Cloudera Machine Learning Workshop

Agenda -

- Understand how to setup and get started with Cloudera Machine Learning
- Getting started with ML projects, writing code, using various editors, building experiments, deploying models and applications
- Learn about MLOps - Model Metrics and Model Governance
- Administering the environment, users and overall security for CML

Steps

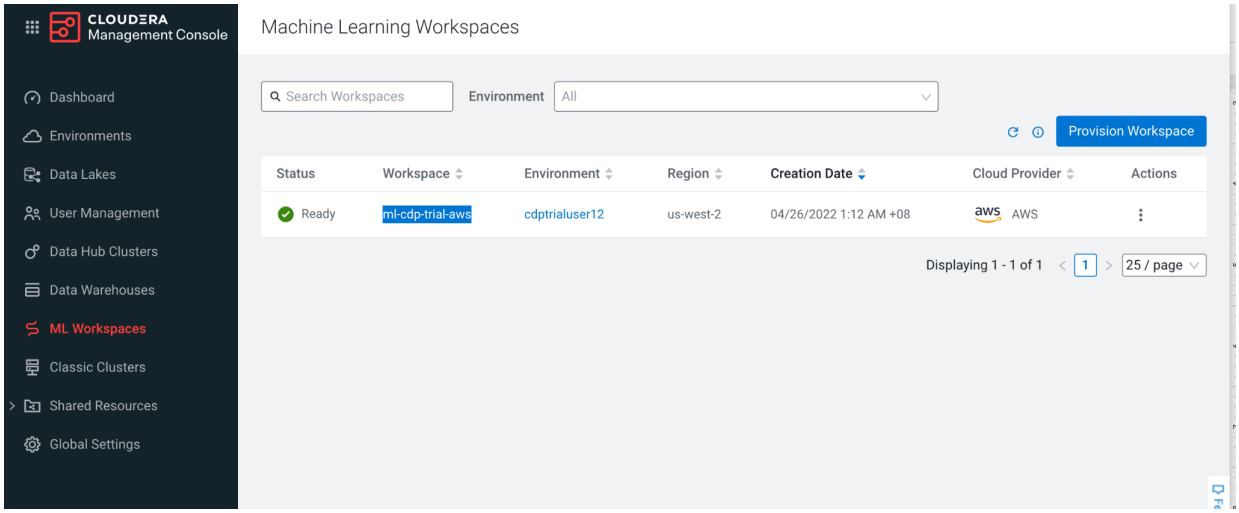
1. **Open CDP, using the "admin" user within the Test Drive link.**

Below is the workshop link, it will only be available for a short time after the workshop. You can register for an extra 48 hours following the workshop through this link.

https://trycdp.com/lease/credentials/200c33214076fa91765eddc2d4c3f91765344427032626cf121eb9650def6657

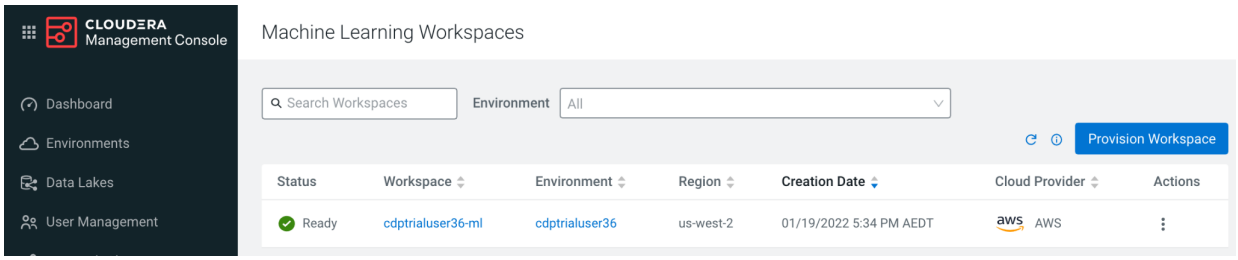
email: manishm@cloudera.com

2. **Click the "Machine Learning" tile within the CDP Home Screen**

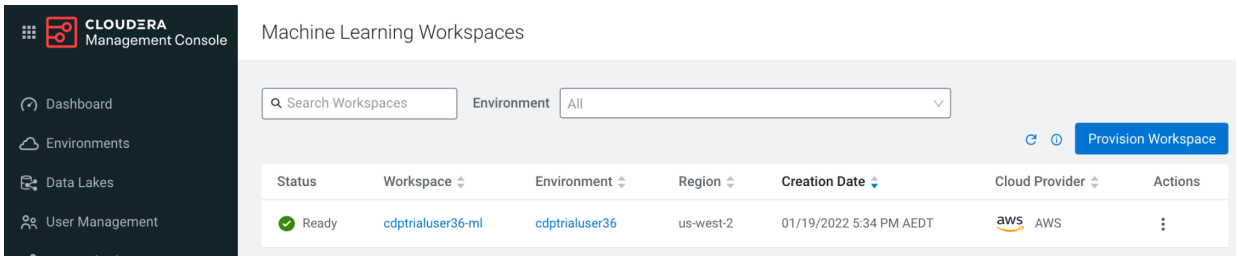


3. **Click on the workspace called " ml-cdp-trial-aws" which is created for you**

Workspaces are the heart of the Cloudera Machine Learning (CML)

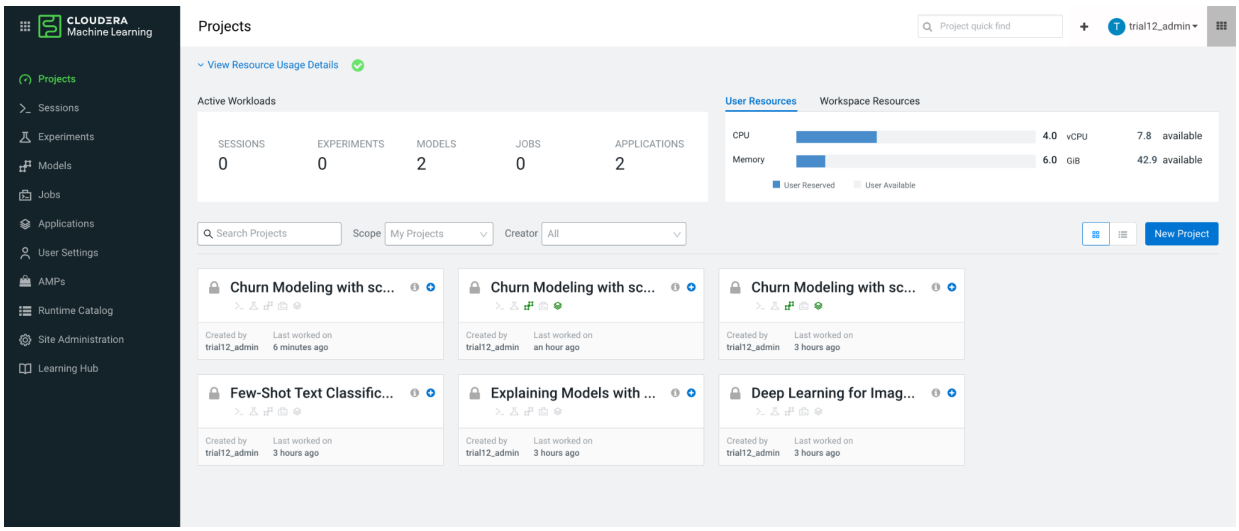


A Workspace is a small cluster that runs on a kubernetes service to provide teams of data scientists to develop, test, train, and ultimately deploy machine learning models. **Click into the Workspace by clicking the Workspace name.**

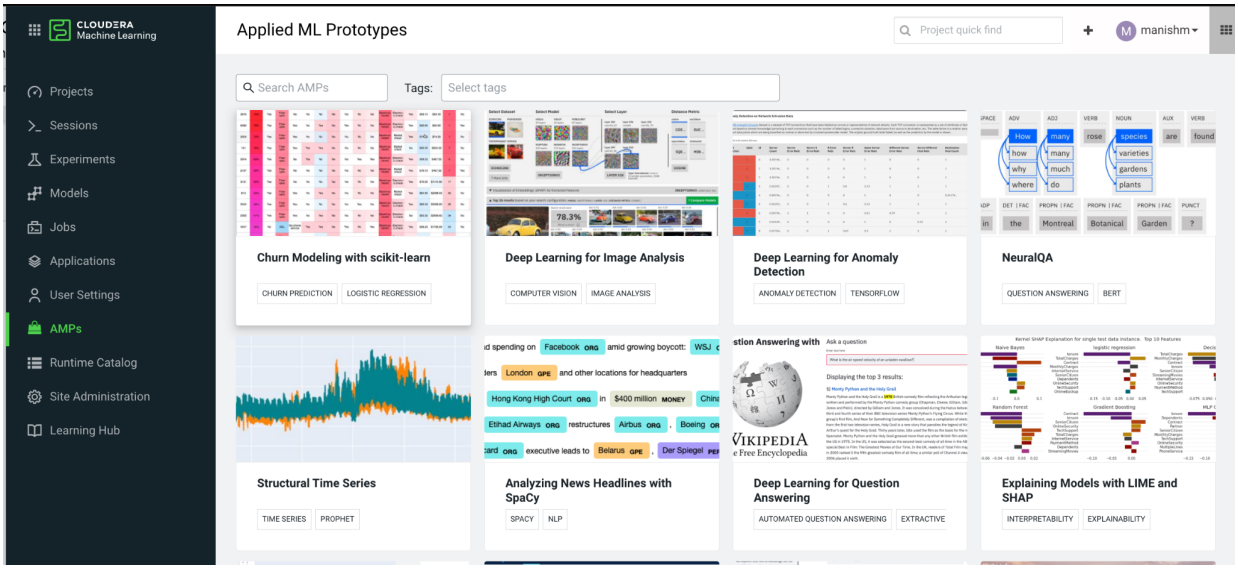


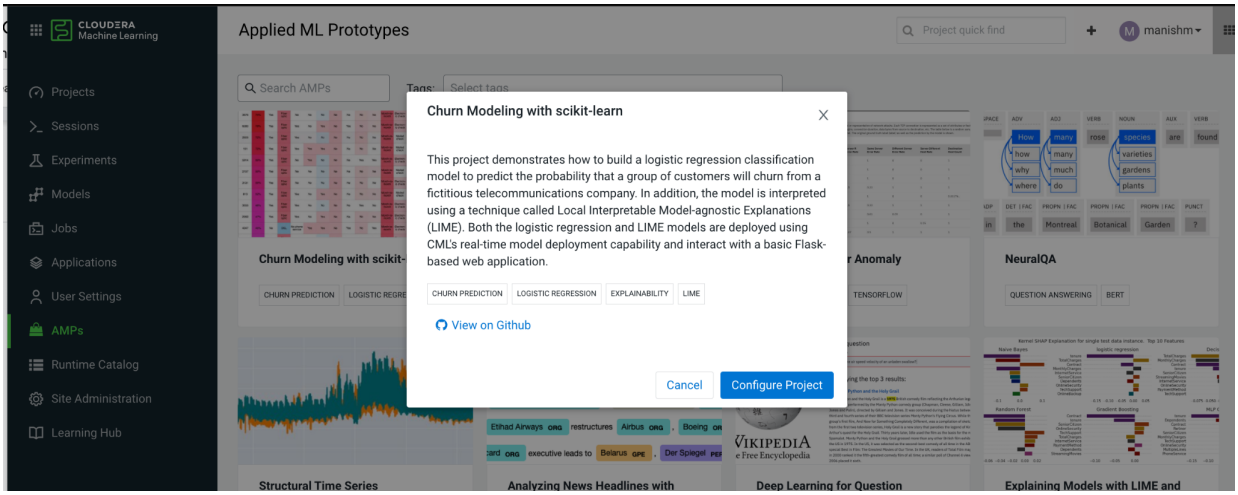
4. **Here you can see all projects created**

You can visualize all of the Projects and Resources that are part of the Projects page. Next we will create a Project where we will develop and deploy models along with other CML features.



5. **Let's make a new project for you - Click on AMP's**





6. **Click on the "Churn Modeling with scikit-learn" AMP and say configure project**

Change the table name to something like **"churn_prototype_<username>"** and execute all the steps.

Configure Project: Churn Modeling with scikit-learn - trial12_admin 2

AMP Name: **ML Churn Prototype (v2)**

Prototype to demonstrate building a churn model on CML

Environment Variables

The settings below were defined by the AMP:

Name	Value	Description
DATA_LOCATION	data/churn_prototype	Relative path that will be used to store the data used for this prototype. This should be a location you have write access to, and which is suitable for non-production data.
HIVE_DATABASE	default	Name of the Hive database that will be used to create the Hive table used for this prototype. This should be a Hive database you have write access to, and which is suitable for non-production data.
HIVE_TABLE	churn_prototype_username	Name of the Hive table that will be created and populated with the data used for this prototype. If the table already exists, the prototype will assume it already contains the data for this prototype.

Default Engine:

Runtime | Engine

Runtime

Editor	Kernel	Edition	Version
Workbench	Python 3.7	Standard	2021.12

Enable Spark — Spark 2.4.7 - CDP 7.2.11 - CDE 1.13 - HO...

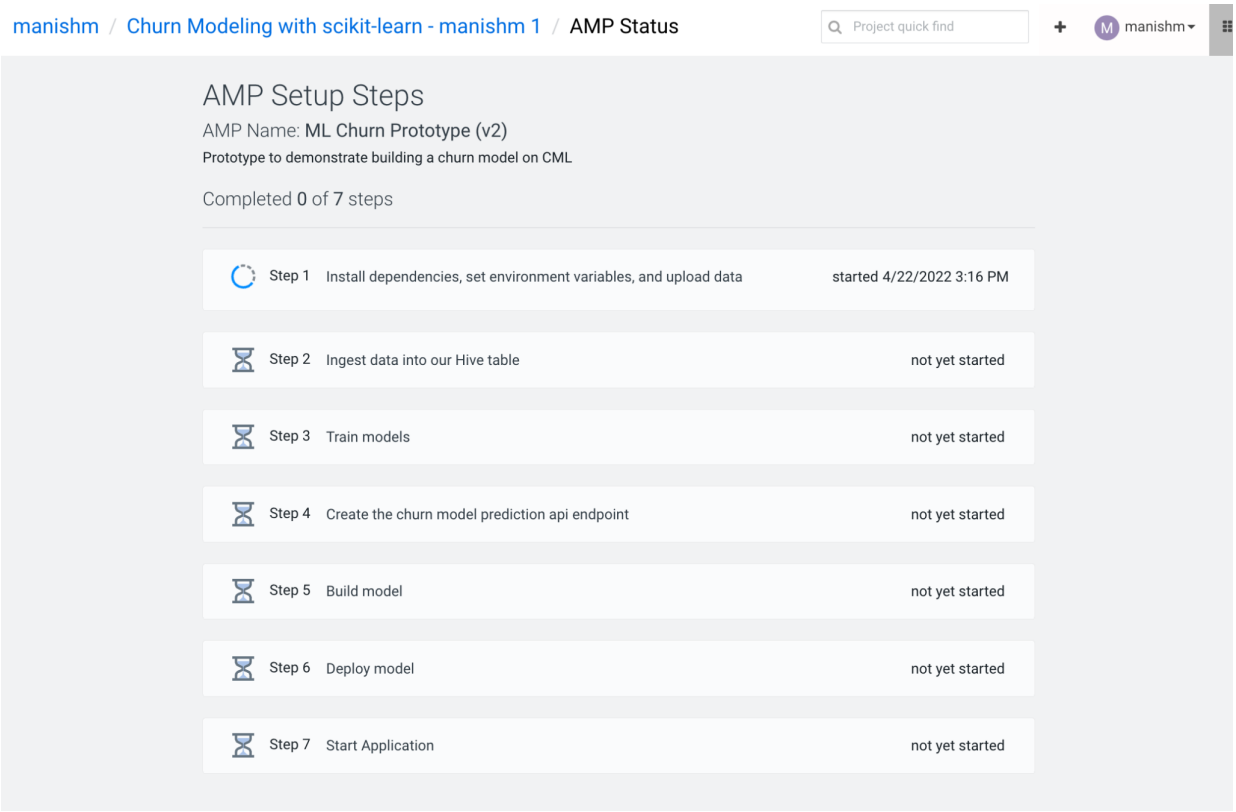
Runtime Image

- docker.repository.cloudera.com/cloudera/cdsw/ml-runtime-workbench-python3.7-standard:2021.12.1-b17

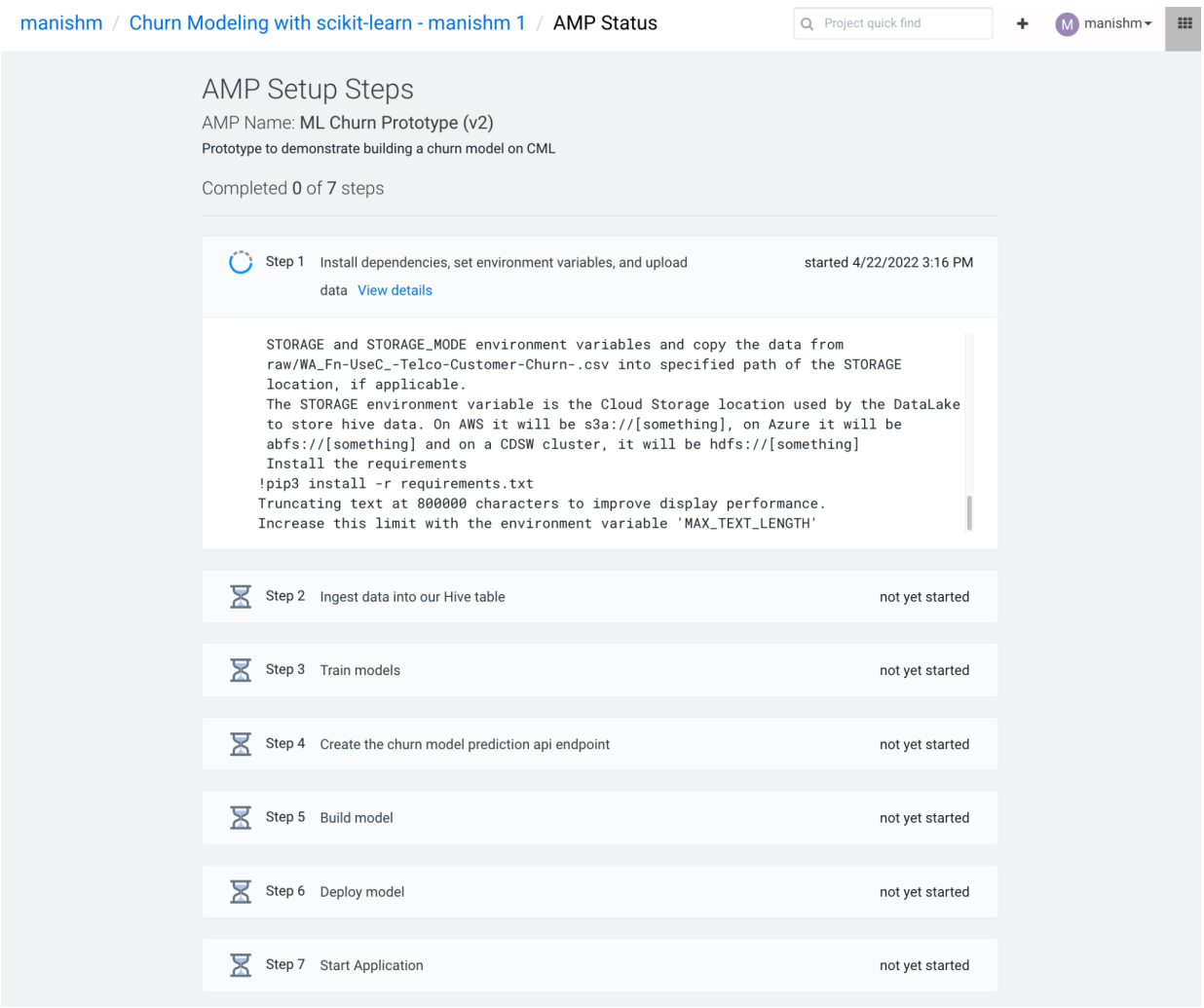
⚠ The runtime addons required for this AMP is not present. Here is the list of runtime addons required for this AMP: Spark 2.4.7 - CDP 7.2.10 - CDE 1.11

Setup Steps

☑ Execute AMP setup steps



7. **A project will get created called “Churn Modeling with scikit-learn - <username> 1”. Wait for all the steps to finish**



Wait for all the steps to finish. I will take approximately 10 minutes to complete.

AMP Setup Steps

AMP Name: **ML Churn Prototype (v2)**

Prototype to demonstrate building a churn model on CML

Completed **4** of **7** steps

Step 1 Install dependencies, set environment variables, and upload data View details — completed 4/22/2022 3:21 PM

```
the qualifiedName of the hive_table object representing
        metadata:                                                    #
this is a predefined key for additional metadata
            query: "select * from historical_data"                   #
suggested use case: query used to extract training data
            training_file: "code/4_train_models.py"                  #
suggested use case: training file used
    """
with open("lineage.yml", "w") as lineage:
    lineage.write(yaml_text)
```

Step 2 Ingest data into our Hive table View details — completed 4/22/2022 3:23 PM

```
learning/cloud/jobs-pipelines/topics/ml-creating-a-job.html).
 > You can create a Job with specified command line arguments or environment
variables.
 > Jobs can be triggered by the completion of other jobs, forming a
 > [Pipeline](https://docs.cloudera.com/machine-learning/cloud/jobs-
pipelines/topics/ml-creating-a-pipeline.html)
 > You can configure the job to email individuals with an attachment, e.g. a csv
report which your
 > script saves at: `/home/cdsw/job1/output.csv`.
 Try running this script `1_data_ingest.py` for use in such a Job.
```

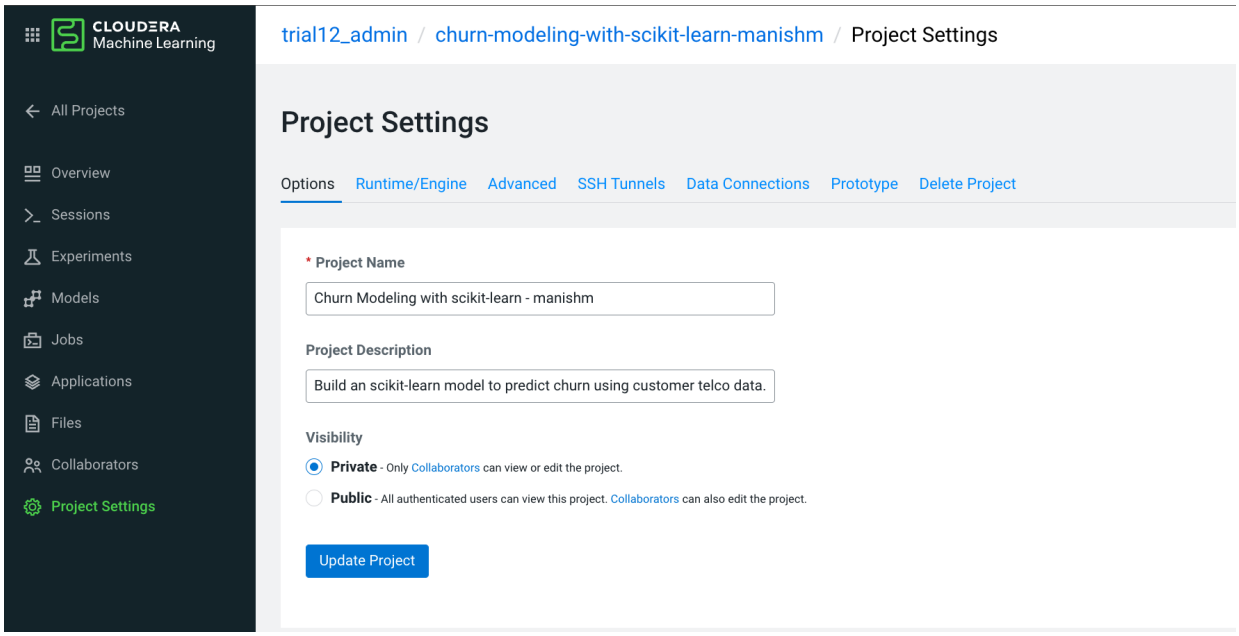
Step 3 Train models View details — completed 4/22/2022 3:23 PM

```
 Notice also that any script that will run as an Experiment can also be run as a Job
or in a Session.
 Our provided script can be run with the same settings as for Experiments.
 A common use case is to **automate periodic model updates**.
 Jobs can be scheduled to run the same model training script once a week using the
latest data.
 Another Job dependent on the first one can update the model parameters being used
in production
 if model metrics are favorable.
```

8. **Once finished let's look at the overview of the project**

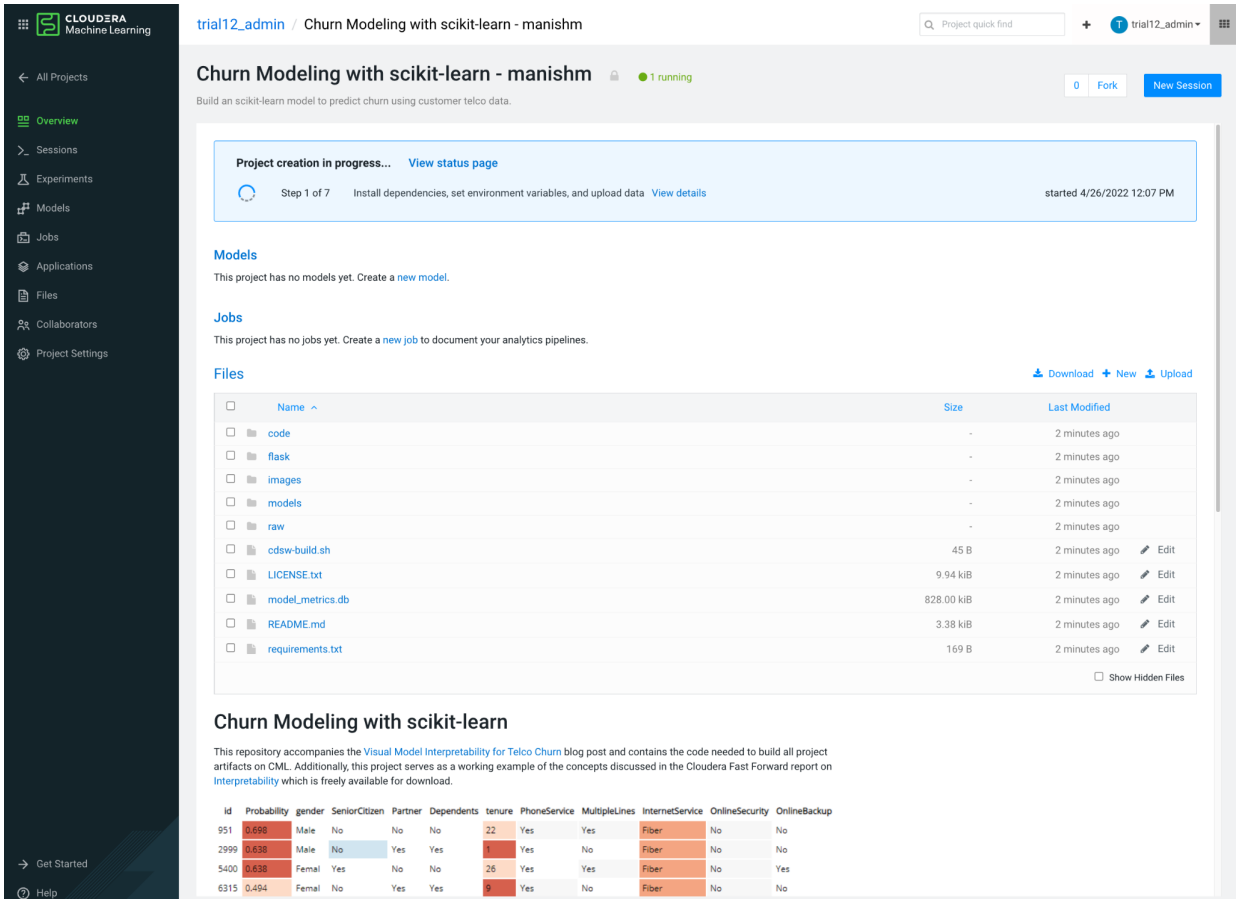
Overview gives you access to all the features of a CML project. Initially it is good to start on the management components of a project. The project has deployed Jobs, Models and Applications

In project settings, Lets rename the project to something like - “Churn Modeling with scikit-learn - <you name>”

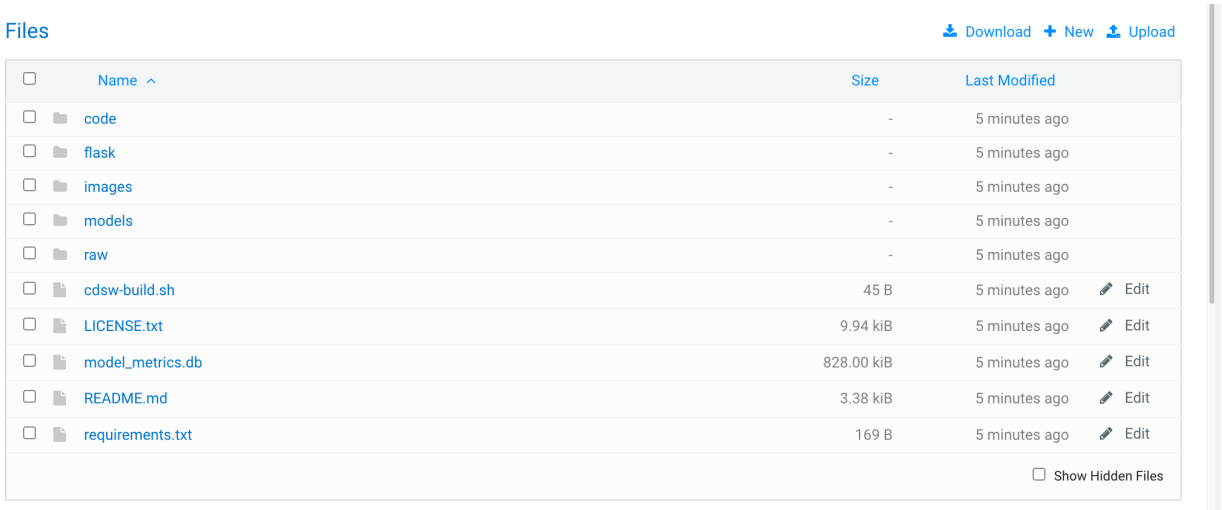


9. Project Basics

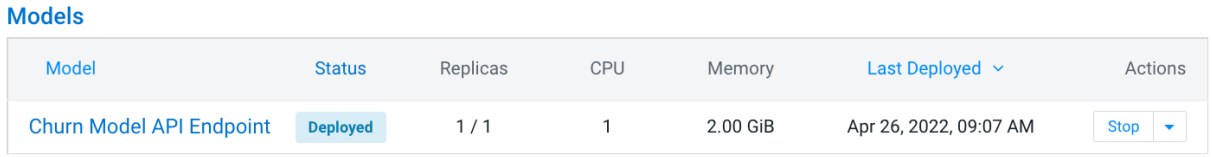
a. Overview



b. Code

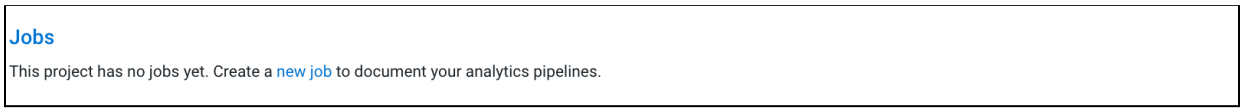


c. Models



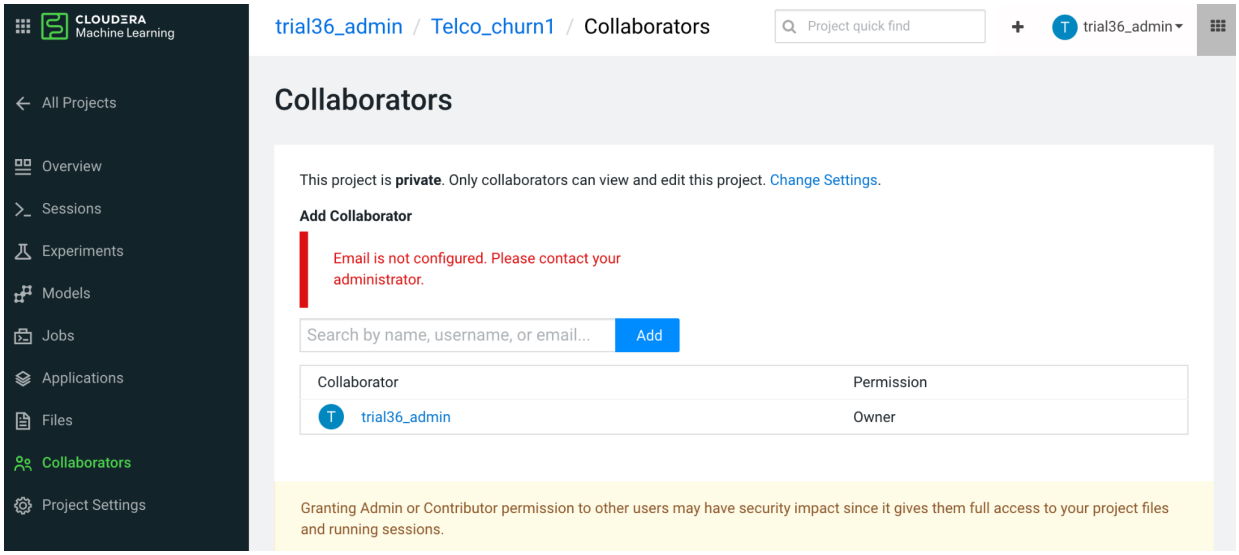
d. Jobs

We will create a job later.



e. Project Collaborators

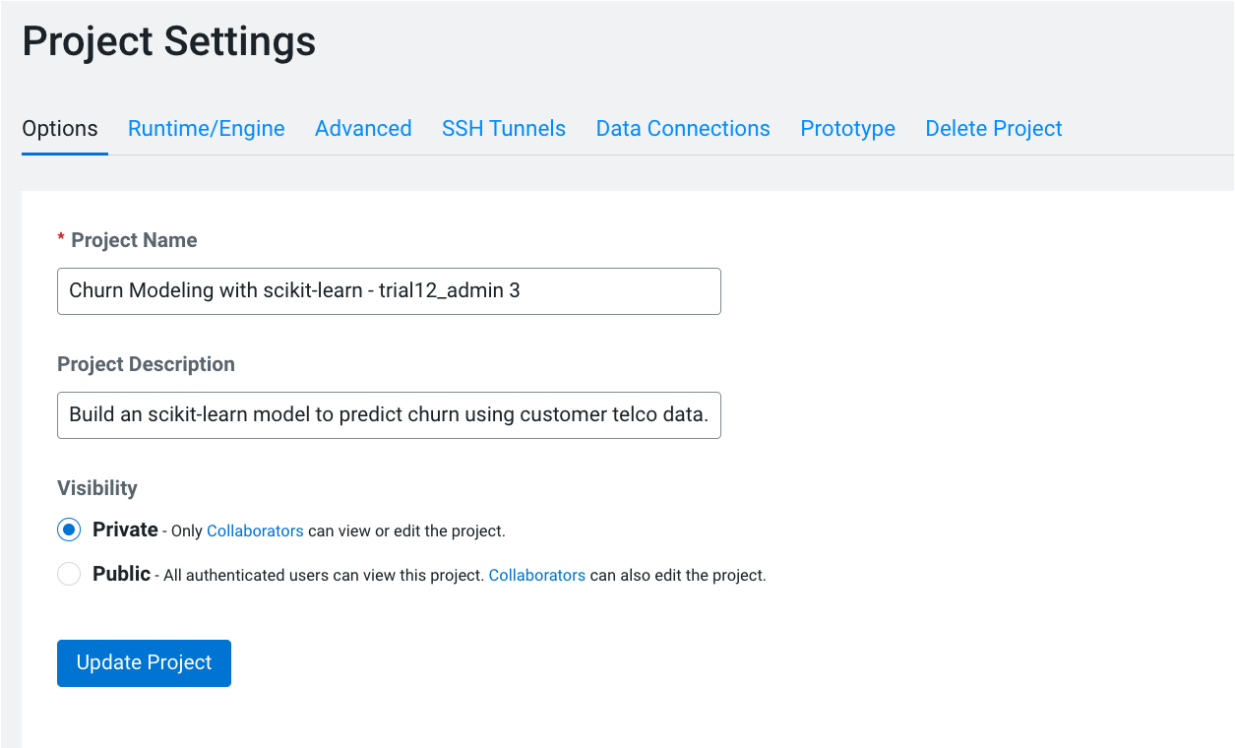
You can give access to other users with certain permissions for the encompassing project so teams of users can collaborate together. You can set up Admins, Contributor, Operator, and Viewer permissions. (Since email is not configured this won't work in this environment)



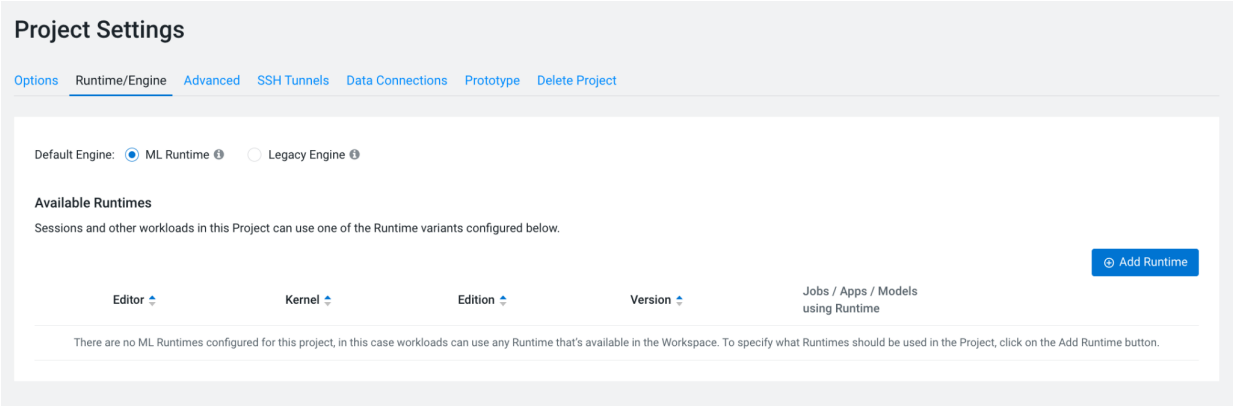
10. **Project Settings**

Taking a look at Project Settings, this is where you can define several options for the current project. You have the ability to define different engines where your code in CML will run. There are project variables that can be defined and used throughout your code. SSH tunnels can also be configured to connect to other services as needed. More details can be found in our docs here.

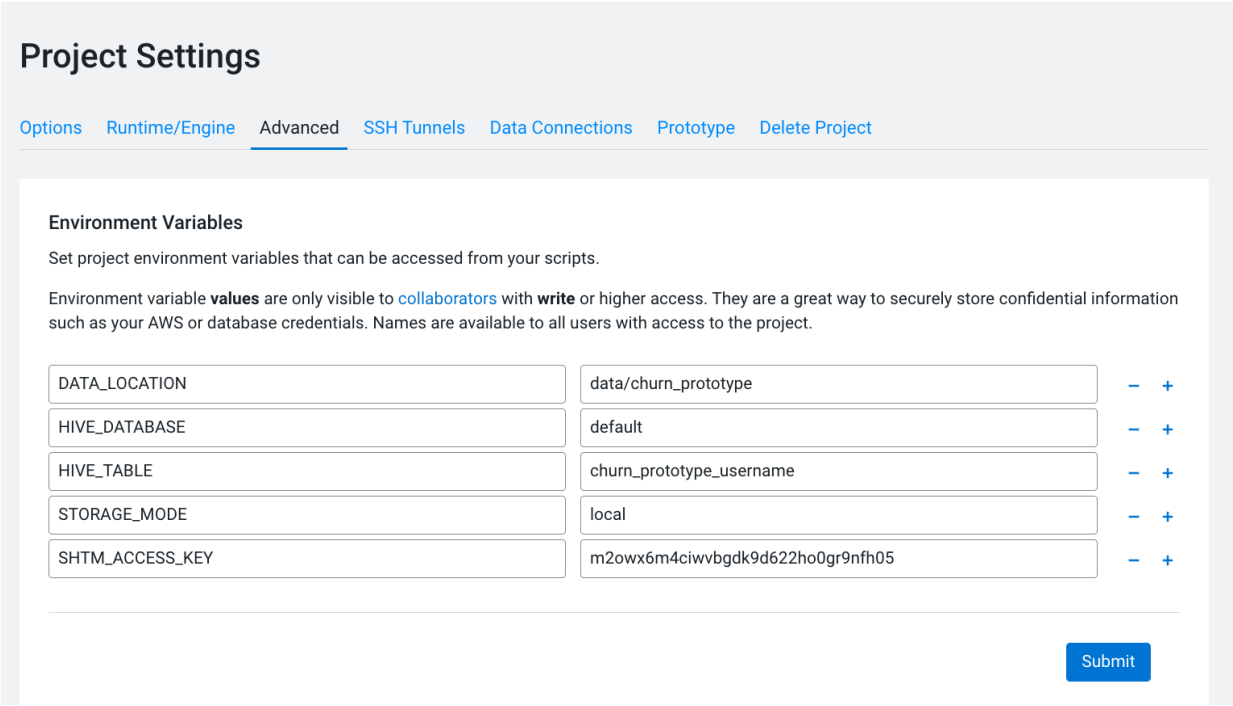
Change Project Names -



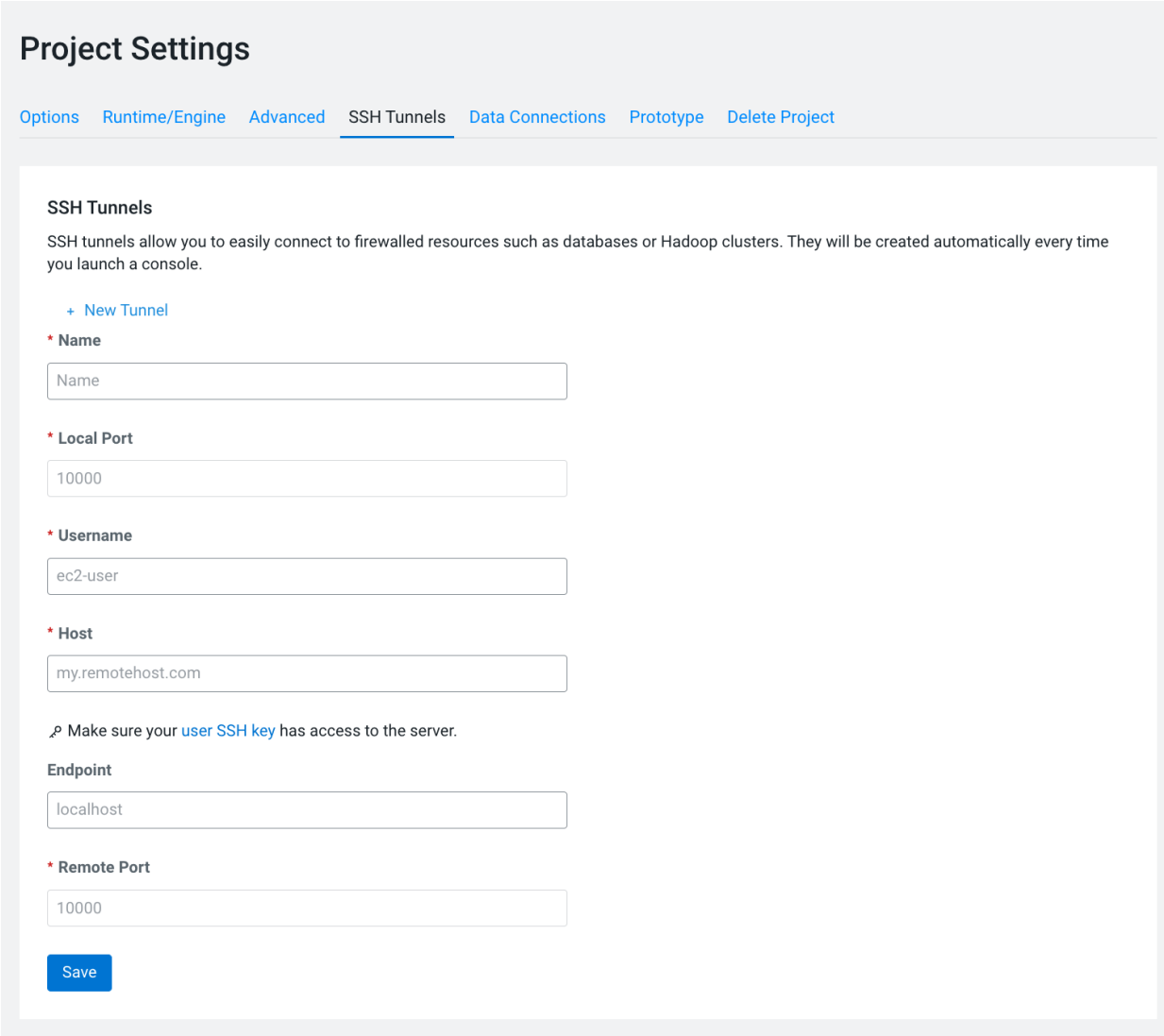
Change Runtimes -



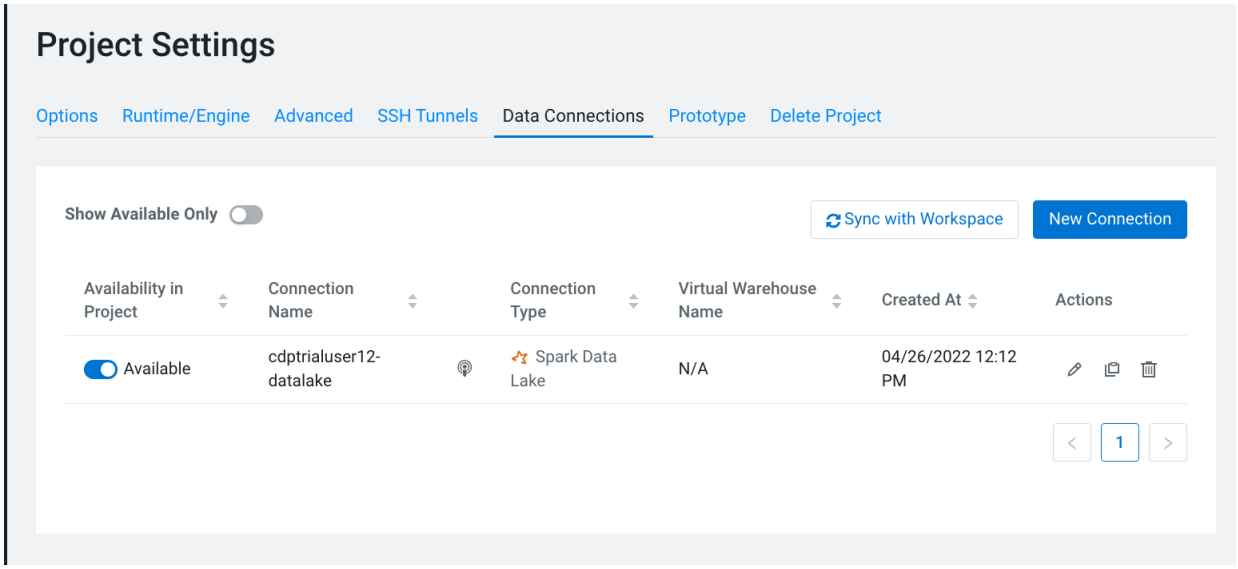
Set Variables -



Use SSH Tunneling -



Data Connections -



11. Lets see how to run some code

a. Running Code in Workbench

Sessions allow you to perform actions such as run R or Python code. They also provide access to an interactive command prompt and terminal. Sessions will be built on a specified Engine Image, which is a docker container that is deployed onto the Workspace. In addition you can specify how many resources are used per session. From the Overview page click on New Session. You can select the CPU, Memory and GPU for each session along with editors, and other addons.

Start A New Session

New version available

There is a new version of Base Image available. Latest engine image is: "Default engine image".

Update Engine

Session Name

Untitled Session

Runtime

Editor ⓘ

Workbench

Kernel ⓘ

Python 3.7

Edition ⓘ

Standard

Version

2021.12

Configure additional runtime options in Project Settings.

Enable Spark ⓘ

Spark 2.4 - CDP 7.2.8 (TP)

Runtime Image

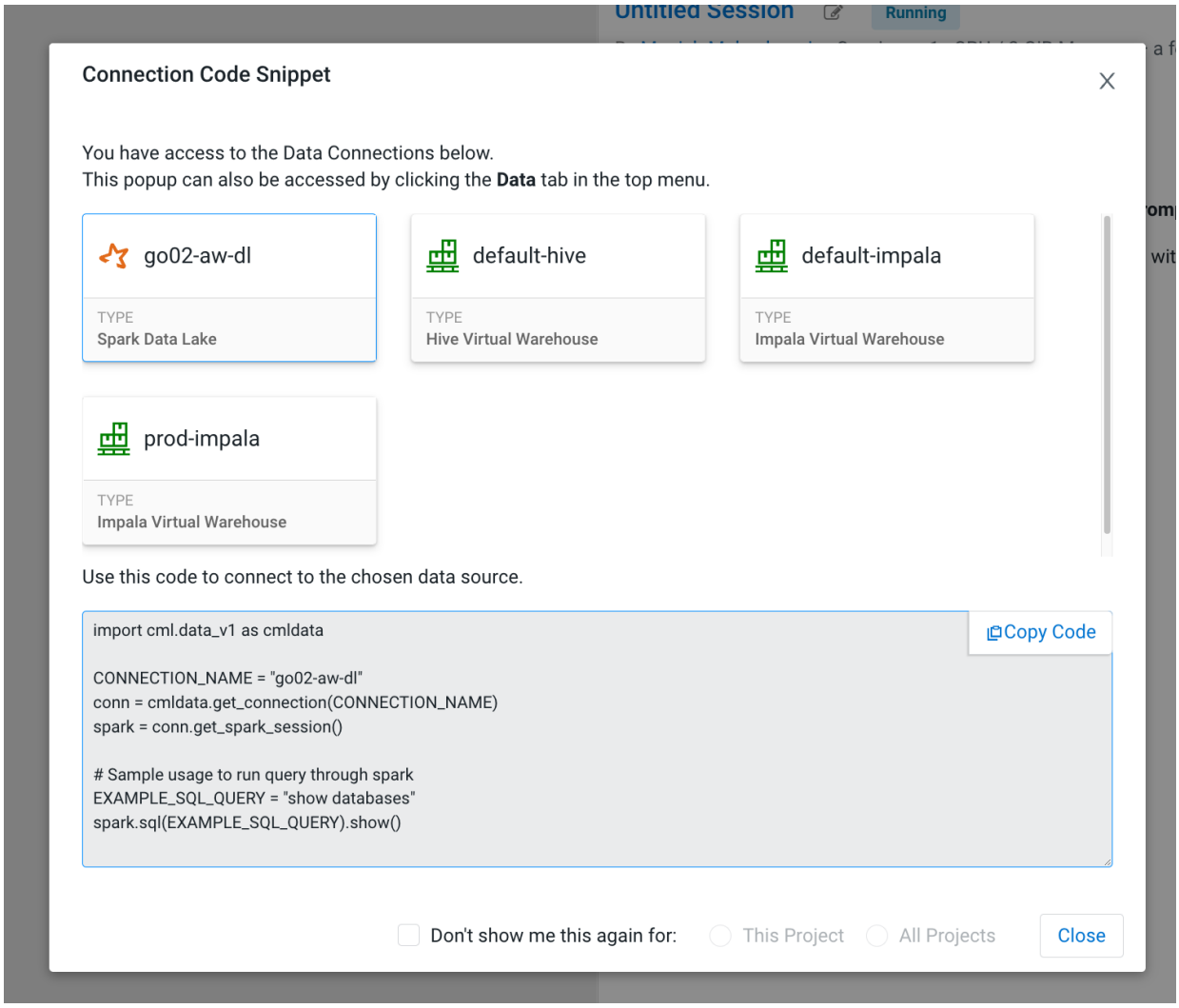
- docker.repository.cloudera.com/cloudera/cdsw/ml-runtime-workbench-python3.7-standard:2021.12.1-b17

Resource Profile

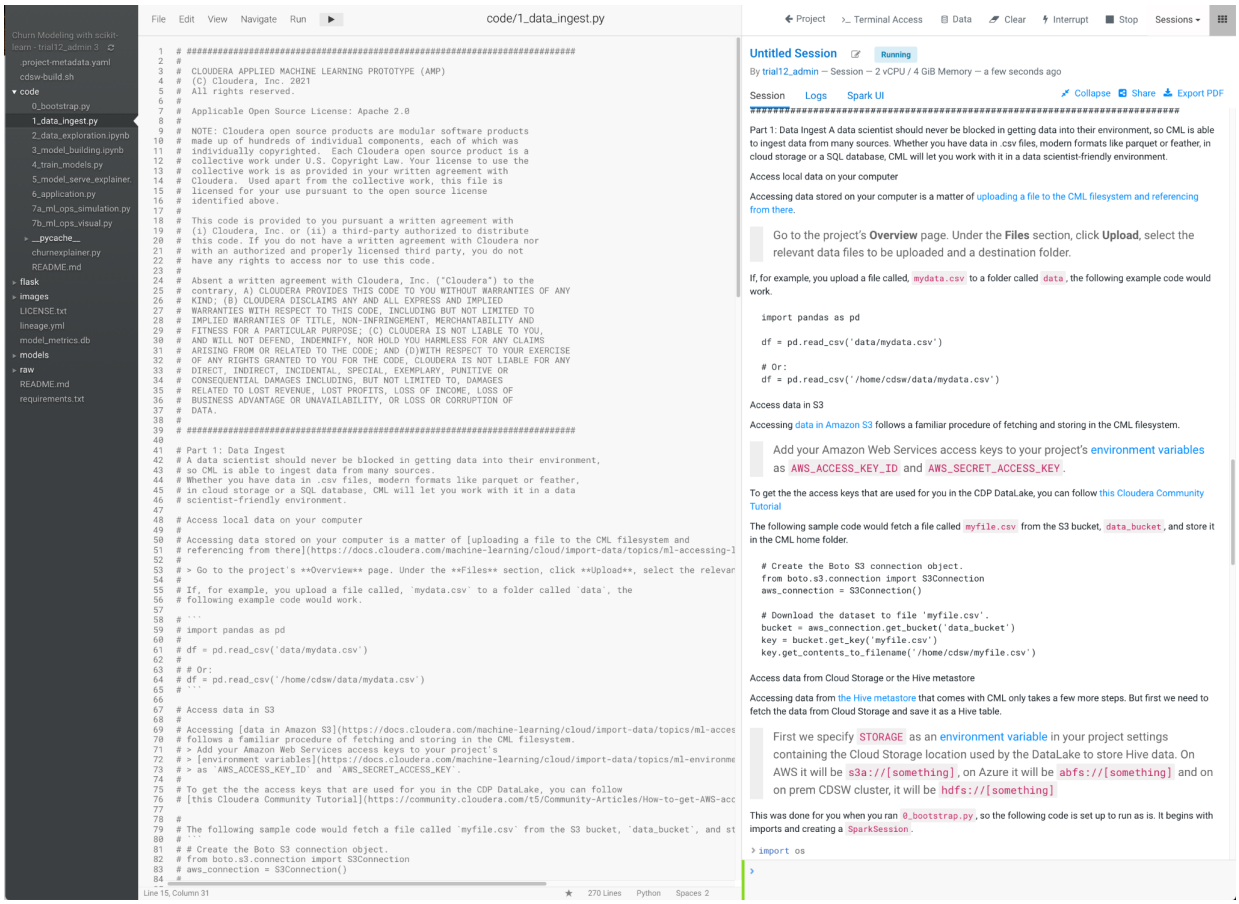
1 vCPU / 2 GiB Memory

0 GPUs

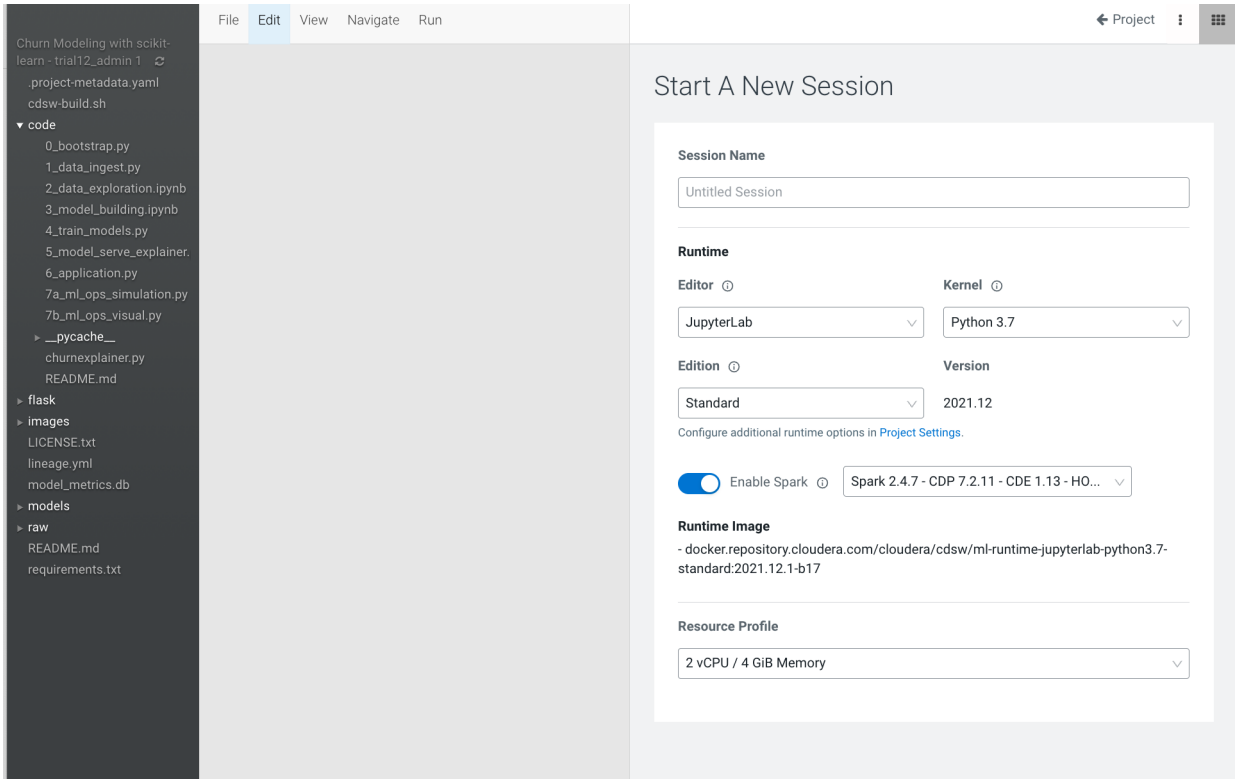
Connection snippets allow you to connect to “Cloudera Datawarehouse” to pull data from Impala/Hive. You can copy code as needed or just close the window.

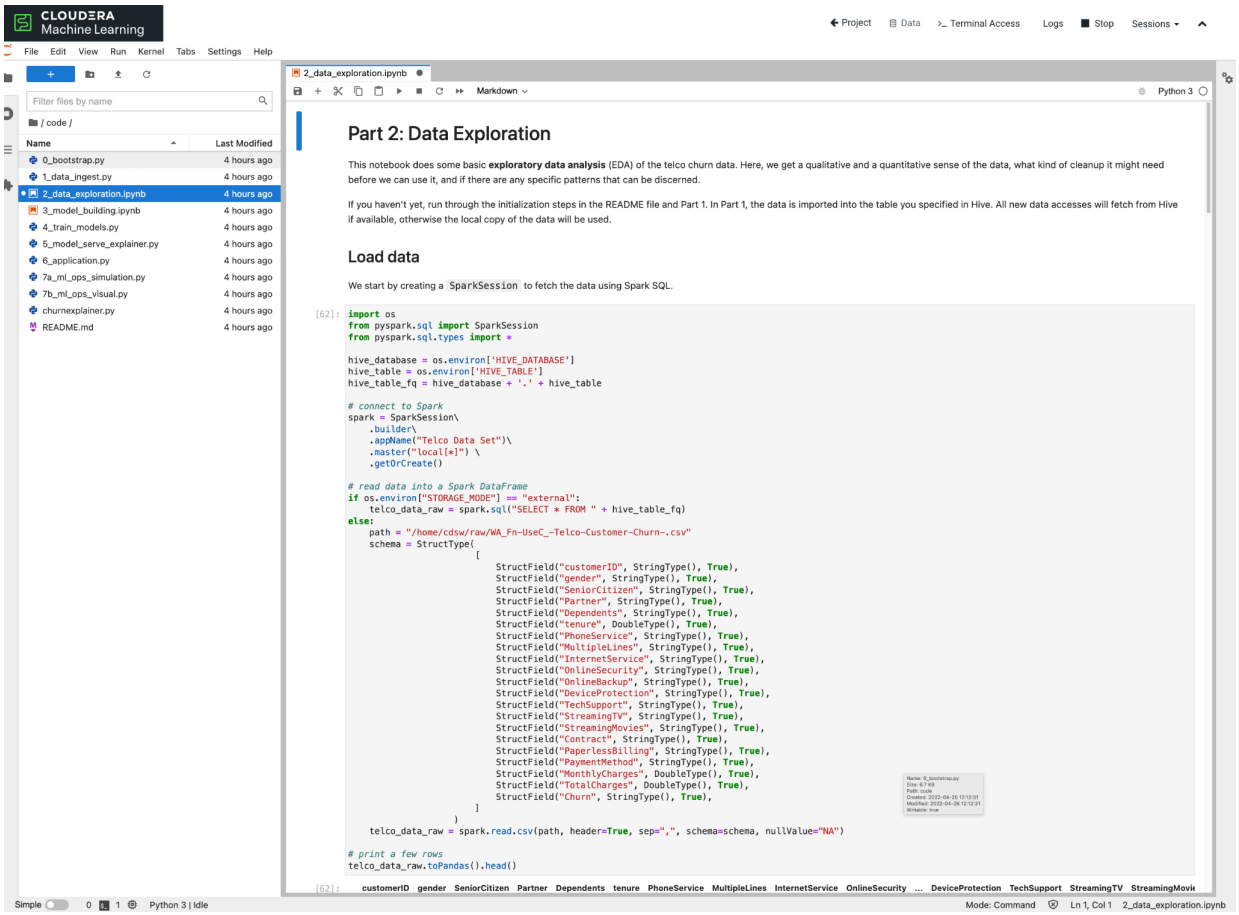


Run the script of code/1_data_ingest.py and see how you can run code in Workbench.


code/1_data_ingest.py
Untitled Session
Running
Part 1: Data Ingest A data scientist should never be blocked in getting data into their environment, so CML is able to ingest data from many sources. Whether you have data in .csv files, modern formats like parquet or feather, in cloud storage or a SQL database, CML will let you work with it in a data scientist-friendly environment.
Access local data on your computer
Accessing data stored on your computer is a matter of uploading a file to the CML filesystem and referencing from there.
Go to the project's Overview page. Under the Files section, click Upload, select the relevant data files to be uploaded and a destination folder.
If, for example, you upload a file called, mydata.csv to a folder called data, the following example code would work.
import pandas as pd
df = pd.read_csv('data/mydata.csv')
# Or:
df = pd.read_csv('/home/cdsw/data/mydata.csv')
Access data in S3
Accessing data in Amazon S3 follows a familiar procedure of fetching and storing in the CML filesystem.
Add your Amazon Web Services access keys to your project's environment variables as AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY.
To get the the access keys that are used for you in the CDP DataLake, you can follow this Cloudera Community Tutorial
The following sample code would fetch a file called myfile.csv from the S3 bucket, data_bucket, and store it in the CML home folder.
# Create the Boto S3 connection object.
from boto.s3.connection import S3Connection
aws_connection = S3Connection()
# Download the dataset to file 'myfile.csv'.
bucket = aws_connection.get_bucket('data_bucket')
key = bucket.get_key('myfile.csv')
key.get_contents_to_filename('/home/cdsw/myfile.csv')
Access data from Cloud Storage or the Hive metastore
Accessing data from the Hive metastore that comes with CML only takes a few more steps. But first we need to fetch the data from Cloud Storage and save it as a Hive table.
First we specify STORAGE as an environment variable in your project settings containing the Cloud Storage location used by the DataLake to store Hive data. On AWS it will be s3a://[something], on Azure it will be abfs://[something] and on on prem CDSW cluster, it will be hdfs://[something]
This was done for you when you ran 0_bootstrap.py, so the following code is set up to run as is. It begins with imports and creating a SparkSession.
> import os

b. Running Code in Jupyter Lab





12. Creating Jobs

Lets see how you can create and schedule jobs.

trial12_admin / Churn Modeling with scikit-learn - trial12_admin 3 / Jobs / New Job

Create a Job

General

Name

Run Data Ingestion

Script

code/1_data_ingest.py

Arguments

Arguments

Runtime

Editor JupyterLab

Kernel Python 3.7

Edition Standard

Version 2021.12

Configure additional runtime options in Project Settings.

Enable Spark Spark 2.4.7 - CDP 7.2.11 - CDE 1.13 - HO...

Runtime Image

- docker.repository.cloudera.com/cloudera/cdsw/ml-runtime-jupyterlab-python3.7-standard:2021.12.1-b17

Schedule

Manual

Resource Profile

2 vCPU / 4 GiB Memory

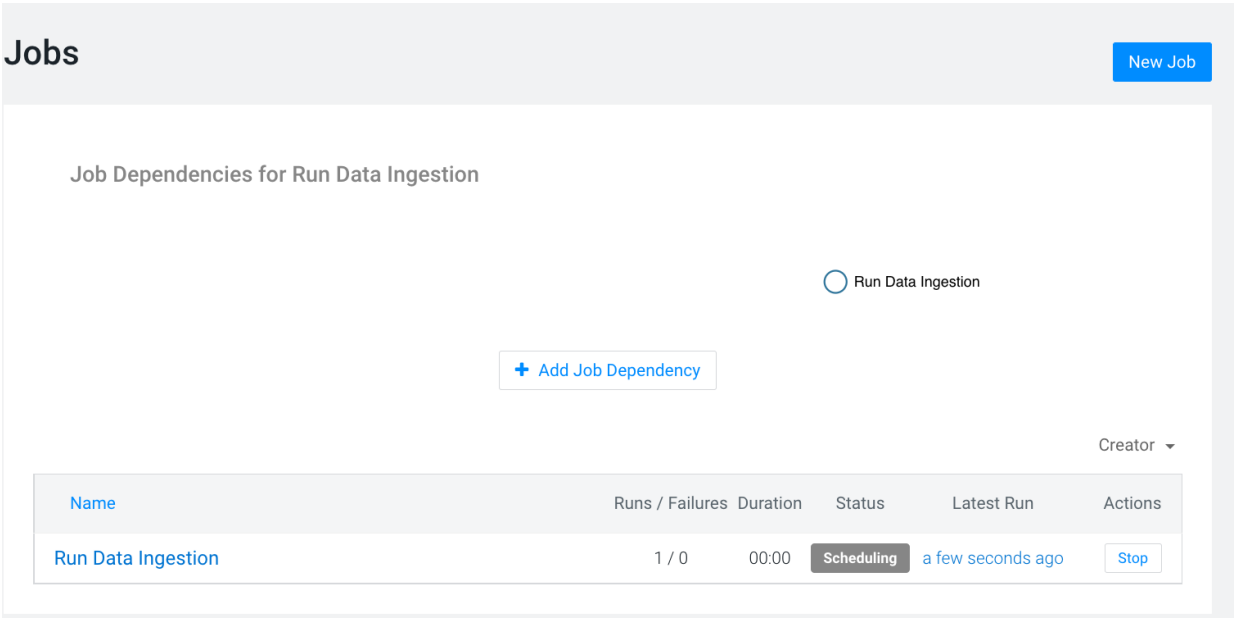
Timeout In Minutes (optional) 30 **Kill on Timeout**

Jobs exceeding timeout send warning email if notifications enabled.

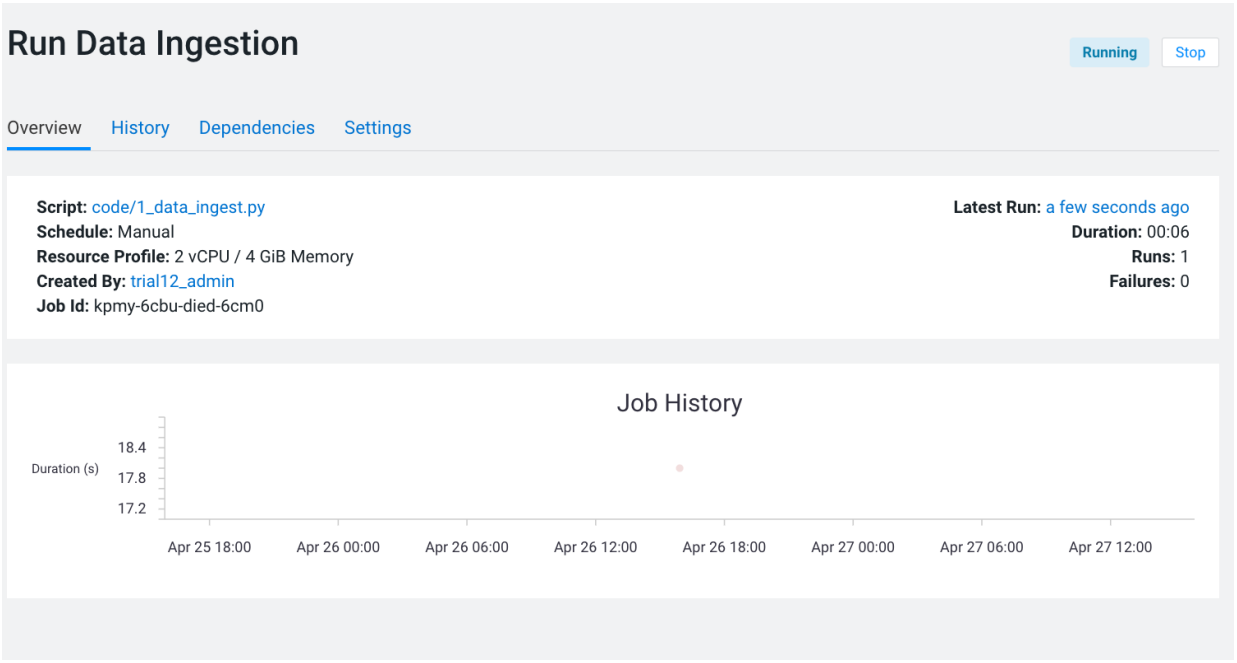
Environment Variables

Name	Value	Actions
		Add

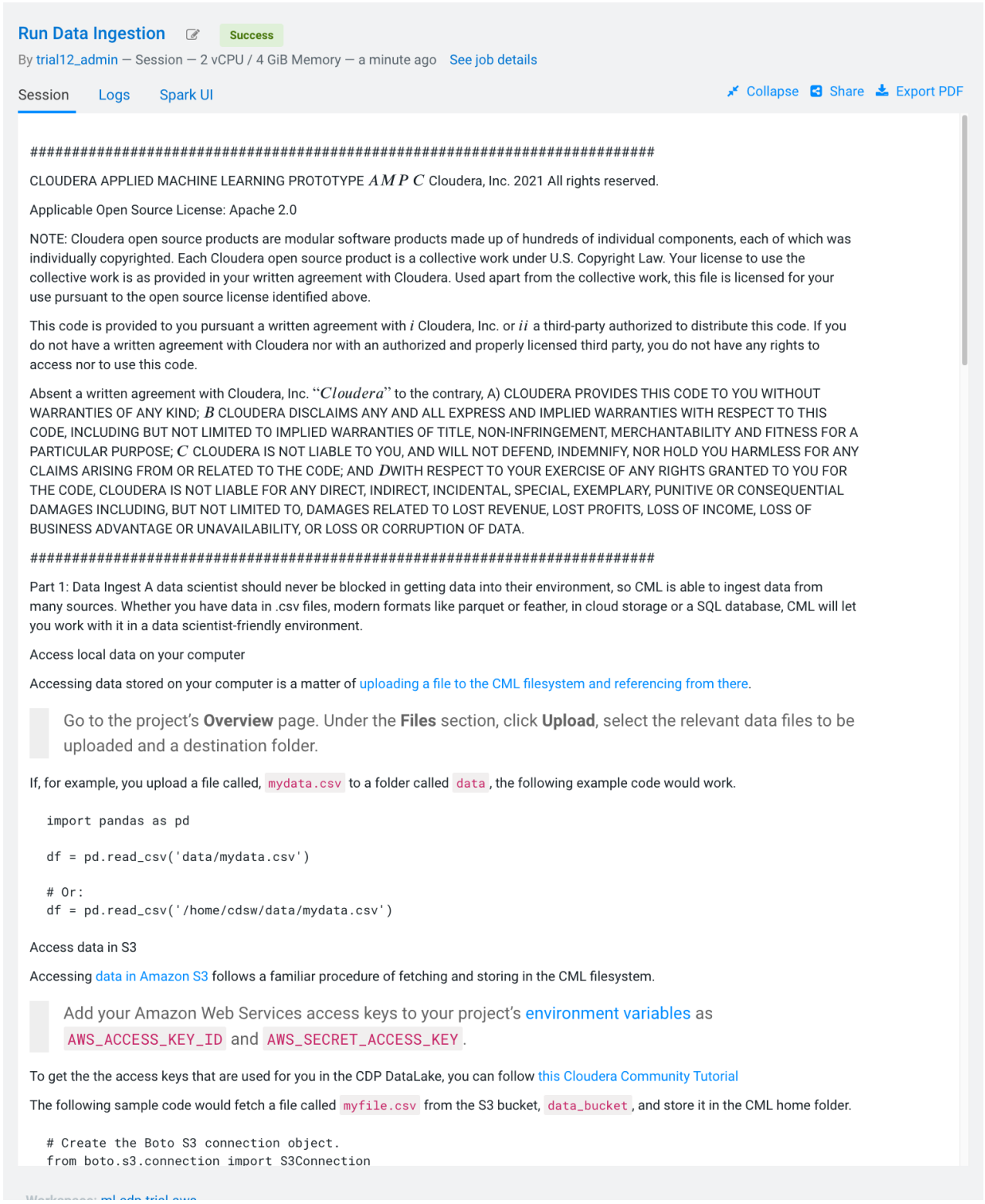
Environment variables will override the project environment.



Look at Job History



Look and share Job Output for each run -



Run Data Ingestion Success

By trial12_admin — Session — 2 vCPU / 4 GiB Memory — a minute ago See job details

Session Logs Spark UI Collapse Share Export PDF

##

CLOUDERA APPLIED MACHINE LEARNING PROTOTYPE $AMPC$ Cloudera, Inc. 2021 All rights reserved.

Applicable Open Source License: Apache 2.0

NOTE: Cloudera open source products are modular software products made up of hundreds of individual components, each of which was individually copyrighted. Each Cloudera open source product is a collective work under U.S. Copyright Law. Your license to use the collective work is as provided in your written agreement with Cloudera. Used apart from the collective work, this file is licensed for your use pursuant to the open source license identified above.

This code is provided to you pursuant a written agreement with i Cloudera, Inc. or ii a third-party authorized to distribute this code. If you do not have a written agreement with Cloudera nor with an authorized and properly licensed third party, you do not have any rights to access nor to use this code.

Absent a written agreement with Cloudera, Inc. "$Cloudera$" to the contrary, A) CLOUDERA PROVIDES THIS CODE TO YOU WITHOUT WARRANTIES OF ANY KIND; B CLOUDERA DISCLAIMS ANY AND ALL EXPRESS AND IMPLIED WARRANTIES WITH RESPECT TO THIS CODE, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE; C CLOUDERA IS NOT LIABLE TO YOU, AND WILL NOT DEFEND, INDEMNIFY, NOR HOLD YOU HARMLESS FOR ANY CLAIMS ARISING FROM OR RELATED TO THE CODE; AND DWITH RESPECT TO YOUR EXERCISE OF ANY RIGHTS GRANTED TO YOU FOR THE CODE, CLOUDERA IS NOT LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, PUNITIVE OR CONSEQUENTIAL DAMAGES INCLUDING, BUT NOT LIMITED TO, DAMAGES RELATED TO LOST REVENUE, LOST PROFITS, LOSS OF INCOME, LOSS OF BUSINESS ADVANTAGE OR UNAVAILABILITY, OR LOSS OR CORRUPTION OF DATA.

##

Part 1: Data Ingest A data scientist should never be blocked in getting data into their environment, so CML is able to ingest data from many sources. Whether you have data in .csv files, modern formats like parquet or feather, in cloud storage or a SQL database, CML will let you work with it in a data scientist-friendly environment.

Access local data on your computer

Accessing data stored on your computer is a matter of uploading a file to the CML filesystem and referencing from there.

> Go to the project's **Overview** page. Under the **Files** section, click **Upload**, select the relevant data files to be uploaded and a destination folder.

If, for example, you upload a file called, `mydata.csv` to a folder called `data`, the following example code would work.

```
import pandas as pd

df = pd.read_csv('data/mydata.csv')

# Or:
df = pd.read_csv('/home/cdsw/data/mydata.csv')
```

Access data in S3

Accessing data in Amazon S3 follows a familiar procedure of fetching and storing in the CML filesystem.

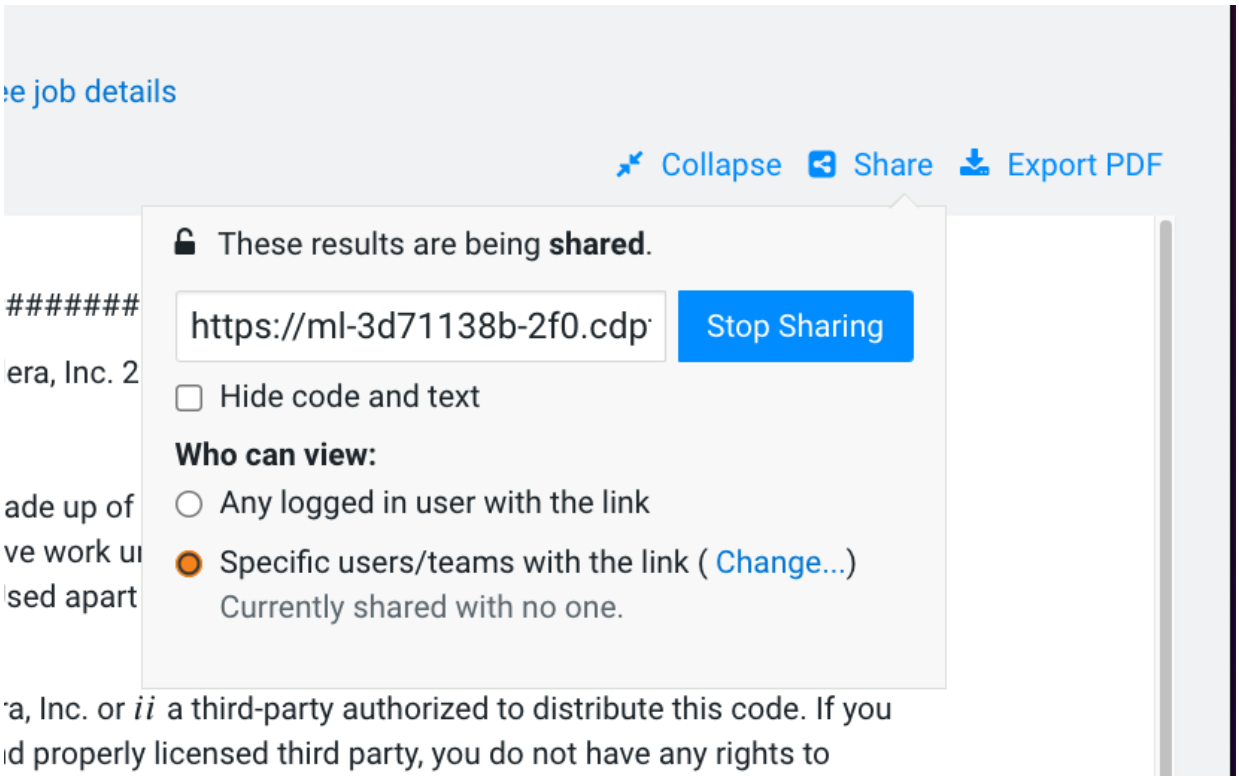
> Add your Amazon Web Services access keys to your project's environment variables as `AWS_ACCESS_KEY_ID` and `AWS_SECRET_ACCESS_KEY`.

To get the the access keys that are used for you in the CDP DataLake, you can follow this Cloudera Community Tutorial

The following sample code would fetch a file called `myfile.csv` from the S3 bucket, `data_bucket`, and store it in the CML home folder.

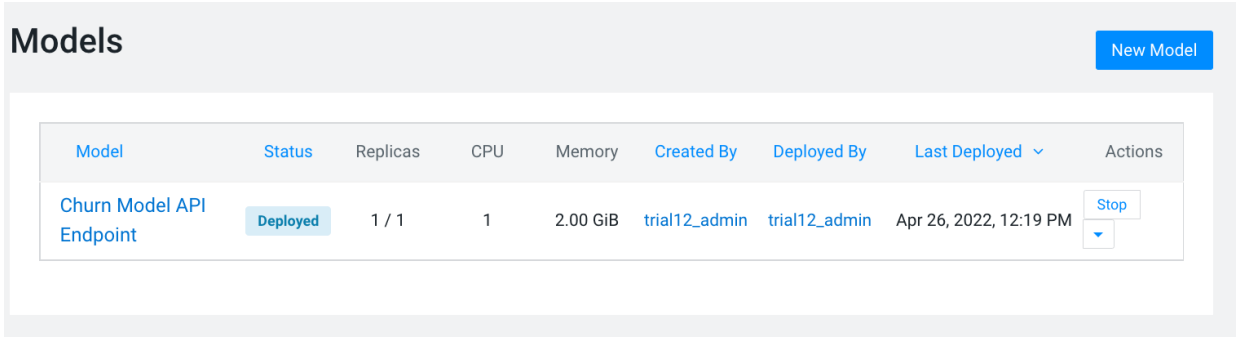
```
# Create the Boto S3 connection object.
from boto.s3.connection import S3Connection
```

Share job output and export PDF's -

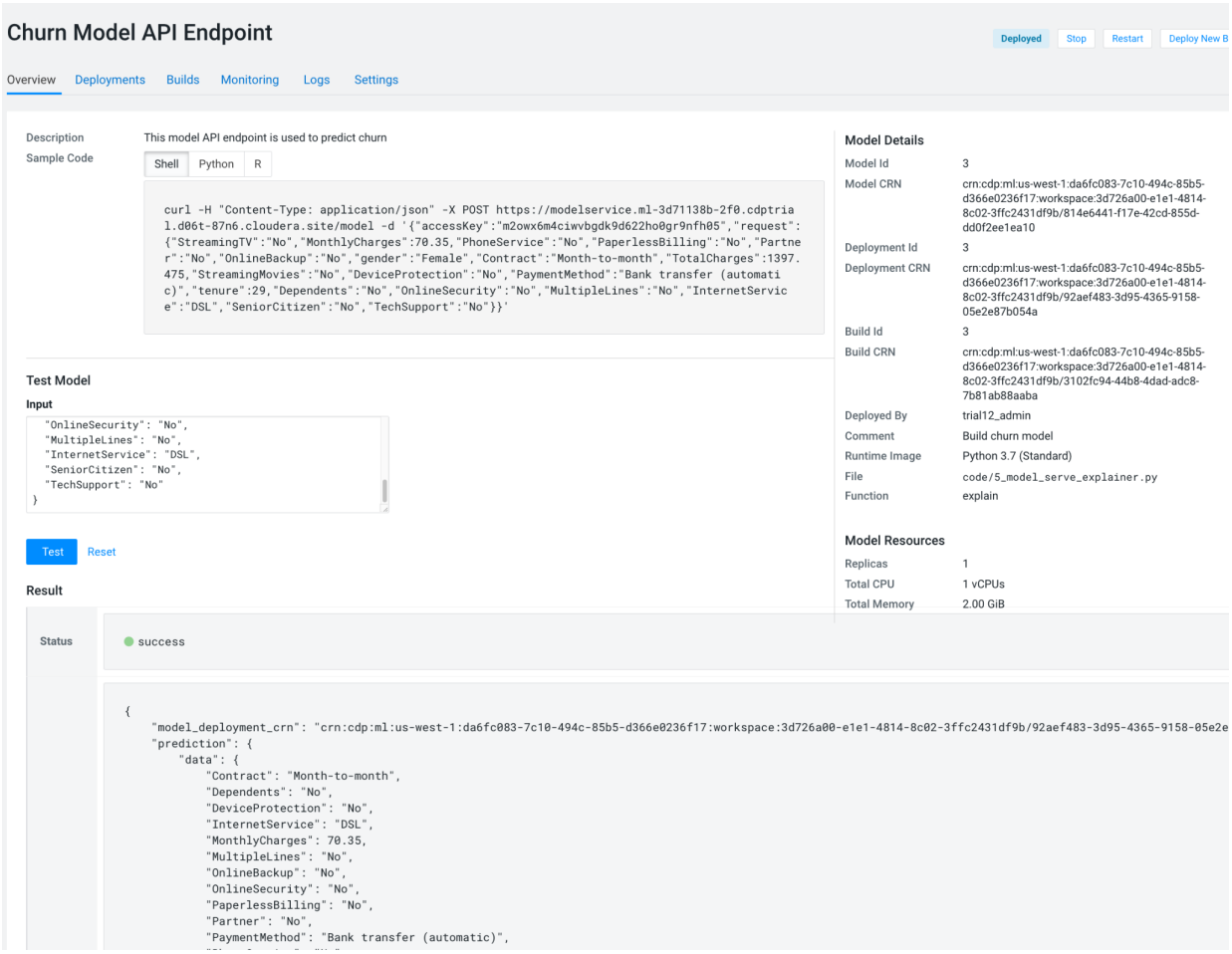


13. Models

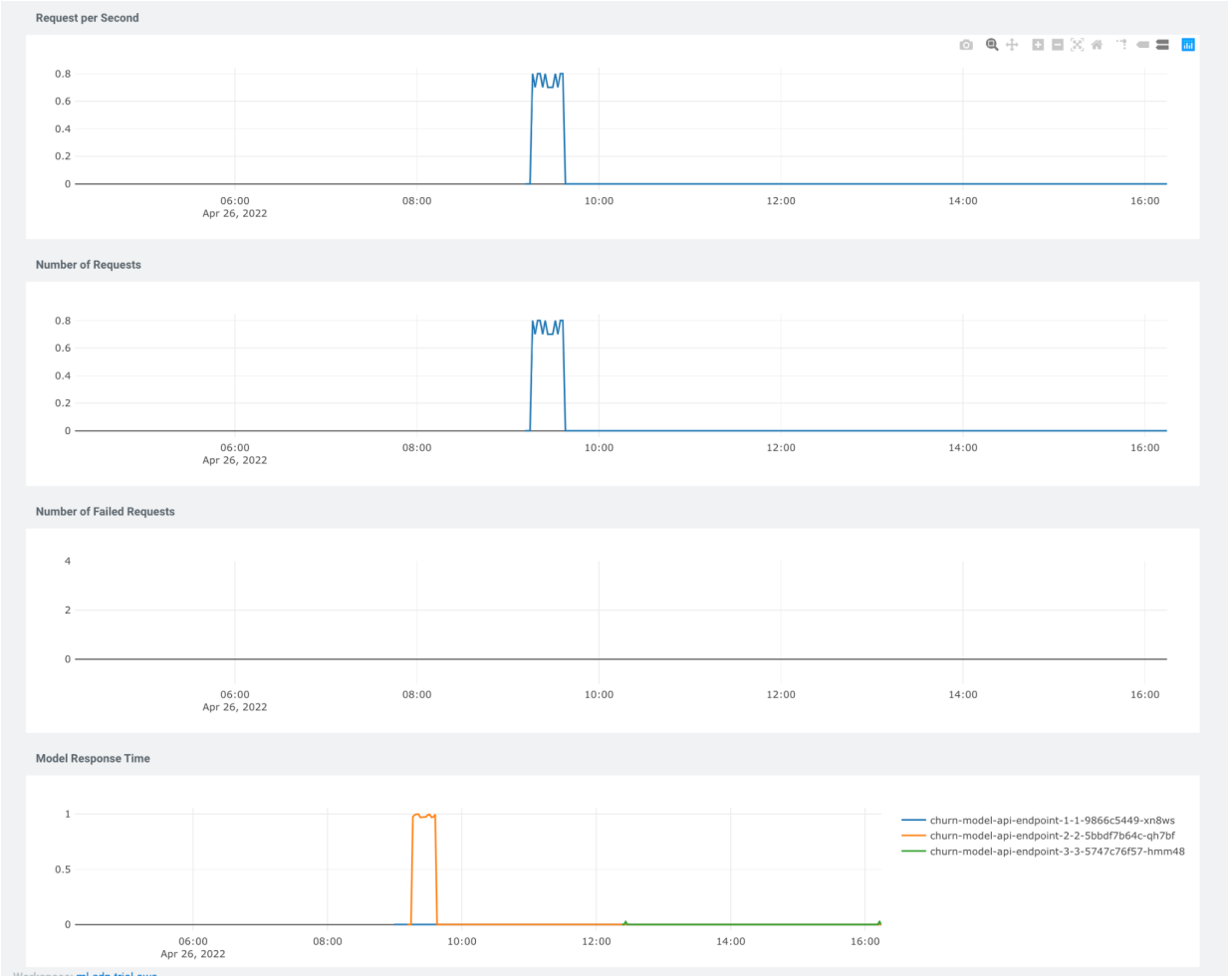
Click on the deployed model in the project and click on Test.



The model executes and returns the results. Models can be used for various use cases to provide an authenticated and access controlled rest api for other applications to send their data to scoring.



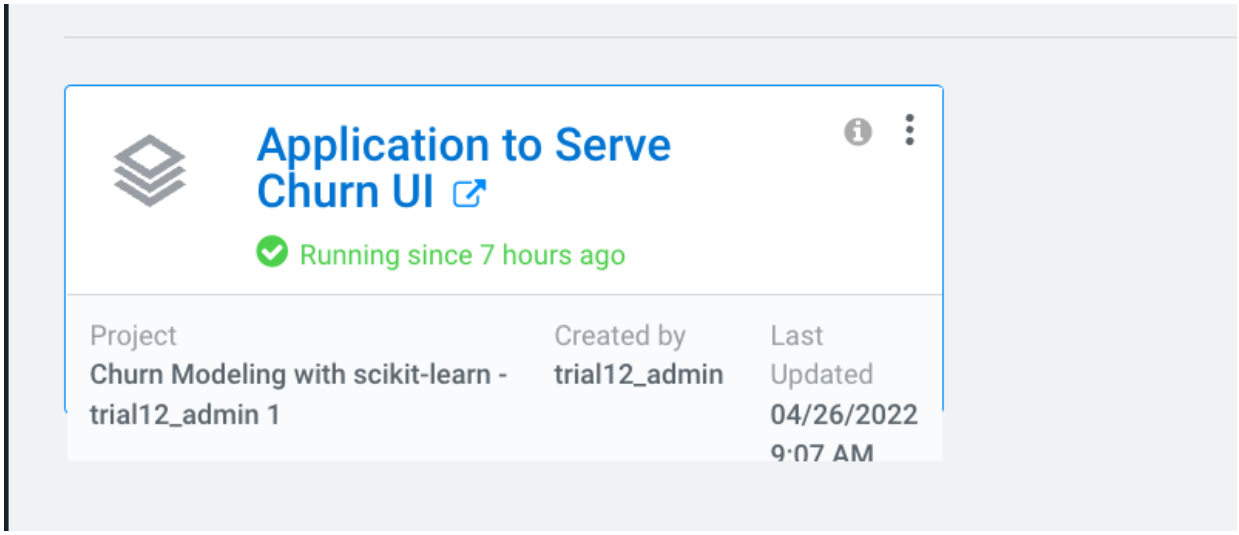
Models can be monitored to provide visibility into how they are responding to incoming requests.

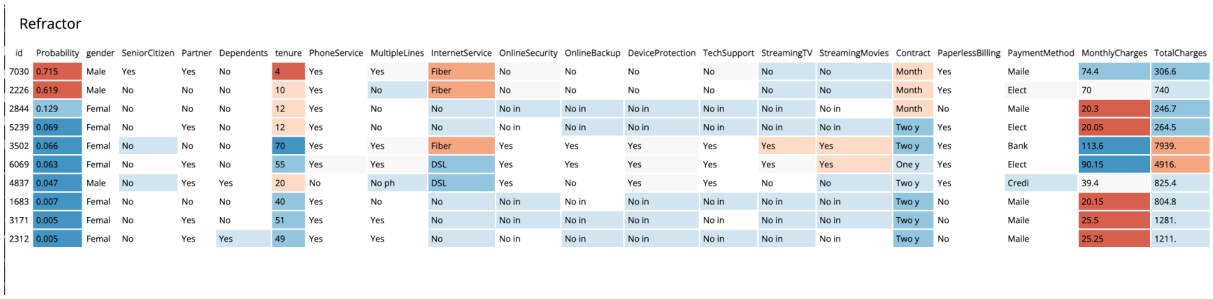


14. Applications

Applications can be built to expose custom UI's to users and consumers of the application. E.g of applications are H2O UI, ML Flow, Cloudera Data Viz, Custom applications in flask, etc etc

Click to open the application UI





id	Probability	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBackup	DeviceProtection	TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges
7030	0.715	Male	Yes	Yes	No	4	Yes	Yes	Fiber	No	No	No	No	No	No	Month	Yes	Maile	74.4	306.6
2226	0.619	Male	No	No	No	10	Yes	No	Fiber	No	No	No	No	No	No	Month	Yes	Elect	70	740
2844	0.129	Femal	No	No	No	12	Yes	No	No	No in	No in	No in	No in	No in	No in	Month	No	Maile	20.3	246.7
5239	0.069	Femal	No	Yes	No	12	Yes	No	No	No in	No in	No in	No in	No in	No in	Two y	Yes	Elect	20.05	264.5
3502	0.066	Femal	No	No	No	70	Yes	Yes	Fiber	Yes	Yes	Yes	Yes	Yes	Yes	Two y	Yes	Bank	113.6	7939.
6069	0.063	Femal	No	Yes	No	55	Yes	Yes	DSL	Yes	Yes	Yes	Yes	Yes	Yes	One y	Yes	Elect	90.15	4916.
4837	0.047	Male	No	Yes	Yes	20	No	No ph	DSL	Yes	No	Yes	Yes	No	No	Two y	Yes	Credi	39.4	825.4
1683	0.007	Femal	No	No	No	40	Yes	No	No	No in	No in	No in	No in	No in	No in	Two y	No	Maile	20.15	804.8
3171	0.005	Femal	No	Yes	No	51	Yes	Yes	No	No in	No in	No in	No in	No in	No in	Two y	No	Maile	25.5	1281.
2312	0.005	Femal	No	Yes	Yes	49	Yes	Yes	No	No in	No in	No in	No in	No in	No in	Two y	No	Maile	25.25	1211.

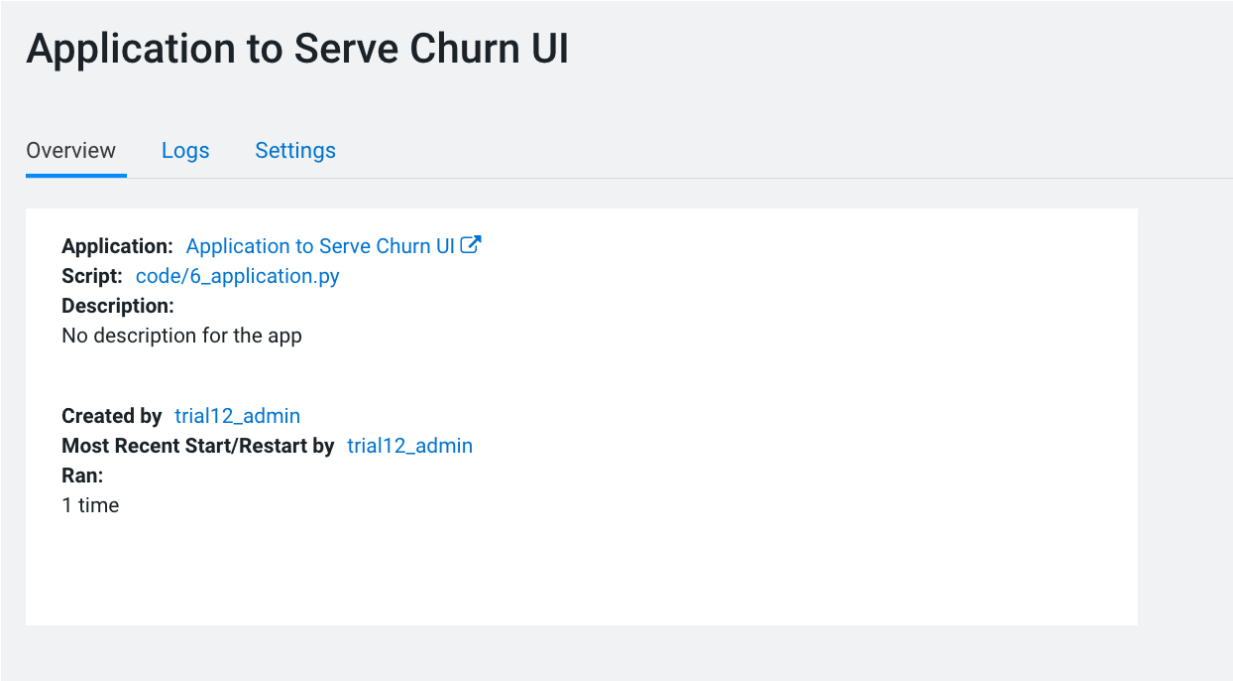
Applications can be interactive to change the result on user provided input an can be exposed to non authenticated users also to consume the application

Single Prediction View

Churn Probability 0.014

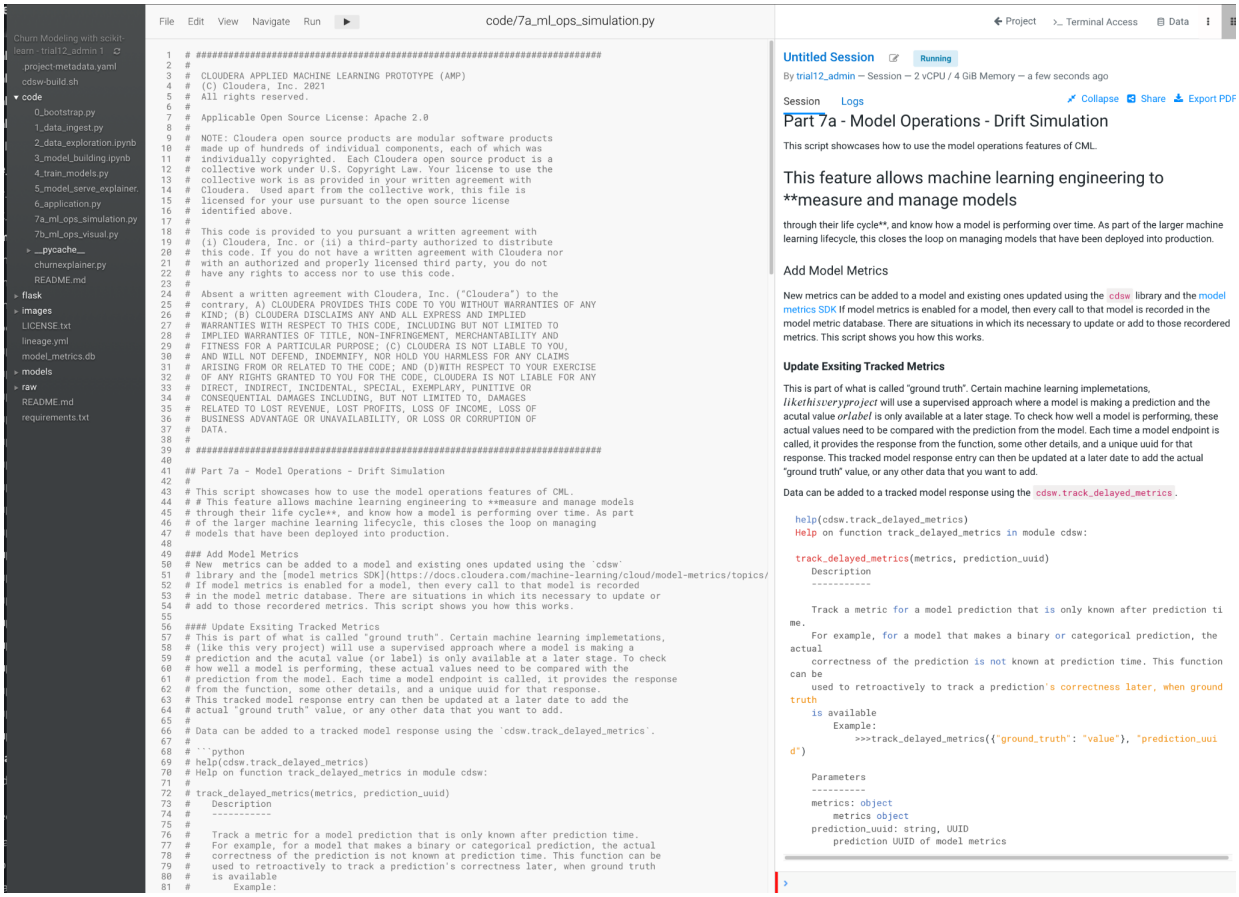
Contract	Month-to-month	0.12	Month-to-month	One year	Two year			
Dependents	Yes	0	No	Yes				
DeviceProtection	No internet service	-0.05	No	No internet service	Yes			
InternetService	No	-0.07	DSL	Fiber optic	No			
MonthlyCharges	25.25	0.31	mean 64.80	min 18.25	max 118.75		Submit	
MultipleLines	No phone service	0	No	No phone service	Yes			
OnlineBackup	No internet service	-0.06	No	No internet service	Yes			
OnlineSecurity	No internet service	-0.05	No	No internet service	Yes			
PaperlessBilling	No	0	No	Yes				
Partner	Yes	0	No	Yes				
PaymentMethod	Mailed check	0	Bank transfer (automatic)	Credit card (automatic)	Electronic check	Mailed check		
PhoneService	Yes	0	No	Yes				
SeniorCitizen	No	0	No	Yes				
StreamingMovies	No internet service	0	No	No internet service	Yes			
StreamingTV	No internet service	-0.05	No	No internet service	Yes			
TechSupport	No internet service	-0.05	No	No internet service	Yes			
TotalCharges	1211.65	-0.08	mean 2283.30	min 18.80	max 8684.80		Submit	
gender	Female	0	Female	Male				
tenure	49	-0.14	mean 32.42	min 1.00	max 72.00		Submit	

Application Details -

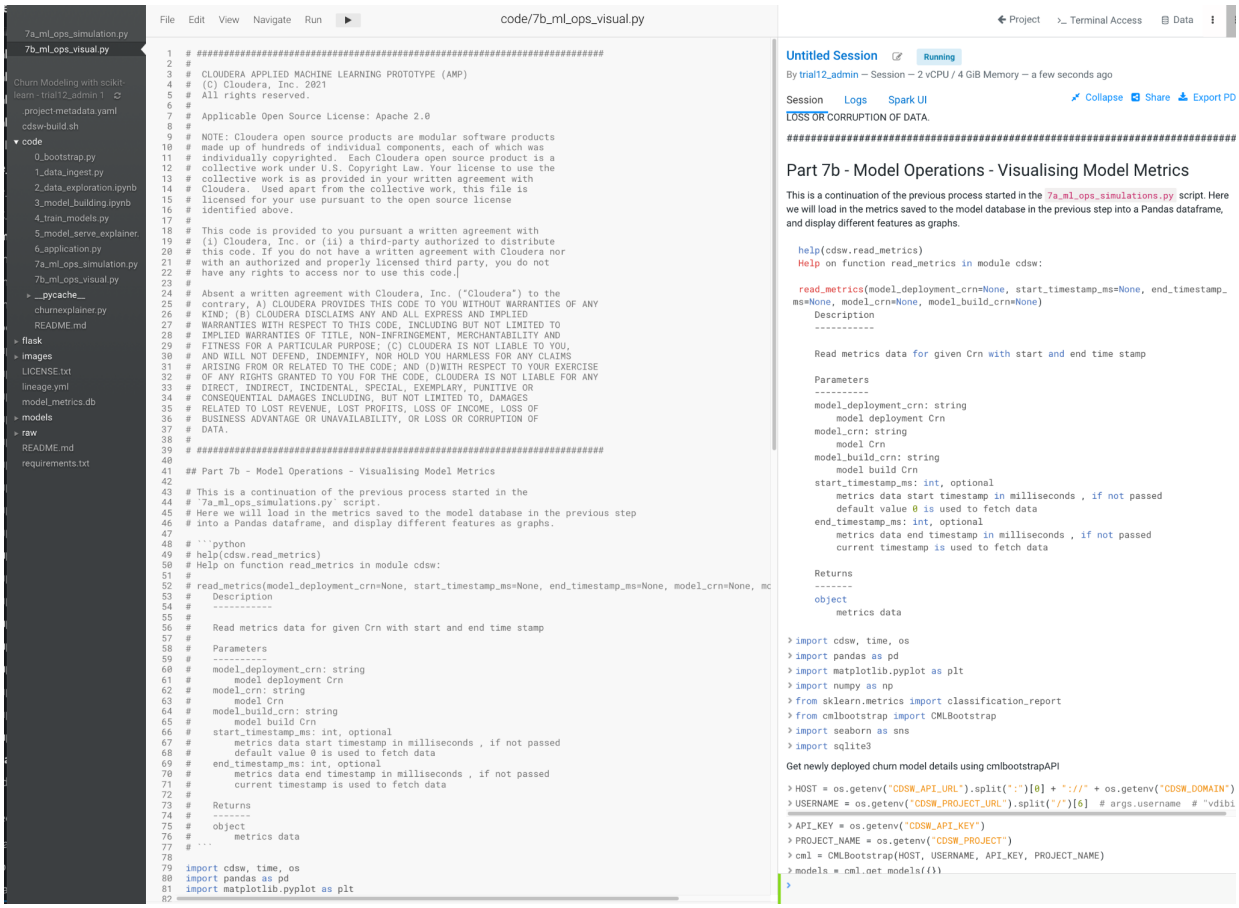


15. ML Operations and Atlas Integration

Let's see how we can test the accuracy of the deployed model. First open the Workbench and run the code/7a_ml_ops_simulation.py. It would take approx 15 minutes to run. I will generate a random data set and invoke the model to generate the scores and store the predicted value (p values) in a database



Once complete, run the code/7b_ml_ops_visual.py. This code will generate the actual values (a values) and compare them against the predicted values (p values) generated in the first step and build comparison graphs

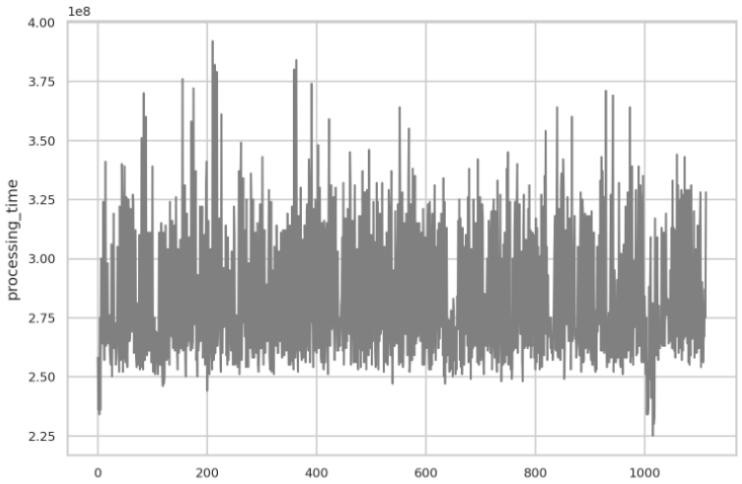


Comparison of model predicted values against the actual values generated -

Plot processing time

```
time_metrics = metrics_df.dropna(subset=["processing_time"]).sort_values(
    "startTimeStampMs"
)
sns.lineplot(
    x=range(len(prob_metrics)), y="processing_time", data=prob_metrics, color="grey"
)
```

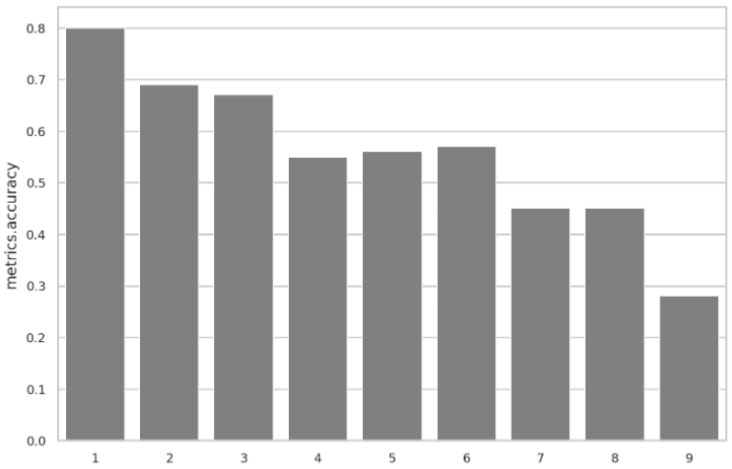
```
<AxesSubplot:ylabel='processing_time'>
```



Plot model accuracy drift over the simulated time period

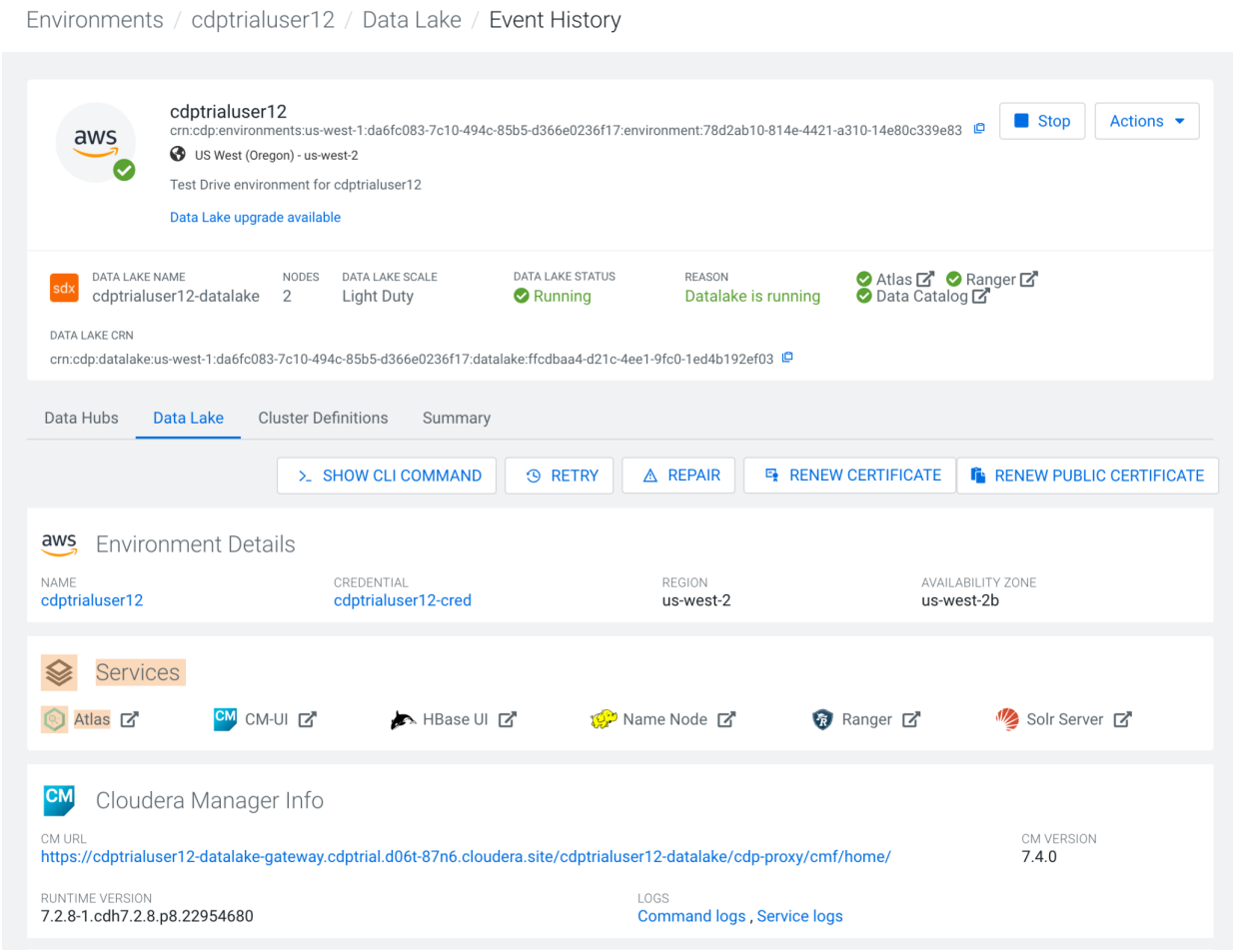
```
agg_metrics = metrics_df.dropna(subset=["metrics.accuracy"]).sort_values(
    "startTimeStampMs"
)
sns.barplot(
    x=list(range(1, len(agg_metrics) + 1)),
    y="metrics.accuracy",
    color="grey",
    data=agg_metrics,
)
```

```
<AxesSubplot:ylabel='metrics.accuracy'>
```

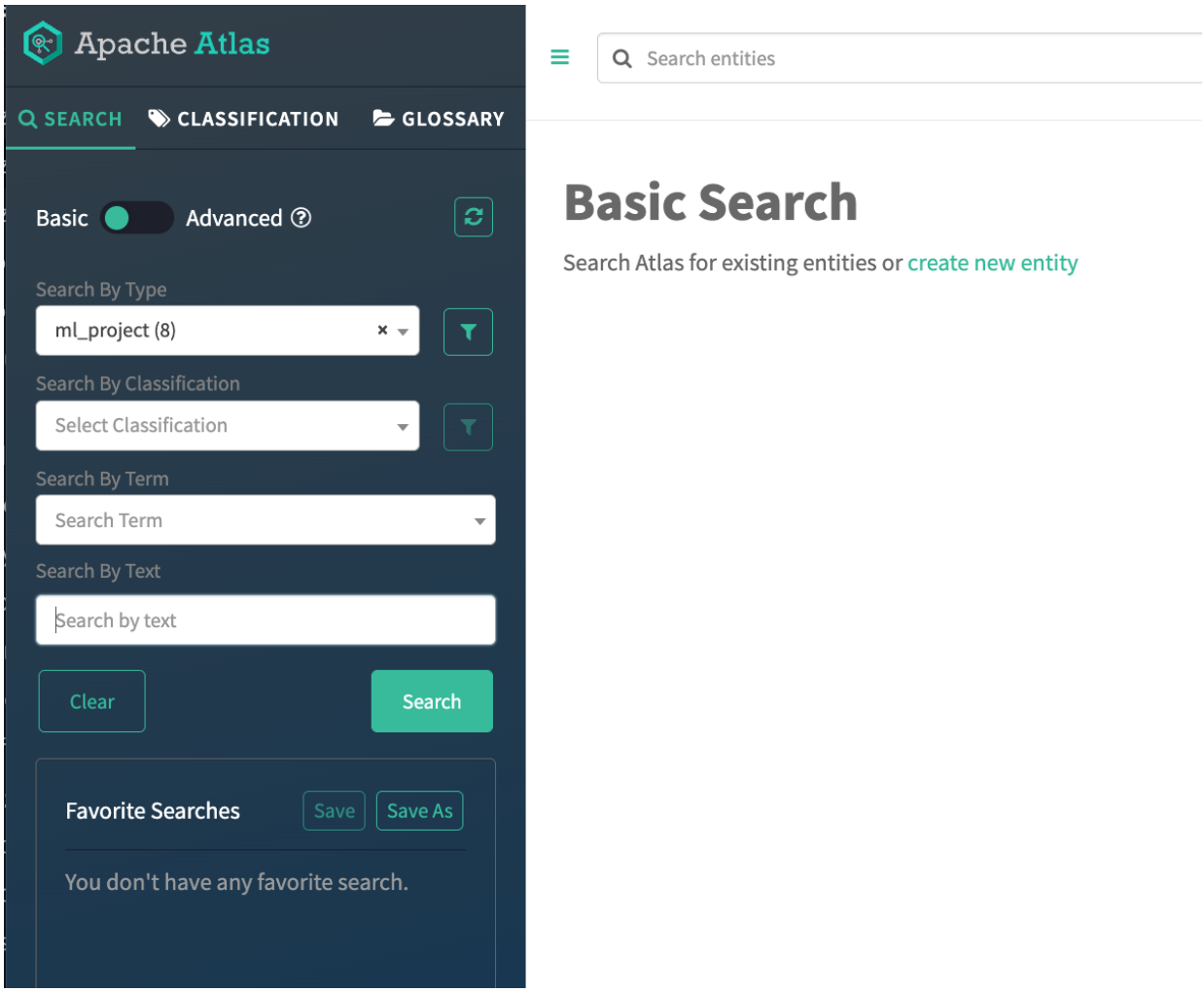


16. Model Lineage and Governance

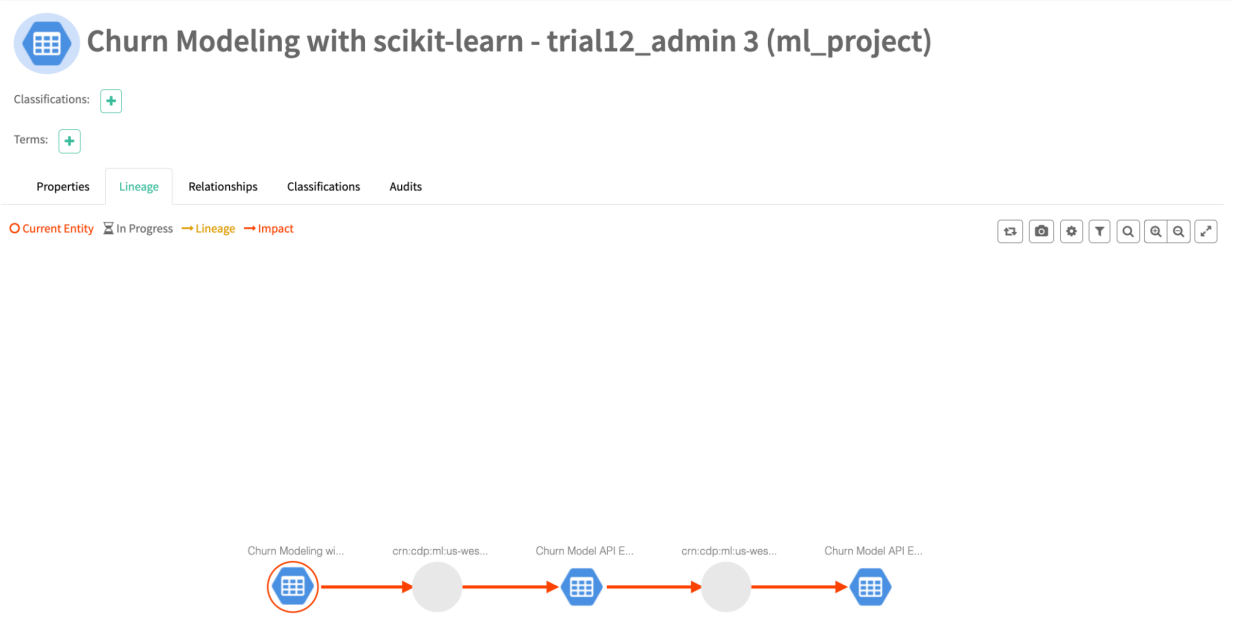
Go the the environment > Data Lake and open Atlas Link -



In atlas, on the left seach for ML Projects -

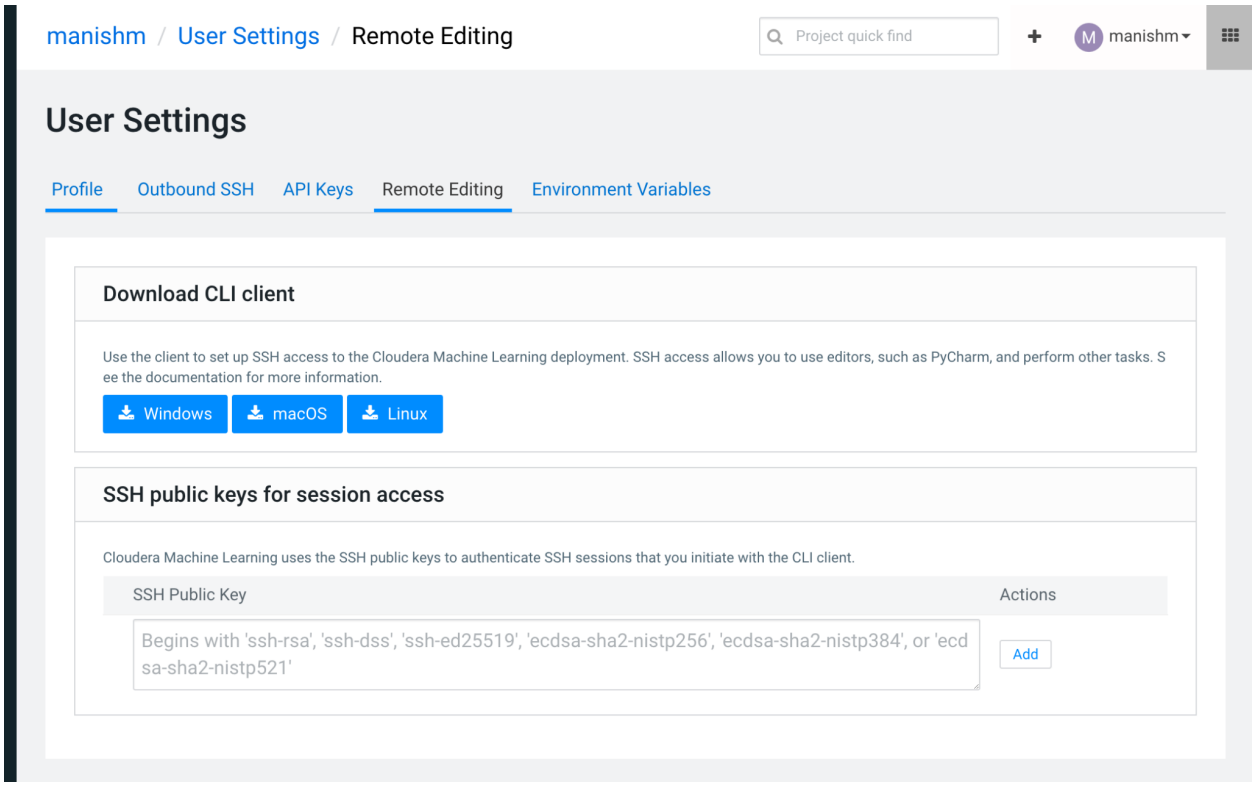


You can see the model, the data sets and the lineage of the model creation in Atlas



17. Remote Code Editing and Terminal + CLI Options

You can download the cli for Window, MacOS and Linux from the CML UI and use it for remote cli based access to submit jobs and run sessions in CML




```
MacBook-Pro :: cdppc/cml » ./cdswctl --help
Usage:
  cdswctl [OPTIONS] <command>

Cloudera Data Science Workbench command-line client 2.0.0.55615
This client is in beta and is subject to change in the future.

Help Options:
  -h, --help  Show this help message

Available commands:
  engine-images   Work with CDSW engine images
  jobruns         Work with CDSW Job Runs
  jobs            Work with CDSW Jobs
  login           Log in
  models          Work with CDSW models
  projects        Work with CDSW projects
  runtime-addons  Work with ML runtime-addons
  runtimes        Work with ML runtimes
  sessions        Work with CDSW sessions
  ssh-endpoint    Forward SSH port to session
  version         Show version
```

18. ML Flow Demo

Make a new project using this github as the base -
https://github.com/myloginid/experiments-with-mlflow

* Project Name

mlflow-demo

Project Description

https://github.com/myloginid/mlflow-demo

Project Visibility

- (•) Private - Only added collaborators can view the project
- ( ) Public - All authenticated users can view this project.

Initial Setup

Blank | Template | AMPs | Local Files | Git

Git URL of Project

https://github.com/myloginid/mlflow-demo

Runtime setup

Basic | Advanced

Basic configuration adds the most commonly used Editors for the Kernel of your choice. To fine-tune the Editors available in the project, choose the Advanced tab.

Kernel

Python 3.7

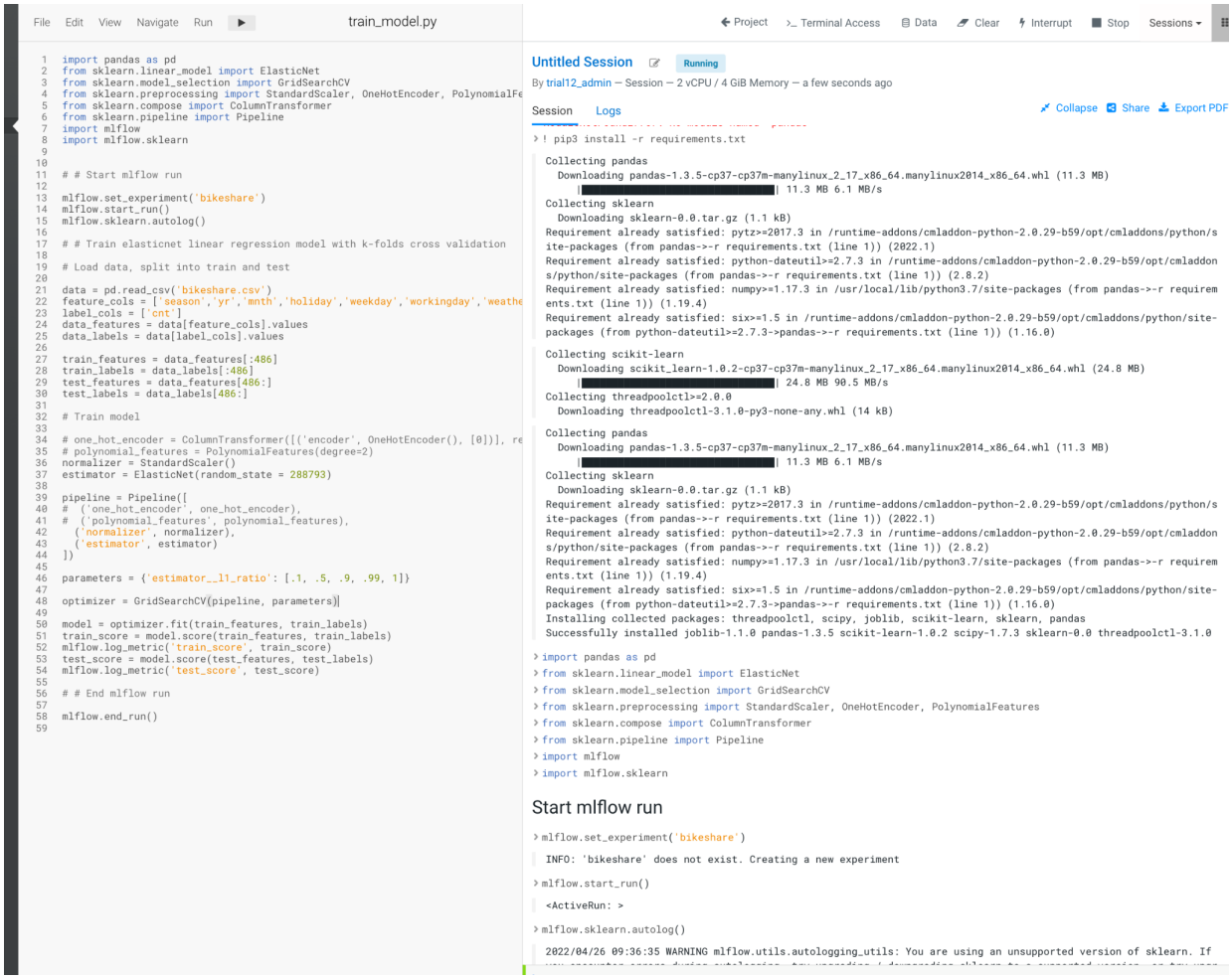
[ ] Add GPU enabled Runtime variant

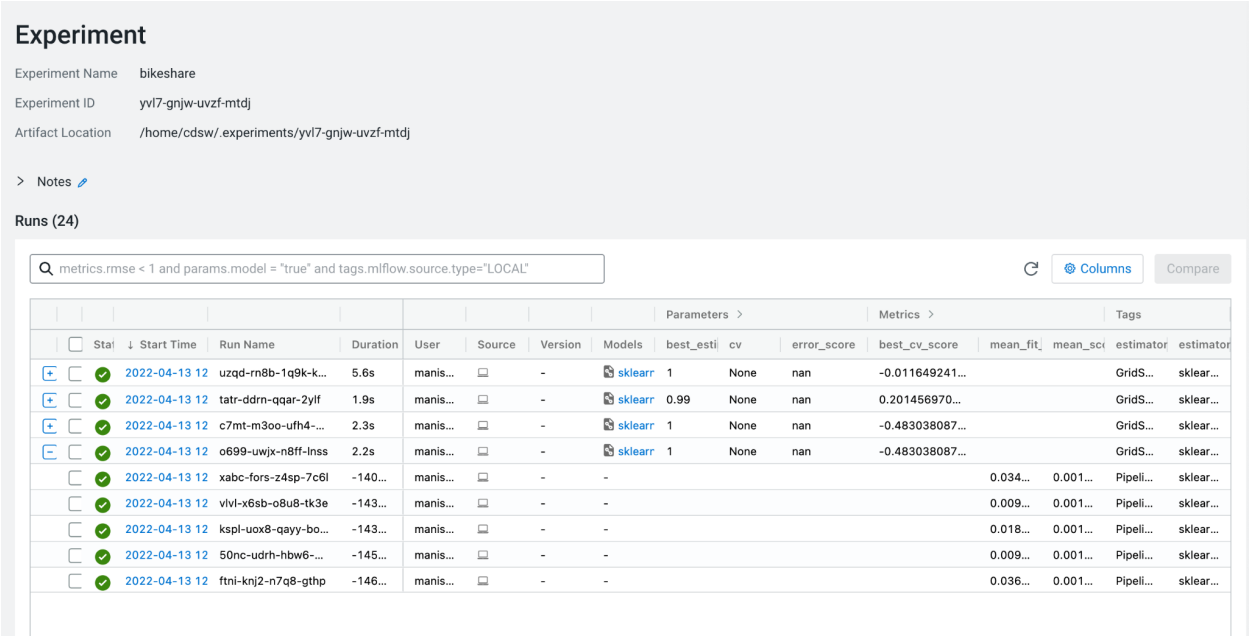
These runtimes will be added to the project:

JupyterLab - Python 3.7 - Standard - 2021.12

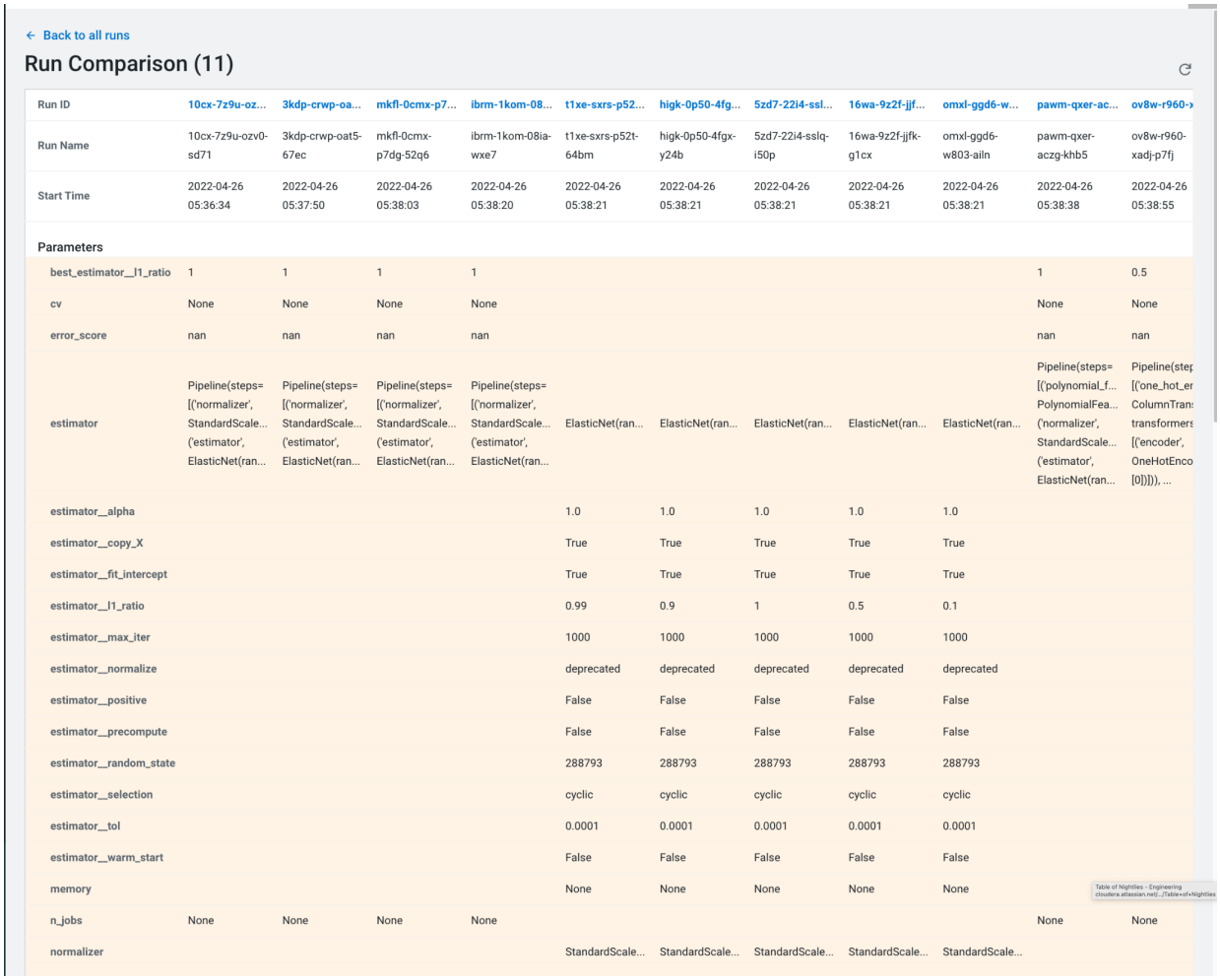
Workbench - Python 3.7 - Standard - 2021.12

Start a Workbench session and run train_model.py





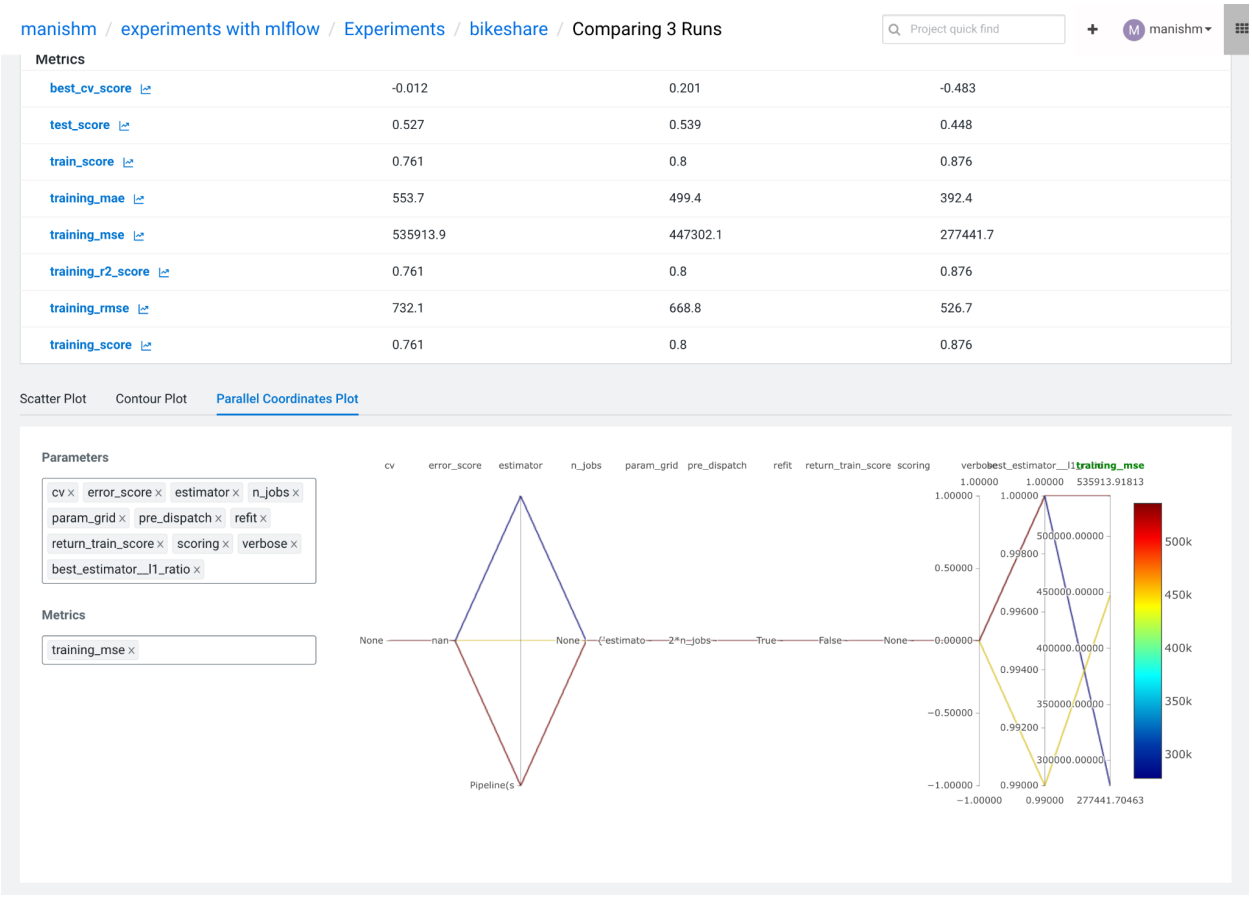
Compare the results -



← Back to all runs

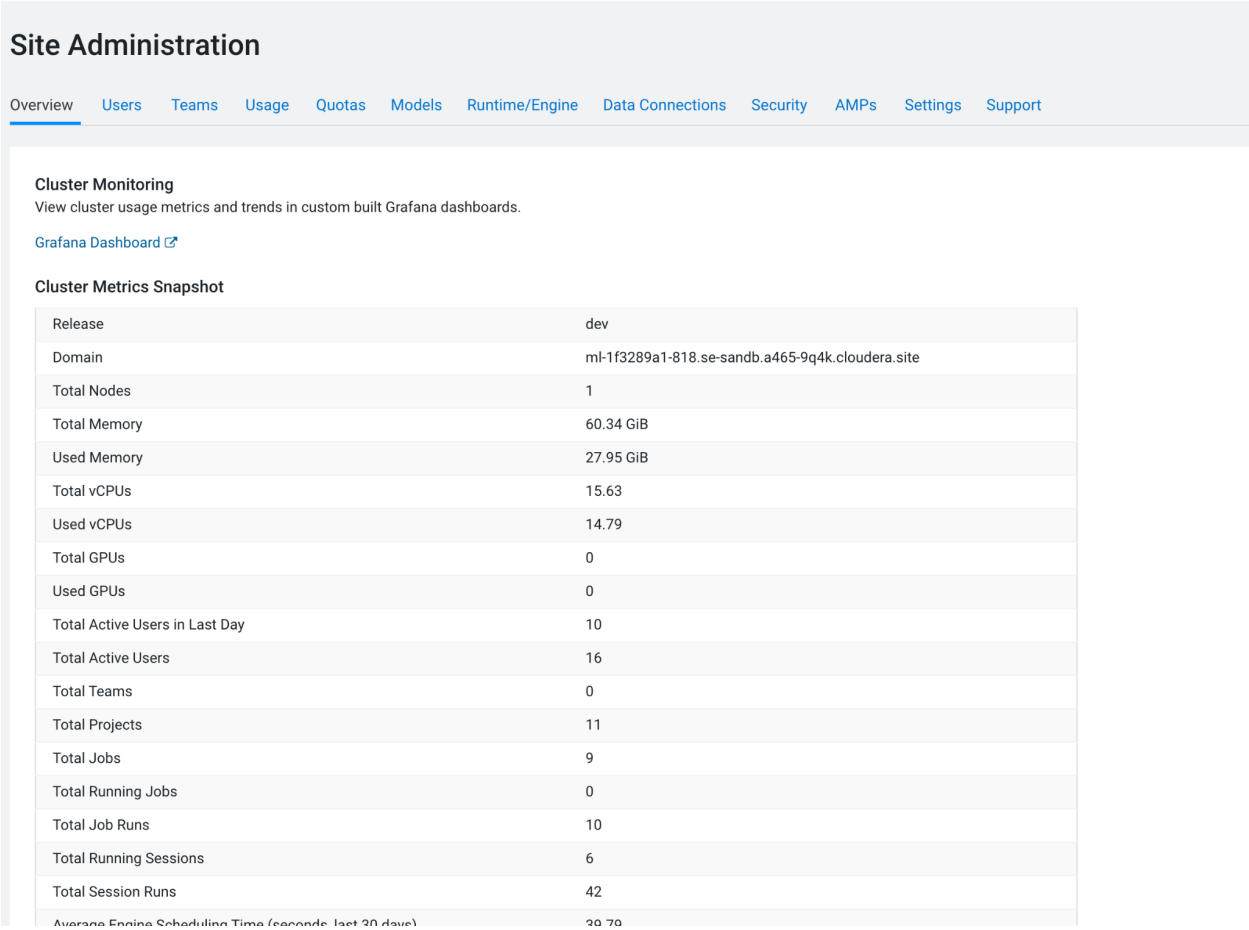
Run Comparison (11)

Run ID	10cx-7z9u-oz...	3kdp-crwp-oa...	mkfl-0cmx-p7...	ibrm-1kom-08...	t1xe-sxrs-p52...	higk-0p50-4fg...	5zd7-22i4-ssl...	16wa-9z2f-jjf...	omxl-ggd6-w...	pawm-qxer-ac...	ov8w-r960-x...
Run Name	10cx-7z9u-ozv0-sd71	3kdp-crwp-oat5-67ec	mkfl-0cmx-p7dg-52q6	ibrm-1kom-08ia-wxe7	t1xe-sxrs-p52t-64bm	higk-0p50-4fgx-y24b	5zd7-22i4-ssIq-i50p	16wa-9z2f-jjfk-g1cx	omxl-ggd6-w803-ailn	pawm-qxer-aczg-khb5	ov8w-r960-xadj-p7fj
Start Time	2022-04-26 05:36:34	2022-04-26 05:37:50	2022-04-26 05:38:03	2022-04-26 05:38:20	2022-04-26 05:38:21	2022-04-26 05:38:21	2022-04-26 05:38:21	2022-04-26 05:38:21	2022-04-26 05:38:21	2022-04-26 05:38:38	2022-04-26 05:38:55
Parameters											
best_estimator__l1_ratio	1	1	1	1						1	0.5
cv	None	None	None	None						None	None
error_score	nan	nan	nan	nan						nan	nan
estimator	Pipeline(steps=[('normalizer', StandardScale... ('estimator', ElasticNet(ran...	Pipeline(steps=[('normalizer', StandardScale... ('estimator', ElasticNet(ran...	Pipeline(steps=[('normalizer', StandardScale... ('estimator', ElasticNet(ran...	Pipeline(steps=[('normalizer', StandardScale... ('estimator', ElasticNet(ran...	ElasticNet(ran...	ElasticNet(ran...	ElasticNet(ran...	ElasticNet(ran...	ElasticNet(ran...	Pipeline(steps=[('polynomial_f... PolynomialFea... ('normalizer', StandardScale... ('estimator', ElasticNet(ran...	Pipeline(step [('one_hot_er ColumnTran transformers [('encoder', OneHotEnco [0])])), ...
estimator__alpha					1.0	1.0	1.0	1.0	1.0		
estimator__copy_X					True	True	True	True	True		
estimator__fit_intercept					True	True	True	True	True		
estimator__l1_ratio					0.99	0.9	1	0.5	0.1		
estimator__max_iter					1000	1000	1000	1000	1000		
estimator__normalize					deprecated	deprecated	deprecated	deprecated	deprecated		
estimator__positive					False	False	False	False	False		
estimator__precompute					False	False	False	False	False		
estimator__random_state					288793	288793	288793	288793	288793		
estimator__selection					cyclic	cyclic	cyclic	cyclic	cyclic		
estimator__tol					0.0001	0.0001	0.0001	0.0001	0.0001		
estimator__warm_start					False	False	False	False	False		
memory					None	None	None	None	None		
n_jobs	None	None	None	None						None	None
normalizer					StandardScale...	StandardScale...	StandardScale...	StandardScale...	StandardScale...		

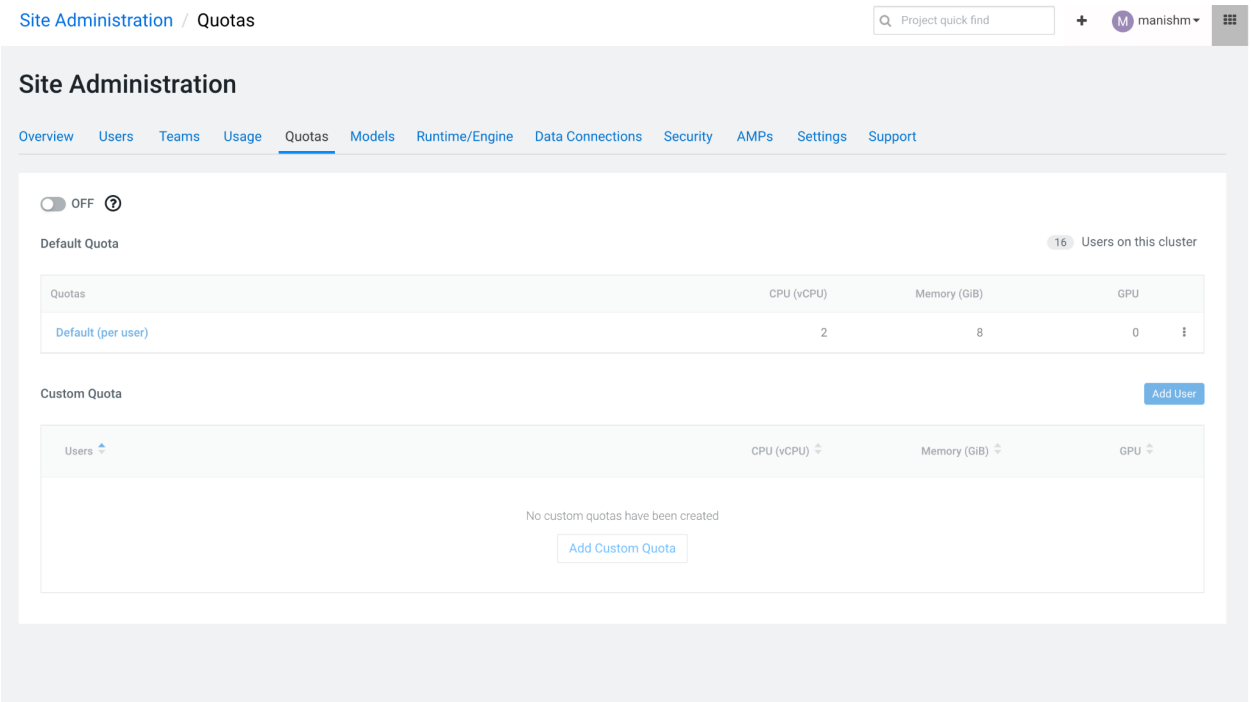


19. CDSW Administration

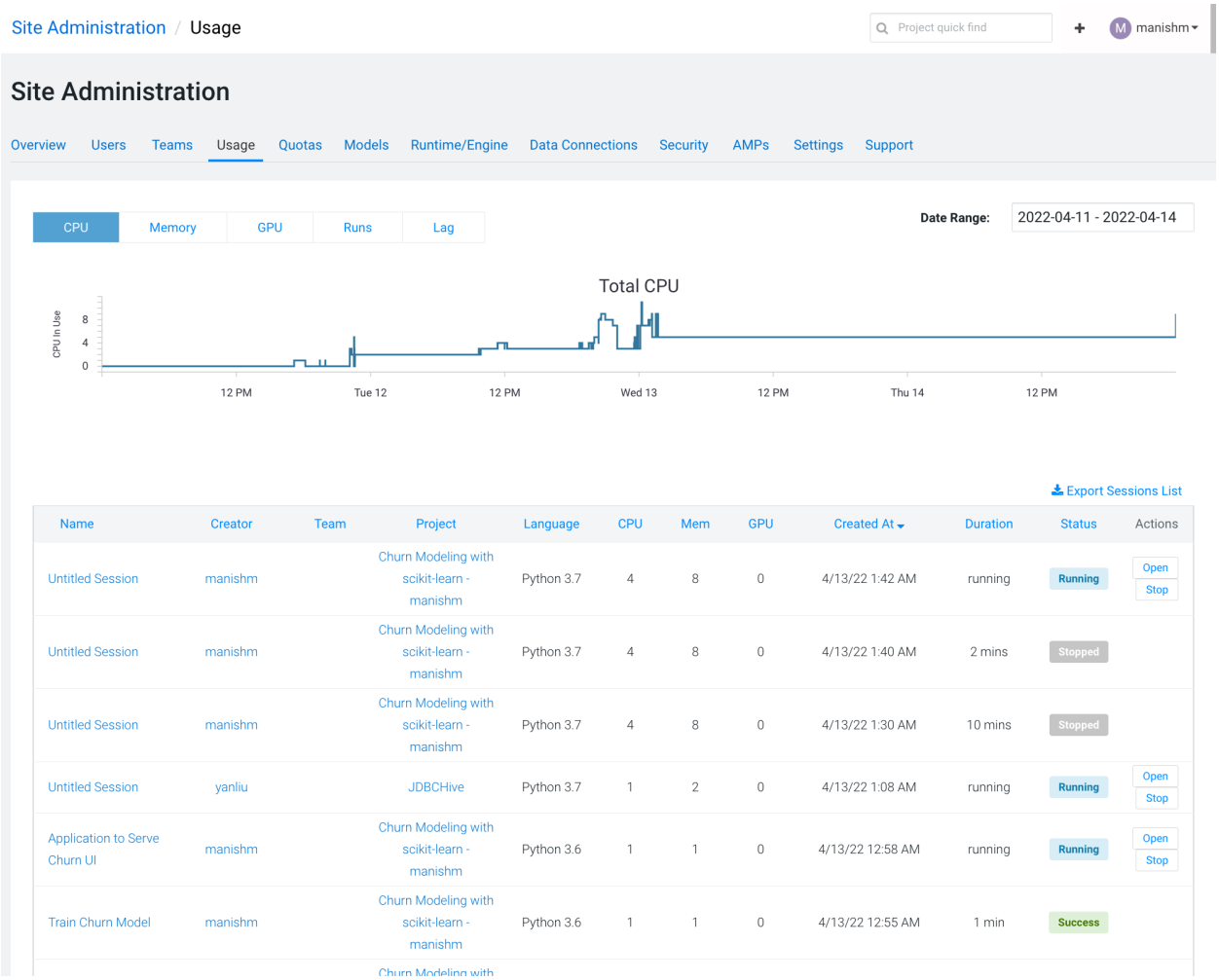
a. Usage Overview


Site Administration
Overview
Users
Teams
Usage
Quotas
Models
Runtime/Engine
Data Connections
Security
AMPs
Settings
Support
Cluster Monitoring
View cluster usage metrics and trends in custom built Grafana dashboards.
Grafana Dashboard
Cluster Metrics Snapshot
Release dev
Domain ml-1f3289a1-818.se-sandb.a465-9q4k.cloudera.site
Total Nodes 1
Total Memory 60.34 GiB
Used Memory 27.95 GiB
Total vCPUs 15.63
Used vCPUs 14.79
Total GPUs 0
Used GPUs 0
Total Active Users in Last Day 10
Total Active Users 16
Total Teams 0
Total Projects 11
Total Jobs 9
Total Running Jobs 0
Total Job Runs 10
Total Running Sessions 6
Total Session Runs 42

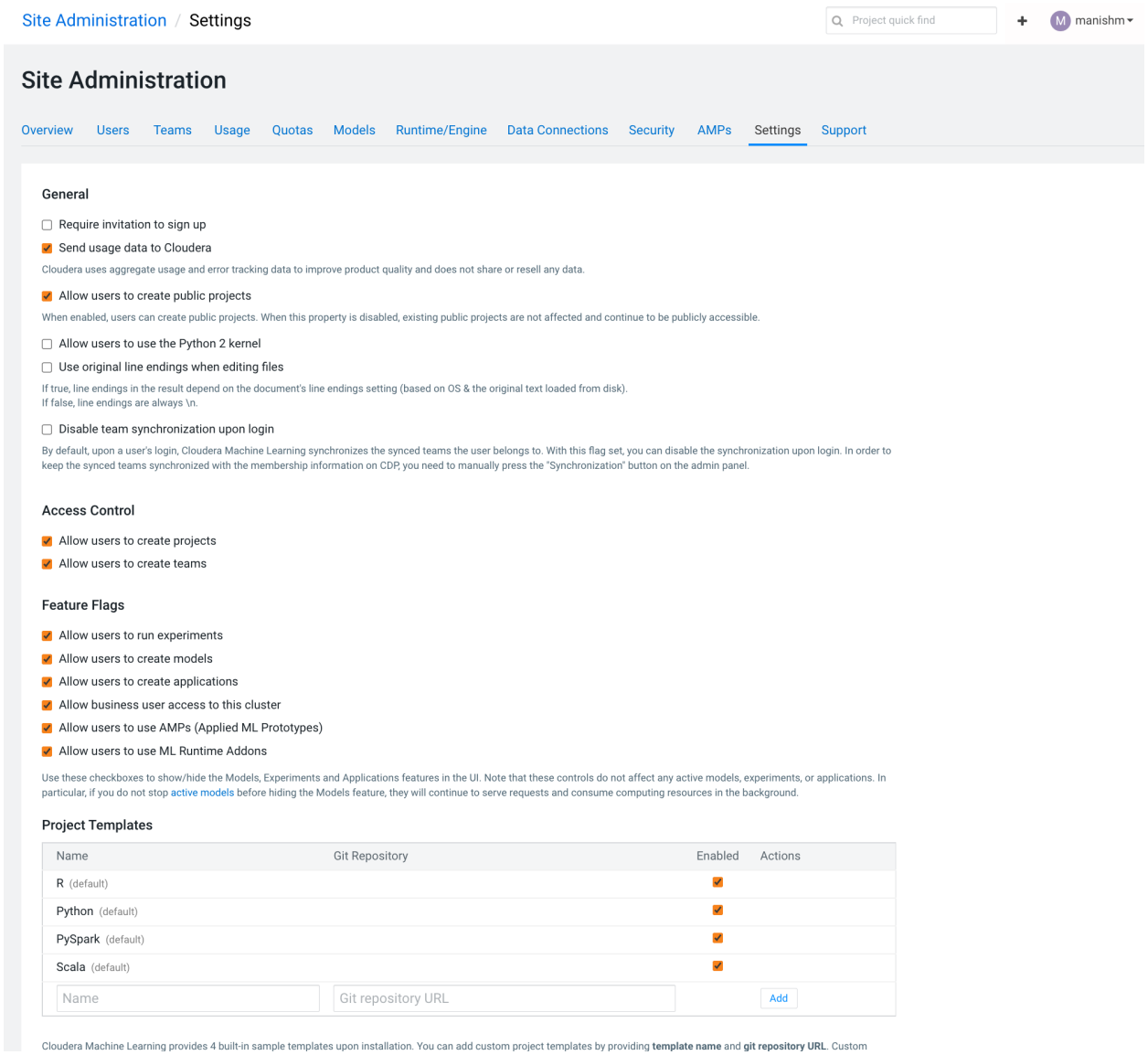
b. Quotas



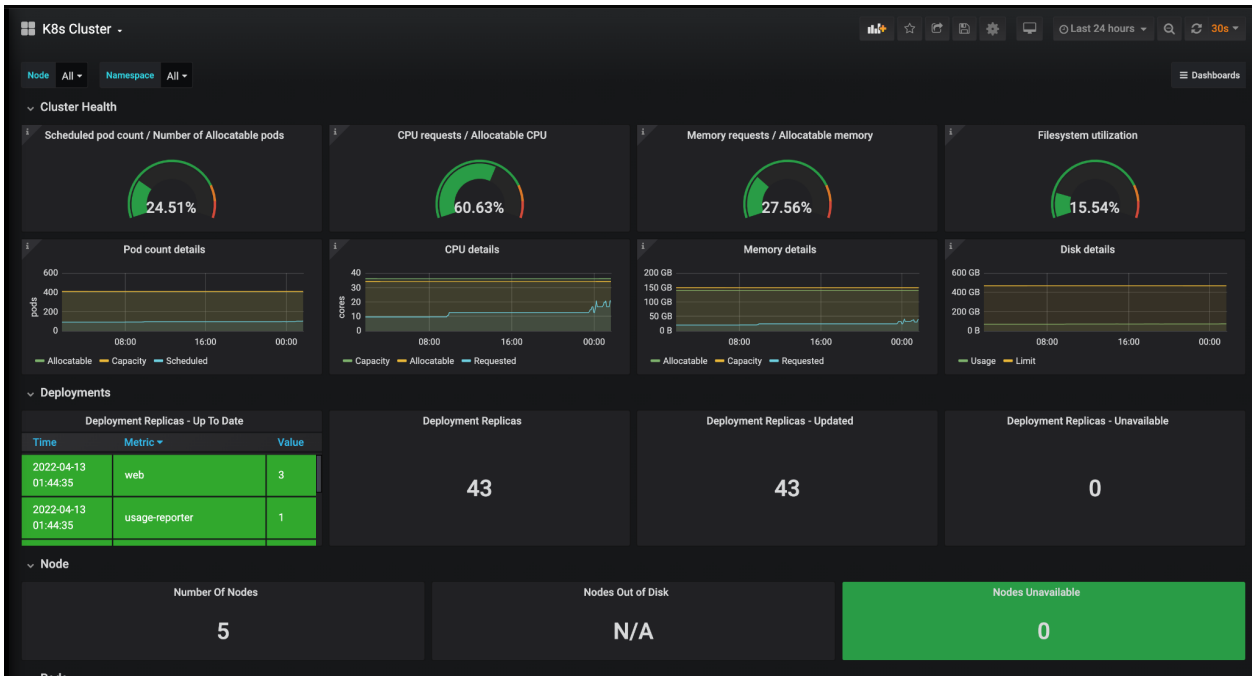
c. Detailed Usage


Site Administration / Usage
Project quick find
manishm
Site Administration
Overview Users Teams Usage Quotas Models Runtime/Engine Data Connections Security AMPs Settings Support
CPU Memory GPU Runs Lag
Date Range: 2022-04-11 - 2022-04-14
Total CPU
CPU In Use
8
4
0
12 PM
Tue 12
12 PM
Wed 13
12 PM
Thu 14
12 PM
Export Sessions List
Name Creator Team Project Language CPU Mem GPU Created At Duration Status Actions
Untitled Session manishm Churn Modeling with scikit-learn - manishm Python 3.7 4 8 0 4/13/22 1:42 AM running Running Open Stop
Untitled Session manishm Churn Modeling with scikit-learn - manishm Python 3.7 4 8 0 4/13/22 1:40 AM 2 mins Stopped
Untitled Session manishm Churn Modeling with scikit-learn - manishm Python 3.7 4 8 0 4/13/22 1:30 AM 10 mins Stopped
Untitled Session yanliu JDBCHive Python 3.7 1 2 0 4/13/22 1:08 AM running Running Open Stop
Application to Serve Churn UI manishm Churn Modeling with scikit-learn - manishm Python 3.6 1 1 0 4/13/22 12:58 AM running Running Open Stop
Train Churn Model manishm Churn Modeling with scikit-learn - manishm Python 3.6 1 1 0 4/13/22 12:55 AM 1 min Success

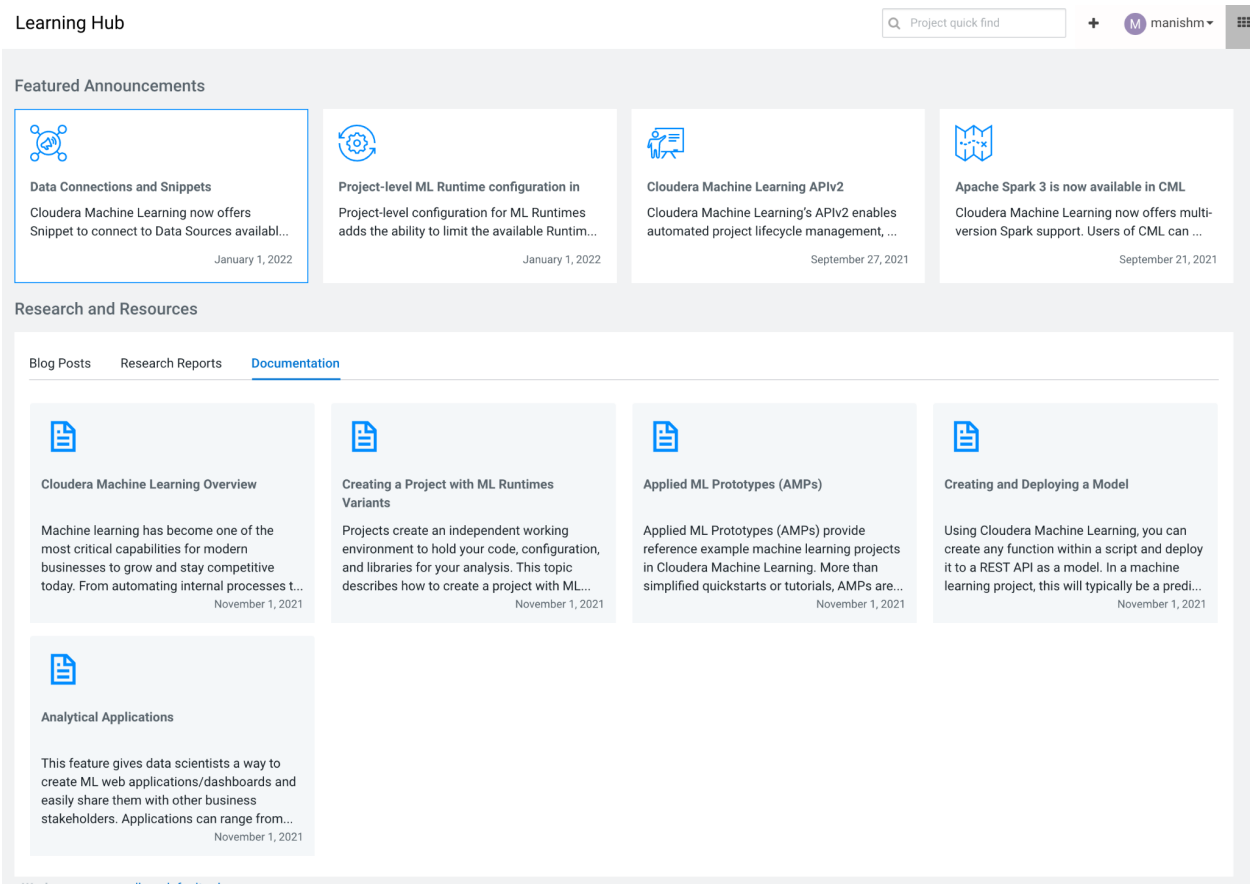
d. Settings



e. Grafana Dashboard



20. Learning Hub


Learning Hub
Project quick find
manishm
Featured Announcements
Data Connections and Snippets
Cloudera Machine Learning now offers Snippet to connect to Data Sources availabl...
January 1, 2022
Project-level ML Runtime configuration in
Project-level configuration for ML Runtimes adds the ability to limit the available Runtim...
January 1, 2022
Cloudera Machine Learning APIv2
Cloudera Machine Learning's APIv2 enables automated project lifecycle management, ...
September 27, 2021
Apache Spark 3 is now available in CML
Cloudera Machine Learning now offers multi-version Spark support. Users of CML can ...
September 21, 2021
Research and Resources
Blog Posts
Research Reports
Documentation
Cloudera Machine Learning Overview
Machine learning has become one of the most critical capabilities for modern businesses to grow and stay competitive today. From automating internal processes t...
November 1, 2021
Creating a Project with ML Runtimes Variants
Projects create an independent working environment to hold your code, configuration, and libraries for your analysis. This topic describes how to create a project with ML...
November 1, 2021
Applied ML Prototypes (AMPs)
Applied ML Prototypes (AMPs) provide reference example machine learning projects in Cloudera Machine Learning. More than simplified quickstarts or tutorials, AMPs are...
November 1, 2021
Creating and Deploying a Model
Using Cloudera Machine Learning, you can create any function within a script and deploy it to a REST API as a model. In a machine learning project, this will typically be a predi...
November 1, 2021
Analytical Applications
This feature gives data scientists a way to create ML web applications/dashboards and easily share them with other business stakeholders. Applications can range from...
November 1, 2021