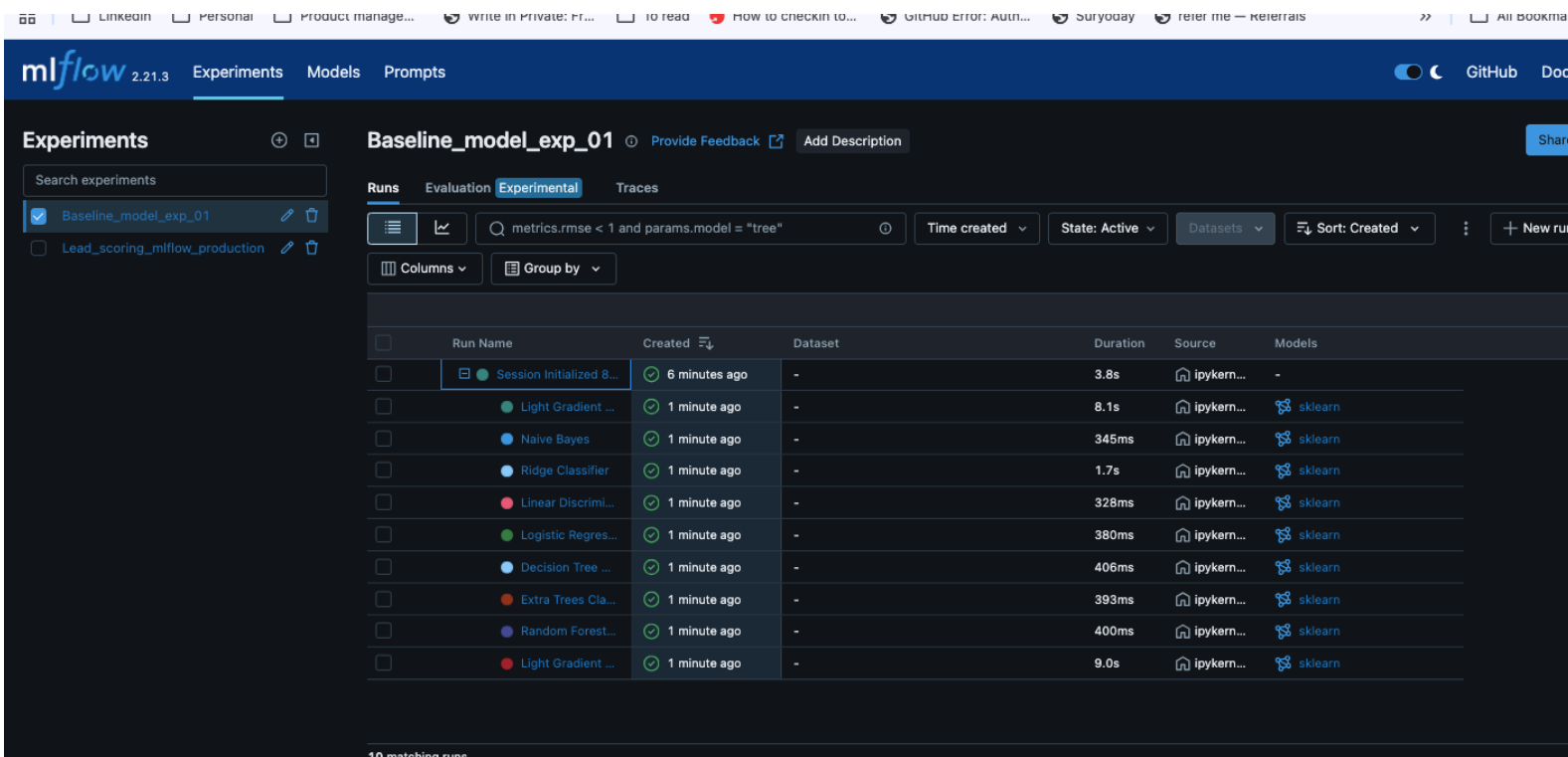


Model Experimentation

Screenshot of MLflow UI

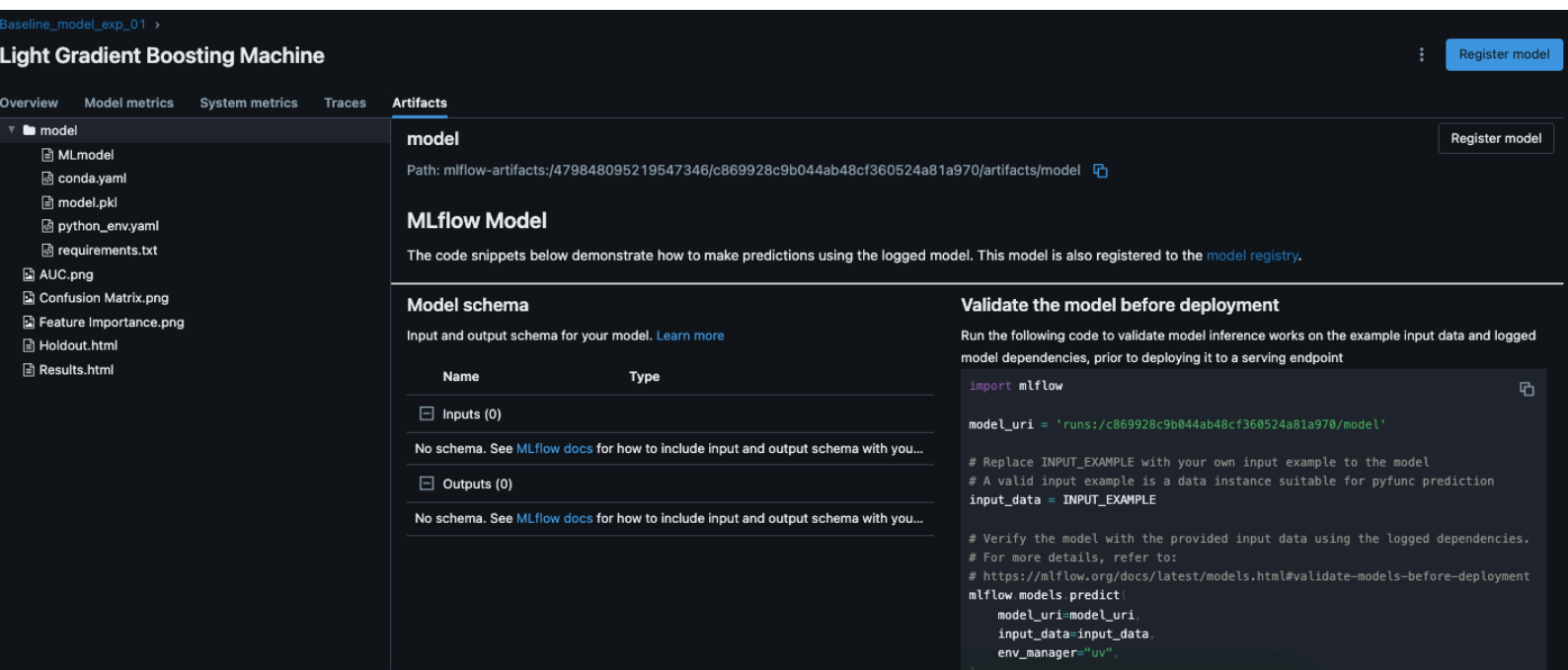
Screenshot of all experiments



The screenshot shows the MLflow UI interface. The top navigation bar includes 'Experiments', 'Models', and 'Prompts'. The left sidebar shows a list of experiments, with 'Baseline_model_exp_01' selected. The main content area displays the details for 'Baseline_model_exp_01', including a search bar, filters, and a table of runs. The table lists various models like 'Session Initialized 8...', 'Light Gradient ...', 'Naive Bayes', 'Ridge Classifier', 'Linear Discrimi...', 'Logistic Regres...', 'Decision Tree ...', 'Extra Trees Cla...', 'Random Forest...', and 'Light Gradient ...'. Each row shows the run name, creation time, dataset, duration, source, and models used.

Run Name	Created	Dataset	Duration	Source	Models
Session Initialized 8...	6 minutes ago	-	3.8s	ipykern...	-
Light Gradient ...	1 minute ago	-	8.1s	ipykern...	sklearn
Naive Bayes	1 minute ago	-	345ms	ipykern...	sklearn
Ridge Classifier	1 minute ago	-	1.7s	ipykern...	sklearn
Linear Discrimi...	1 minute ago	-	328ms	ipykern...	sklearn
Logistic Regres...	1 minute ago	-	380ms	ipykern...	sklearn
Decision Tree ...	1 minute ago	-	406ms	ipykern...	sklearn
Extra Trees Cla...	1 minute ago	-	393ms	ipykern...	sklearn
Random Forest...	1 minute ago	-	400ms	ipykern...	sklearn
Light Gradient ...	1 minute ago	-	9.0s	ipykern...	sklearn

Screenshot of one experiment with all artefacts visible



The screenshot shows the MLflow UI interface for a specific experiment, 'Baseline_model_exp_01'. The left sidebar shows a list of artifacts, including 'model', 'MLmodel', 'conda.yaml', 'model.pkl', 'python_env.yaml', 'requirements.txt', 'AUC.png', 'Confusion Matrix.png', 'Feature Importance.png', 'Holdout.html', and 'Results.html'. The main content area displays the details for the 'model' artifact, including the path, the MLflow Model schema, and the code snippets for predicting using the logged model.

Light Gradient Boosting Machine

Overview Model metrics System metrics Traces **Artifacts**

model

Path: mlflow-artifacts:/479848095219547346/c869928c9b044ab48cf360524a81a970/artifacts/model

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

Input and output schema for your model. [Learn more](#)

Name	Type
Inputs (0)	
No schema. See MLflow docs for how to include input and output schema with you...	
Outputs (0)	
No schema. See MLflow docs for how to include input and output schema with you...	

Validate the model before deployment

Run the following code to validate model inference works on the example input data and logged model dependencies, prior to deploying it to a serving endpoint

```
import mlflow

model_uri = 'runs:/c869928c9b044ab48cf360524a81a970/model'

# Replace INPUT_EXAMPLE with your own input example to the model
# A valid input example is a data instance suitable for pyfunc prediction
input_data = INPUT_EXAMPLE

# Verify the model with the provided input data using the logged dependencies.
# For more details, refer to:
# https://mlflow.org/docs/latest/models.html#validate-models-before-deployment
mlflow.models.predict(
    model_uri=model_uri,
    input_data=input_data,
    env_manager="uv",
```

Screenshot of MLflow UI after dropping features

Screenshot of all experiments

The screenshot shows the MLflow UI interface. The top navigation bar includes the MLflow logo, version 2.21.3, and tabs for Experiments, Models, and Prompts. On the right, there are links for GitHub and Docs, and a dark mode toggle. The left sidebar shows a list of experiments: Lead_Scoring_Training_Pipeline, Baseline_model_exp_02 (selected), Baseline_model_exp_01, and Lead_scoring_mlflow_production. The main area displays the details for 'Baseline_model_exp_02', including a search bar with the query 'metrics.rmse < 1 and params.model = "tree"', filters for 'Time created', 'State: Active', and 'Datasets', and a '+ New run' button. Below this is a table of runs with columns: Run Name, Created, Dataset, Duration, Source, and Models. The table lists 13 runs, including 'Session Initialized 6...', 'Light Gradient ...', 'Naive Bayes', 'Linear Discrimi...', 'Ridge Classifier', 'Logistic Regres...', 'Decision Tree ...', 'Extra Trees Cla...', 'Random Forest...', 'Light Gradient ...', and 'CatBoost Class...'. At the bottom, it states '43 matching runs'.

Run Name	Created	Dataset	Duration	Source	Models
Session Initialized 6...	1 day ago	-	-	ipykern...	-
Light Gradient ...	1 day ago	-	10.5s	ipykern...	sklearn
Light Gradient ...	1 day ago	-	9.0s	ipykern...	sklearn
Naive Bayes	1 day ago	-	0.5s	ipykern...	sklearn
Linear Discrimi...	1 day ago	-	0.5s	ipykern...	sklearn
Ridge Classifier	1 day ago	-	425ms	ipykern...	sklearn
Logistic Regres...	1 day ago	-	432ms	ipykern...	sklearn
Decision Tree ...	1 day ago	-	406ms	ipykern...	sklearn
Extra Trees Cla...	1 day ago	-	431ms	ipykern...	sklearn
Random Forest...	1 day ago	-	478ms	ipykern...	sklearn
Light Gradient ...	1 day ago	-	462ms	ipykern...	sklearn
CatBoost Class...	1 day ago	-	437ms	ipykern...	sklearn

Screenshot of one experiment with all artefacts visible

The screenshot shows the MLflow UI interface for the 'Baseline_model_exp_02' experiment. The left sidebar shows the experiment name and a list of artifacts: MLmodel, conda.yaml, model.pkl, python_env.yaml, requirements.txt, AUC.png, Confusion Matrix.png, Feature Importance.png, Holdout.html, and Results.html. The main area displays the details for the 'Light Gradient Boosting Machine' experiment, including a 'Register model' button. Below this is a table of artifacts with columns: Name, Type, and Path. The table lists the following artifacts: MLmodel, conda.yaml, model.pkl, python_env.yaml, requirements.txt, AUC.png, Confusion Matrix.png, Feature Importance.png, Holdout.html, and Results.html. The 'MLflow Model' section shows the code snippets for making predictions using the logged model. The 'Model schema' section shows the input and output schema for the model. The 'Validate the model before deployment' section shows the code snippets for validating the model inference works on the example input data and logged dependencies, prior to deploying it to a serving endpoint.

Name	Type	Path
MLmodel	Model	Path: mlflow-artifacts:/432885645426391555/59be74396eb44b23ba5e3f463e94855f/artifacts/model
conda.yaml	File	
model.pkl	File	
python_env.yaml	File	
requirements.txt	File	
AUC.png	Image	
Confusion Matrix.png	Image	
Feature Importance.png	Image	
Holdout.html	File	
Results.html	File	

Lead_Scoring_Training_Pipeline >
redolent-shrike-62

Run ID: dd24ea882b22491d9ed86c7b2aad676a Date: 2025-04-07 03:50:58 Source: airflow Git Commit: db887dea27d98a162f0002270db2067406102773

User: arunprakash Duration: 5.7s Status: FINISHED Lifecycle Stage: active

> Description [Edit](#)

> Datasets

> Parameters (20)

▼ Metrics (4)

Name	Value
Precision 🔗	0.706
Recall 🔗	0.837
auc 🔗	0.74
test_accuracy 🔗	0.741

Registered Models 🔗 Create Model

Filter registered models by name or tags 🔍 🔍

Name [⬆]	Latest version	Staging	Production	Created by	Last modified	Tags
LightGBM	Version 1	—	—		2025-04-07 03:51:...	—

Screen of Production experiment with artifacts visible

mlflow 2.8.1 Experiments Models GitHub Docs

Experiments ⊕ 🔍

Search Experiments

☐ Default 🔗 🗑

☐ Lead_Scoring_Training_Pipeline 🔗 🗑

☒ Lead_scoring_mlflow_product... 🔗 🗑

Lead_scoring_mlflow_production 🔗 Provide Feedback 🔗 Share

Experiment ID: 2 Artifact Location: /Users/arunprakash/Downloads/Assignment/03_inference_pipeline/mlruns/2

> Description [Edit](#)

Q metrics.rmse < 1 and params.model = "tree" 🔍 Time created ⌵ State: Active ⌵ Sort: Created ⌵ ⋮ 🔗 🗑 🔄 New run

Columns ⌵

Table Chart Evaluation **Experimental**

<input type="checkbox"/>	<input type="radio"/>	Run Name	Created tags: 'mlflow.runName'	Dataset	Duration	Source	Models
<input type="checkbox"/>	<input type="radio"/>	melodic-croc-24	🟢 3 hours ago	-	50.0min	ipykerne...	LightGBM/3
<input type="checkbox"/>	<input type="radio"/>	funny-finch-700	🟢 3 hours ago	-	1.0s	ipykerne...	-

Lead_scoring_mflow_production >

melodic-croc-24

Run ID: 235eadd7fad43a9ab4881a62b4e4ed0

Date: 2025-04-08 18:51:37

Source: ipykernel_launcher.py

User: arunprakash

Duration: 50.0min

Status: FINISHED

Lifecycle Stage: active

> Description [Edit](#)

> Datasets

> Parameters (20)

> Metrics (4)

> Tags

> Artifacts

model

MLmodel

conda.yaml

model.pkl

python_env.yaml

registered_model_meta

requirements.txt

Full Path: /Users/arunprakash/Downloads/Assignment/03_inference_pipeline/miruns/2/235eadd7fad43a9ab4881a62b4e4ed0/artifacts...

LightGBM, v3

Registered on 2025/04/08

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

Input and output schema for your model. [Learn more](#)

Make Predictions

Predict on a Spark DataFrame:

Screenshot of Model Registry

Registered Models > LightGBM >

Version 3

Registered At: 2025-04-08 18:51:48

Stage: Production

Last Modified: 2025-04-08 20:06:10

Source Run: melodic-croc-24

> Description [Edit](#)

> Tags

> Schema

Name	Type
No schema. See MLflow docs for how to include input and output schema with your model.	

Screenshot of Airflow successful DAG Graph

DAG: Lead_scoring_inference_pipeline

inference pipeline of Lead Scoring system

Schedule: @hourly

Next Run ID: 2025-04-08, 19:00:00 UTC

08/04/2025

08:52:13 PM

All Run Types

All Run States

Clear Filters

Auto-refresh

25

Press **SHIFT** + **/** for Shortcuts

deferred failed queued removed restarting running scheduled shutdown skipped success up_for_reschedule up_for_retry upstream_failed no_status

DAG

Run

Task

Lead_scoring_inference_pipeline / > 2024-04-09, 00:00:00 UTC / encoding_categorical_variables

Clear task

Mark state as...

Filter DAG by task

Details

Graph

Gantt

Code

Event Log

Logs

XCom

Task Duration

Layout: Left -> Right

encoding_categorical_variables

checking_input_features

generating_models_prediction

checking_model_prediction_ratio

Screenshot of Airflow UI grid

