ORACLE
University

Oracle Cloud Infrastructure Data Science Professional

Activity Guide

S1104293GC10

Learn more from Oracle University at ***education.oracle.com***

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

Trademark Notice

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

Third-Party Content, Products, and Services Disclaimer

This documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

1108212023

Table of Contents

Workshop Design and Setup: Data Science Project and Notebooks

Lab 01 Practices

Estimated Time: 40 mins

Setup Instructions for the Lab

In this practice, you will create the environment required for running this lab and the subsequent labs. This usually includes:

a. Creating a compartment

b. Creating a User

c. Adding a User to a User Group

d. Creation of a Dynamic Group

e. Creation of policies that allows User Groups and Dynamic Groups permissions to conduct operations in a tenancy, usually a user compartment

Creating a Compartment

a. From the navigation menu**,** select **Identity** under Identity & Security.

b. Click **Compartments**.

c. Click **Create Compartment**.

d. Enter the following in the dialog box:

- **Name**: Provide a name.
- **Description:** Provide a suitable description.
- Click **Create.**

Creating a User

a. From the navigation menu**,** select **Identity** under Identity & Security.

b. Click **Users**.

c. Enter the following in the dialog box:

- **Name**: Provide a name.
- **Description:** Provide a suitable description.
- **Email**: Optional
- **Confirm Email**: Provide the same email ID.
- Click **Create.**

Adding a User to a User Group

a. From the navigation menu**,** select **Groups** under Identity & Security.

b. Click **Create Group**.

c. Enter the following details in the dialog box:

- **Name**: <DataScienceGroup>
- **Description**: Provide a suitable description.

d. Add the user created to this group.

Create Dynamic Group

a. From the navigation menu**,** select **Dynamic Groups** under Identity & Security.

b. Click **Create Dynamic Group**.

c. Enter the following details in the dialog box:

- **Name**: <DataScienceGroup>
- **Description**: Provide a suitable description.

d. Under matching rules, add the following rules:

```
ALL {resource.type='datasciencenotebooksession',
resource.compartment.id='<replace-with-your-compartment-ocid>'}

ALL {resource.type='datasciencejobrun',
resource.compartment.id='<replace-with-your-compartment-ocid>'}

ALL {resource.type='datasciencemodeldeployment',
resource.compartment.id='<replace-with-your-compartment-ocid>'}
```

Notes

- Replace with the OCID of the compartment where you will create your notebook session. You can retrieve the OCID of the compartment by copying the partially displayed OCID when you click your **Compartment** name.
- Refer to Dynamic Groups documentation if required: https://docs.oracle.com/en-us/iaas/Content/Identity/Tasks/managingdynamicgroups.htm

IAM Policies

Policies can be set at the tenancy level or compartment level. Policies set at the tenancy level will be applicable to all the compartment within the parent tenancy. For better isolation you will set IAM policies at the compartment level. If you are an administrator of your OCI account, you will have required permissions to create policy in your compartment. In case you are not an administrator, and your compartment was created by an administrator reach out to your administrator to make sure you can create a policy in your compartment.

Add Policies

a. From the navigation menu**,** select **Policies** under Identity & Security.

b. Click **Create Policy**, and fill out the dialog box with the following details:

- Name: <Data Science Project>
- Description: <Provide suitable Description>
- Compartment:<select-your-compartment>
- Select **Manual Editor**

c. Include the following statements

```
Allow group <GRPname> to manage data-science-family in
compartment <student compartment name>

Allow group <GRPname> to manage log-groups in compartment
<student compartment name>

Allow group <GRPname> to use log-content in compartment <student
compartment name>

Allow group <GRPname> to manage buckets in compartment <student
compartment name>

Allow group <GRPname> to manage objects in compartment <student
compartment name>

Allow group <GRPname> to manage objectstorage-namespaces in
compartment <student compartment name>
```

```
Allow service datascience to use virtual-network-family in
compartment <student compartment name>

Allow dynamic-group <DynamicGroup> to manage data-science-family
in compartment   <student compartment name>

Allow dynamic-group <DynamicGroup> to manage log-groups in
compartment <student compartment name>

Allow dynamic-group <DynamicGroup> to use log-content in
compartment <student compartment name>

Allow service datascience to manage log-groups in compartment
<student compartment name>

Allow service datascience to manage log-content in compartment
<student compartment name>
```

For example, if the name of your user group is DataScienceGroup and name of your compartment is my-compartment, the policy statement will look like,

```
Allow group DataScienceGroup to manage buckets in compartment
my-compartment
```

For example, if the name of your dynamic group is DataScienceDynamicGroup and name of your compartment is my-compartment, the policy statement will look like,

```
Allow dynamic-group DataScienceDynamicGroup to manage log-groups
in compartment my-compartment
```

d. Click **Create.**

Get Started

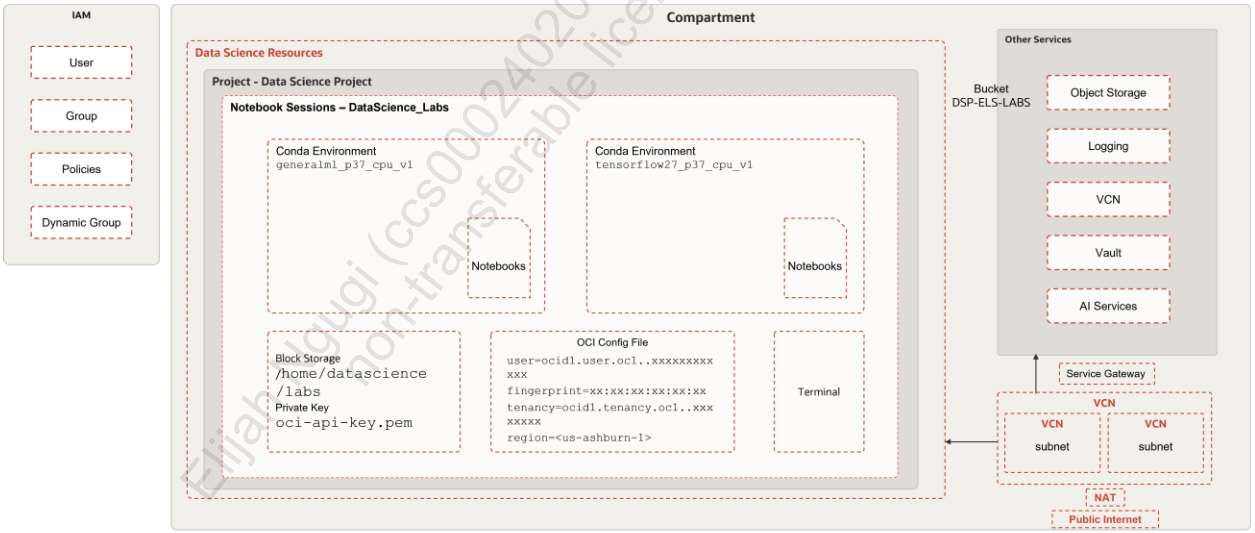
Overview

Data Science is a fully managed and serverless platform for data science teams to build, train, and manage machine learning models in Oracle Cloud Infrastructure.

Projects are collaborative workspaces for organizing and documenting Data Science assets, such as notebook sessions and models. In this workshop, each participant creates their own project. Data Science notebook sessions are interactive coding environments for building and training models. Notebook sessions come with many pre-installed open source, and Oracle-developed machine learning and data science packages. In this workshop, each user has a notebook session containing several notebooks. Condas are collections of specific libraries and other resources that simplify library management. They are used to run notebooks and deployed models.

When a notebook session is created or activated, the compute instance, block storage, VCN, and subnet are provisioned and configured. These resources can be changed by deactivating a notebook session, and then selecting a different configuration when activating the notebook session again.

Note: The size of the block storage can only be increased.



This lab will first guide you through creating a Data Science Project, creating a notebook session in your project, and then copying a repository to your newly created data science notebook session environment.

In this lab, you'll:

- Create your first Data Science project
- Create a Data Science notebook session
- Download course notebooks and data sets to your notebook session
- Install and use a conda environment

Prerequisites

- You must have access to the OCI Console and your Oracle Cloud account.

Create Your First Data Science Project

In this practice, you will learn to create a Data Science project.

Tasks

1. From the navigation menu, under **Analytics & AI**, select **Data Science** under Machine Learning.
2. In the left pane, under **List Scope**, note that, your allocated compartment is preselected.
3. Click **Create Project**, and fill out the dialog box with the following details:
   - **Name**: *<Data Science Project>*
   - **Description**: *<Provide suitable Description>*
4. Click **Create.**
5. The project gets created and listed on the Projects page.

Note: The project page lists project information along with resources contained in the project, such as notebook sessions, models, jobs, and model deployments.

Create and Open a Notebook Session

In this practice, you will learn how to create a notebook session and then open a notebook within it.

Tasks

1. From the navigation menu, select **Data Science** under Analytics & AI.
2. Click *< your -Data Science Project>*.
3. Click **Notebook Sessions** under Resources.
4. Click **Create Notebook Session**.
5. The Create Notebook Session dialog box opens. Fill out the details:
   - **Compartment**: <*your-compartment*>
   - **Name**: <*DataScience_Labs*>
   - **Compute Shape:** Click **Change shape**.
   - **Shape Series:** Intel
   - **Shape name: VM. Standard.E4. Flex**
   - Click **Select shape**.
   - **Number of OCPUs**:2
   - **Block storage size:** 50 GB
   - **Networking type:** Default Networking
6. Click **Create.**
7. Check the **View detail page** after clicking Create.
8. While the notebook session is being created, you will be taken to the detail page. The notebook session will take about 4 minutes to provision. When it is ready, the Open button will be enabled.
9. Click **Open** to open your notebook session.

Download Course Notebook and Data Sets to Your Notebook Session

After completing the earlier tasks (creating and opening a notebooks session), perform the following steps to download the notebook and data sets from the bucket:

Tasks

1. Download the required lab file in the zipped format from the Student Guides tab to your local storage.

2. Extract the files.

3. From the notebook session, upload the extracted files to `</home/datascience/>` folder.

Install and Use a Conda Environment

In this practice, you will learn to install and use conda environments in the Data Science notebook. Use the Conda Environment Explorer tool in the notebook session to install the required conda environment. After the environment is installed, you will run a notebook in a conda environment kernel.

Tasks

Before you can use a conda environment in your notebook session, you need to install it. You will install a prebuilt Data Science Conda Environment.

1. Install a conda environment for General Machine Learning for CPUs on Python 3.7.
   a. On the Launcher tab, click **Environment Explorer**.
   b. On the **Environment Explorer** tab, scroll down until you find the **General Machine Learning for CPUs on Python 3.8.** (If you see no results, use the Refresh button on the right side of the filter bar of Environment Explorer.)
   c. Click the **caret** on the right side and copy the `install` command.
   d. Go back to the Launcher tab and select **Terminal** to open a terminal window.
   e. **Paste the command** into the terminal window and press **Enter** to execute it. The command that you previously copied is:
   ```
   $ odsc conda install -s generalml_p38_cpu_v1
   ```
   f. You will receive a prompt related to what version number you want. Press **Enter** to select the default.
   g. It takes about 3-5 minutes for the conda package to be installed.
   h. You can confirm that the conda environment has been successfully installed by going back to the Launcher tab and see that a new kernel has been created with the slugname of the conda environment.
   i. Select the kernel and click **Create Notebook.**

   Note: If you continue, with other labs, retain the notebook session and follow the instructions in the respective lab documents, else follow the Purge instructions for deactivating the Notebook session.

Purge Instructions

Deactivating Notebook Sessions

1. From the navigation menu, under **Analytics & AI**, select **Data Science**.

2. You are now on the Projects page.

3. Click *<your-Data Science-project>*.

4. Click **Notebook sessions** under Resources**.**

5. Click the **Actions** icon under *<your-notebook-session>*, and then click **Deactivate.**

   Note: You can use the project and notebook session created in this lab for the subsequent labs. It is best practice to Deactivate the notebook session to save cost. You can Deactivate the notebook session, until required in the subsequent labs.

Machine Learning Life Cycle: Build, Train, Publish to Model Catalog, and Deploy a ML Model

Lab 02 Practices

Estimated Time: 40 minutes

Setup Instructions for the Lab

The setup to run this practice is already completed in Lab 01. Activate the notebook session to continue working with this Lab.

Steps to Activate and Open Notebook session:

a. In the navigation menu, navigate to Machine Learning under Analytics & AI and click **Data Science**.

b. Select your <*compartment-name*>.

c. Click the <*DataScienceProject*> created in Lab01.

d. Select the <*Notebook session*> used in Lab01.

e. Click the three dots on the right to open the Actions menu. Select **Activate**.

f. Click **Open** to open the notebook session.

   Note: In case, you are creating a new Project and Notebook session, refer to the instructions given in Lab1.

Get Started

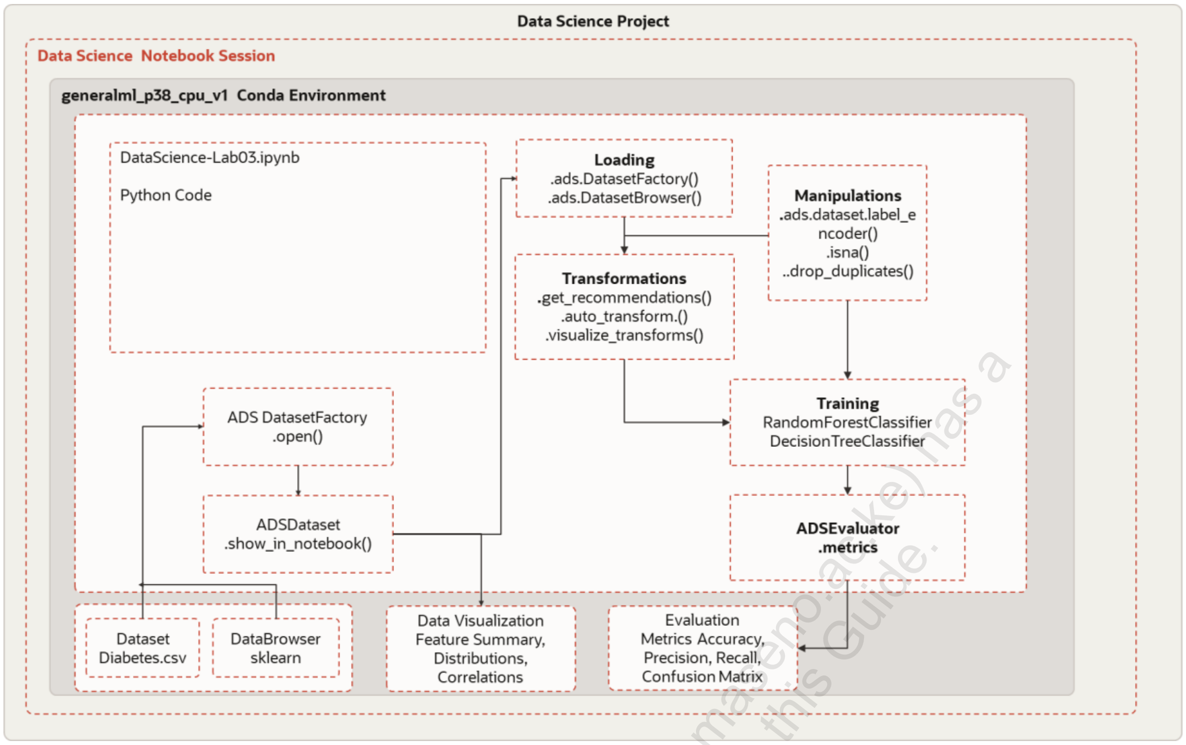
Overview

This lab will guide you through an entire Machine Learning Lifecycle Model. To start with, the environment is set up. This is followed by a practical example of how to build, train, and save a machine learning model. Then is the model catalog. The model catalog is a central repository of model artifacts and metadata. In the Oracle Cloud Infrastructure Data Science service, the model catalog lets data scientists share and control access to the models and deploy them as REST endpoints via the Data Science Model Deployment service. The model catalog is centered on documentation, reproducibility, and auditability — these are critical concepts in ML experimentation.

> **Note:** Model reproducibility and auditability are very real concerns of regulated industries that need to comply with audit rules and other traceability requirements.

The model that is in the model catalog is deployed as a REST endpoint using the Data Science Model Deployment service. A common use case for doing this is when an application needs to perform inference, but you want to keep the model separate from the application. You will use ADS to deploy the model from a notebook.

The dataset used in the lab practice, is a Binary Classification dataset used to predict employee retention. While this example uses a classifier, the steps learned with this exercise are applicable to many other types of machine learning models.



In this lab, you'll:

a. Open and explore DataScience_Lab02.ipynb.

b. Model Catalog the trained model and Deploy model for real-time inference.

Prerequisites

- Successful completion of Setup as described in Lab 01.
- Ensure to select the condo environment **General Machine Learning for CPUs on Python 3.8** following the similar instructions for setting conda environment in Lab 01.

Open and Explore DataScience_Lab02.ipynb

A notebook has been prepared containing all the necessary Python code to Build, Train and save the machine learning model for Binary Classification for Predicting Employee Attrition with ADS.

Tasks

1. In the file browser, navigate to the directory `</home/datascience/>`. This directory was created in Lab 01 when you uploaded the Lab files.

2. Open the notebook `DataScience_Lab02.ipynb` by double-clicking it. A new tab opens in the workspace on the right.

3. Notice in the upper right corner of the notebook tab, it displays the name of the conda environment being used by this notebook. Confirm that the name you see is the **slugname** of the **General Machine Learning for CPUs on Python 3.8 conda environment** (generalml_p38_cpu_v1).

4. Scroll through each cell in the notebook and read the explanations. When you encounter a code cell, execute it (using shift + enter) and view the results. For executable cells, the ""[]"" changes to a "[*]" while executing, then a number when complete "[1]". (If you run short on time, you can use the Run menu to run the remaining cells and the review the results.

When a model is saved with ADS, a bunch of metadata is automatically extracted about your model object and stored in the model catalog. You will review some of the key extracted metadata.

1. Select your <*DataScienceProject* >.

2. On the project details page, click **Models** option available under Resources**.**

3. Click the model <*model_name*> you just saved with the `DataScience_Lab2`.

4. On the left pane under Resources, you can see details <*data science project*>of Model provenance, Model taxonomy, Model introspection, and input and output data schema**.** Examine each of these.

5. Click **Model Provenance** under Resources**.** Model Provenance is a piece of documentation that helps you improve the model reproducibility and auditability.

6. Click the tab **Model training Source Code**. It gives the git reference to the training source code for the Model created.

7. Under Resources, click **Model Taxonomy.** Model taxonomy allows you to describe the model you are saving to the model catalog. You can use pre-allocated fields or create custom attributes.

8. Under Resources, click **Model introspection**. Model introspection captures the results of the tests run on the client side before the model is saved to the model catalog.

9. Under Resources, click **Input and Output data schema.** The input schema is the sequence of steps to pass as an input to an orchestration API to enable actual workflow execution. The output schema specifies what and how the data results are returned. This can be visualized from the input and output data schema tabs respectively.

10. A simple deployment configuration is created which results in an active deployment. This takes some time to become active.

11. To invoke your Deployed Model, under Resources, click **Invoking Your Model**.

12. Click **Copy** next to your model HTTP endpoint URI.

13. Go back to the notebook `DataScience_Lab02.ipynb` and paste the URI in the notebook cell (under the header Invoking your Model with the Predict from Console Model Deployment) where the URI variable is assigned.

14. Run the notebook again. You will successfully invoke a new model endpoint!

Purge Instructions

Deleting Model Deployments

1. From the navigation menu, select **Analytics & AI** and click on **Data Science**.
2. You are now on Projects page.
3. Select the **Data Science project** with the model deployment (to be deleted).
4. Click **Model deployments** under Resources section.
5. Click **Actions** icon, and then select **Terminate**.
6. When prompted for confirmation, provide the deployment name (as entered) and click **Delete**.

Deleting Models

1. From the navigation menu, select **Analytics & AI** and click on **Data Science**.
2. You are now on Projects page.
3. Select the Data Science project with the model (to be deleted).
4. Click **Models** under Resources section.
5. Click **Actions** icon, and then select **Delete**.
6. When prompted for confirmation, provide the model's name (as entered) and click **Delete**.

Deactivating Notebook Sessions

1. From the navigation menu, under **Analytics & AI**, click on **Data Science**.
2. You are now on Projects page.
3. Select the Data Science project with the notebook sessions (to be deactivated).
4. Click **Notebook sessions** under Resources section**.**

5. Click **Actions** icon, and then select **Deactivate**.

   Note: It is best practice to Deactivate the notebook session to save cost. You can Deactivate the notebook session, until required in the subsequent labs.

This completes the cleanup of the artifacts for this lab.

EDA Scenarios and Data Cleaning: Data Preprocessing and Visualization

Lab 03 Practices

Estimated Time: 45 minutes

Setup Instructions for the Lab

The setup to run this practice is already completed in Lab 01. Activate the notebook session to continue working with this Lab.

Steps to Activate and Open Notebook session:

a. From the navigation menu, navigate to Machine Learning under Analytics & AI and click **Data Science**.

b. Select your <*compartment-name*>.

c. Click the <*DataScienceProject*> created in Lab01.

d. Select the <*Notebook session*> used in Lab01.

e. Click the three dots on the right to open the Actions menu. Select **Activate**.

f. Click **Open** to open the notebook session.

Get Started

Overview

This lab will guide you will go through different Exploratory Data Analysis (EDA) scenarios. The most important element in any data science project is the data itself. It is extremely important that this data is as clean as possible. Before training the Machine Learning Model, this data preprocessing phase involves Data Transformation and Manipulation. This is the most time-consuming step in the entire lifecycle of a Machine Learning Model.

Preprocessing of data involves combining attributes or columns, Data imputation, Data cleaning, Dummy variables Encoding, Outlier Detection, Feature Scaling, Feature Engineering, Feature Selection and Feature Extraction. ADSDatasetFactory Object has rich features for automating data preprocessing. It provides features like auto_transform, suggest_recommendations, Visualize_transforms. Finally, the purge instructions are used to clean-up all the resources created during this lab.

The dataset used in the lab practice is Diabetes dataset.

In this lab, you'll:

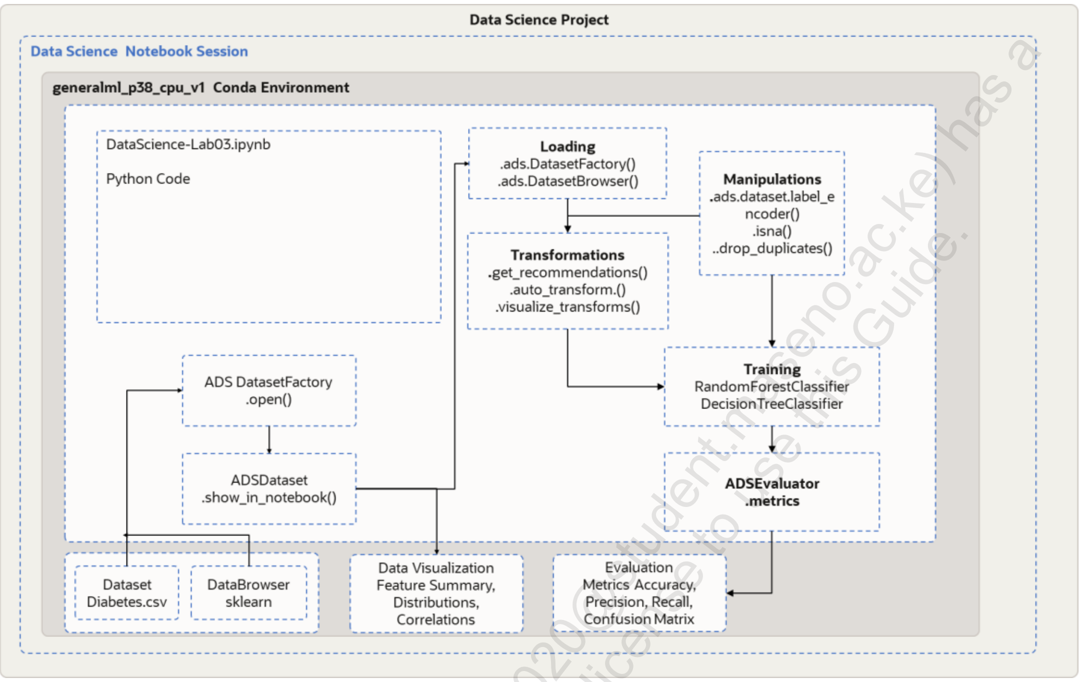
a. Open the Notebook and explore DataScience_Lab03.ipynb.

Prerequisites

- Successful completion of Setup as per Lab01.
- Ensure to select the condo environment **General Machine Learning for CPUs on Python 3.8** following the similar instructions for setting conda environment in Lab 01.

Open the Notebook and Explore `DataScience_Lab03.ipynb`

A notebook has been prepared containing all the necessary Python code for Exploratory Data Analysis. The Diabetes dataset is used, and the model is trained using Decision Tree Classifier and Random Forest.



Tasks

1. In the file browser, navigate to the directory `</home/datascience/>.` This directory was created in Lab 01 when you downloaded the bucket content in labs folder.

2. Open the notebook `DataScience_Lab03.ipynb` by double-clicking it. A new tab opens in the workspace on the right.

3. Notice in the upper right corner of the notebook tab, it displays the name of the conda environment being used by this notebook. Confirm that the name you see is the **slugname** of the **conda environment** is generalml_p38_cpu_v1.

4. Scroll through each cell in the notebook and read the explanations. When you encounter a code cell, execute it (using shift + enter), and view the results. For executable cells, the ""[]"" changes to a "[*]" while executing, then a number when complete "[1]". (If you run short on time, you can use the Run menu to run the remaining cells and the review the results.

Purge Instructions

Deactivating Notebook Sessions

1. From the navigation menu, under **Analytics & AI**, click on **Data Science**.

2. You are now on Projects page.

3. Select the Data Science project with the notebook sessions (to be deactivated).

4. Click **Notebook sessions** under Resources section**.**

5. Click **Actions** icon, and then select **Deactivate**.

   Note: It is best practice to Deactivate the notebook session to save cost. You can Deactivate the notebook session, until required in the subsequent labs.

This completes the cleanup of the artifacts for this lab.

Machine Learning Life Cycle: Automation Using Data Science Job

Lab 04 Practices

Estimate Time: 45 minutes

Setup Instructions for the Lab

The setup to run this lab is already completed in Lab 01. Activate the notebook session to continue working with this Lab.

Steps to Activate and Open Notebook session:

a. In the Navigation Menu, navigate to **Machine Learning** under **Analytics & AI** and click **Data Science**.

b. Select your <*compartment-name*>.

c. Click the ***<DataScienceProject>*** created in Lab01.

d. Select the ***<Notebook session>*** used in Lab01.

e. Click the **three dots** on the right to open the Actions menu. Select **Activate**.

f. Click **Open** to open the notebook session.

Get Started

Overview

Data Science Jobs enable custom tasks because you can apply any use case you have, such as data preparation, model training, hyperparameter tuning, batch inference, and so on.

Using jobs, you can:

- Run machine learning (ML) or data science tasks outside of your notebook sessions in JupyterLab.
- Operationalize discrete data science and machine learning tasks as reusable runnable operations
- Automate your typical MLOps or CI/CD pipeline
- Execute batches or workloads triggered by events or actions
- Get batch, mini batch, or distributed batch job inference

After the steps are completed, you can automate the process of data exploration, model training, and deploying and testing using jobs. With a single change in the data preparation or model training, experiments with hyperparameter tunings could be run as Job and independently tested.

Jobs are of two parts, a job and a job run:

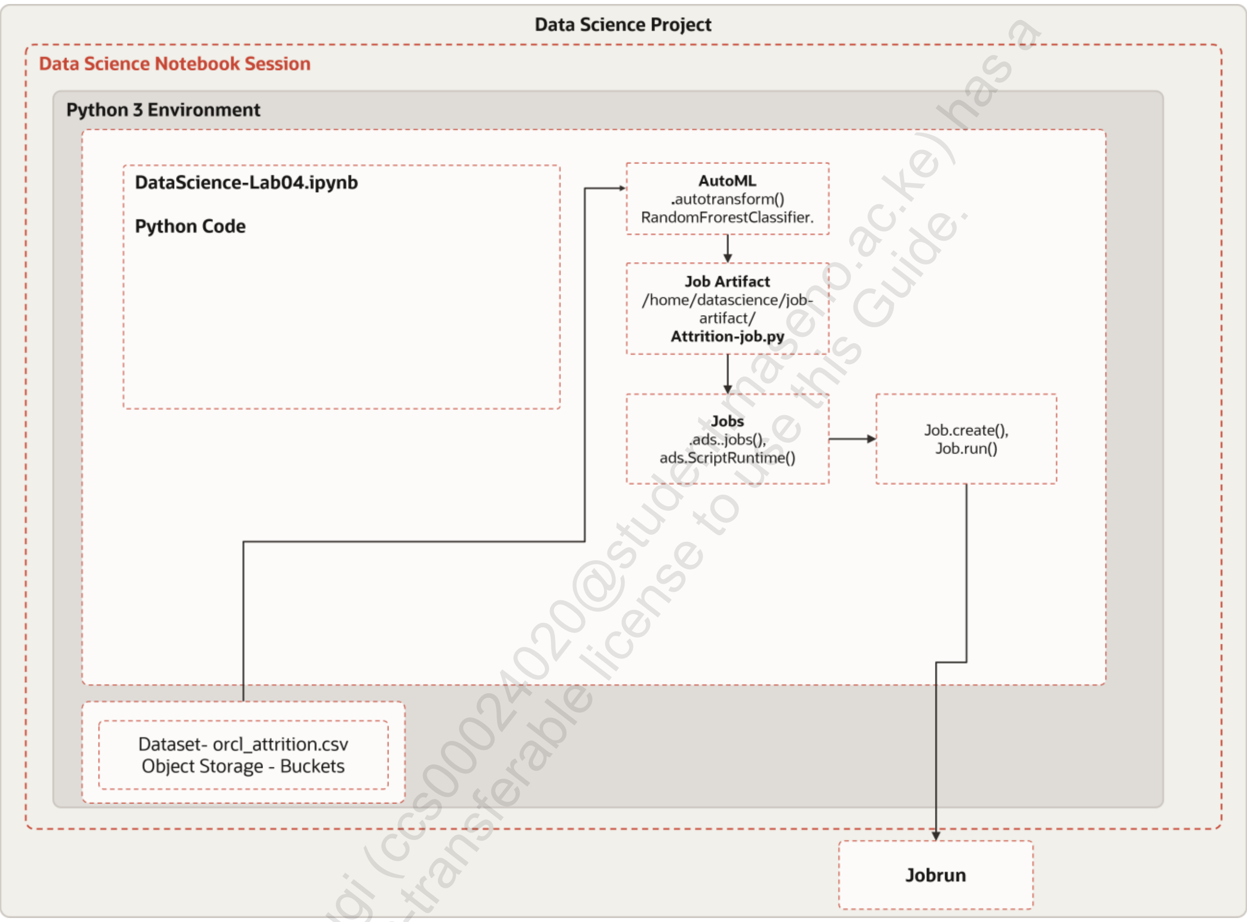
Job

A job is a template that describes the task. It contains elements like the job artifact that is immutable and can't be modified after it's uploaded to a job. Also, the job contains information about the Compute shapes the job runs on, logging options, block storage, and other options. You can add environment variables or CLI arguments to jobs to be unique or similar for all your future job runs. You can override these variables and arguments in job runs.

You can edit the Compute shape in the job and between job runs. For example, if you notice that you want to execute a job run on more powerful shape, you can edit the job Compute shape, and then start a new job run.

Job Run

A job run is the actual job processor. In each job run, you can override some of the job configuration, and most importantly the environment variables and CLI arguments. You can have the same job with several sequentially or simultaneously started job runs with different parameters. For example, you could experiment with how the same model training process performs by providing different hyperparameters.



In this lab, you'll:

a. Open and explore DataScience_Lab04.ipynb.

Prerequisites

- Successful completion of Setup given in Lab1.
- Ensure to select the condo environment **General Machine Learning for CPUs on Python 3.8**

Execute the `DataScience_Lab4.ipynb`

A notebook has been prepared containing all the necessary Python code to train and save the same machine learning model as in lab 3 but this time we will run the training script as a Data Science job.

Tasks

1. In the file browser, navigate to the directory </home/datascience/>. This directory was created in Lab 01 when you downloaded the bucket content in labs folder.
2. Open the notebook **`DataScience_Lab04.ipynb`** by double-clicking it. A new tab opens in the workspace on the right.
3. Notice in the upper right corner of the notebook tab, it displays the name of the **conda environment** being used by this notebook. Confirm that the name you see is the **slugname** of the **General Machine Learning for CPUs on Python 3.8** (`generalml_p38_cpu_v1`).
4. Scroll through each cell and read the explanations. When you encounter a code cell, execute it (using shift + enter) and view the results. For executable cells, the ""[ ]"" changes to a "[*]" while executing, then a number when complete "[1]". (If you run short on time, you can use the Run menu to run the remaining cells and the review the results.)

On successful execution of the notebook, a job is created and executed by creating a job run.

Purge Instructions

Deleting Model Deployments

1. Click the navigation menu, select **Analytics & AI** and click on **Data Science**.
2. You are now on **Projects** page.
3. Choose the **Data Science** project with the model deployment (to be deleted).
4. Click **Model deployments** under **Resources** section.
5. Click **Actions** icon, and then select **Terminate**.
6. When prompted for confirmation, provide the deployment name (as entered) and click **Delete**.

Deleting Models

1. Click the navigation menu , select **Analytics & AI** and click on **Data Science**.
2. You are now on **Projects** page.
3. Choose the **Data Science** project with the model (to be deleted).
4. Click **Models** under Resources section.
5. Click **Actions** icon, and then select **Delete**.
6. When prompted for confirmation, provide the model name (as entered) and click **Delete**.

Deleting Jobs

1. Click the navigation menu, select **Analytics & AI** and click on **Data Science**.
2. You are now on **Projects** page.
3. Choose the **Data Science** project with the model deployment (to be deleted).
4. Click **Jobs** under Resources section.
5. Click **Actions** icon, and then select **Delete**.

6. When prompted for confirmation, provide the **attrition-model-training-job** (as entered) and click **Delete**.

Deleting Notebook Sessions

1. Sign into **Oracle Cloud Infrastructure**.
2. Click the navigation menu , select **Analytics & AI** and click on **Data Science**.
3. You are now on **Projects** page.
4. Choose the **Data Science** project with the **notebook sessions** (to be deleted)
5. Click **Notebook sessions** under **Resources section.**
6. Click **Actions** icon, and then select **Terminate**.

This completes the cleanup of the artifacts on OCI tenancy/domain.

Explore Using OCI AI: Anomaly Detection Service

Lab 5 Practices

Estimated Time: 30 minutes

Setup Instructions for the Lab

The setup to run this lab is already completed in Lab 01.

a. Ensure that the following datasets are already downloaded and extracted to the locak Storage.

   - demo-training-data.csv
   - demo-testing-data.json

b. In addition to the policies mentioned in Lab 01, ensure the following policies are set at the compartment level.

```
Allow group GRP<name> to manage ai-service-anomaly-detection-
project in compartment <student compartment name>

Allow group GRP<name> to manage ai-service-anomaly-detection-data-
asset in compartment <student compartment name>

Allow group GRP<name> to manage ai-service-anomaly-detection-model
in compartment <student compartment name>
```

Get Started

Overview

Anomaly Detection is the identification of rare items, events, or observations in data that differ significantly from the expectation. This can be used for several scenarios like asset monitoring, maintenance, and prognostic surveillance in industries such as utility, aviation, transportation, and manufacturing. The core algorithm of our Anomaly Detection service is an Oracle-patented multivariate time-series anomaly detection algorithm originally developed by Oracle Labs and had been successfully used in several industries for prognosis analysis.

The Oracle Cloud Infrastructure Anomaly Detection will create customized Machine Learning models by taking the data uploaded by users, using the core algorithm to train the model, and hosted in the cloud to be ready for detection. Users can then send new data to the detection endpoints to get detected anomaly results.

The dataset used in the practice has training data and testing data. Training data has 10 signals with timestamp column and with 10000 observations. The training dataset is anomaly free and is uploaded to the bucket in a csv format. The testing data is a json file, with 10 signals and timestamp and has 100 observations.

In this lab, you'll:

a. Create anomaly detection project.

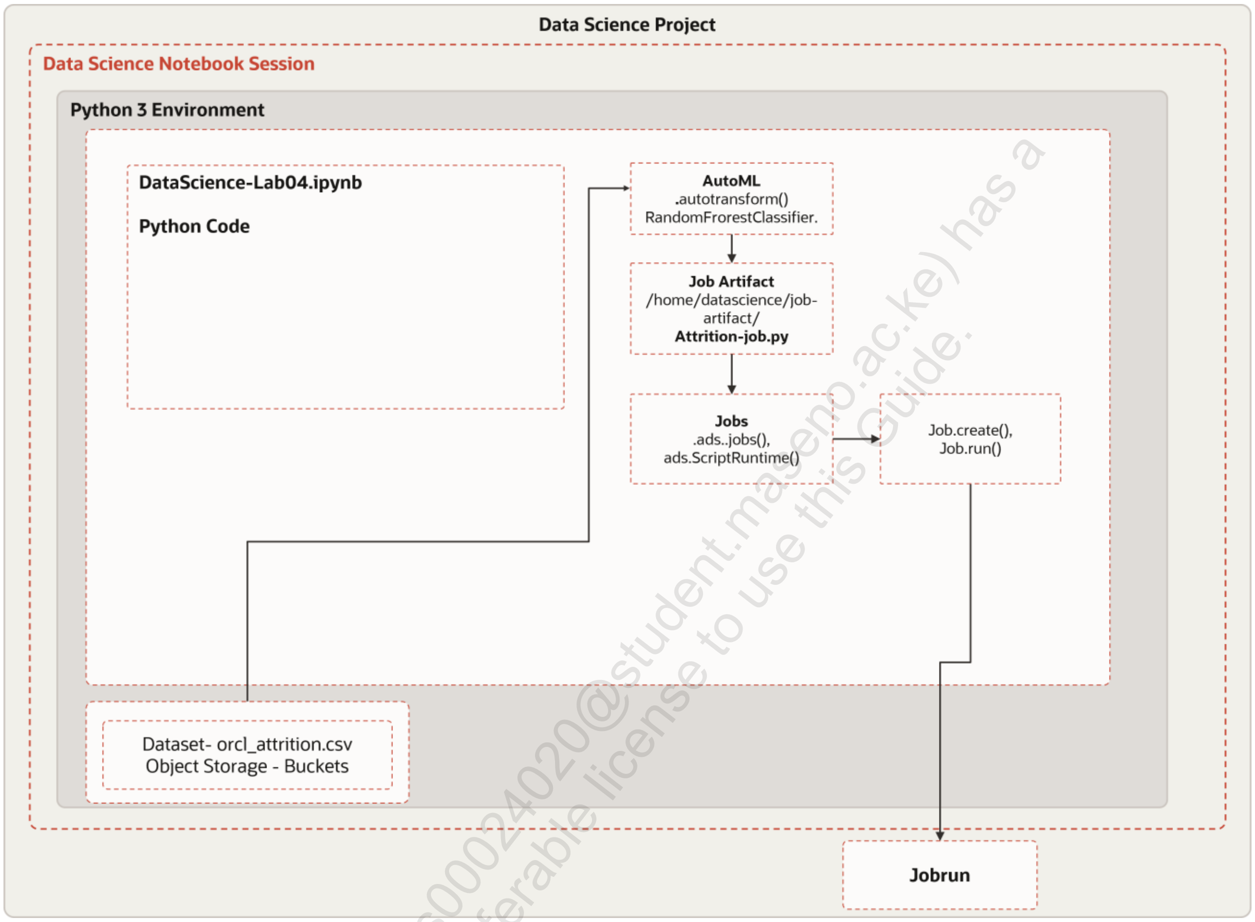
b. Train Anomaly Detection Model and Detect

Prerequisites

- Successful completion of Setup.

Create Anomaly Detection Project

You will learn to create an anomaly detection project, and upload data into Oracle Object Storage. You will also learn to create data asset to refer data in Oracle Object Storage.



Tasks

1. From the navigation menu, navigate to Analytics & AI, and click **Anomaly Detection** under AI Services.

2. Click **Create Project**. Fill in the following details in the dialog box:

   - **Compartment:** <*your-compartment-name*>
   - **Name:** Anomaly Detection (optional)
   - **Description:** Provide a suitable description.

   Note: Download demo-testing-data.json from DSP-LABS bucket.

3. Click **Create**. Once the project is created successfully, it will show up in the project pane.

4. Click *<your-project-name>*

5. Data Assets are uploaded through an Object Store. You will learn how to upload the data asset into an Object Store.
   a. From the navigation menu, Click Storage and select **Buckets**.
   b. Select *<your-compartment-name>.*
   c. Fill out the dialog box with the following details
      - **Bucket name:** Provide a name.
      - **Default Storage Tier:** Standard.

   d. Click **Create**.
   e. Bucket detail window would be visible. Click **Upload**.
   f. In the upload dialog box, that opens, fill in the following details.
      - **Storage Tier:** Standard
      - **Choose Files from your Computer:** Select demo-training-data.csv
   g. Click **Upload**.

6. From the navigation menu, navigate to Analytics & AI, and click **Anomaly Detection** under AI Services.

7. Click the **Project Name** which is already created.

8. Under Details, select Data Assets and click Create **Data Assets.** The Create Data Asset dialog box opens.

9. Fill in the dialog box with the following details:
   - **Compartment:** *<your-compartment-name>.*
   - **Name:** *<data-asset-name>* Optional
   - **Type*:*** Oracle Object Storage
   - **Choose bucket in Compartment:** *<the_bucket_created>*
   - **Training data:** demo-training-data.csv

10. Click **Create**.

    Note: After a few seconds, the data asset will be shown in the data asset main panel.

Train Anomaly Detection Model and Detect

Here you will learn to train the Anomaly Detection Model and make predictions with new data.

Tasks

1. Ensure the proper compartment and the project created in the previous step are selected.

2. Under Details, select Models and click **Create and Train Model.**

3. When the dialog box opens, select the radio button **Choose and existing data asset.**

4. Under Training Data Asset, under drop down menu, select <*data-asset-name*> created in the previous step.

5. Click **Next.** This takes us to the “Train Model” form with parameter selections. Fill in the dialog box with the following details:
   - **Select Model Compartment:** <*your-compartment-name*>.
   - **Name:** Optional
   - **Target False Alarm Probability:** 0.05
   - **Training Fraction Ratio:** 0.7
   - Click **Create**.

   Note: For this dataset, it takes 10 – 15 minutes to train the model.

6. Once model is trained, the status changes to Active. Click on the model created. Now under Anomalies, click **Detect Anomalies**.

7. The Detect Anomalies dialog box opens. Select the **demo-testing-data.json** file or drag and drop it or from the local filesystem and click **Detect**.

8. The detection result will return immediately, and you have the option to select the column to see related anomalies.

9. Use the drop wizard to select a column to see anomalies.

Purge Instructions

Delete Project

1. Click the navigation menu, select Analytics & AI and click on **Anomaly Detection** under AI Services.
2. You are now on **Projects** page.
3. Choose the **Anomaly Detection** project to be deleted.
4. Click the three dots on the right to open the Actions menu. Select **Delete**.
5. The delete project dialog box opens. When prompted fill in the project name and click **delete**.

Delete Bucket

1. Click the navigation menu, select Storage, and click on **Buckets**.
2. You are now on **Object Storage & Archive Storage** page.
3. Choose the **Bucket** to be deleted.
4. Ensure your compartment is chosen.
5. Click the three dots on the right to open the Actions menu. Select **Delete**.
6. The delete project dialog box opens. When prompted fill in the bucket name and click **delete**.