

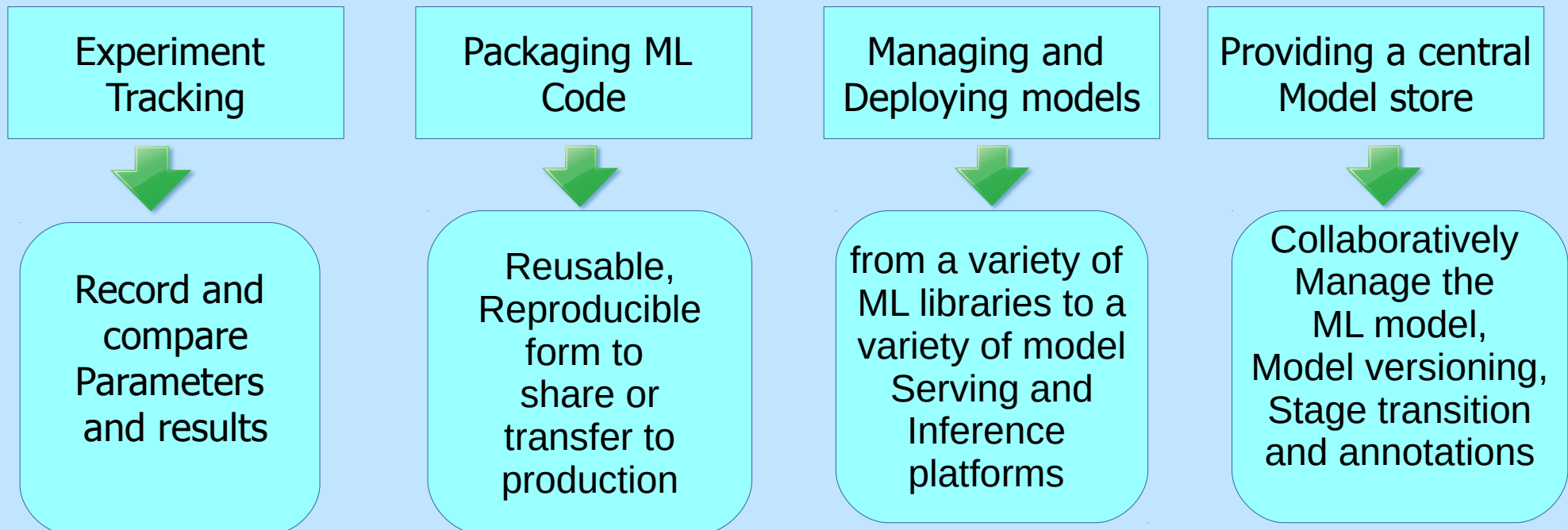
MLflow

Using MLflow for Experiment Tracking and Model Management for Diamond data set

Introduction to MLflow

MLflow is an open source platform for managing the end-to-end machine learning lifecycle.

It tackles four primary functions:



Introduction to Experiment Tracking

Terminologies:

- 1.Experiment
- 2.Run
- 3.Metadata (i.e. Tags, Parameters, Metrics)
- 4.Artifacts (i.e. Output files associated with experiment runs)

What do you want to track for each Experiment Run?

- 1.Training and Validation Data Used
- 2.Hyperparameters
- 3.Metrics
- 4.Models

Why Track?

Organization Optimization Reproducibility

Uses of ML flow

Tool - MLflow

MLFlow helps you to organize your experiments into runs.

MLflow keeps track of:

- Tags

- Parameters

- Metrics

- Models

- Artifact

- Source code, Start and End Time, Authors etc

Working on MLflow

Step 1 - Import MLFlow

```
import mlflow
```

Step 2 - Set the tracker and experiment

```
mlflow.set_tracking_uri(DATABASE_URI)
```

```
mlflow.set_experiment("EXPERIMENT_NAME")
```

Step 3 - Start a experiment run

```
with mlflow.start_run():
```

Step 4 - Logging the metadata

```
mlflow.set_tag(KEY, VALUE)
```

```
mlflow.log_param(KEY, VALUE) mlflow.log_metric(KEY, VALUE)
```

Step 5 - Logging the model and other files (2 ways)

```
Way 1 – mlflow.<FRAMEWORK>.log_model(MODEL_OBJECT,artifact_path="PATH")
```

```
Way 2 - mlflow.log_artifact(LOCAL_PATH, artifact_path="PATH")
```

Working on MLflow

Requirements :

Source:

https://github.com/KusumaChalla/Innomatics_Research_Lab_Intenship2022/tree/main/Machine%20Learning/Experiments

.ipynb_checkpoints	20-09-2022 15:36	File folder	
data	20-09-2022 12:44	File folder	
mlruns	21-09-2022 00:51	File folder	
pickle_files	20-09-2022 16:50	File folder	
experiments.ipynb	21-09-2022 11:55	IPYNB File	62 KB
mlflow.db	21-09-2022 11:59	Data Base File	224 KB

**Contains
diamond.csv**

**Contains the experiment id
folder i.e set_experiment()
and logs of artifacts files and
created while we run mlflow**

**Contain pickle files like
labelencoder, scaler
pickle files**

Contains mlflow code

**Created when
mlflow ui --backend-store-uri sqlite:///mlflow.db is
run on cmd terminal after mlflow installation and its the
sqlite database that logs the models**

1. Install mlflow

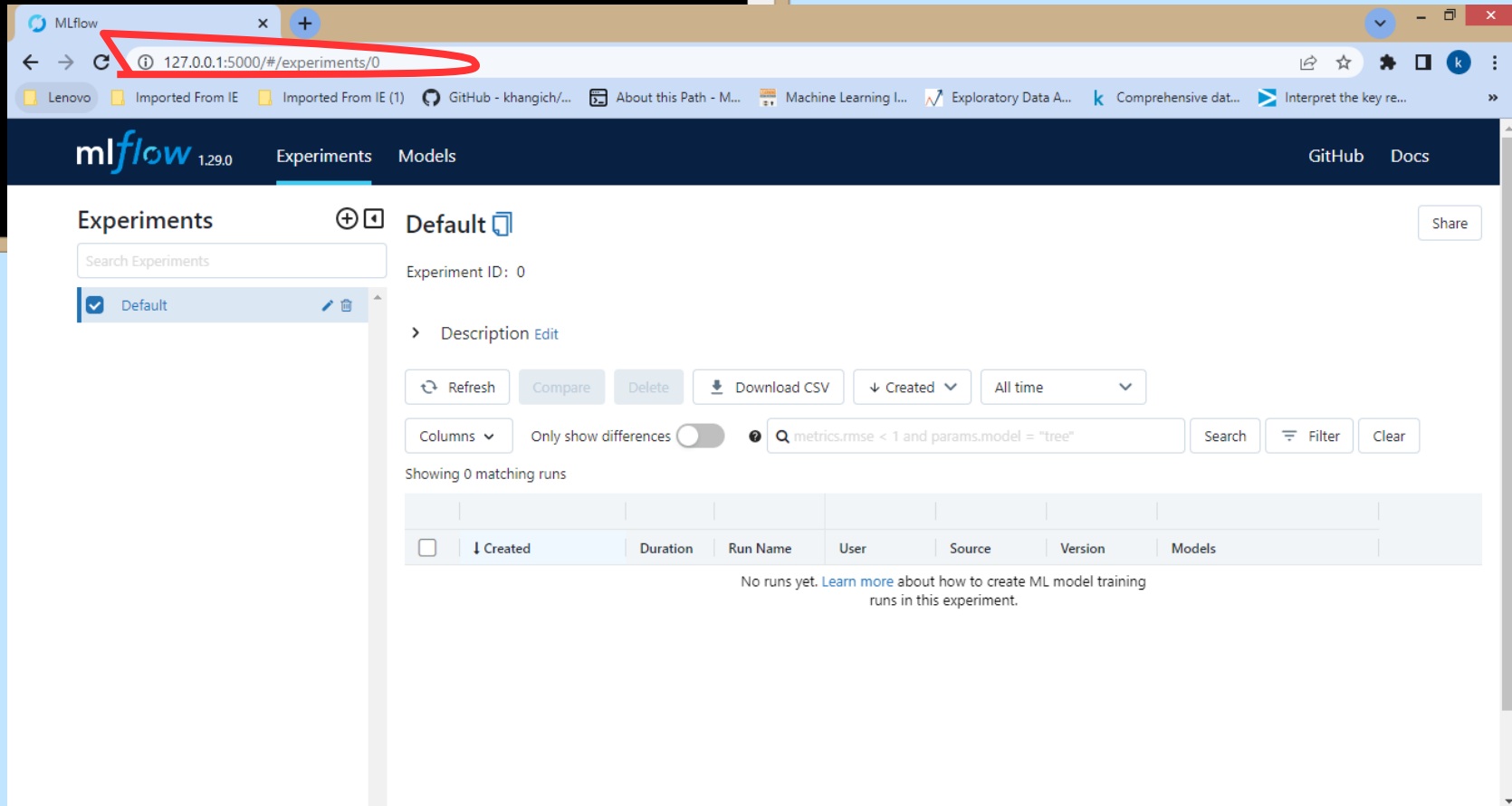
pip install mlflow

mlflow ui --backend-store-uri sqlite:///mlflow.db

```
C:\WINDOWS\system32\cmd.exe - mlflow ui --backend-store-uri sqlite:///mlflow.db
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\kusuma>cd C:\Users\kusuma\Desktop\Innomatics Lab Intership\AppDev\Diamendprice_StreamlitApp\Experiments

C:\Users\kusuma\Desktop\Innomatics Lab Intership\AppDev\Diamendprice_StreamlitApp\Experiments>mlflow ui --backend-store-uri sqlite:///mlflow.db
INFO:waitress:erving on http://127.0.0.1:5000
```



2.Run the MLFlow code

Experiment run ¶

```
import mlflow
```

```
mlflow.set_tracking_uri("sqlite:///mlflow.db")
```

```
mlflow.set_experiment("Diamond Price Prediction")
```



The screenshot shows the MLflow web interface in a browser. The address bar indicates the URL is 127.0.0.1:5000/#/experiments/2. The interface has a dark blue header with the MLflow logo (1.29.0) and tabs for 'Experiments' and 'Models'. On the right of the header are links for 'GitHub' and 'Docs'. The main content area is titled 'Diamond_Price Prediction' with a 'Share' button. Below the title, it shows 'Experiment ID: 2'. There is a 'Description Edit' section. A list of experiments is shown on the left, with 'Diamond_Price Prediction' selected and highlighted by a red rectangle. Below this, there are buttons for 'Refresh', 'Compare', 'Delete', 'Download CSV', and a dropdown for 'Created' set to 'All time'. A search bar contains the query 'metrics.rmse < 1 and params.model = "tree"'. Below the search bar, it says 'Showing 0 matching runs'. A table with columns 'Created', 'Duration', 'Run Name', 'User', 'Source', 'Version', and 'Models' is shown, but it is empty. At the bottom, a message states 'No runs yet. Learn more about how to create ML model training runs in this experiment.'

Desktop/Innomatics Lab Intensi... experiments - Jupyter Notebook MLflow

127.0.0.1:5000/#/experiments/2

Lenovo Imported From IE Imported From IE (1) GitHub - khangich/... About this Path - M... Machine Learning I... Exploratory Data A... Comprehensive dat... Interpret the key re...

mlflow 1.29.0 Experiments Models GitHub Docs

Experiments (+) Diamond_Price Prediction

Search Experiments

☐ Default ☒ Diamond_Price Prediction

Experiment ID: 2

Description Edit

Refresh Compare Delete Download CSV ↑ Created All time

Columns Only show differences ☐ Q metrics.rmse < 1 and params.model = "tree" Search Filter Clear

Showing 0 matching runs

	Created	Duration	Run Name	User	Source	Version	Models
--	---------	----------	----------	------	--------	---------	--------

No runs yet. [Learn more](#) about how to create ML model training runs in this experiment.

3.Run the experiments/models

Ex :KneighborsRegressor model



Experiment 2 -Training KNeighborsRegressor

```
In [41]: with mlflow.start_run():|
mlflow.set_tag("dev", "Kusuma")
mlflow.set_tag("algorithm", "KNN")
# Log the data for each run using log_param, log_metric, log_model
mlflow.log_param("data-path", "data/diamonds.csv")
k = 5
mlflow.log_param("n_neighbors", k)
knn_regressor = KNeighborsRegressor(n_neighbors=k)
knn_regressor.fit(X_train_transformed, y_train)
y_test_pred = knn_regressor.predict(X_test_transformed)
MAE = metrics.mean_absolute_error(y_test, y_test_pred)
mlflow.log_metric("Mean Absolute Error", MAE)
mlflow.sklearn.log_model(knn_regressor, artifact_path="models")
mlflow.log_artifact("pickle_files/label_encoderfile.pkl")
mlflow.log_artifact("pickle_files/standard_scaler.pkl")
```

The screenshot shows the MLflow web interface with the following components:

- Header:** MLflow 1.29.0, Experiments, Models, GitHub, Docs.
- Left Sidebar:** Search Experiments, Default, Diamond_Price Prediction (selected).
- Main Content:** Diamond_Price Prediction, Experiment ID: 2, Description Edit.
- Buttons:** Refresh, Compare, Delete, Download CSV, Created (dropdown), All time (dropdown).
- Search Bar:** Columns, Only show differences (toggle), Search, Filter, Clear.
- Table:** Showing 1 matching run. The table has columns: Created, Duration, Run Name, User, Sc, Ve, Models, Mean, Parameters, Tags. The first row is highlighted with a red box.
- Buttons:** Load more.

Created	Duration	Run Name	User	Sc	Ve	Models	Mean	Parameters	Tags
36 seconds ago	23.4s	placid-be...	kusuma			sklea	40...	da... 5	Kusuma KNN

4.Run all the experiments and check parallelly the Mlflow UI to track the experiments

The screenshot shows the MLflow 1.29.0 Experiments page for an experiment named 'Diamond_Price Prediction'. The interface includes a sidebar with a search bar and a list of experiments, where 'Diamond_Price Prediction' is selected. The main panel displays the experiment details, including the Experiment ID (2), a description, and a table of runs. The table is filtered to show 5 matching runs based on the search criteria 'metrics.rmse < 1 and params.model = "tree"'. The runs are sorted by creation time, with the most recent run at the top. A red box highlights the table of runs.

Experiments **Diamond_Price Prediction** Share

Search Experiments

☐ Default ☒ Diamond_Price Prediction

Experiment ID: 2

> Description Edit

Refresh Compare Delete Download CSV Created All time

Columns Only show differences metrics.rmse < 1 and params.model = "tree" Search Filter Clear

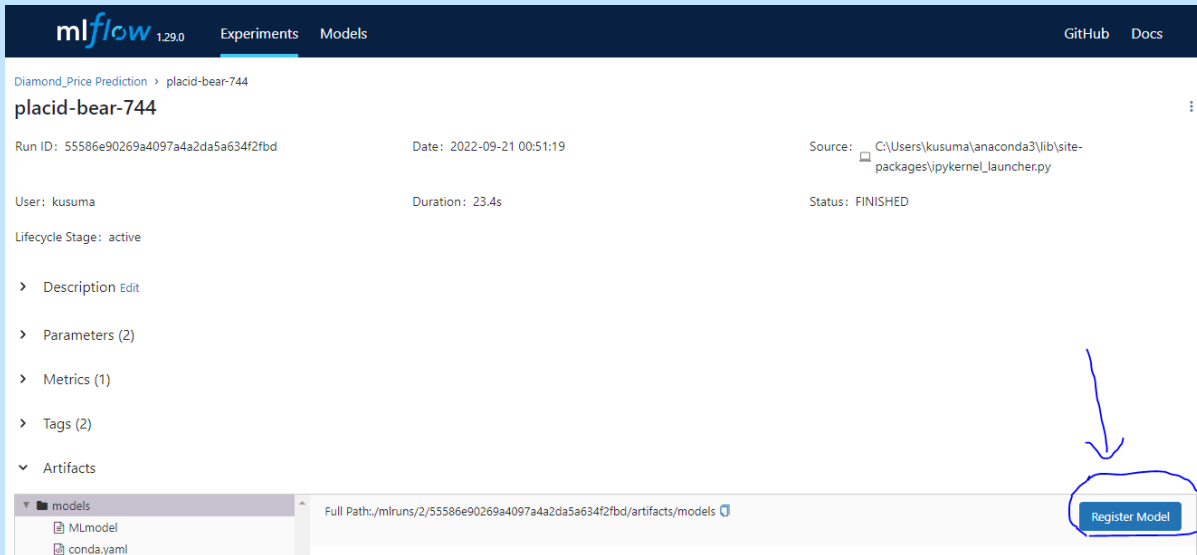
Showing 5 matching runs

				Metrics	Parameters >	Tags			
<input type="checkbox"/>	↓ Created	Duration	Run Name	Mean Absolute	data-path	learning_rate	max_depth	dev	algorithm
<input type="checkbox"/>	✓ 28 seconds ago	17.8s	suave-goat-...	621.4	data/diamo...	-	8	Kusuma	DecisionTree...
<input type="checkbox"/>	✓ 1 minute ago	17.8s	crawling-fly-...	869	data/diamo...	-	-	Kusuma	LinearRegre...
<input type="checkbox"/>	✓ 2 minutes ago	39.4s	omniscient-...	438.4	data/diamo...	-	30	Kusuma	RandomFor...
<input type="checkbox"/>	✓ 3 minutes ago	42.8s	suave-vole-...	297	data/diamo...	0.1	5	Kusuma	GradientBoo...
<input type="checkbox"/>	✓ 14 minutes ago	23.4s	placid-bear-...	404.7	data/diamo...	-	-	Kusuma	KNN

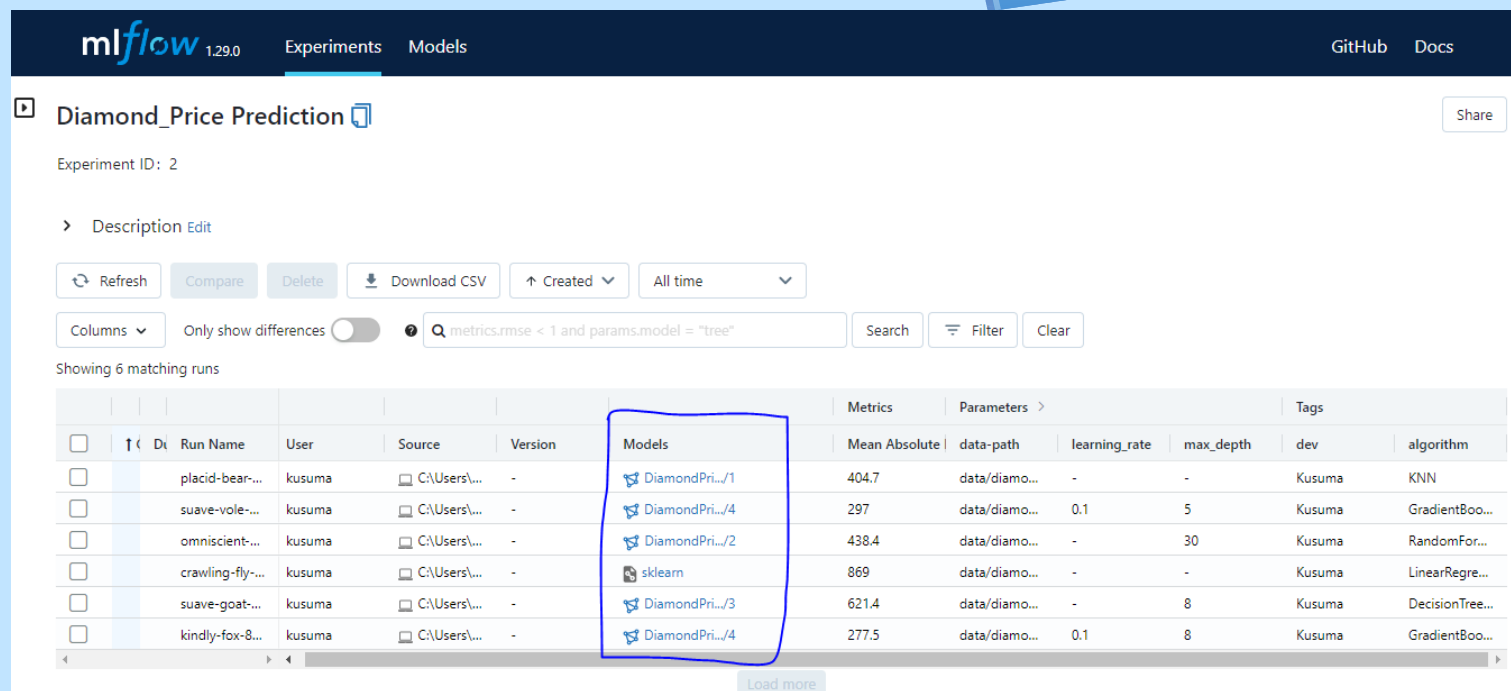
Load more

5. Select the experiments you want to register model

EX: Registered the experiments under a model name




The screenshot shows the mlflow 1.29.0 interface. The 'Experiments' tab is active, displaying the details for experiment 'placid-bear-744'. The experiment was created by 'kusuma' on '2022-09-21 00:51:19' with a 'Duration' of '23.4s' and a 'Status' of 'FINISHED'. The source is 'C:\Users\kusuma\anaconda3\lib\site-packages\ipykernel_launcher.py'. The lifecycle stage is 'active'. On the left, there are expandable sections for 'Description', 'Parameters (2)', 'Metrics (1)', 'Tags (2)', and 'Artifacts'. The 'Artifacts' section is expanded, showing a file named 'models'. A blue arrow points to the 'Register Model' button in the bottom right corner of the experiment details panel.



The screenshot shows the mlflow 1.29.0 interface. The 'Models' tab is active, displaying a list of registered models for the 'Diamond_Price Prediction' experiment. The experiment ID is '2'. The interface includes buttons for 'Refresh', 'Compare', 'Delete', 'Download CSV', and a filter dropdown set to 'All time'. A search bar contains the query 'metrics.rmse < 1 and params.model = "tree"'. The table shows 6 matching runs. A blue box highlights the 'Models' column, which lists the registered model names for each run.

	Run Name	User	Source	Version	Models	Metrics	Parameters	Tags			
						Mean Absolute	data-path	learning_rate	max_depth	dev	algorithm
<input type="checkbox"/>	placid-bear-...	kusuma	C:\Users\...	-	DiamondPri.../1	404.7	data/diamo...	-	-	Kusuma	KNN
<input type="checkbox"/>	suave-vole-...	kusuma	C:\Users\...	-	DiamondPri.../4	297	data/diamo...	0.1	5	Kusuma	GradientBoo...
<input type="checkbox"/>	omniscient-...	kusuma	C:\Users\...	-	DiamondPri.../2	438.4	data/diamo...	-	30	Kusuma	RandomFor...
<input type="checkbox"/>	crawling-fly-...	kusuma	C:\Users\...	-	sklearn	869	data/diamo...	-	-	Kusuma	LinearRegre...
<input type="checkbox"/>	suave-goat-...	kusuma	C:\Users\...	-	DiamondPri.../3	621.4	data/diamo...	-	8	Kusuma	DecisionTree...
<input type="checkbox"/>	kindly-fox-8...	kusuma	C:\Users\...	-	DiamondPri.../4	277.5	data/diamo...	0.1	8	Kusuma	GradientBoo...

6. Check the registered models

 1.29.0

ExperimentsModels

GitHubDocs

Registered Models > DiamondPrice_predict

DiamondPrice_predict





Created Time: 2022-09-21 01:24:31Last Modified: 2022-09-21 01:28:08




> Description [Edit](#)

> Tags

> Versions

AllActive 0Compare

<input type="checkbox"/>	Version	Registered at	Created by	Stage	Description
<input type="checkbox"/>	 Version 1	2022-09-21 01:24:32		None	
<input type="checkbox"/>	 Version 2	2022-09-21 01:25:05		None	
<input type="checkbox"/>	 Version 3	2022-09-21 01:25:39		None	
<input type="checkbox"/>	 Version 4	2022-09-21 01:28:08		None	



7. Name the registered models with their respective functional name and assign a stage transaction for the model

The screenshot displays the MLflow web interface for managing models. The top navigation bar includes the MLflow logo (1.29.0), links to Experiments and Models, and GitHub/Docs links. The breadcrumb trail indicates the current view is for a registered model named 'DiamondPrice_predict', specifically 'Version 4'.

Key details for this model version include:

- Registered At:** 2022-09-21 01:28:08
- Source Run:** [kindly-fox-825](#)
- Last Modified:** 2022-09-21 01:28:08

The 'Stage' dropdown menu is highlighted with a red box, showing options to transition the model to 'Staging' (yellow button), 'Production' (green button), or 'Archived' (blue button). The current stage is 'None'.

The 'Tags' section is also highlighted with a red box, showing a table with columns 'Name', 'Value', and 'Actions'. The table is currently empty, displaying 'No tags found.' Below the table, there is an input field for 'algorithm' with the value 'GradientBoostingRegressi' and an 'Add' button.

The 'Schema' section is visible below the tags, showing a table with columns 'Name' and 'Type'. A message at the bottom states: 'No schema. See [MLflow docs](#) for how to include input and output schema with your model.'

8. Check the registered model and staging state

The screenshot shows the mlflow 1.29.0 interface. The top navigation bar includes 'mlflow 1.29.0', 'Experiments', and 'Models' (which is active). On the right, there are links for 'GitHub' and 'Docs'. Below the navigation bar, the breadcrumb 'Registered Models > DiamondPrice-Prediction' is shown. The main heading is 'DiamondPrice-Prediction'. Below this, the 'Created Time' is '2022-09-20 21:34:04' and the 'Last Modified' time is '2022-09-21 01:18:28'. There are links for 'Description' and 'Edit'. Below these, there is a 'Tags' section. The 'Versions' section is expanded, showing a list of versions with checkboxes, version names, registered times, and stage labels. The stages are 'None', 'Staging', 'Archived', and 'Production'.

Registered Models > DiamondPrice-Prediction

DiamondPrice-Prediction

Created Time: 2022-09-20 21:34:04 Last Modified: 2022-09-21 01:18:28

> Description Edit

> Tags

▼ Versions All Active 2 Compare

<input type="checkbox"/>	Version	Registered at	Created by	Stage	Description
<input type="checkbox"/>	✓ Version 4	2022-09-21 01:18:28		None	
<input type="checkbox"/>	✓ Version 3	2022-09-20 21:45:28		Staging	
<input type="checkbox"/>	✓ Version 2	2022-09-20 21:44:57		Archived	
<input type="checkbox"/>	✓ Version 1	2022-09-20 21:34:05		Production	

References:

<https://www.mlflow.org/docs/latest/index.html>

<https://github.com/bansalkanav>