

Wine Quality MLOps Lab - Screenshot Submission

Submitted by: Sami

Date: June 4, 2025

Model Versions (wine_quality model)

This screenshot shows the wine_quality model registered in Unity Catalog with multiple versions. Version 4 is promoted to production. This fulfills the requirement to demonstrate model versioning and promotion.

Catalog

Serverless Starter Warehouse Serverless S

Type to search...

My organization

workspace

default

wine_quality

information_schema

system

Delta Shares Received

samples

Catalog Explorer > workspace > default >

wine_quality

Serve this model!

Overview Details Permissions

Description

AI generate Add

About this model

Owner skamacho@gmail.com

Deployment job

To manage the model lifecycle, link a deployment job. A deployment job can automatically trigger steps such as evaluation, approval, and deployment each time a new model version is created. [Learn more.](#)

Connect deployment job

Tags

Add tags

Versions

Status	Version	Tags	Aliases	Deployme...	Active endpo...	Comment
✔	Version 4		@ production			
✔	Version 3		@ archived			
✔	Version 2					
✔	Version 1					

Activity log

Time	Version	Log
2025-06-04 13:30:00	Version 4	Model version 4 registered by user skamacho@gmail.com.
2025-06-04 13:29:05	Version 3	Model version 3 registered by user skamacho@gmail.com.
2025-06-04 13:26:46	Version 2	Model version 2 registered by user skamacho@gmail.com.
2025-06-04 13:25:47	Version 1	Model version 1 registered by user skamacho@gmail.com.

Experiments with AUC Scores

This screenshot displays the Experiments interface, where multiple model runs are tracked and compared. The auc metric is shown, allowing evaluation of model performance.

Experiments >

MLOps on Databricks (1)

Send feedback

Add Description

Permissions

MLflow 3 is now available! Try out the new features and provide feedback. [Learn more](#)

Runs

Models

Preview

Evaluation

Monitoring

Beta

Traces

metrics.rmse < 1 and params.model = "tree"

Time created

State: Active

Datasets

Sort: Created

Columns

Expand rows

Group by

	Run Name	Created	Dataset	Duration	Source	Models
<input type="checkbox"/>	xgboost_models	16 minutes ago	-	23.0s	MLOps o...	-
<input type="checkbox"/>	untuned_random_for...	17 minutes ago	-	2.5s	MLOps o...	workspace.default.wine_qu...
<input type="checkbox"/>	xgboost_models	20 minutes ago	-	25.3s	MLOps o...	-
<input type="checkbox"/>	untuned_random_for...	20 minutes ago	-	2.3s	MLOps o...	workspace.default.wine_qu...

Show more columns (17 total)

MLflow Run Details - Parameters and Metrics

This detailed view shows parameters (like n_estimators) and the auc metric (0.854). It confirms successful experiment logging and model registry linkage.

Experiments > /Users/skamacho@gmail.com/MLOps on Databricks (1) >

untuned_random_forest [Send feedback](#)

⋮

Reproduce Run

Model regist

Overview

Model metrics

System metrics

Traces

Evaluation results ⓘ

Artifacts

Description [✎](#)

No description

Details

Created at	Jun 04, 2025, 02:02 PM
Created by	skamacho@gmail.com
Experiment ID	3720338771827826 🔗
Status	🟢 Finished
Run ID	9b86c52d6a86420fabcf568668b49782 🔗
Duration	2.7s
Datasets used	—
Tags	Add tags
Source	📁 MLOps on Databricks (1)
Logged models	🔗 pyfunc
Registered models	🔗 workspace.default.wine_quality v5

Metrics (1)

🔍 Search metrics

Metric	Latest	Min	Max
auc	0.8540300975814177	0.8540300975814177	0.8540300975814177

Parameters (1)

🔍 Search parameters

Parameter	Value
n_estimators	10

MLflow Artifacts Tab

The Artifacts tab includes all necessary model files: MLmodel, conda.yaml, and python_model.pkl.

This confirms the model is fully saved and reproducible.

Experiments > /Users/skamacho@gmail.com/MLops on Databricks (1) >

untuned_random_forest [Send feedback](#)

Overview

Model metrics

System metrics

Traces

Evaluation results

Artifacts

random_forest_model

metadata

MLmodel

conda.yaml

python_env.yaml

python_model.pkl

requirements.txt

random_forest_model

Path: dbfs:/databricks/mlflow-tracking/3720338711827826/9086c52d6a86420fabcf568668b49782/artifacts/random_forest_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 (12)	
fixed_acidity (required)	double
volatile_acidity (required)	double
citric_acid (required)	double
residual_sugar (required)	double
chlorides (required)	double
free_sulfur_dioxide (required)	double
total_sulfur_dioxide (required)	double
density (required)	double
pH (required)	double
sulphates (required)	double
alcohol (required)	double

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:/9086c52d6a86420fabcf568668b49782/random_forest_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",
)
```

Make Predictions

Predict on a Pandas DataFrame:

```
import mlflow
logged_model = "runs:/9086c52d6a86420fabcf568668b49782/random_forest_model"

# 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))
```

Predict on a Spark DataFrame:

```
import mlflow
from pyspark.sql.functions import struct, col
logged_model = "runs:/9086c52d6a86420fabcf568668b49782/random_forest_model"

# 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)

# Predict on a Spark DataFrame.
df.withColumn("predictions", loaded_model(struct("col", df.columns)))
```

workspace.default.wine_quality, v5

Registered on Jun 04, 2025, 02:03 P

Feature Importance Visualization

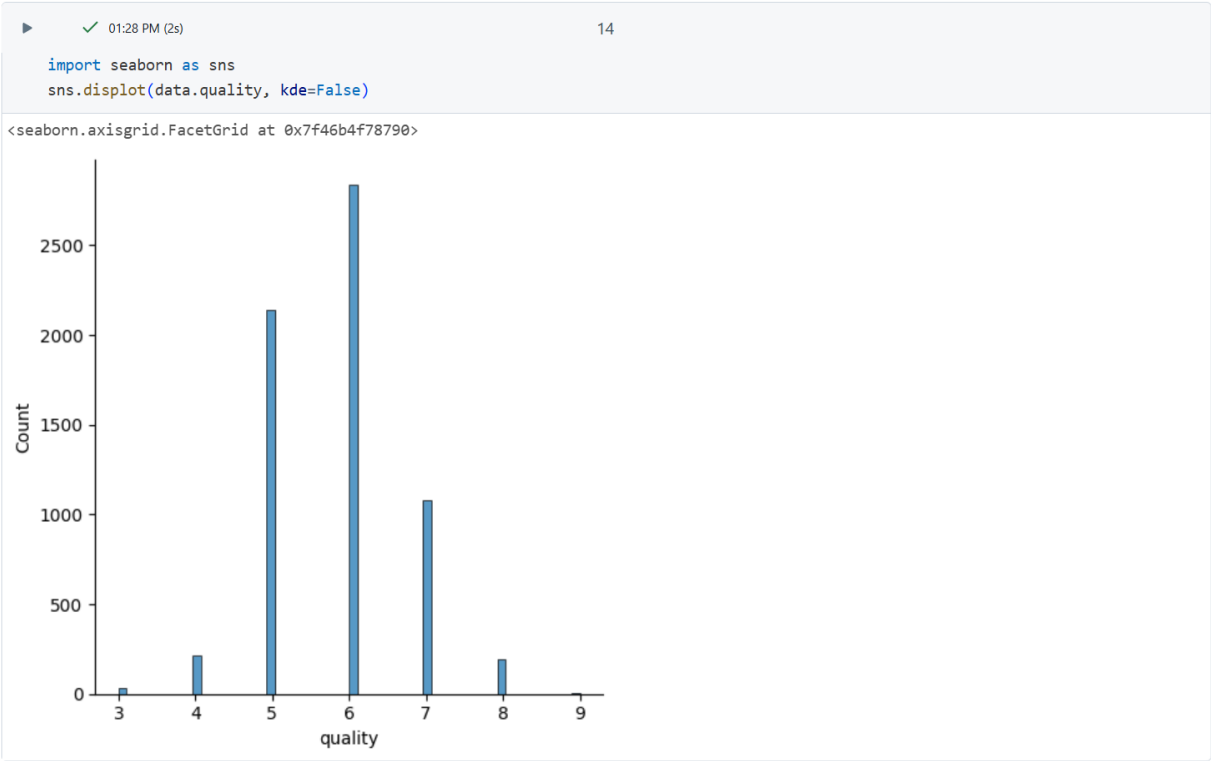
This screenshot highlights which features most influenced the model's decisions. Alcohol and density ranked highest.



Key Visualization - Wine Quality Distribution

This histogram supports the binary classification decision by showing how wine ratings cluster. A threshold of 7 was used to define 'high quality'.

Visualize Data



Looks like quality scores are normally distributed between 3 and 9.

Define a wine as high quality if it has quality ≥ 7 .

Confirmation

All required screenshots have been provided and labeled with descriptions. Each fulfills the assignment criteria.