXECUTIONS **PREVIEW** SCHEDULES **PREVIEW** SCHEDULED RUNS

Create and use Jupyter Notebooks with a notebook instance. Notebook instances have JupyterLab pre-installed and are configured with GPU-enabled machine learning frameworks. [Learn more](#)

Filter Enter property name or value

	Instance name	Zone	Auto-upgrade	Environment	Machine type	GPUs	Permission	Last modified
<input type="checkbox"/>	mlops2	us-central1-a	TensorFlow:2.3	4 vCPUs, 15 GB RAM	None	Service account	Sep 15, 2021, 10:05:03 PM	

OPEN JUPYTERLAB

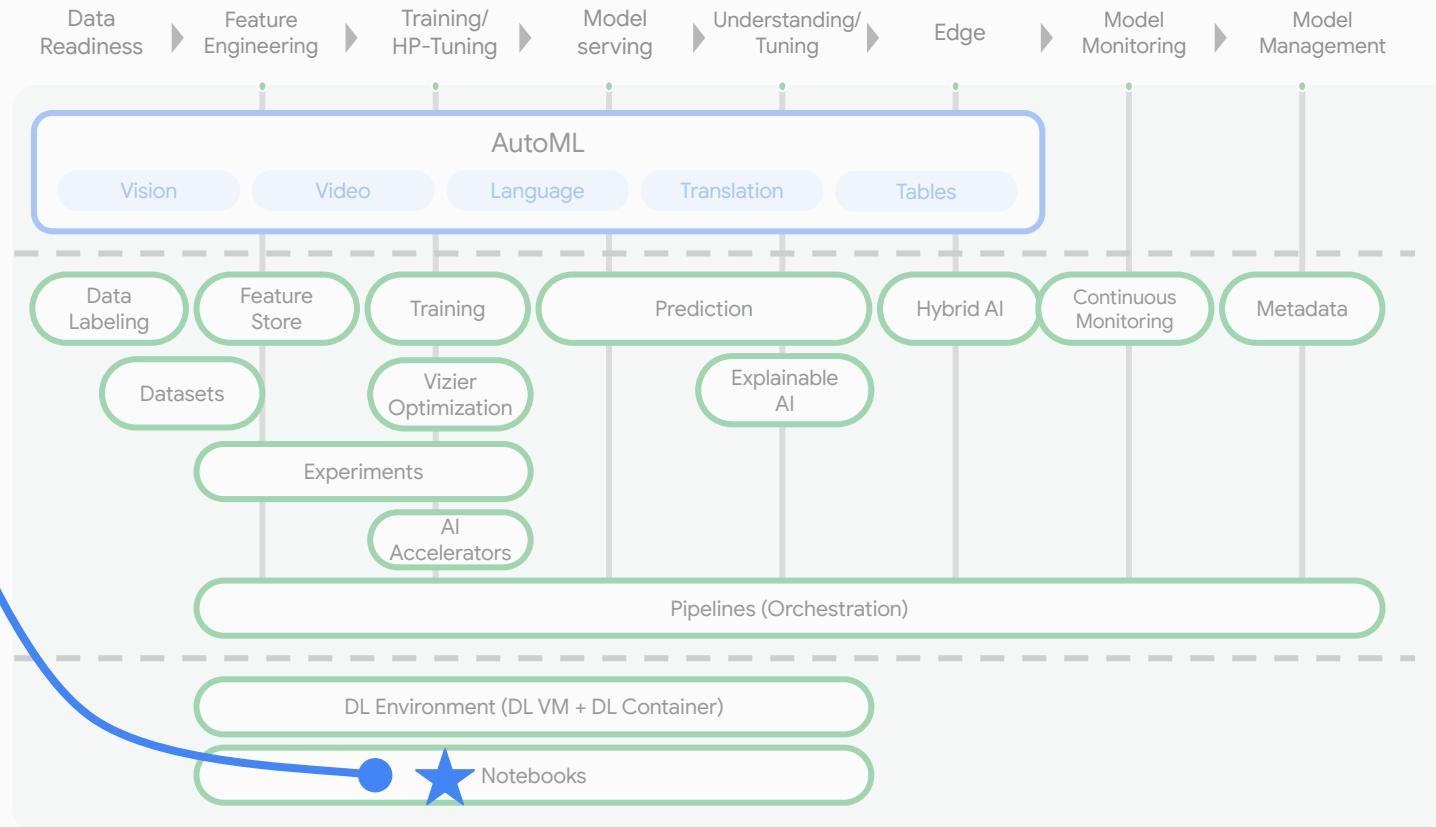
The JupyterLab interface shows a file tree on the left with notebooks like '00 - Environment Setup.ipynb' and '01 - BigQuery ...'. The main area displays code snippets for setup and imports.

```
[24]: REGION = 'us-central1'  
PROJECT_ID = 'statmike-mlops'  
DATANAME = 'digits'  
  
[25]: BUCKET = PROJECT_ID  
  
[26]: from google.cloud import storage  
import pandas as pd  
from sklearn import datasets
```

Mode: Command L1, Col 1 Show debug panel

01

Vertex AI Overview



Vertex AI

Dashboard

Datasets

Features

Labeling tasks

Notebooks

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

Marketplace

The screenshot shows a Jupyter Notebook interface with a sidebar navigation bar at the top. The sidebar includes options like File, Edit, View, Run, Kernel, Git, Tabs, Settings, and Help. Below the sidebar is a file tree showing a directory structure under '/vertex-ai-mlops/'. A yellow arrow points from the 'Notebooks' item in the sidebar to the 'Notebooks' tab in the file tree. The main content area displays a notebook titled '01 - BigQuery - Table Data'. The notebook content includes:

01 - BigQuery - Table Data Source

Use BigQuery to load and prepare data for machine learning:

Prerequisites:

- 00 - Environment Setup

Overview:

- Setup BigQuery
 - Create a Dataset
 - Use BigQuery Python Client
 - Create Tables
 - Copy from another Project:Dataset
 - SQL with BigQuery
 - Load data from GCS with BigQuery
 - BigQuery Python
 - Run SQL Queries to prepare data for analysis

Below the overview, there's a list of files in the directory:

- architectures
- Dev
- temp
- 00 - Environment Setup
- 01 - BigQuery - Table Data.ipynb
- 02a - Vertex AI ...
- 02b - Vertex AI ...
- 02c - Vertex AI ...
- 03a - BigQuery ...
- 03b - Vertex AI ...
- 04a - Vertex AI ...
- 05 - Vertex AI ...
- 05a - Vertex AI ...
- 05b - Vertex AI ...
- 05c - Vertex AI ...
- 05d - Vertex AI ...
- 05e - Vertex AI ...
- 05f - Vertex AI ...
- 06 - Vertex AI >...
- 07 - Vertex AI >...
- readme.md
- XX - Cleanup.ip...

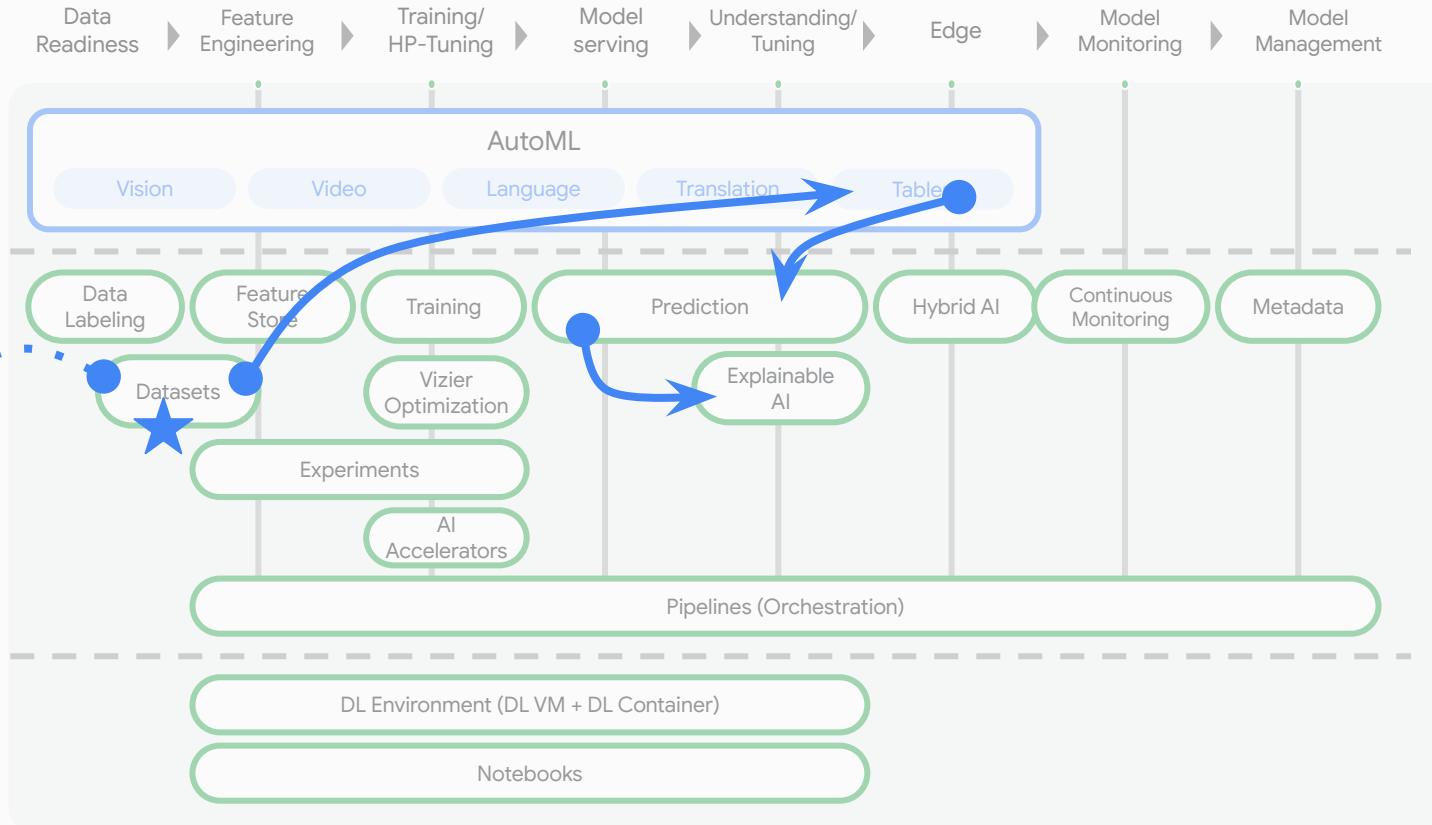
At the bottom of the sidebar, it says '0 19 Git: idle Python 3 | Idle'.

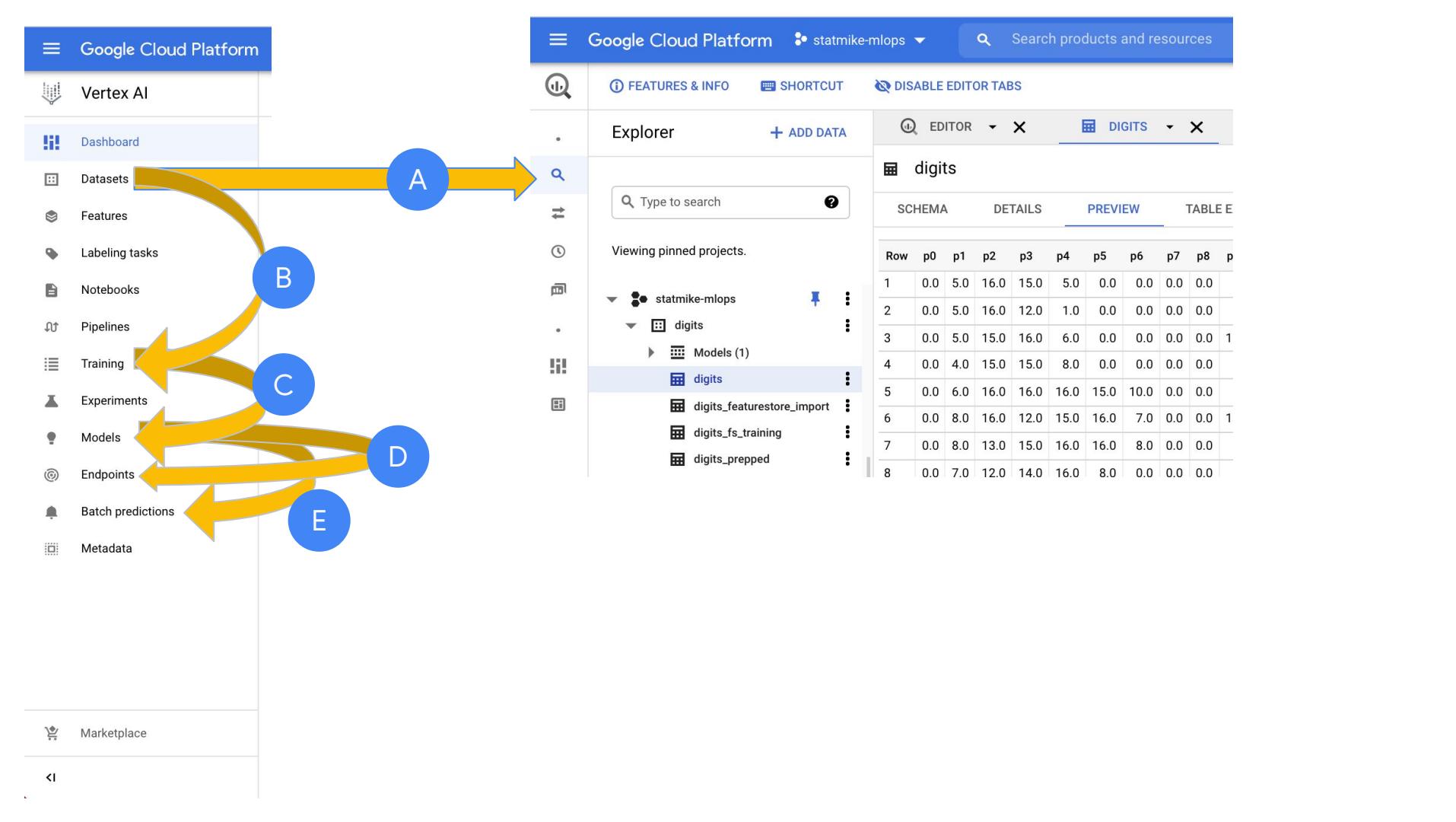
The screenshot shows the Google Cloud Platform console with a search bar at the top. Below the search bar are buttons for 'FEATURES & INFO', 'SHORTCUT', and 'DISABLE EDITOR TABS'. The main area is titled 'Explorer' and shows a pinned project 'statmike-mlops' containing a dataset 'digits'. The 'DETAILS' tab is selected, showing a table schema with columns Row, p0, p1, p2, p3, p4, p5, p6, p7, p8, p9. The 'PREVIEW' tab is also visible. A large yellow arrow points from the 'Notebooks' tab in the Jupyter interface to the 'DIGITS' tab in the Google Cloud Platform interface, indicating a connection between the two environments.

Row	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9
1	0.0	5.0	16.0	15.0	5.0	0.0	0.0	0.0	0.0	0.0
2	0.0	5.0	16.0	12.0	1.0	0.0	0.0	0.0	0.0	0.0
3	0.0	5.0	15.0	16.0	6.0	0.0	0.0	0.0	0.0	1
4	0.0	4.0	15.0	15.0	8.0	0.0	0.0	0.0	0.0	0.0
5	0.0	6.0	16.0	16.0	16.0	15.0	10.0	0.0	0.0	0.0
6	0.0	8.0	16.0	12.0	15.0	16.0	7.0	0.0	0.0	1
7	0.0	8.0	13.0	15.0	16.0	16.0	8.0	0.0	0.0	0.0
8	0.0	7.0	12.0	14.0	16.0	8.0	0.0	0.0	0.0	0.0

02a

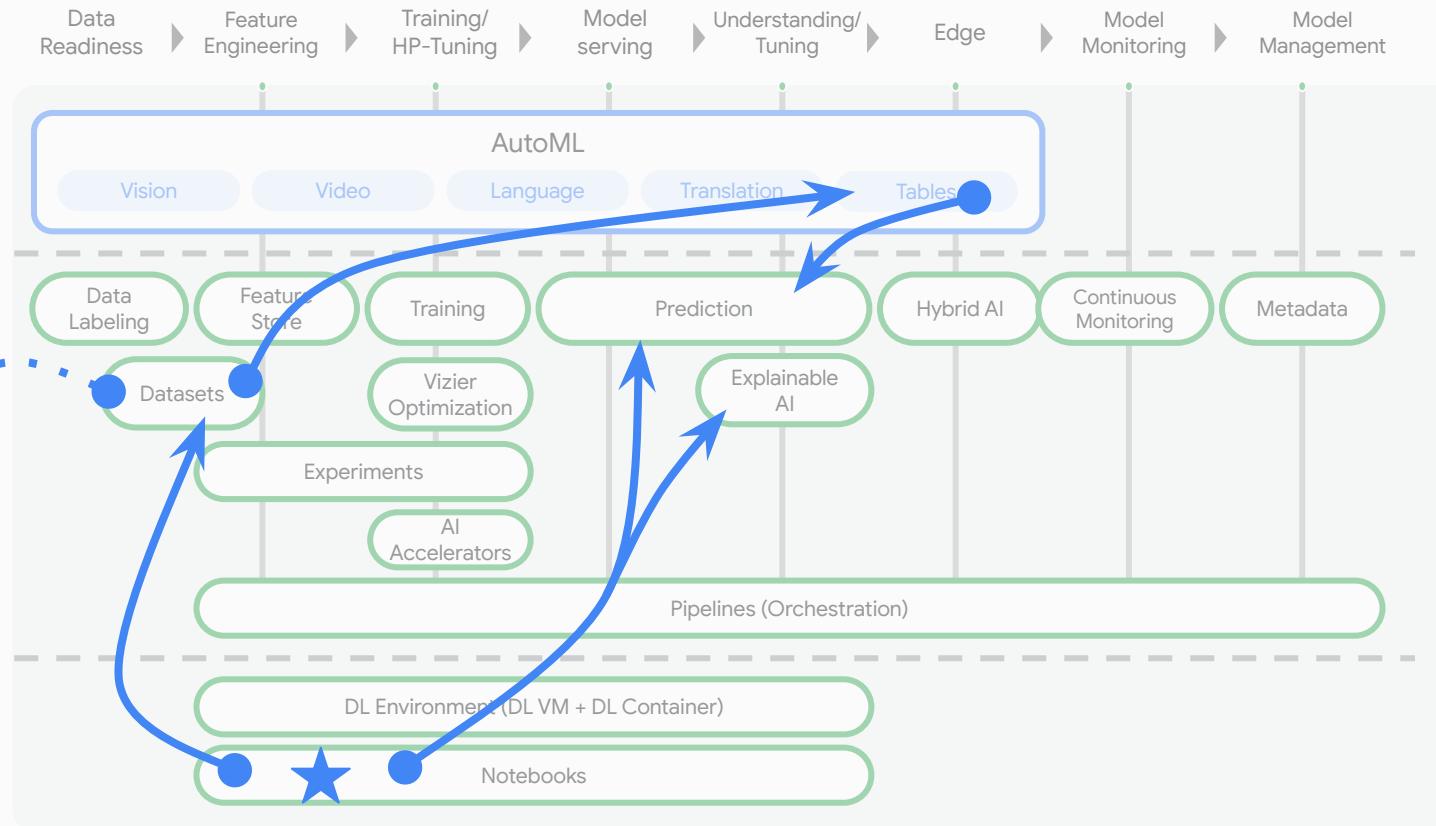
Vertex AI Overview

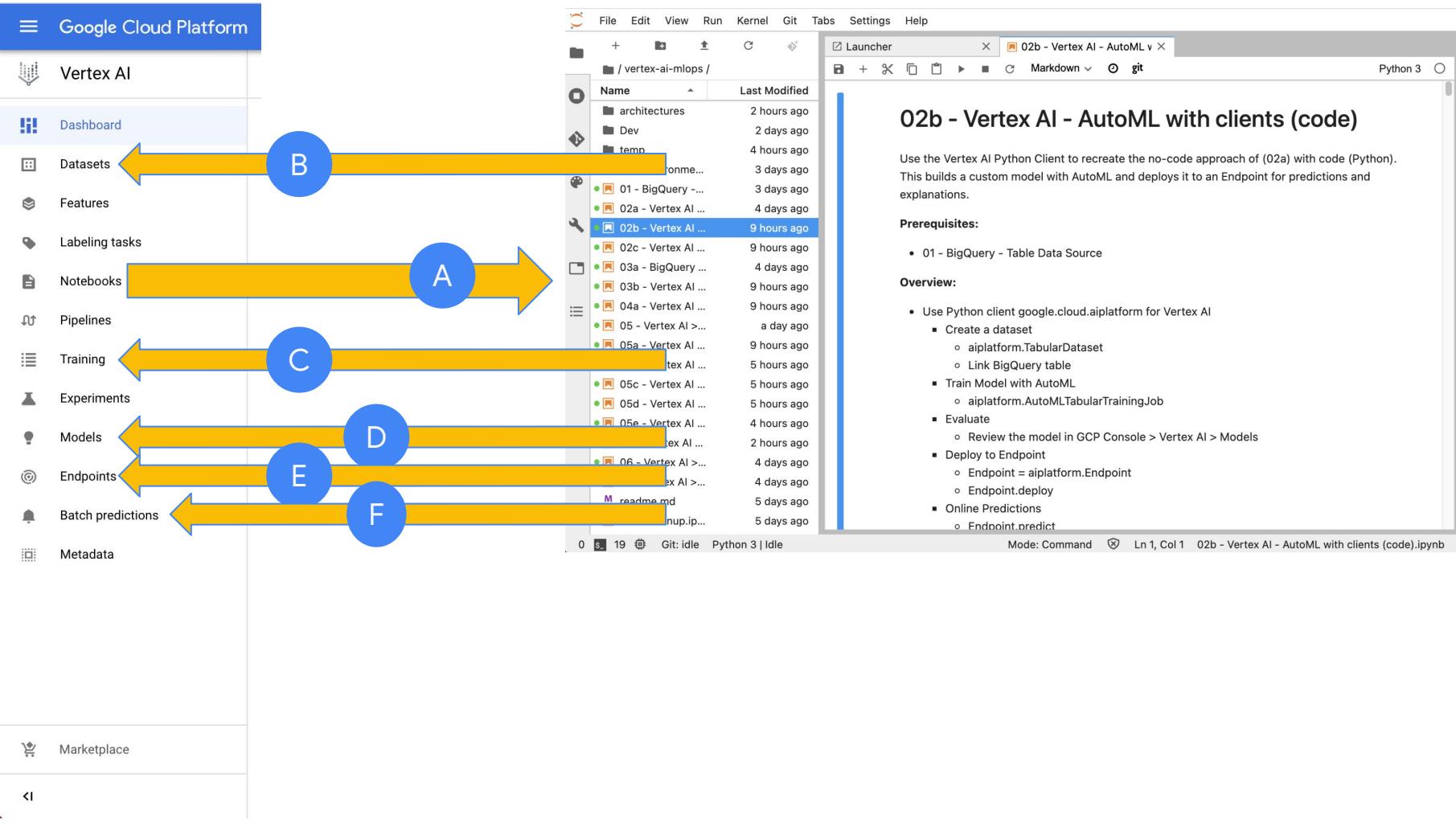




02b

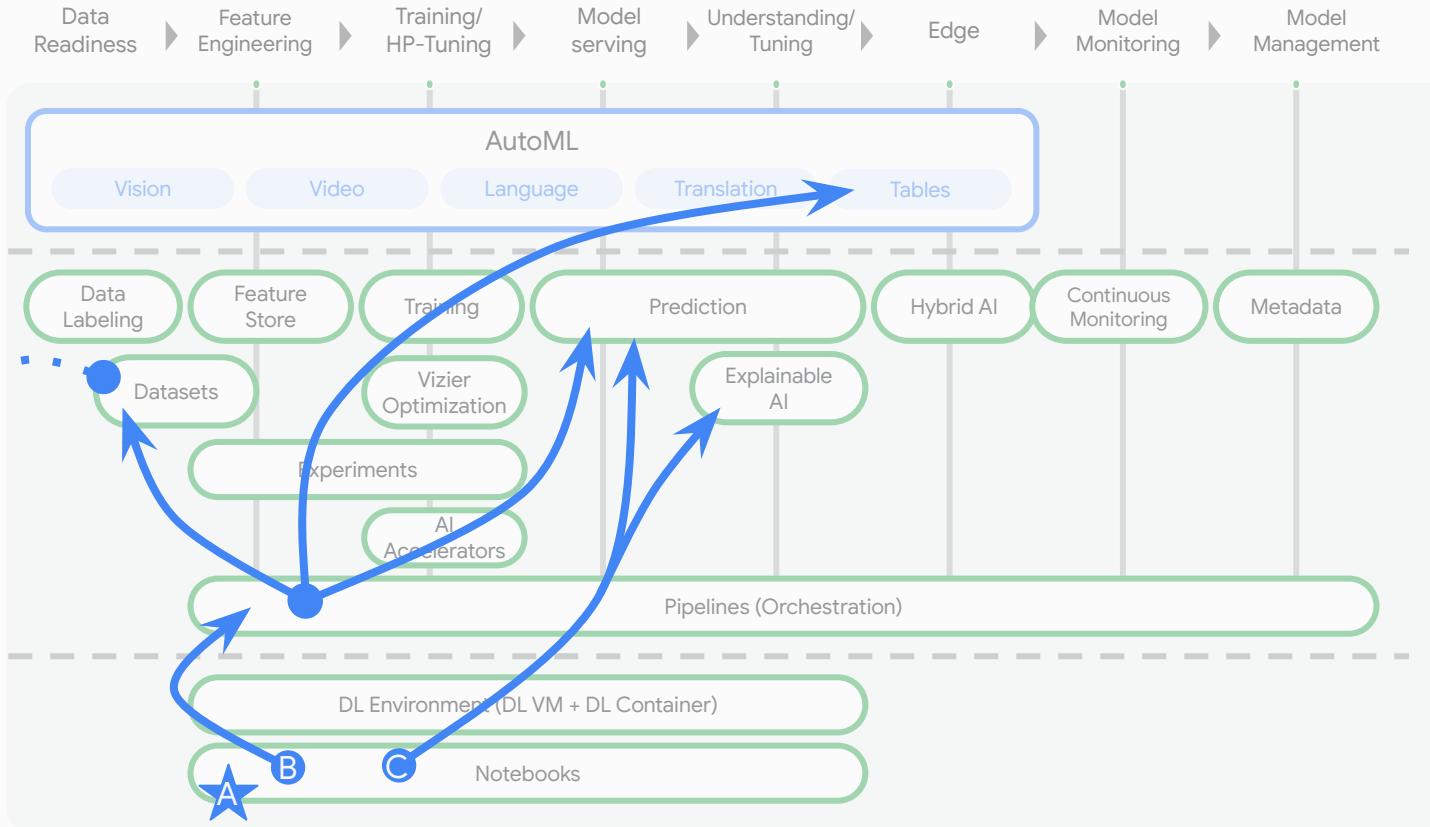
Vertex AI Overview

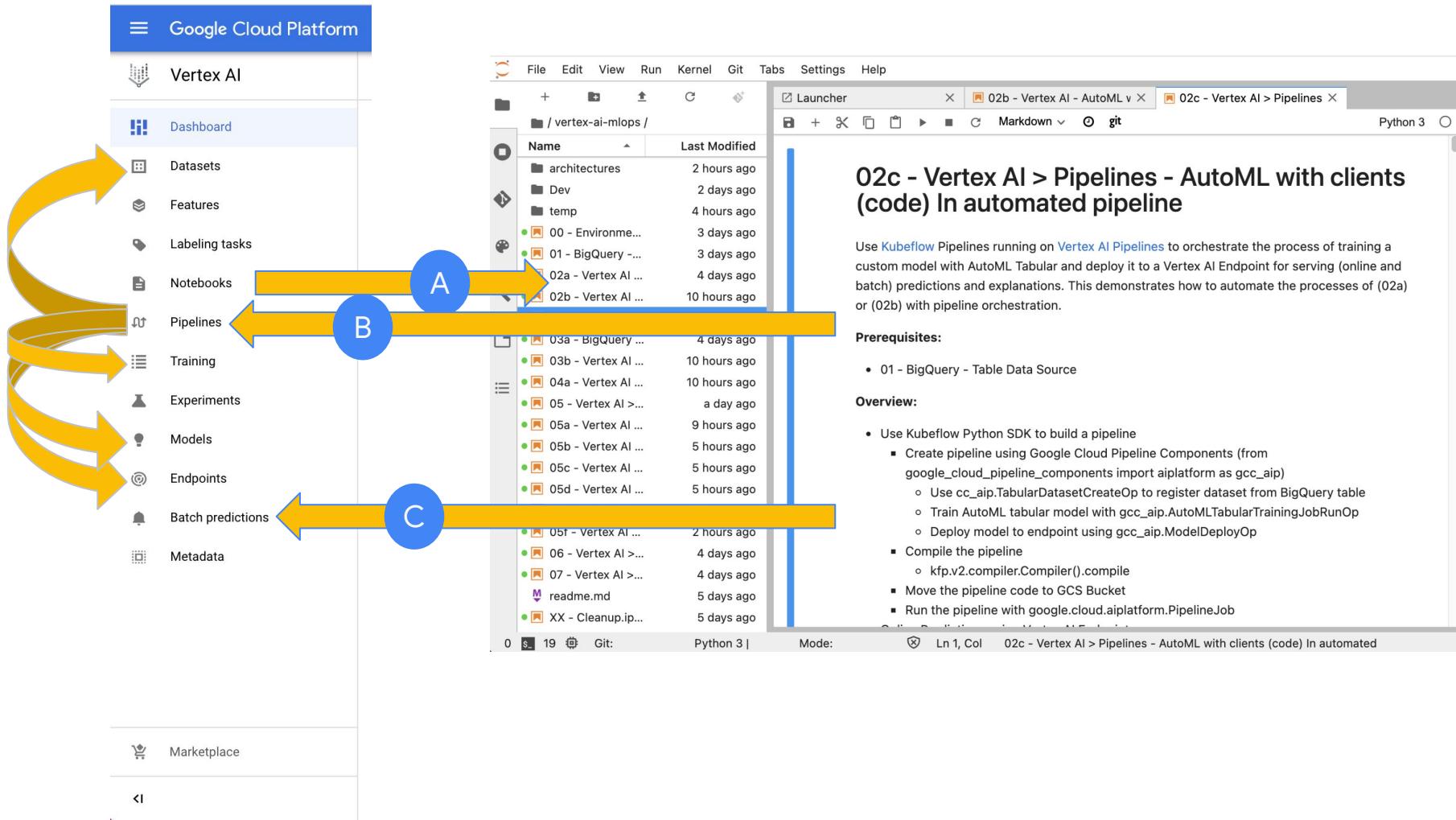




02c

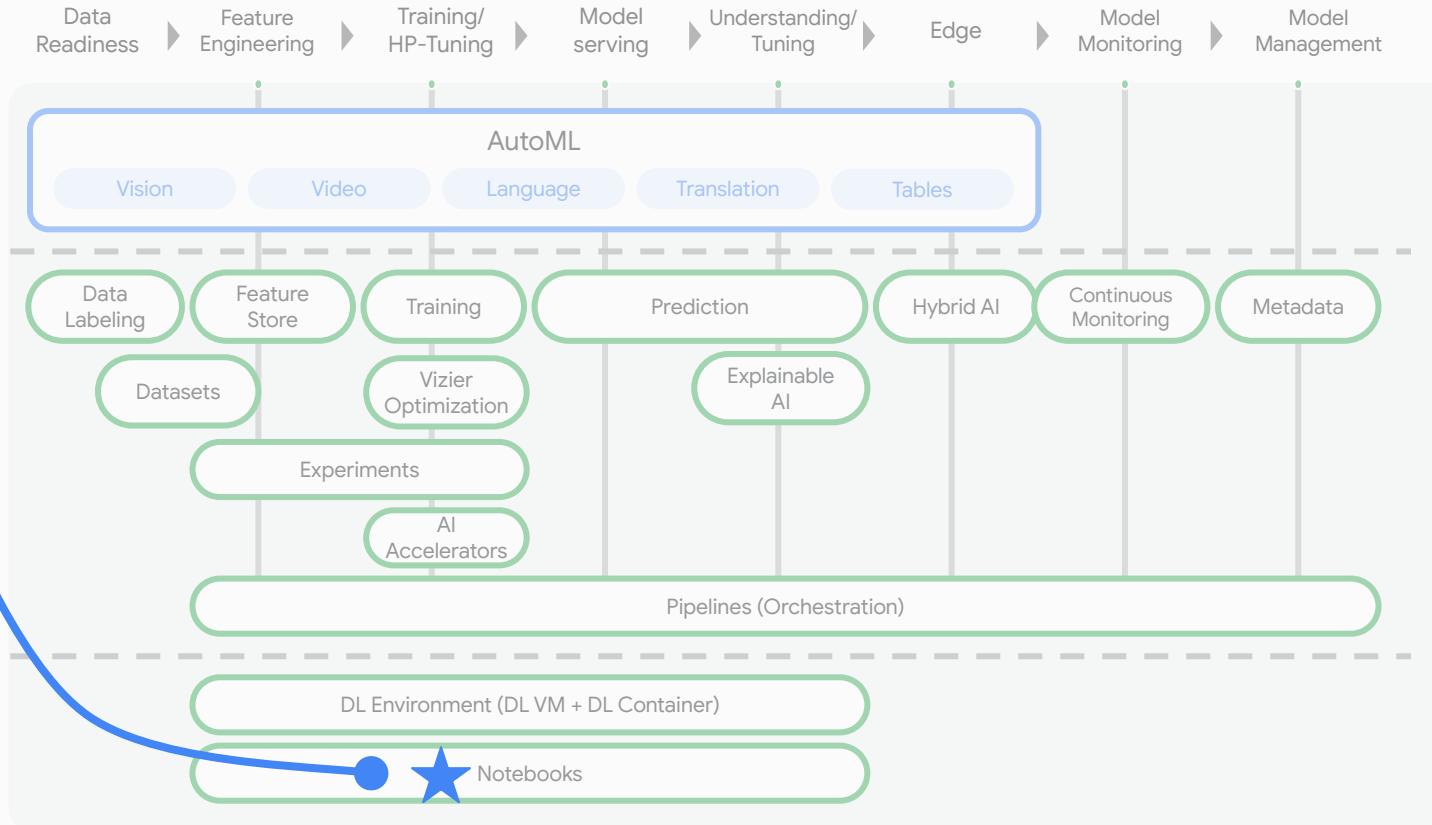
Vertex AI Overview





03a

Vertex AI Overview



Vertex AI

Dashboard

Datasets

Features

Labeling tasks

Notebooks

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

Marketplace

Google Cloud Platform statmike-mlops Search products and resources

FEATURES & INFO SHORTCUT DISABLE EDITOR TABS

Editor Digits DIGITS_LR

Type to search

Viewing pinned projects.

statmike-mlops digits Models (1) digits_lr digits digits_featurestore_import digits_fs_training digits_prep

Loss Duration (sec)

10 Training loss: 0.011 Evaluation loss: 0.014

File Edit View Run Kernel Git Tabs Settings Help

/ vertex-ai-mlops/

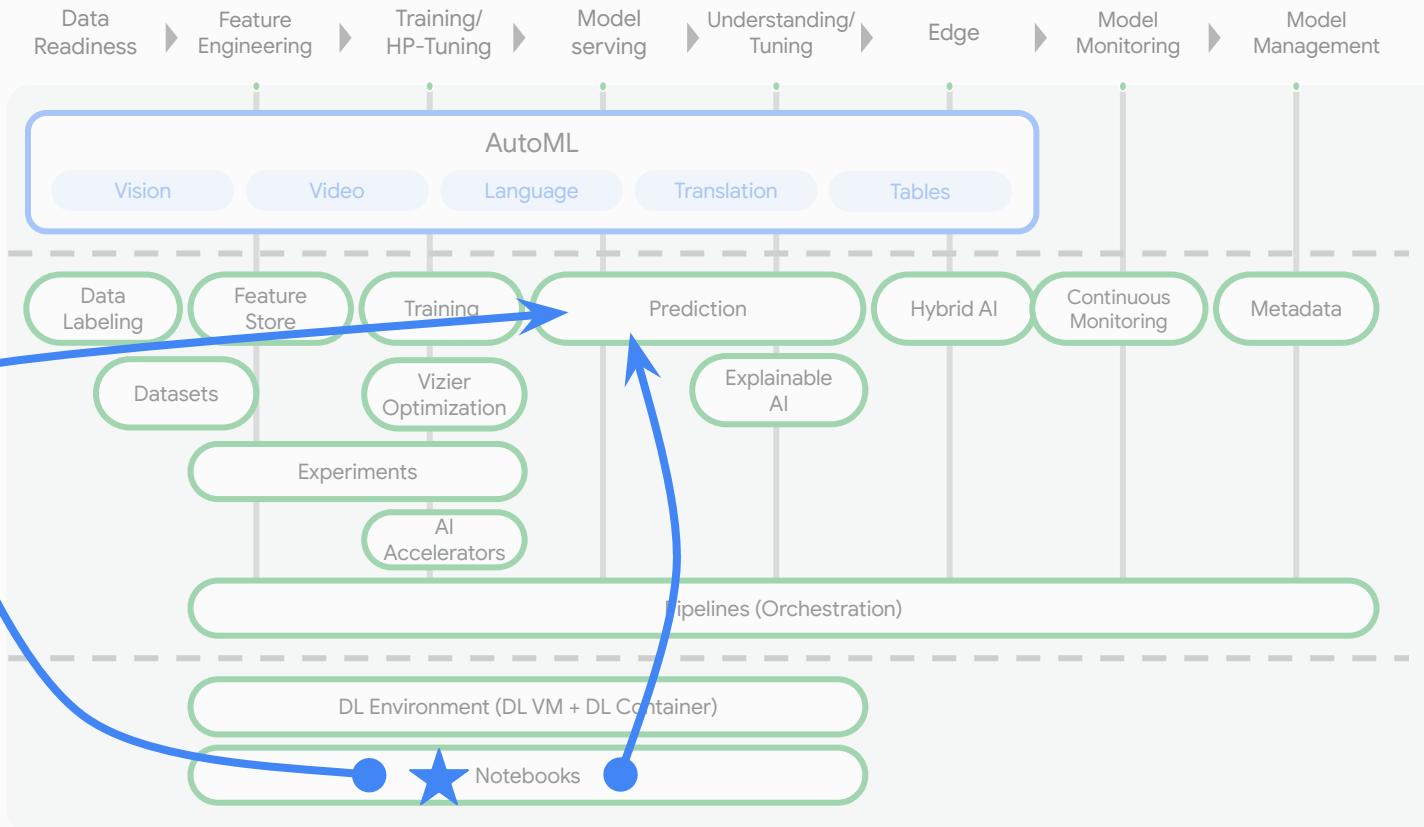
Name	Last Modified
architectures	3 hours ago
Dev	2 days ago
temp	5 hours ago
00 - Environment...	3 days ago
01 - BigQuery -...	3 days ago
02a - Vertex AI ...	4 days ago
02b - Vertex AI ...	10 hours ago
02c - Vertex AI ...	10 hours ago
03a - BigQuery ...	4 days ago

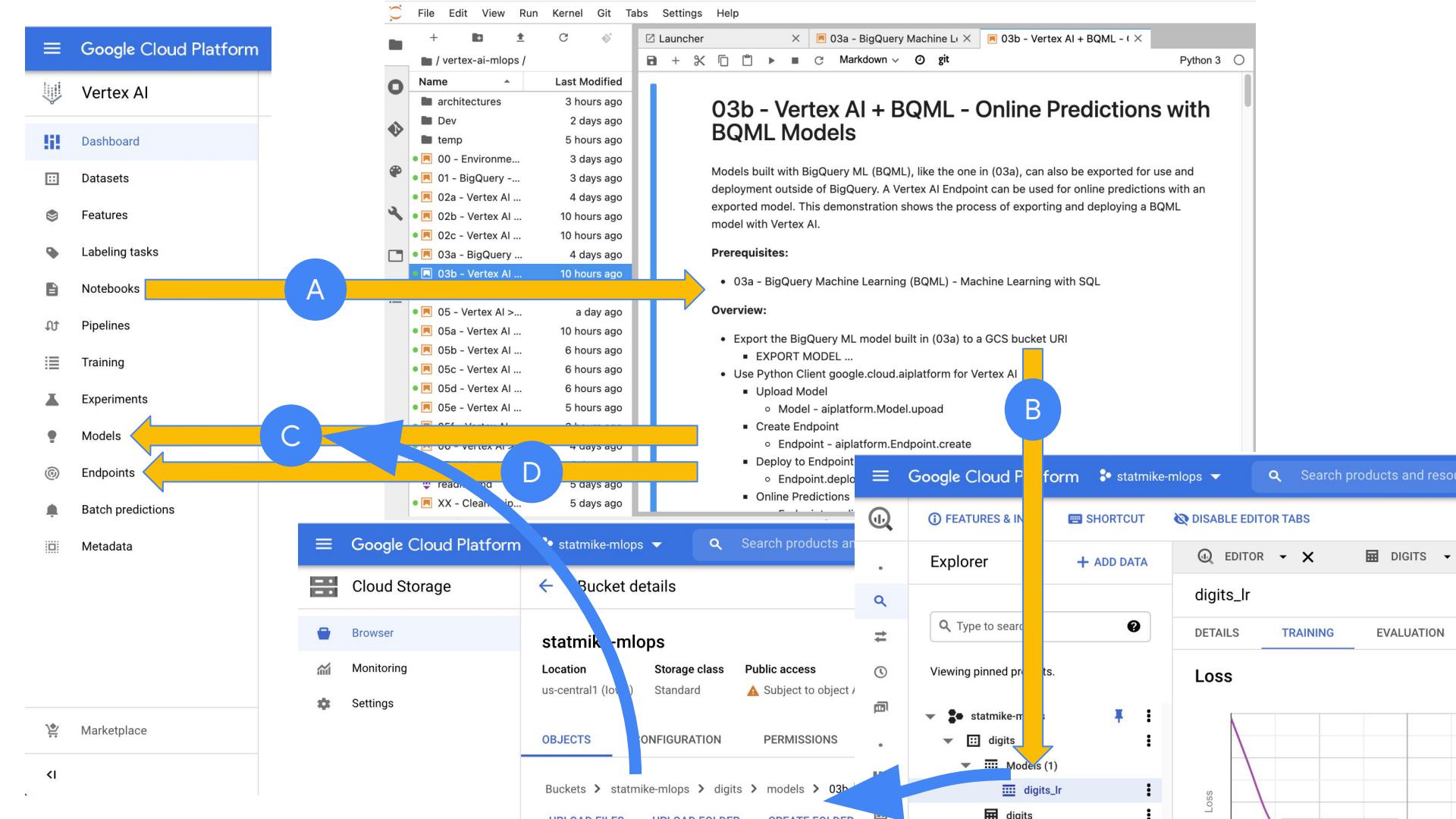
03a - BigQuery Machine Learning (BQML) - Machine Learning with SQL

BigQuery has a number of machine learning algorithms callable directly from SQL. This gives the convenience of using the common language of SQL to "CREATE MODEL ...". The library of available models is constantly growing and covers supervised, unsupervised, and time series methods as well as functions for evaluation - even anomaly detection from results, explainability and hyperparameter tuning. A great starting point for seeing the scope of available methods is [https://cloud.google.com/bigquery-machine-learning](#).

03b

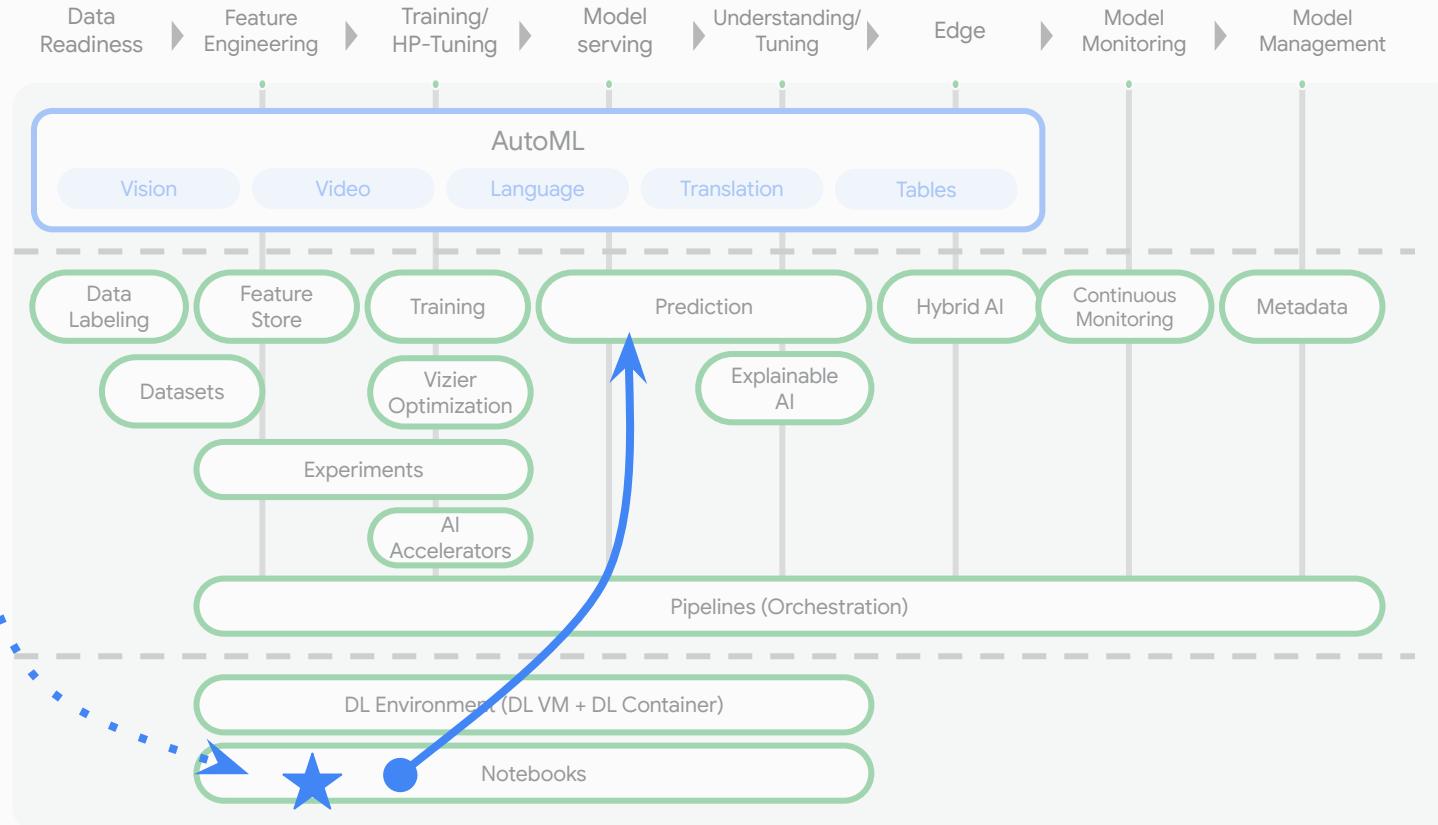
Vertex AI Overview





04a

Vertex AI Overview



A

B

C

D

04a - Vertex AI > Notebooks - Models Built in Notebooks with Tensorflow

Where a model gets trained is where it consumes computing resources. With Vertex AI, you have choices for configuring the computing resources available at training. This notebook is an example of an execution environment. When it was set up there were choices for machine type and accelerators (GPUs).

This notebook shows training a model directly within the runtime of the notebook. Then the model is saved and moved to GCS for deployment to a Vertex AI endpoint. The endpoint can then make predictions. The model training is done with [Tensorflow](#), specifically [Keras](#). In this notebook, we will show a neural network approach to logistic regression. The training data being used is a table in BigQuery using [Tensorflow I/O](#).

Prerequisites:

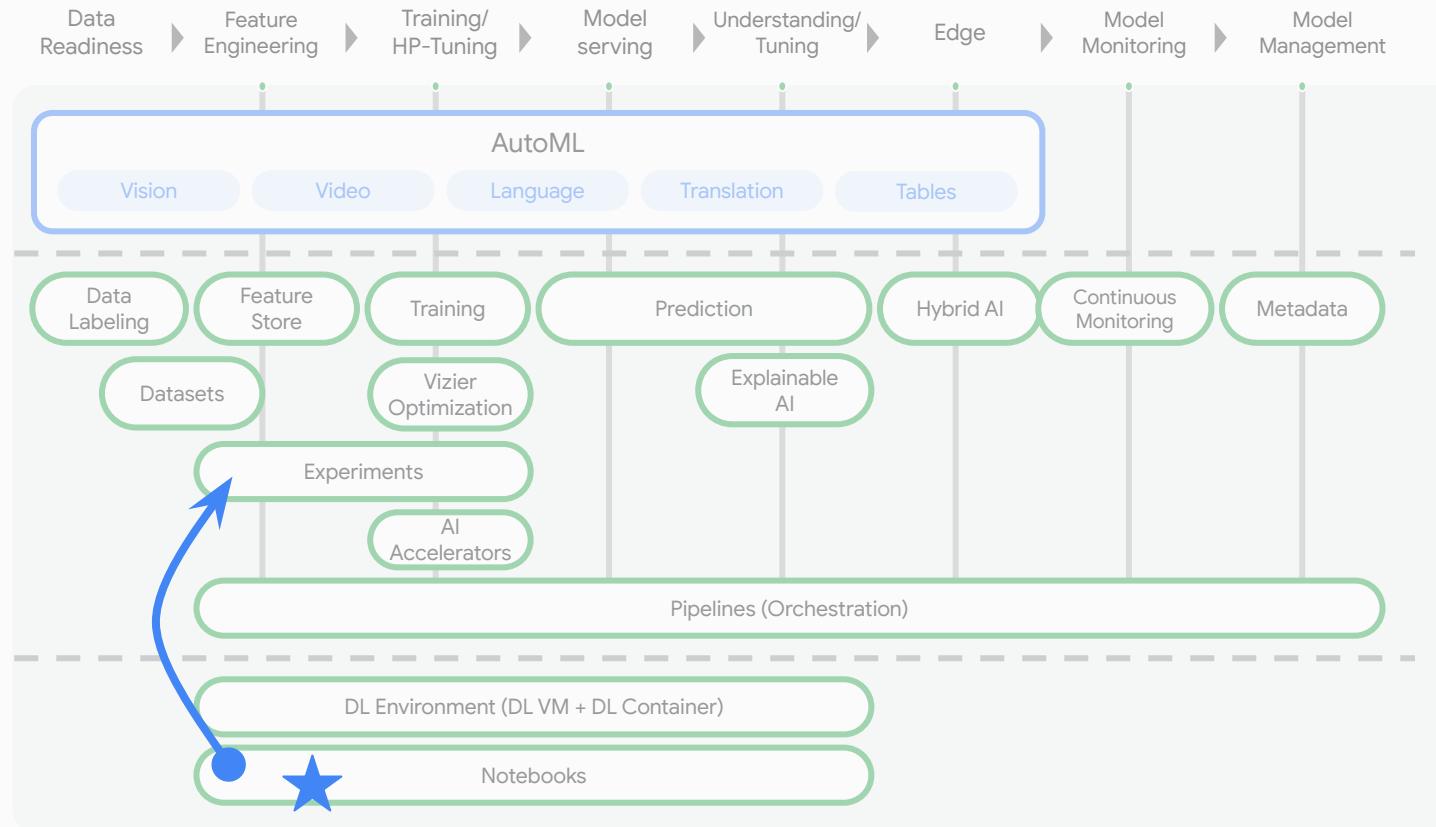
- 01 - BigQuery - Table Data Source

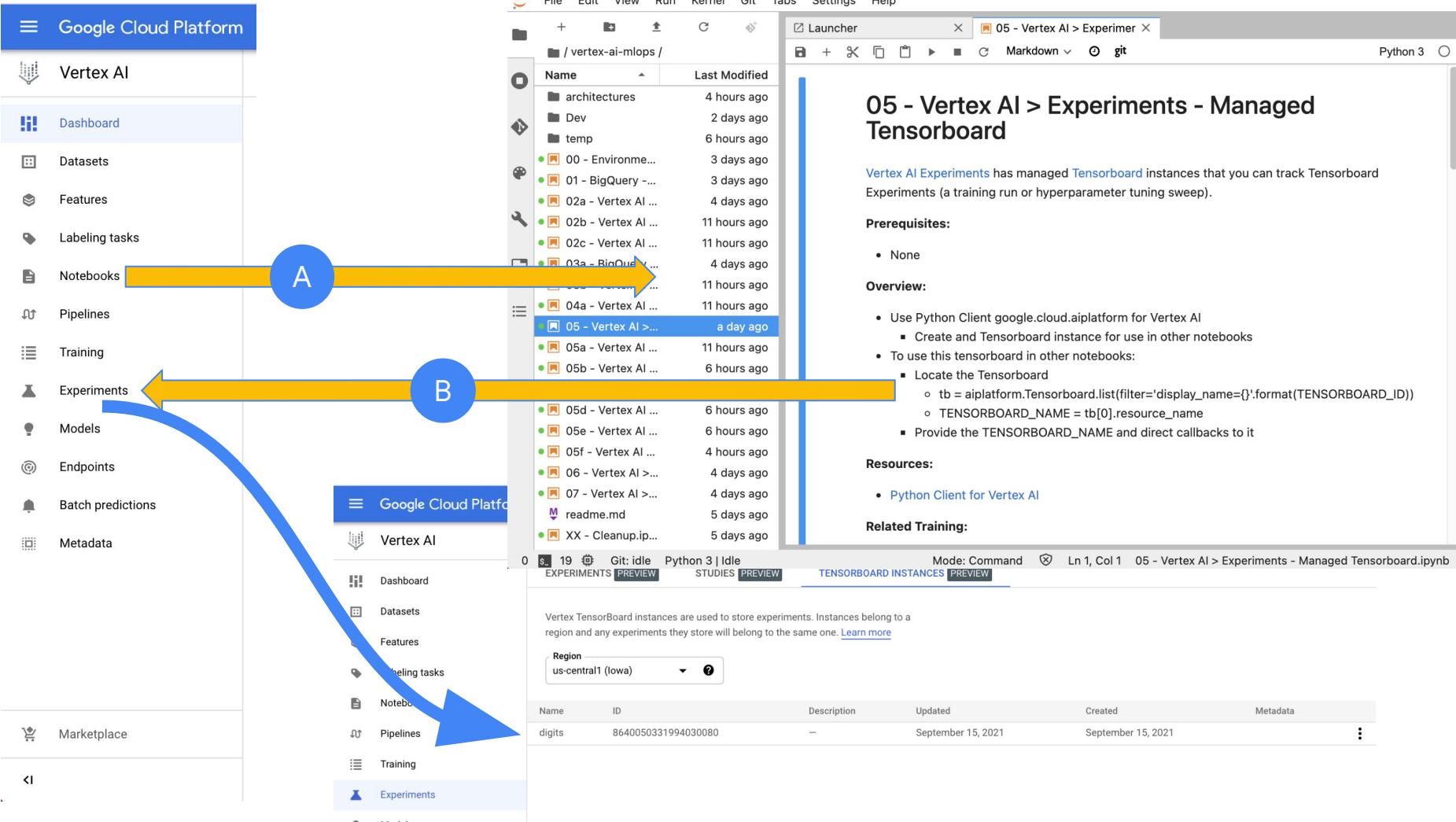
Overview:

- Use Python Client for BigQuery
 - Read the tables schema from BigQuery INFORMATION_SCHEMA
 - Prepare the feature information for Tensorflow
- Define a function that remaps the input data into features and target variables where target is one-hot encoded (classification model with 10 classes)

Mode: Command Ln 1, Col 1 04a - Vertex AI > Notebooks - Models Built in Notebooks with Tensorflow.ipynb

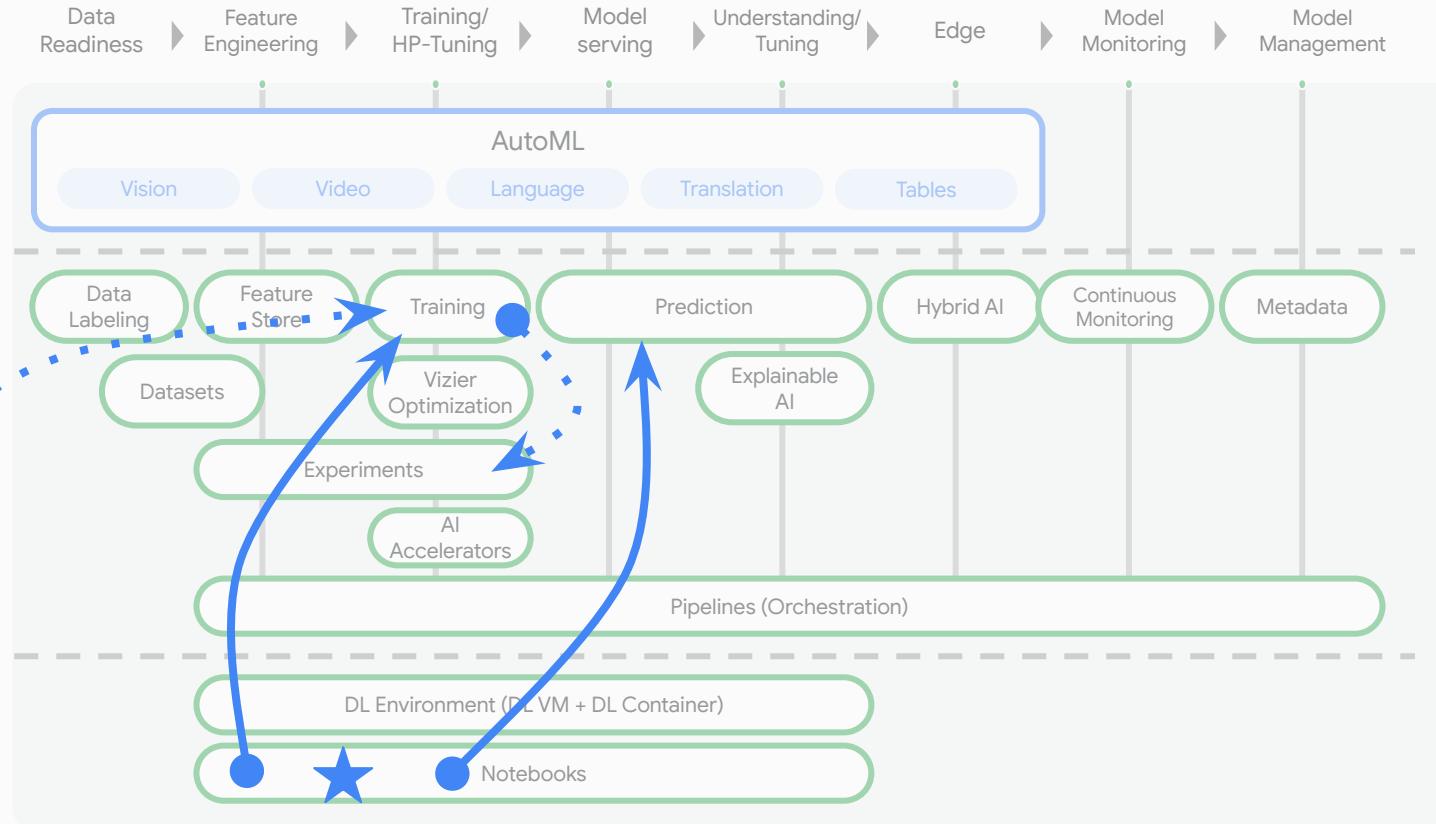
Vertex AI Overview

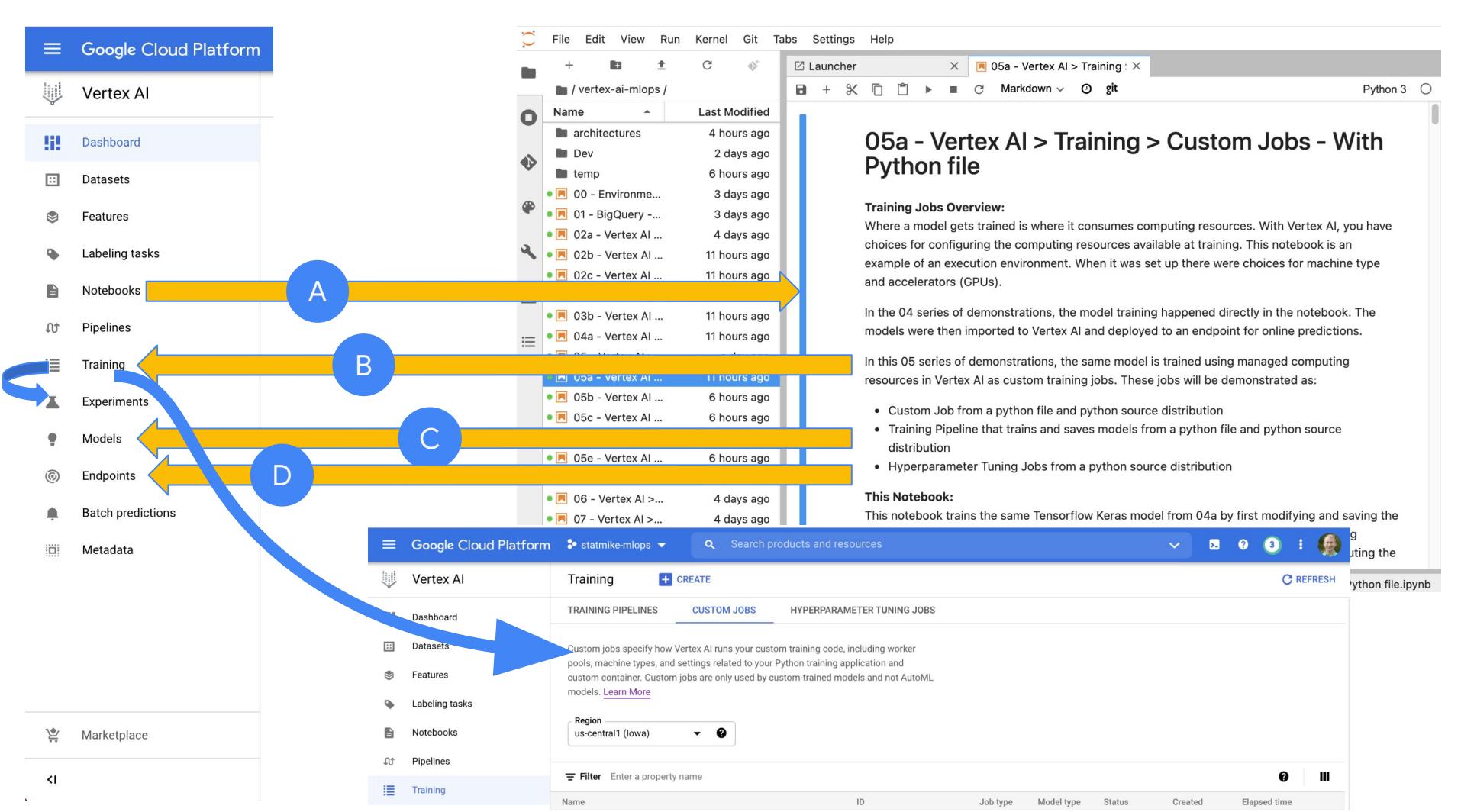




05a

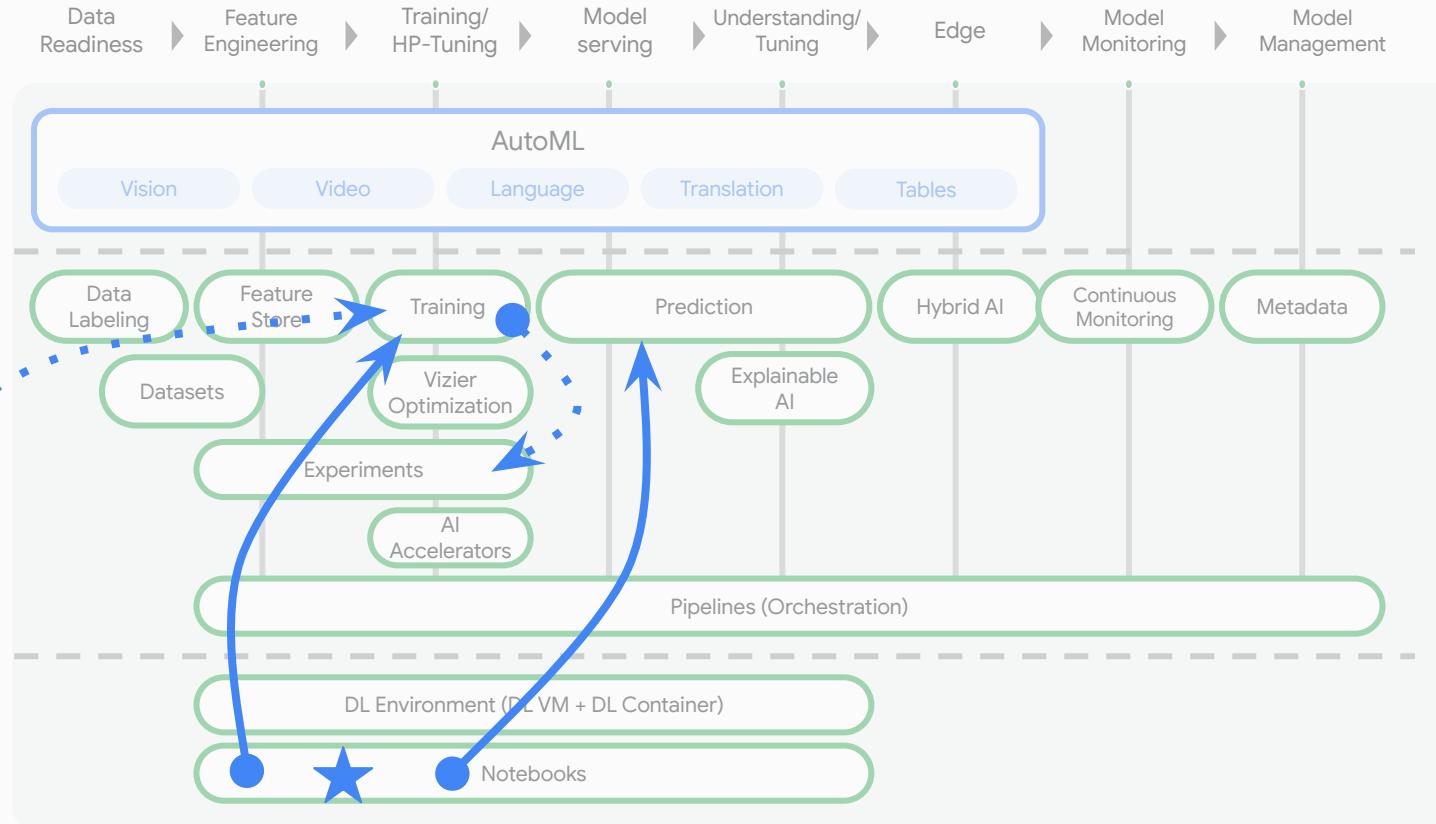
Vertex AI Overview





05b

Vertex AI Overview



Google Cloud Platform

Vertex AI

- Dashboard
- Datasets
- Features
- Labeling tasks
- Notebooks
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Marketplace

05b - Vertex AI > Training > Custom Jobs - With Python Source Distribution

Training Jobs Overview:
 Where a model gets trained is where it consumes computing resources. With Vertex AI, you have choices for configuring the computing resources available at training. This notebook is an example of an execution environment. When it was set up there were choices for machine type and accelerators (GPUs).

In the 04 series of demonstrations, the model training happened directly in the notebook. The models were then imported to Vertex AI and deployed to an endpoint for online predictions.

In this 05 series of demonstrations, the same model is trained using managed computing resources in Vertex AI as custom training jobs. These jobs will be demonstrated as:

- Custom Job from a python file and python source distribution
- Training Pipeline that trains and saves models from a python file and python source distribution
- Hyperparameter Tuning Jobs from a python source distribution

This Notebook:
 This notebook trains the same Tensorflow Keras model from 04a by first modifying and saving the

Google Cloud Platform

Vertex AI

Training

Custom Jobs

Custom jobs specify how Vertex AI runs your custom training code, including worker pools, machine types, and settings related to your Python training application and custom container. Custom jobs are only used by custom-trained models and not AutoML models. [Learn More](#)

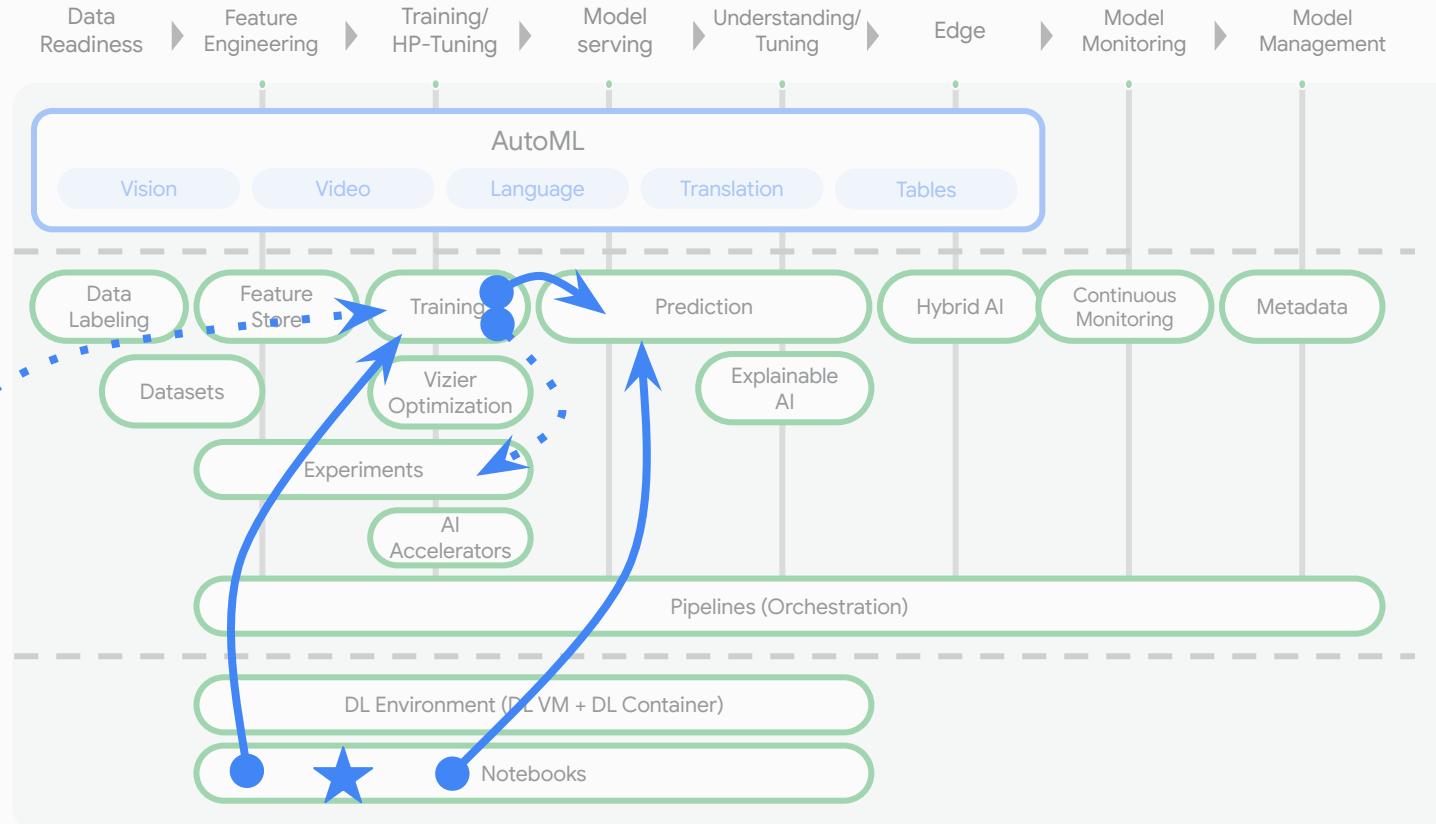
Region: us-central1 (Iowa)

Filter: Enter a property name

Name	ID	Job type	Model type	Status	Created	Elapsed time
00 - Environme...						
01 - BigQuery ...						
02a - Vertex AI ...						
02b - Vertex AI ...						
02c - Vertex AI ...						
03b - Vertex AI ...						
04a - Vertex AI ...						
05a - Vertex AI ...						
05b - Vertex AI ...						
05c - Vertex AI ...						
05e - Vertex AI ...						
06 - Vertex AI >...						
07 - Vertex AI >...						

05c

Vertex AI Overview



The diagram illustrates the Vertex AI workflow, showing the transition from Notebooks to Training Pipelines. It features four main stages: A (Notebooks), B (Training), C (Models), and D (Endpoints). Stage A is highlighted with a yellow bar and a blue arrow pointing to it from the left. Stage B is also highlighted with a yellow bar. Stage C is shown as a yellow bar with a blue arrow pointing to it from stage B. Stage D is shown as a yellow bar with a blue arrow pointing to it from stage C. A large blue curved arrow points from stage A towards stage D.

Google Cloud Platform

Vertex AI

- Dashboard
- Datasets
- Features
- Labeling tasks
- Notebooks
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Marketplace

05c - Vertex AI > Training : x

Launcher + Markdown git Python 3

05c - Vertex AI > Training Pipelines - With Python file

Training Jobs Overview:

Where a model gets trained is where it consumes computing resources. With Vertex AI, you have choices for configuring the computing resources available at training. This notebook is an example of an execution environment. When it was set up there were choices for machine type and accelerators (GPUs).

In the 04 series of demonstrations, the model training happened directly in the notebook. The models were then imported to Vertex AI and deployed to an endpoint for online predictions.

In this 05 series of demonstrations, the same model is trained using managed computing resources in Vertex AI as custom training jobs. These jobs will be demonstrated as:

- Custom Job from a python file and python source distribution
- Training Pipeline that trains and saves models from a python file and python source distribution
- Hyperparameter Tuning Jobs from a python source distribution

This Notebook: An extension of 05a

Google Cloud Platform statmike-mlops Search products and resources

Vertex AI

Training + CREATE

TRAINING PIPELINES CUSTOM JOBS HYPERPARAMETER TUNING JOBS

Training pipelines are the primary model training workflow in Vertex AI. You can use training pipelines to create an AutoML-trained model or a custom-trained model. For custom-trained models, training pipelines orchestrate custom training jobs and hyperparameter tuning with additional steps like adding a dataset or uploading the model to Vertex AI for prediction serving. [Learn More](#)

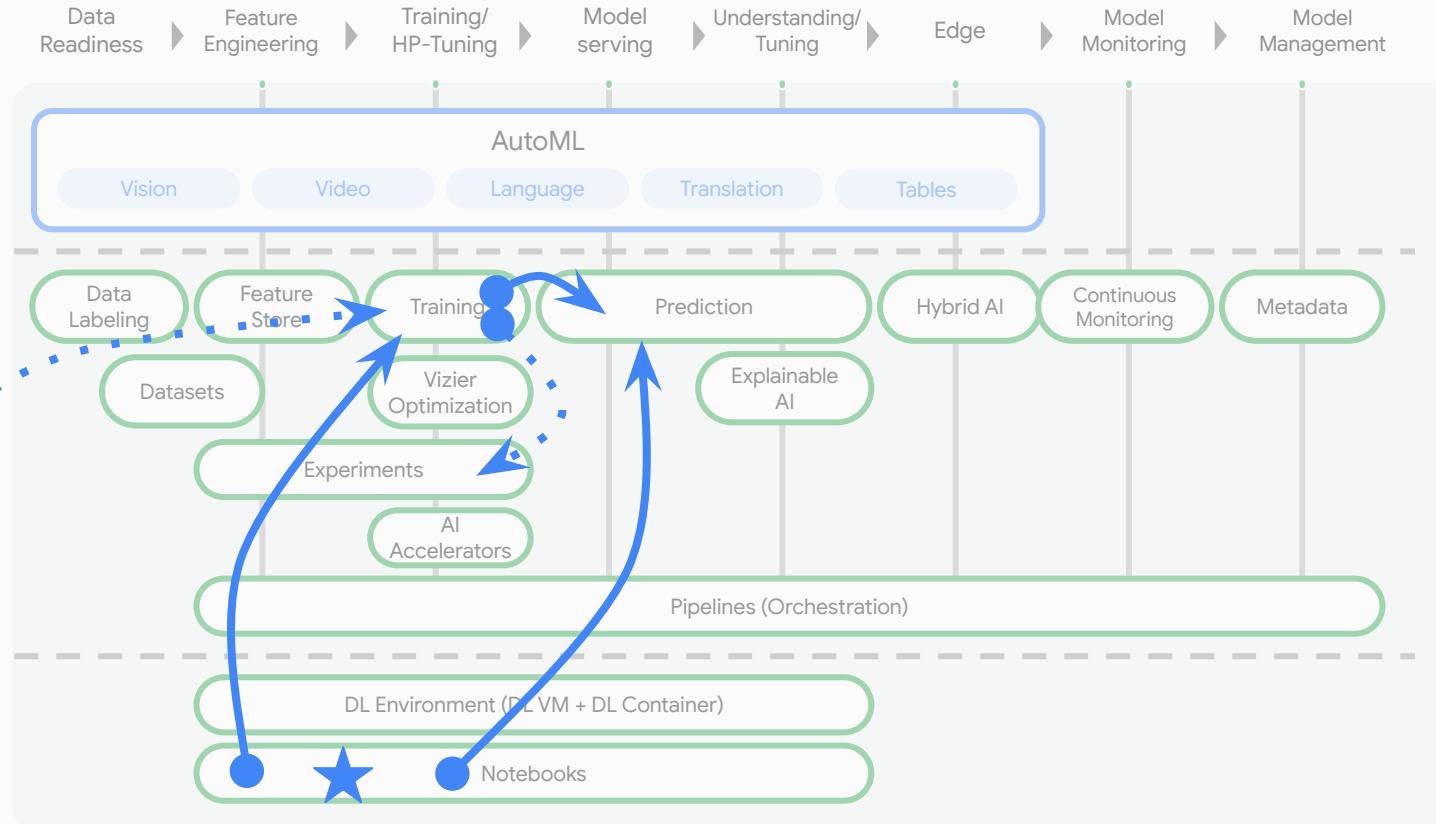
Region us-central1 (Iowa)

Filter Enter a property name

Name	ID	Job type	Model type	Status	Created	Elapsed time
00 - Environme...						
01 - BigQuery ...						
02a - Vertex AI ...						
02b - Vertex AI ...						
02c - Vertex AI ...						
03b - Vertex AI ...						
04a - Vertex AI ...						
05a - Vertex AI ...						
05b - Vertex AI ...						
05c - Vertex AI ...						
05d - Vertex AI ...						
05e - Vertex AI ...						
06 - Vertex AI >...						

05d

Vertex AI Overview



The diagram illustrates the Vertex AI ecosystem and the transition from Notebooks to Training Pipelines.

Vertex AI Components:

- Notebooks:** Represented by a blue circle labeled A.
- Training:** Represented by a blue circle labeled B.
- Models:** Represented by a blue circle labeled C.
- Endpoints:** Represented by a blue circle labeled D.

Flow and Transitions:

- A blue arrow points from Notebooks (A) to Training (B).
- A blue arrow points from Training (B) to Models (C).
- A blue arrow points from Models (C) to Endpoints (D).
- A large blue curved arrow originates from Notebooks (A) and points to the Training Pipelines section of the Google Cloud Platform interface.

Google Cloud Platform Interface (Top Right):

File Edit View Run Kernel Git Tabs Settings Help

Launcher 05d - Vertex AI > Training x

Markdown git Python 3

05d - Vertex AI > Training Pipelines - With Python Source Distribution

Training Jobs Overview:

Where a model gets trained is where it consumes computing resources. With Vertex AI, you have choices for configuring the computing resources available at training. This notebook is an example of an execution environment. When it was set up there were choices for machine type and accelerators (GPUs).

In the 04 series of demonstrations, the model training happened directly in the notebook. The models were then imported to Vertex AI and deployed to an endpoint for online predictions.

In this 05 series of demonstrations, the same model is trained using managed computing resources in Vertex AI as custom training jobs. These jobs will be demonstrated as:

- Custom Job from a python file and python source distribution
- Training Pipeline that trains and saves models from a python file and python source distribution
- Hyperparameter Tuning Jobs from a python source distribution

This Notebook: An extension of 05b

Google Cloud Platform Interface (Bottom Left):

File Edit View Run Kernel Git Tabs Settings Help

Dashboard / vertex-ai-mlops /

Name Last Modified

- architectures 4 hours ago
- Dev 2 days ago
- temp 6 hours ago
- 00 - Environme... 3 days ago
- 01 - BigQuery ... 3 days ago
- 02a - Vertex AI ... 4 days ago
- 02b - Vertex AI ... 11 hours ago
- 02c - Vertex AI ... 11 hours ago
- 03b - Vertex AI ... 11 hours ago
- 04a - Vertex AI ... 11 hours ago
- 05 - Vertex AI ... 11 hours ago
- 05a - vertex AI ... 11 hours ago
- 05b - Vertex AI ... 7 hours ago
- 05c - Vertex AI ... 7 hours ago
- 05e - Vertex AI ... 6 hours ago
- 06 - Vertex AI >... 4 days ago

Search products and resources

Google Cloud Platform Interface (Bottom Center):

Vertex AI Training + CREATE

TRAINING PIPELINES CUSTOM JOBS HYPERPARAMETER TUNING JOBS

Training pipelines are the primary model training workflow in Vertex AI. You can use training pipelines to create an AutoML-trained model or a custom-trained model. For custom-trained models, training pipelines orchestrate custom training jobs and hyperparameter tuning with additional steps like adding a dataset or uploading the model to Vertex AI for prediction serving. [Learn More](#)

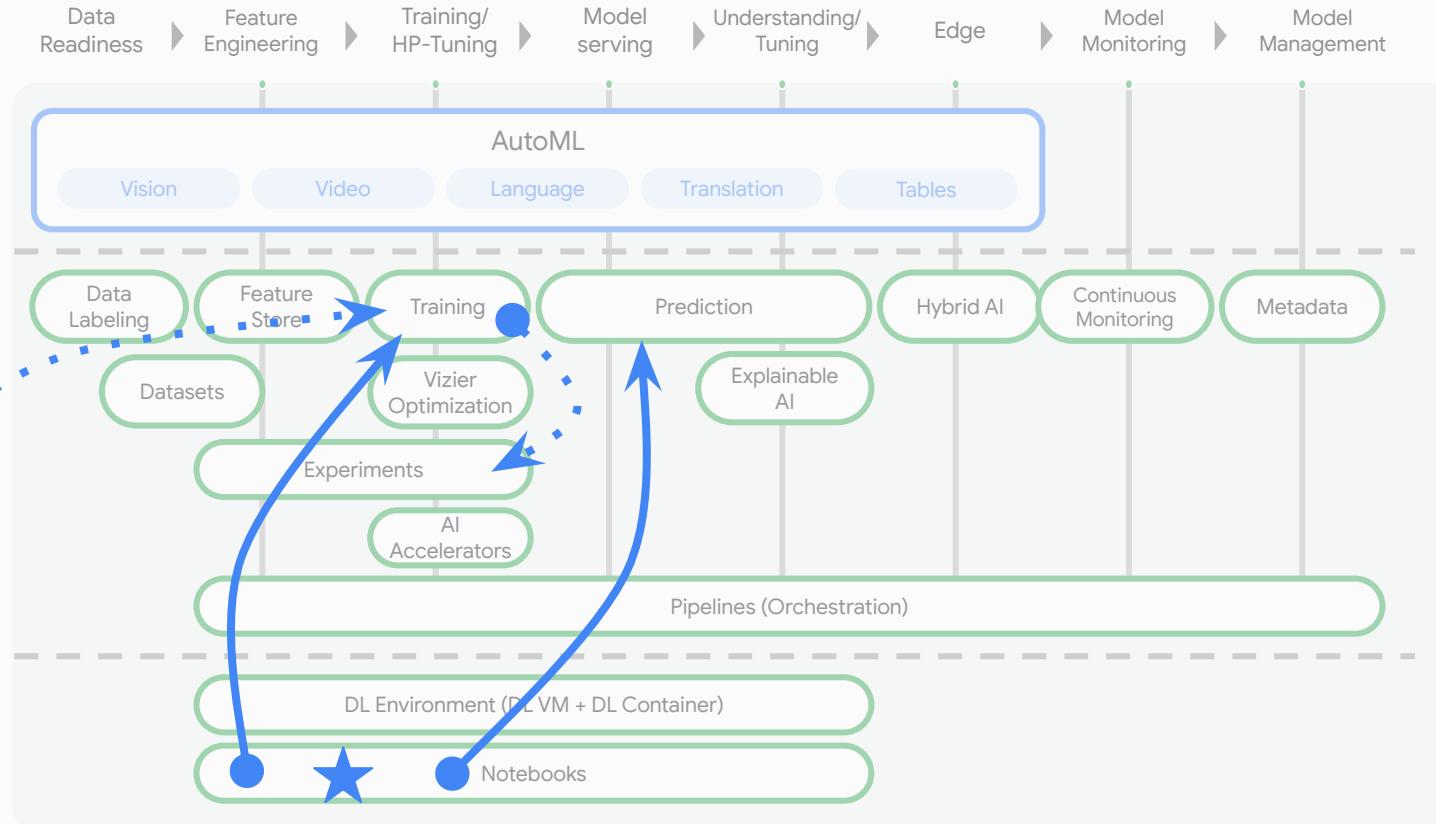
Region us-central1 (Iowa)

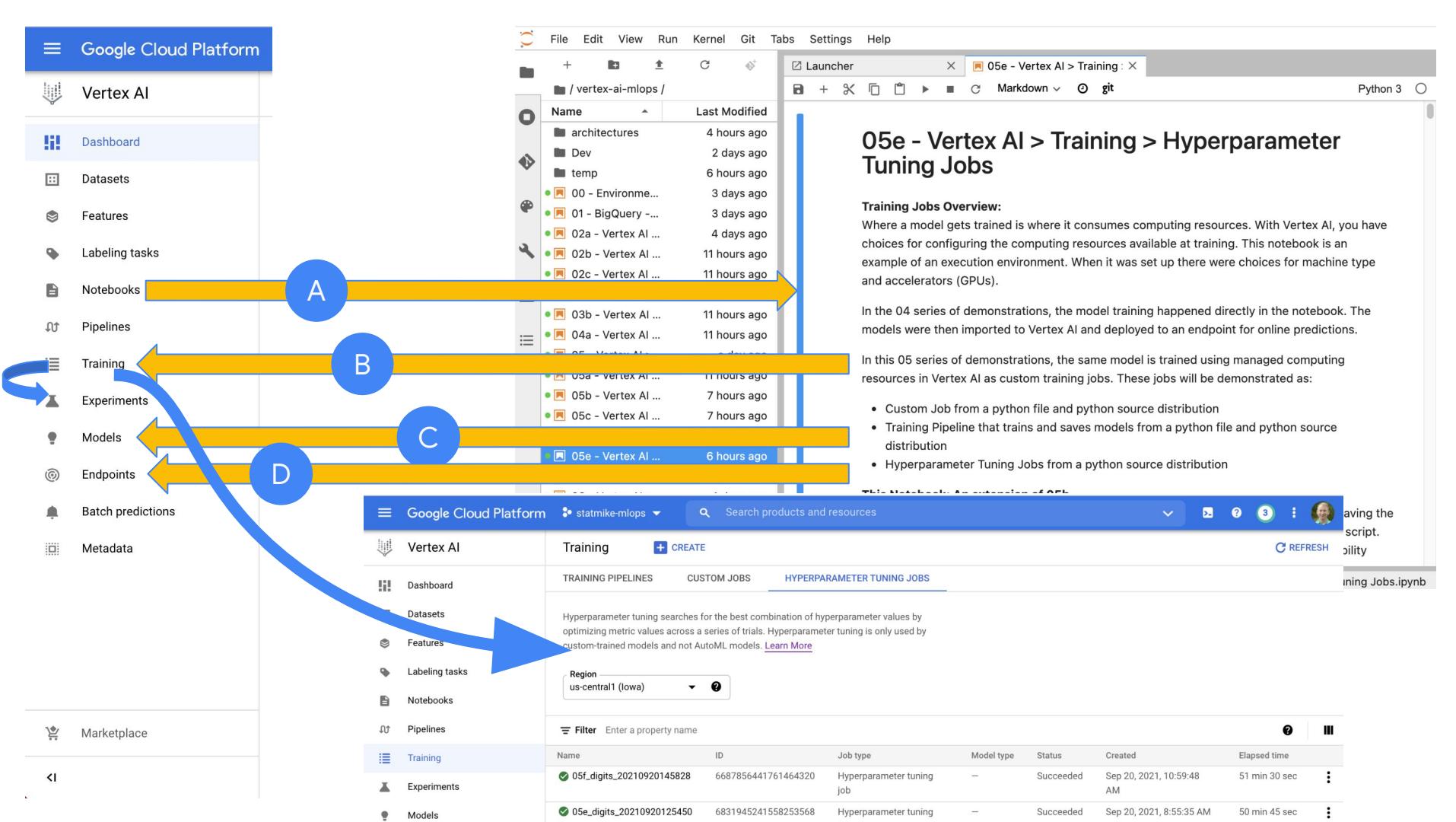
Filter Enter a property name

Name	ID	Job type	Model type	Status	Created	Elapsed time

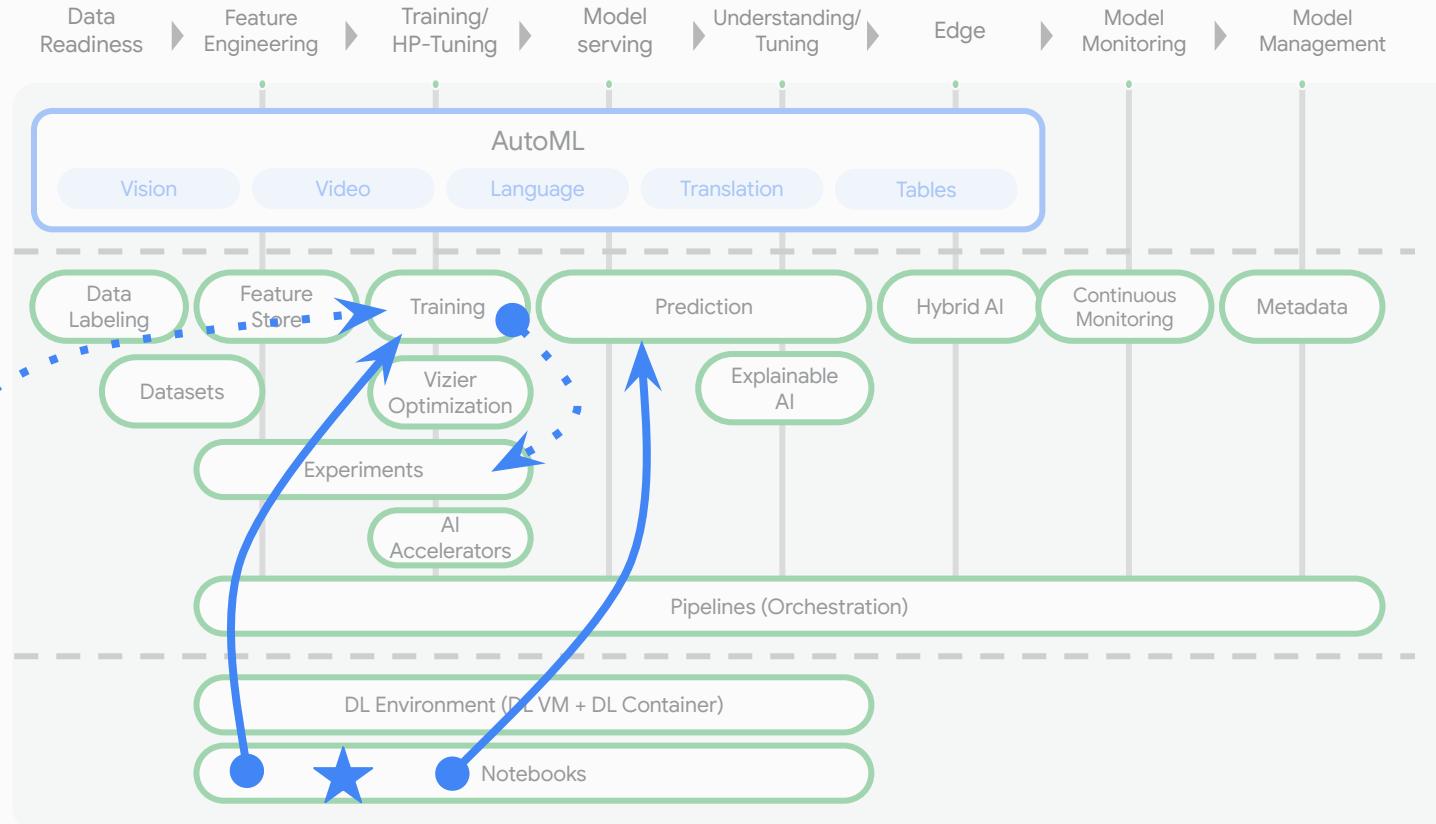
05e

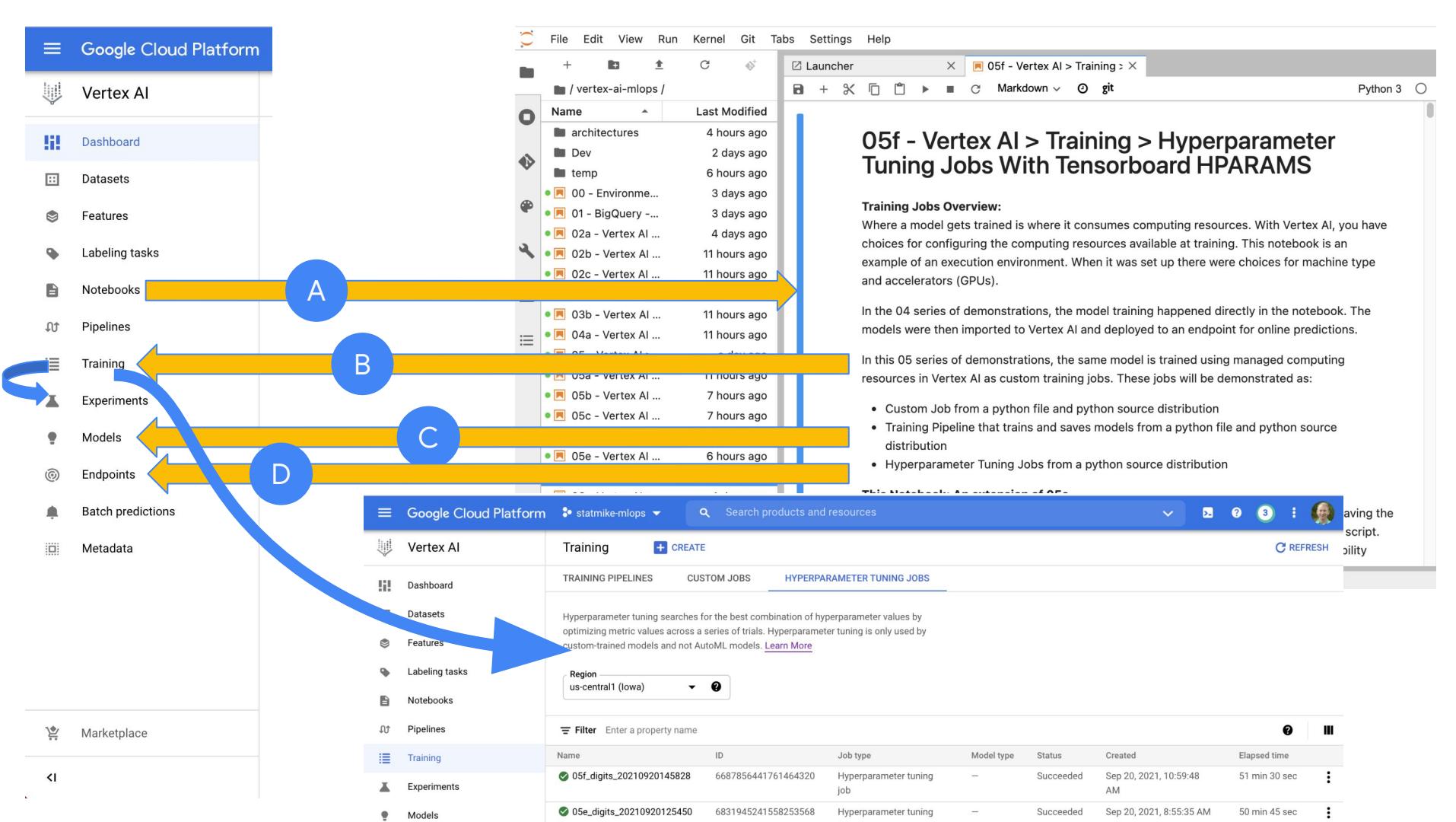
Vertex AI Overview



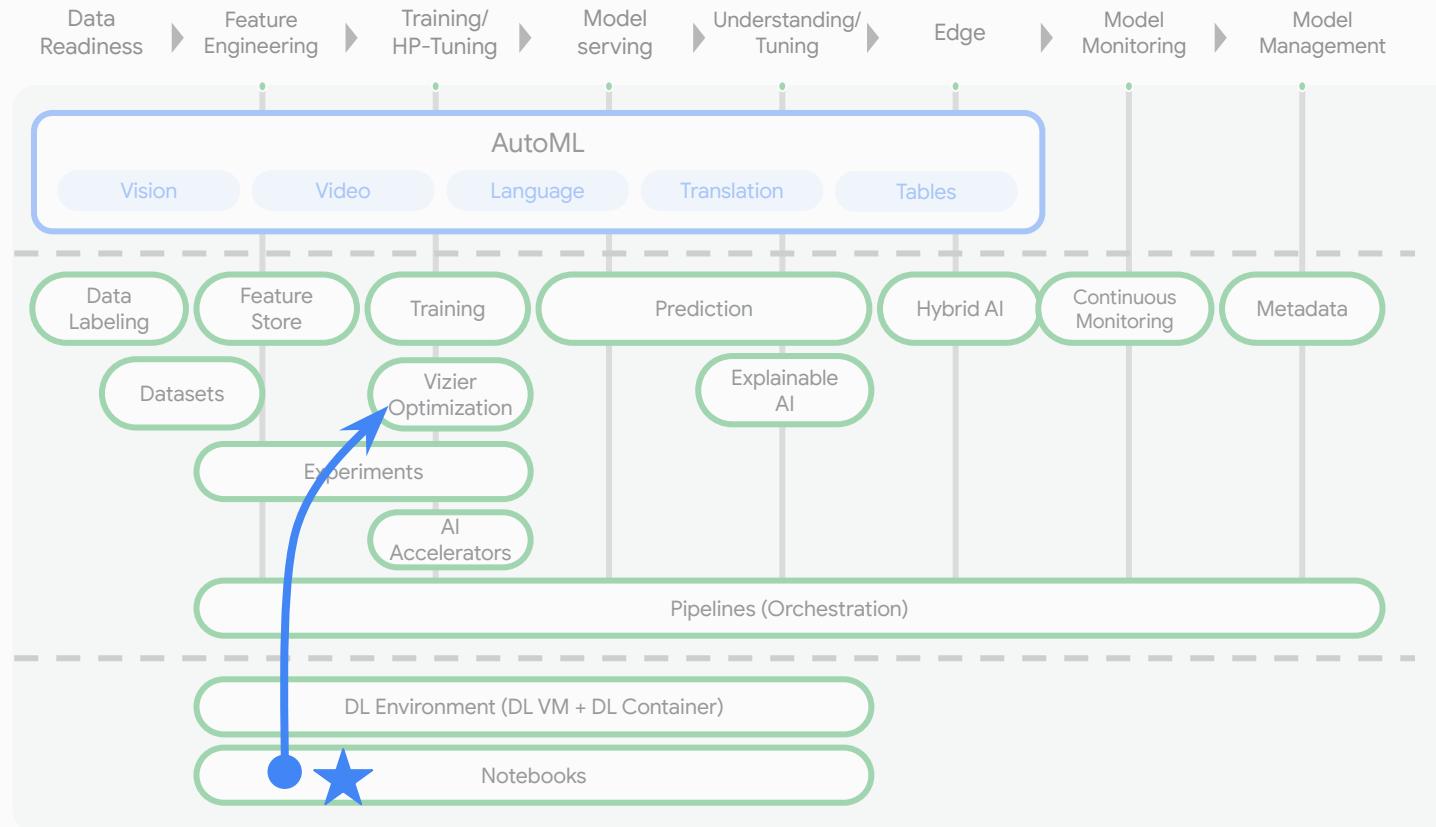


Vertex AI Overview





Vertex AI Overview



Google Cloud Platform

Vertex AI

Dashboard

Datasets

Features

Labeling tasks

Notebooks

Pipelines

Training

Experiments

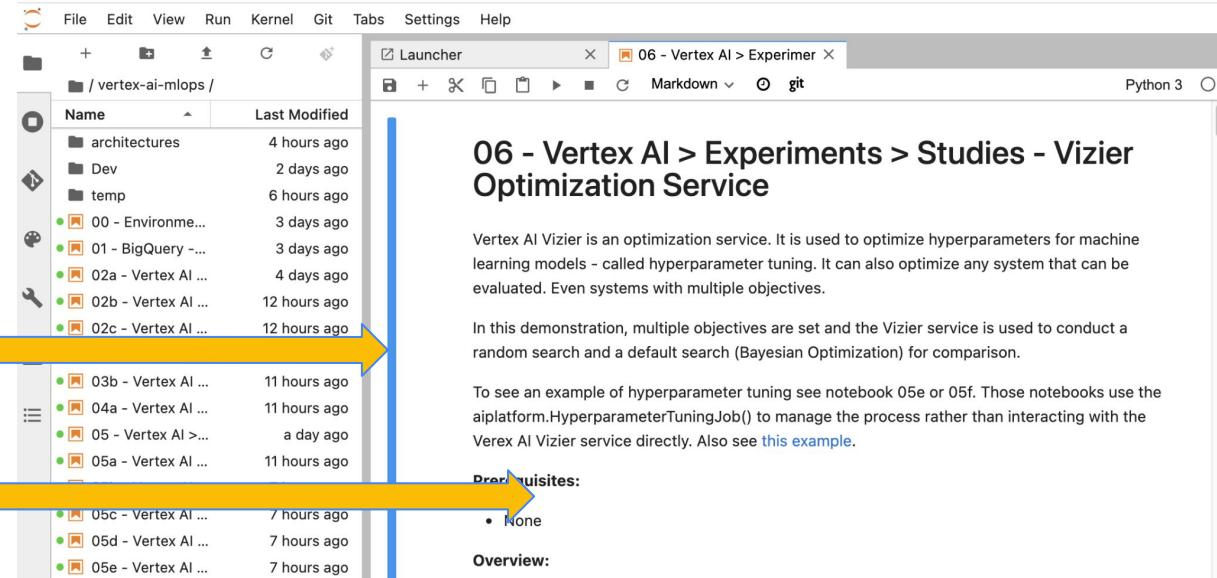
Models

Endpoints

Batch predictions

Metadata

Marketplace



Google Cloud Platform statmike-mlops Search products and resources REFRESH

Vertex AI Studies

EXPERIMENTS PREVIEW STUDIES PREVIEW TENSORBOARD INSTANCES PREVIEW

Vertex Vizier is an optimization service that helps you tune hyperparameters in complex machine learning models. [Learn more](#)

Region us-central1 (Iowa)

Filter Enter property name or value

Study name	ID	Objective	Created
Study_06_Bayesian_Optimization	4214226082825	Minimize "blue" and Maximize "green"	Sep 16, 2021, 11:51:29 AM
Study_06_Random	639592116037	Minimize "blue" and Maximize "green"	Sep 16, 2021, 11:44:46 AM

Vertex AI Overview

