

Model Experimentation

ML Flow UI

All the experiments (Lead_scoring_model_experimentation)

The screenshot shows the MLflow UI interface. At the top, there's a header bar with the MLflow logo, version 1.26.1, and navigation links for Experiments and Models. On the right side of the header are GitHub and Docs links. Below the header, the main content area is titled "Experiments" and shows a specific experiment named "Lead_scoring_model_experimentation". A search bar is present at the top of the experiment view. The main content area displays a table of 10 matching runs. The columns in the table are: Start Time, Duration, Run Name, User, Source, Version, Model, AUC, Accuracy, and F1. The table lists various machine learning models and their performance metrics.

	Start Time	Duration	Run Name	User	Source	Version	Model	AUC	Accuracy	F1
	4 minutes ago		Session Initialized b00c	root	ipykernel_launcher.py	-	sklearn	0.821	0.738	0.762
	28 seconds ago		Light Gradient Boosting Ma...	root	ipykernel_launcher.py	-	sklearn	0.738	0.679	0.728
	1 minute ago		Naive Bayes	root	ipykernel_launcher.py	-	sklearn	0	0.715	0.742
	1 minute ago		Ridge Classifier	root	ipykernel_launcher.py	-	sklearn	0.79	0.715	0.742
	1 minute ago		Linear Discriminant Analysis	root	ipykernel_launcher.py	-	sklearn	0.792	0.717	0.741
	1 minute ago		Logistic Regression	root	ipykernel_launcher.py	-	sklearn	0.817	0.736	0.758
	1 minute ago		Decision Tree Classifier	root	ipykernel_launcher.py	-	sklearn	0.818	0.737	0.758
	1 minute ago		Extra Trees Classifier	root	ipykernel_launcher.py	-	sklearn	0.819	0.737	0.76
	1 minute ago		Random Forest Classifier	root	ipykernel_launcher.py	-	sklearn	0.821	0.738	0.762
	1 minute ago		Light Gradient Boosting Ma...	root	ipykernel_launcher.py	-	sklearn	0.821	0.738	0.762

Light Gradient Boosting Machine Details

The screenshot shows the MLflow UI interface, specifically the details for a run within the "Lead_scoring_model_experimentation" experiment. The top navigation bar includes the MLflow logo, version 1.26.1, and links for Experiments and Models. The main content area is titled "Light Gradient Boosting Machine". It displays basic run metadata: Date (2023-03-22 15:09:12), Status (UNFINISHED), and Source (ipykernel_launcher.py). The Lifecycle Stage is listed as active, and the Parent Run ID is 9c2cbe341d4c47c6bf62b9e42e657b45. Below this, there are sections for "Description" (which is currently empty) and "Parameters". The "Parameters" section shows 20 parameters with their corresponding values, such as boosting_type: gbdt, class_weight: None, colsample_bytree: 1.0, importance_type: split, learning_rate: 0.1, max_depth: -1, min_child_samples: 20, min_child_weight: 0.001, min_split_gain: 0.0, and n_estimators: 100.

Name	Value
boosting_type	gbdt
class_weight	None
colsample_bytree	1.0
importance_type	split
learning_rate	0.1
max_depth	-1
min_child_samples	20
min_child_weight	0.001
min_split_gain	0.0
n_estimators	100

n_estimators	100
n_jobs	-1
num_leaves	31
objective	None
random_state	42
reg_alpha	0.0
reg_lambda	0.0
silent	warn
subsample	1.0
subsample_for_bin	200000
subsample_freq	0

▼ Metrics (8)

Name	Value
AUC ↗	0.821
Accuracy ↗	0.738
F1 ↗	0.762
Kappa ↗	0.476
MCC ↗	0.485
Accuracy ↗	0.738
F1 ↗	0.762
Kappa ↗	0.476
MCC ↗	0.485
Prec. ↗	0.702
Recall ↗	0.833
TT ↗	4.3

▼ Tags (5)

Name	Value	Actions
Run ID	834aab36f6ee483ab1603e 4924f02cf3	🔗 🗑
Run Time	22.43	🔗 🗑
Source	create_model	🔗 🗑
URI	ea4abcc3	🔗 🗑
USI	b00c	🔗 🗑
Name	Value	Add

▼ Artifacts

ML Flow UI after dropping features

All the experiments (Lead_scoring_model_experimentation_optimize)

mlflow 1.26.1 Experiments Models GitHub Docs

Experiments + × Lead_scoring_model_experimentation_optimize

Search Experiments

Default Lead_scoring_model_...

Experiment ID: 1

Description Edit

Refresh Compare Delete Download CSV Start Time All time

Columns Filter Clear

Showing 11 matching runs

Start Time	Duration	Run Name	User	Source	Version	Models	AUC	Accuracy	F1
17 minutes ago		Session initialized f9c4	root	ipykernel_launcher.py	-	sklearn	0.821	0.739	C
23 seconds ago		Light Gradient Boosting Machine	root	ipykernel_launcher.py	-	sklearn	0.821	0.739	C
10 minutes ago		Light Gradient Boosting Machine	root	ipykernel_launcher.py	-	sklearn	0.734	0.673	C
11 minutes ago		Naive Bayes	root	ipykernel_launcher.py	-	sklearn	0.773	0.7	C
11 minutes ago		Ridge Classifier	root	ipykernel_launcher.py	-	sklearn	0	0.7	C
11 minutes ago		Logistic Regression	root	ipykernel_launcher.py	-	sklearn	0.784	0.71	C
11 minutes ago		Decision Tree Classifier	root	ipykernel_launcher.py	-	sklearn	0.817	0.736	C
11 minutes ago		Extra Trees Classifier	root	ipykernel_launcher.py	-	sklearn	0.818	0.736	C
11 minutes ago		Random Forest Classifier	root	ipykernel_launcher.py	-	sklearn	0.819	0.737	C
11 minutes ago		Light Gradient Boosting Machine	root	ipykernel_launcher.py	-	sklearn	0.821	0.739	C

Load more

Light Gradient boosting machine (optimized)

mlflow 1.26.1 Experiments Models GitHub Docs

Lead_scoring_model_experimentation_optimize > Light Gradient Boosting Machine

Light Gradient Boosting Machine

Date: 2023-03-22 16:15:58 Source: ipykernel_launcher.py User: root

Status: UNFINISHED Lifecycle Stage: active Parent Run: 4f807561e817479ebd6867328e78395c

Description Edit

None

Parameters (20)

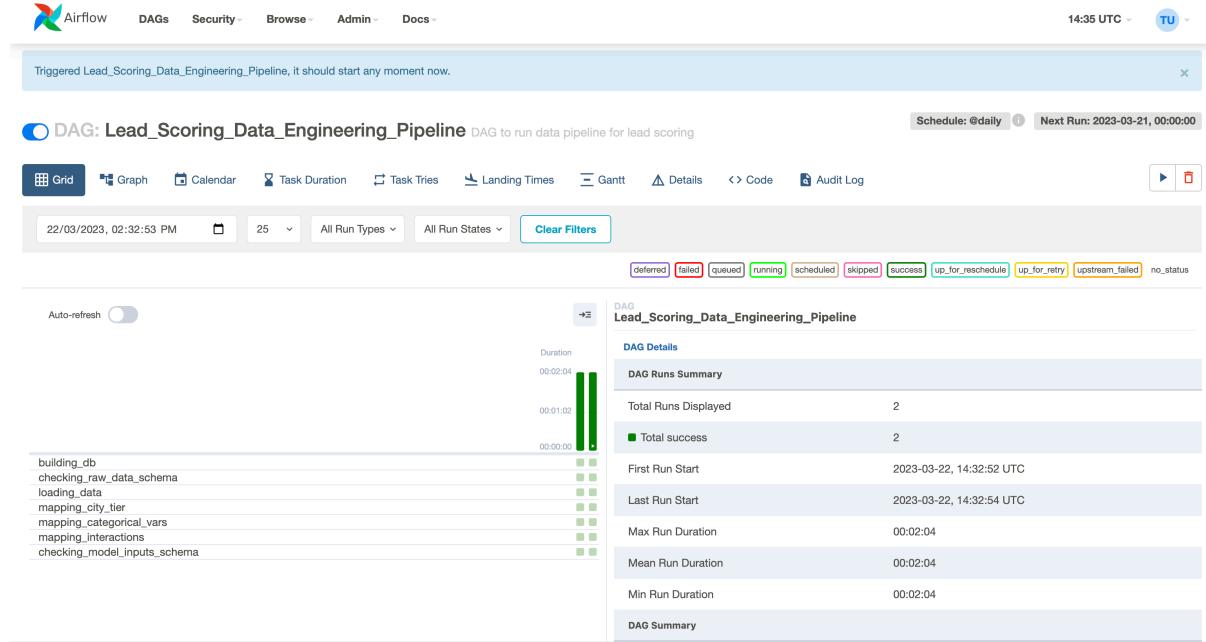
Name	Value
boosting_type	gbdt
class_weight	None
colsample_bytree	1.0
importance_type	split
learning_rate	0.1
max_depth	-1
min_child_samples	20
min_child_weight	0.001
min_split_gain	0.0
n_estimators	100

n_estimators	100	
n_jobs	-1	
num_leaves	31	
objective	None	
random_state	42	
reg_alpha	0.0	
reg_lambda	0.0	
silent	warn	
subsample	1.0	
subsample_for_bin	200000	
subsample_freq	0	
▼ Metrics (8)		
Name	Value	
AUC	0.821	
Accuracy	0.739	
F1	0.762	
Kappa	0.477	
MCC	0.485	
Prec.	0.703	
Kappa	0.477	
MCC	0.485	
Prec.	0.703	
Recall	0.832	
TT	25.33	
▼ Tags (5)		
Name	Value	Actions
Run ID	203161f068bb4eca859d0b e0af515d4eb	
Run Time	537.8	
Source	tune_model	
URI	cdaad403	
USI	f9c4	
<input type="text" value="Name"/>	<input type="text" value="Value"/>	<input type="button" value="Add"/>
▼ Artifacts		

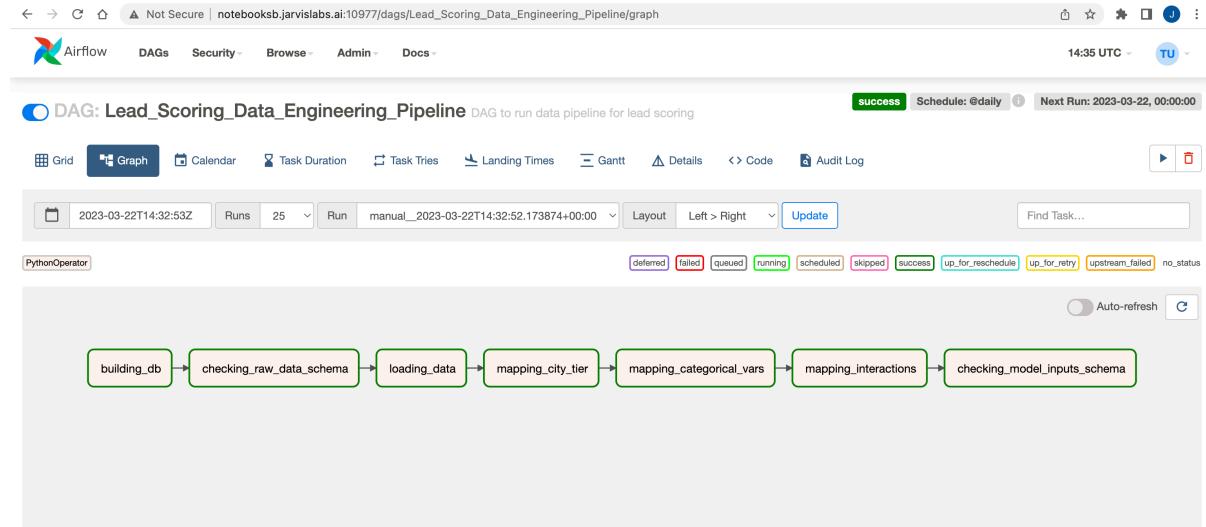
Data Pipeline

Air Flow UI

Airflow DAG Grid (Lead_Scoring_Data_Engineering_Pipeline)



Airflow DAG Graph (Lead_Scoring_Data_Engineering_Pipeline)



List of AirFlow DAGs

Not Secure | notebooksb.jarvislabs.ai:10977/home

14:36 UTC TU

Do not use **SQLite** as metadata DB in production – it should only be used for dev/testing. We recommend using Postgres or MySQL. [Click here](#) for more information.

Do not use **SequentialExecutor** in production. [Click here](#) for more information.

DAGs

All 33	Active 0	Paused 33	Filter DAGs by tag	Search DAGs			
DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Action
Lead_Scoring_Data_Engineering_Pipeline	airflow	2	@daily	2023-03-22, 14:32:52	2023-03-21, 00:00:00	1	[View]
example_bash_operator	airflow	0 * * * *			2023-03-21, 00:00:00	1	[View]
example_branch_datetime_operator	airflow	0 * * * *	@daily		2023-03-21, 00:00:00	1	[View]
example_branch_datetime_operator_2	airflow	0 * * * *	@daily		2023-03-21, 00:00:00	1	[View]
example_branch_dop_operator_v3	airflow	*/1 * * * *			2023-03-22, 14:31:00	1	[View]
example_branch_labels	airflow	0 * * * *	@daily		2023-03-21, 00:00:00	1	[View]
example_branch_operator	airflow	0 * * * *	@daily		2023-03-21, 00:00:00	1	[View]
example_branch_python_operator_decorator	airflow	0 * * * *	@daily		2023-03-21, 00:00:00	1	[View]
example_complex							[View]

Training Pipeline

ML Flow UI

List of experiments (Lead_scoring_mlflow_production)

The screenshot shows the ML Flow UI interface. At the top, there's a header with the ML Flow logo, version 1.26.1, and tabs for 'Experiments' (which is selected) and 'Models'. Below the header, a banner says 'Track machine learning training runs in experiments. Learn more'. A sub-header 'Experiment ID: Lead_scoring_mlflow...' is followed by a 'Description' section with a 'Edit' link. Below this are buttons for 'Refresh', 'Compare', 'Delete', 'Download CSV', and filters for 'Start Time' (set to 'All time') and 'Metrics' (set to 'auc > 0.729'). A search bar filters for 'metrics.rmse < 1 and params.model == "tree"'. The main area shows a table titled 'Showing 1 matching run' with columns: Start Time, Duration, Run Name, User, Source, Version, Models, auc, boosting_type, class_weight, and colsample_bytree. One row is listed: '4 minutes ago', '5.8s', 'run_LightGB', 'root', 'ipykernel...', 'LightGBM/1', '0.729', 'gbdt', 'None', '1.0'. A 'Load more' button is at the bottom.

LightGBM with all artifacts

The screenshot shows the details of a specific experiment run. The top navigation bar includes the ML Flow logo, version 1.26.1, and tabs for 'Experiments' (selected) and 'Models'. The URL shows 'Lead_scoring_mlflow_production > run_LightGB'. The run details are as follows:

- Date: 2023-03-23 13:19:13
- Source: ipykernel_launcher.py
- User: root
- Duration: 5.8s
- Status: FINISHED
- Lifecycle Stage: active

Below the run details, there are sections for 'Description' (with an 'Edit' link) and 'Parameters (20)'. The 'Parameters' table has columns 'Name' and 'Value':

Name	Value
boosting_type	gbdt
class_weight	None
colsample_bytree	1.0
importance_type	split
learning_rate	0.1
max_depth	-1
min_child_samples	20
min_child_weight	0.001
min_split_gain	0.0
n_estimators	100

Not Secure | notebooksb.jarvislabs.ai:21964/#/experiments/1/runs/9db317ff533242229983ae7cfe5f7f5e

n_estimators	100
n_jobs	-1
num_leaves	31
objective	None
random_state	42
reg_alpha	0.0
reg_lambda	0.0
silent	warn
subsample	1.0
subsample_for_bin	200000
subsample_freq	0

▼ Metrics (1)

Name	Value
auc ↗	0.729

▶ Tags

▼ Artifacts

Not Secure | notebooksb.jarvislabs.ai:10976/#/experiments/1/runs/71a6a0a445ac463c87dd1f8f4462e118

▶ Metrics (1)

▶ Tags

▼ Artifacts

▼ models

Full Path: ./mlruns/1/71a6a0a445ac463c87dd1f8f4462e118/artifacts/models ↗
Registered on 2023/03/23

MLflow Model

The code snippets below demonstrate how to make predictions using the logged model. This model is also registered to the [model registry](#).

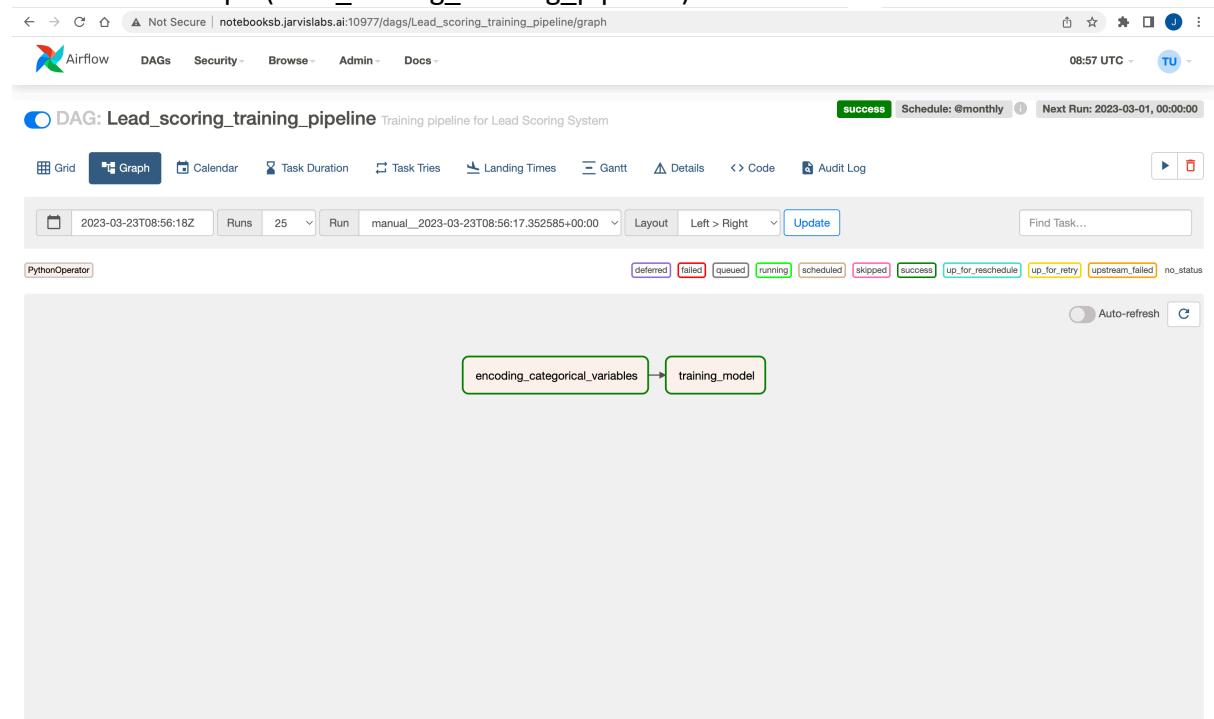
Model schema	Make Predictions
Input and output schema for your model. Learn more Name Type No schema. See MLflow docs for how to include input and output schema with your model.	Predict on a Spark DataFrame: <pre>import mlflow logged_model = 'runs:/71a6a0a445ac463c87dd1f8f4462e118/models' # Load model as a Spark UDF. Override result_type if the model does not return double values. loaded_model = mlflow.pyfunc.spark_udf(spark, model_uri=logged_model, result_type='double') # Predict on a Spark DataFrame. columns = list(df.columns) df.withColumn('predictions', loaded_model(*columns)).collect()</pre>
	Predict on a Pandas DataFrame: <pre>import mlflow logged_model = 'runs:/71a6a0a445ac463c87dd1f8f4462e118/models' # Load model as a PyFuncModel. loaded_model = mlflow.pyfunc.load_model(logged_model) # Predict on a Pandas DataFrame. import pandas as pd loaded_model.predict(pd.DataFrame(data))</pre>

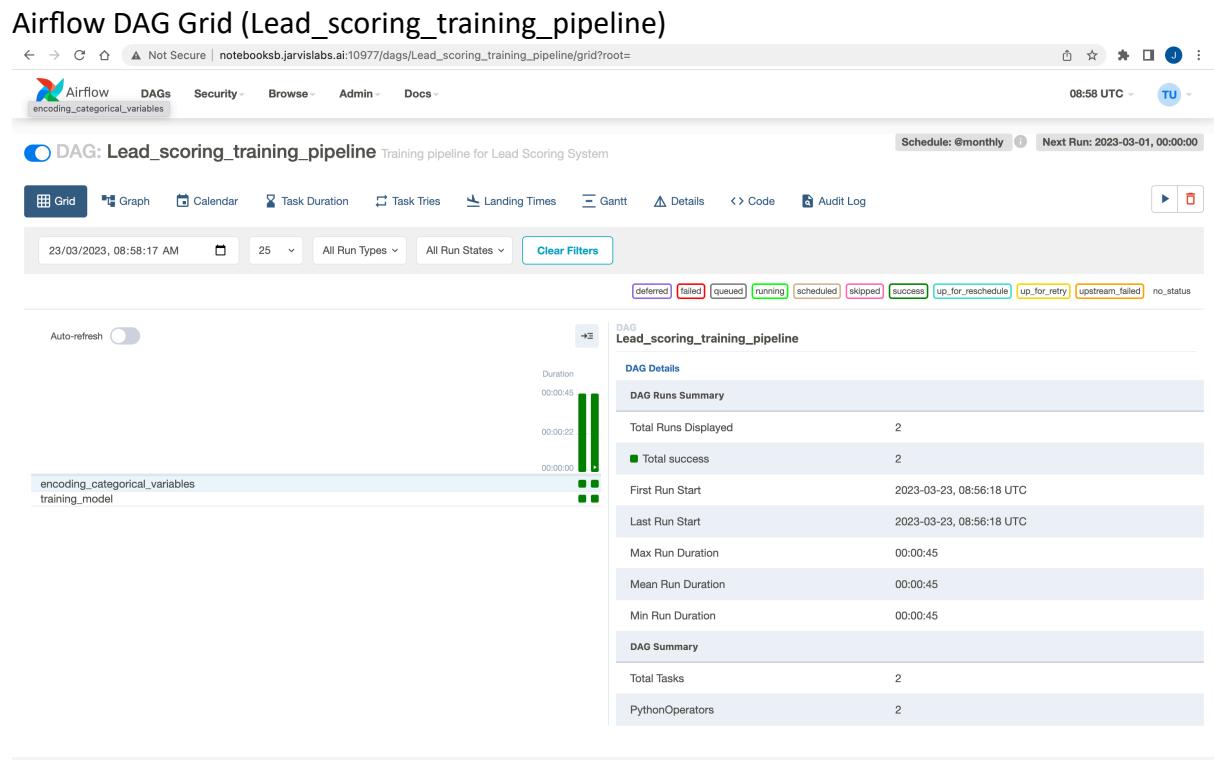
Model Registry with LightGBM in Production

The screenshot shows the mlflow UI interface. At the top, there is a dark header bar with the mlflow logo (1.26.1), navigation links for Experiments and Models, and links for GitHub and Docs. Below the header is a light blue banner with the text "Share and manage machine learning models. Learn more" and a close button (X). A "Create Model" button is located on the left. To the right is a search bar with a magnifying glass icon, a "Search" button, and filter/clear buttons. A table lists registered models. The columns are Name, Latest Version, Staging, Production, Last Modified, and Tags. One entry is shown: "LightGBM" with "Version 1" in the Latest Version column, "Staging" in the Staging column, "Production" in the Production column, "2023-03-23 13:24:21" in the Last Modified column, and no tags. Below the table are navigation icons (back, forward, first, last) and a page size selector (10 / page).

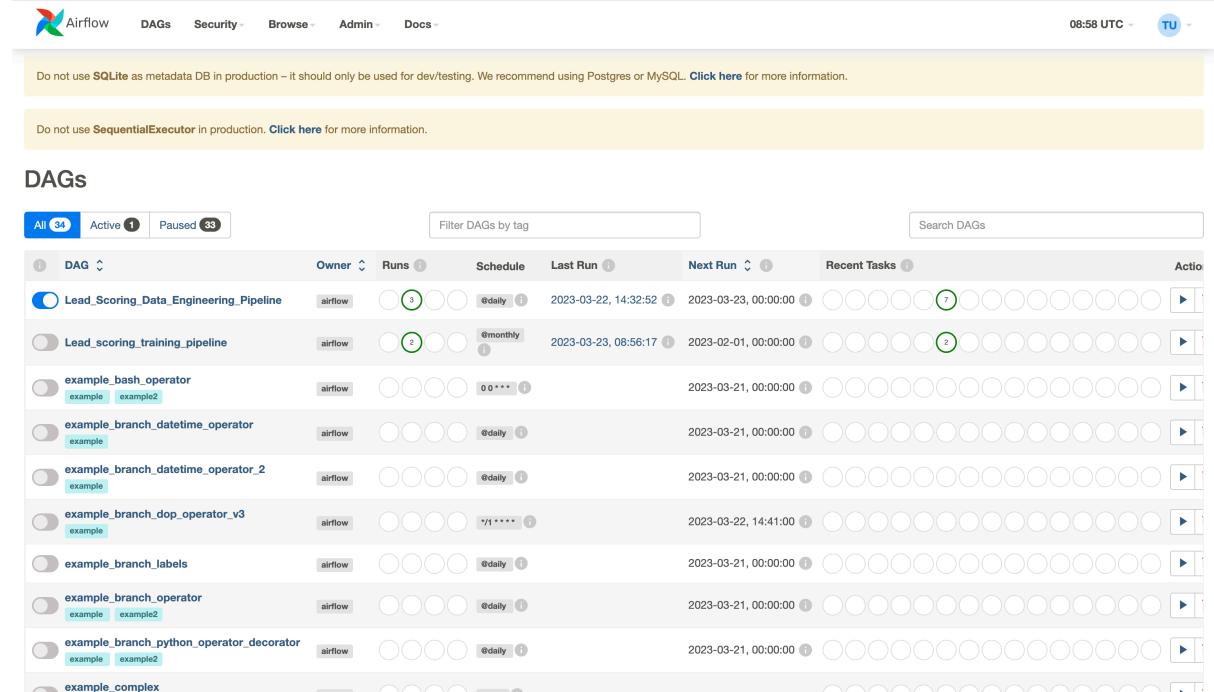
Airflow UI

Airflow DAG Graph (Lead_scoring_training_pipeline)





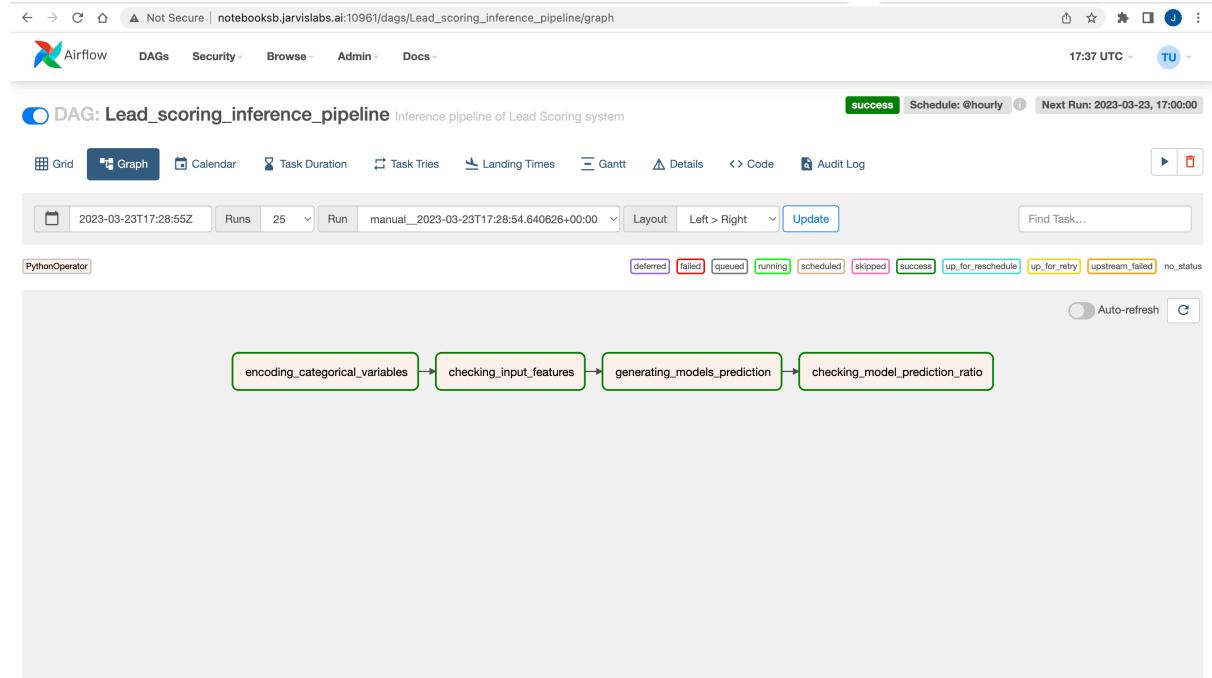
List of all Airflow DAGs



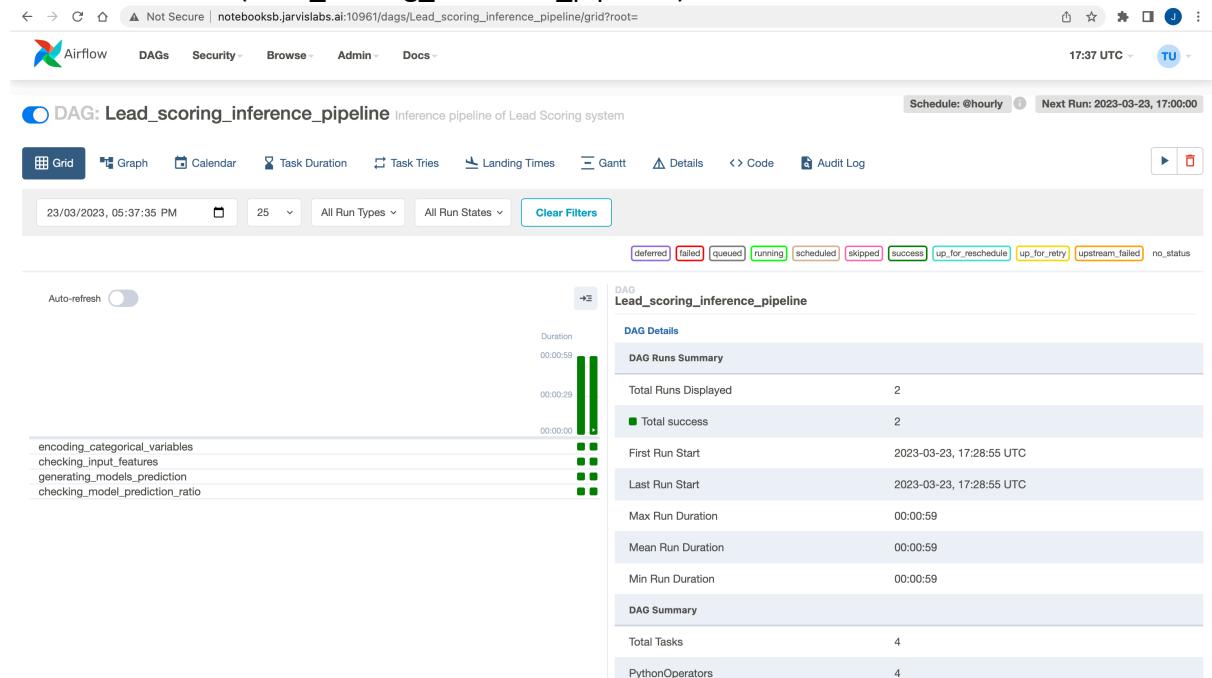
Inference pipeline

Airflow UI

Airflow DAG Graph (Lead_scoring_inference_pipeline)



Airflow DAG Grid (Lead_scoring_inference_pipeline)



List of all Airflow DAGs

Not Secure | notebooksb.jarvislabs.ai:10961/home

17:38 UTC TU

Do not use SQLite as metadata DB in production – it should only be used for dev/testing. We recommend using Postgres or MySQL. [Click here](#) for more information.

Do not use SequentialExecutor in production. [Click here](#) for more information.

DAGs

All 35	Active 3	Paused 32	Filter DAGs by tag	Search DAGs			
DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Action
Lead_Scoring_Data_Engineering_Pipeline	airflow	3	@daily	2023-03-22, 14:32:52	2023-03-23, 00:00:00	7	[View]
Lead_scoring_inference_pipeline	airflow	2	@hourly	2023-03-23, 17:28:54	2023-03-23, 17:00:00	4	[View]
Lead_scoring_training_pipeline	airflow	2	@monthly	2023-03-23, 08:56:17	2023-03-01, 00:00:00	2	[View]
example_bash_operator	example example2	0	* * * * *		2023-03-22, 00:00:00	1	[View]
example_branch_datetime_operator	example	0	@daily		2023-03-22, 00:00:00	1	[View]
example_branch_datetime_operator_2	airflow	0	@daily		2023-03-22, 00:00:00	1	[View]
example_branch_dop_operator_v3	airflow	0	*/* * * * *		2023-03-23, 17:36:00	1	[View]
example_branch_labels	airflow	0	@daily		2023-03-22, 00:00:00	1	[View]
example_branch_operator	airflow	0	@daily		2023-03-22, 00:00:00	1	[View]

Running on Leadsoring_inference.csv

Airflow DAG Grid (Lead_scoring_Data_Engineering_Pipeline)

Not Secure | notebooksb.jarvislabs.ai:10969/dags/Lead_Scoring_Data_Engineering_Pipeline/grid

05:27 UTC TU

DAG: Lead_Scoring_Data_Engineering_Pipeline DAG to run data pipeline for lead scoring

Schedule: @daily Next Run: 2023-03-25, 00:00:00

Grid Graph Calendar Task Duration Task Tries Landing Times Gantt Details Code Audit Log

25/03/2023, 05:26:43 AM All Run Types All Run States Clear Filters

deferred failed queued running scheduled skipped success up_for_reschedule up_for_retry upstream_failed no_status

Auto-refresh

DAG Details

Lead_Scoring_Data_Engineering_Pipeline

DAG Runs Summary

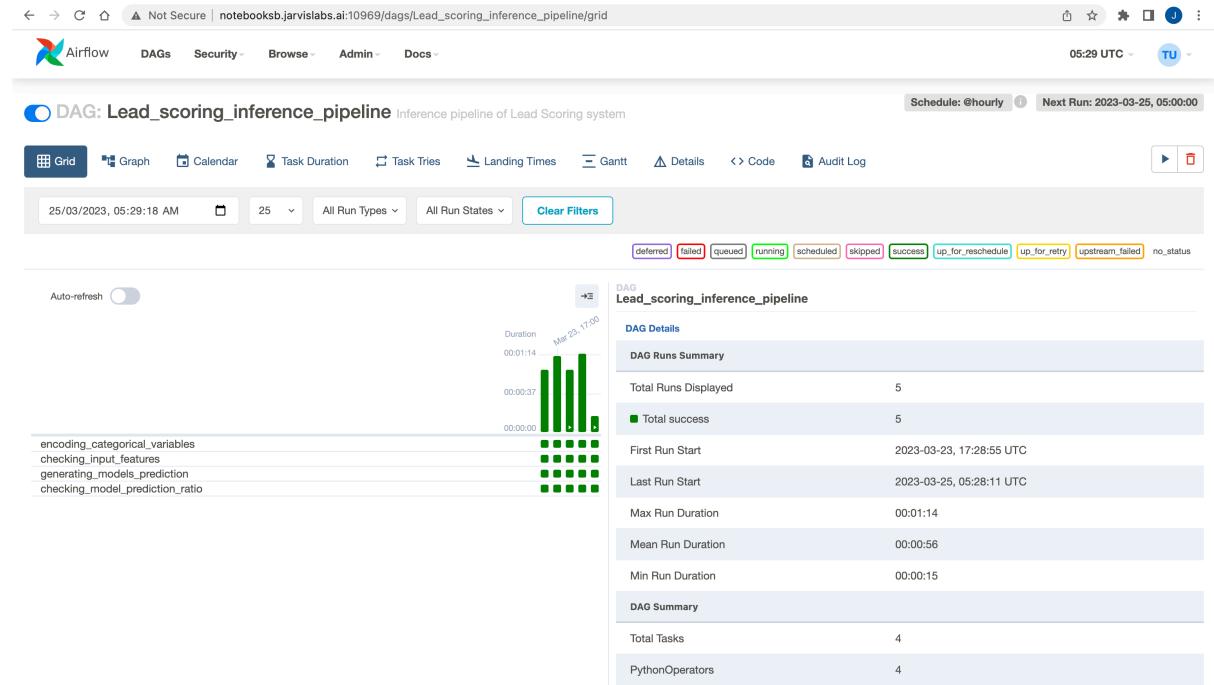
Total Runs Displayed	5
Total success	5
First Run Start	2023-03-22, 14:32:52 UTC
Last Run Start	2023-03-25, 05:26:01 UTC
Max Run Duration	00:02:04
Mean Run Duration	00:01:43
Min Run Duration	00:01:10

DAG Summary

Total Tasks	7
PythonOperators	7

building_db
checking_raw_data_schema
loading_data
mapping_city_tier
mapping_categorical_vars
mapping_interactions
checking_model_inputs_schema

Airflow DAG Grid (Lead_scoring_inference_pipeline)



Output of prediction_distribution.txt after run on Leadscore_inference.csv

The screenshot shows a Jupyter Notebook interface with multiple tabs open. The current tab displays the contents of 'prediction_distribution.txt'. The file contains five lines of text, each representing a prediction distribution. The output is as follows:

```
1 2023-03-23 17:29:48.360966 %of 1 = 0.6487148489304693 %of 2 = 0.3512851510695308
2 2023-03-23 17:29:54.444831 %of 1 = 0.6487148489304693 %of 2 = 0.3512851510695308
3 2023-03-25 05:27:03.482213 %of 1 = 0.6487148489304693 %of 2 = 0.3512851510695308
4 2023-03-25 05:27:15.711802 %of 1 = 0.6487148489304693 %of 2 = 0.3512851510695308
5 2023-03-25 05:28:26.435877 %of 1 = 0.532150864830237 %of 2 = 0.46784913516976295
```

Unit Test

Four test successful run

The screenshot shows a Jupyter Notebook interface. On the left, there is a file browser window titled 'notebooks' showing the contents of the directory '/Assignment / 02_training_pipeline/'. It lists three items: 'notebooks' (modified an hour ago), 'scripts' (modified 2 days ago), and 'INSTRUCTIONS_training.txt' (modified 7 months ago). On the right, there is a terminal window titled 'Terminal 6' showing the output of a pytest run. The output indicates a successful run with 4 passed tests and 3 warnings.

```
root@8a820ec2d078:~/Assignment/01_data_pipeline/unit_test# pytest test_with_pytest.py
=====
platform linux -- Python 3.8.12, pytest-6.2.5, py-1.11.0, pluggy-1.0.0
rootdir: /home/Assignment/01_data_pipeline/unit_test
plugins: asyncio-3.6.1, cov-3.8.0, pythonpath-0.7.4, hypothesis-4.50.8
collected 4 items

test_with_pytest.py .... [100%]

test_with_pytest.py::test_map_categorical_vars
/home/Assignment/01_data_pipeline/unit_test/utils.py:191: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
new_df['first_platform_c'] = "others"

test_with_pytest.py::test_map_categorical_vars
/home/Assignment/01_data_pipeline/unit_test/utils.py:200: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
new_df['first_utm_medium_c'] = "others"

test_with_pytest.py::test_map_categorical_vars
/home/Assignment/01_data_pipeline/unit_test/utils.py:209: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
new_df['first_utm_source_c'] = "others"

-- Docs: https://docs.pytest.org/en/stable/warnings.html
===== 4 passed, 3 warnings in 0.54s =====
root@8a820ec2d078:~/Assignment/01_data_pipeline/unit_test#
```