

author: Prash Medirattaa id: end\_to\_end\_machine\_learning\_with\_dataiku summary: This is a sample Snowflake Guide categories: Getting Started environments: web status: Published feedback link: <https://github.com/Snowflake-Labs/sfguides/issues> tags: Getting Started, Data Science, Data Engineering, Twitter

# End to End Machine learning with Snowflake and Dataiku

---

## Overview

Duration: 5

This Snowflake Quickstart introduces you to the using Snowflake together with Dataiku Cloud as part of a Machine learning project, and build an end-to-end machine learning solution. This lab will showcase seamless integration of both Snowflake and Dataiku at every stage of ML life cycle. We will also use Snowflake Data Marketplace to enrich the dataset.

## Business Problem

Will go through a **supervised machine learning** by building a binary classification model to predict if a lender will default on a loan. **LOAN\_STATUS (yes/no)** considering multiple features.

**Supervised machine learning** is the process of taking a historical dataset with KNOWN outcomes of what we would like to predict, to train a model, that can be used to make future predictions. After building a model we will deploy back to Snowflake for scoring by using Snowpark-java udf.

## Dataset

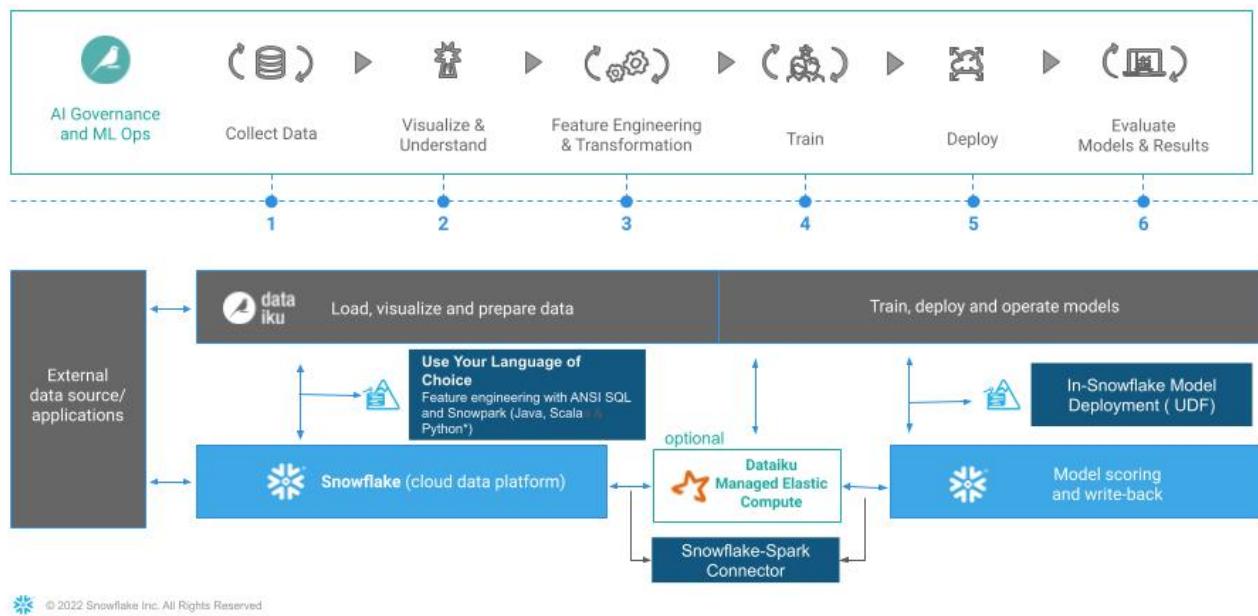
We will be exploring a financial service use of evaluating loan information to predict if a lender will default on a loan. The base data set was derived from loan data from the Lending Club.

In addition to base data, this will then be enriched with unemployment data from Knoema on the Snowflake Data Marketplace.

## What We're Going To Build

We will build a project. The project contains the input datasets from Snowflake. We'll build a data science pipeline by applying data transformations, enriching from Marketplace employment data, building a machine learning model, and deploying it to the Flow. We will then see how you can score the model against fresh data from Snowflake and automate

# Data Science with Snowflake + Dataiku



## Prerequisites

- Familiarity with Snowflake, basic SQL knowledge and Snowflake objects
- Basic knowledge Machine Learning
- Basic knowledge Python, Jupyter notebook for **Bonus**

## What You'll Need During the Lab

To participate in the virtual hands-on lab, attendees need the following:

- A [Snowflake free 30-day trial](#) **ACCOUNTADMIN** access
- Dataiku Cloud trial version via Snowflake's Partner Connect

## What You'll Build

Operational end-to-end ML project using joint capabilities of Snowflake and Dataiku from Data collection to deployment

- Create a Data Science project in Dataiku and perform analysis on data via Dataiku within Snowflake
- The analysis and feature engineering using Dataiku
- Create, run, and evaluate simple Machine Learning models in Dataiku, measure their performance and interpret
- Building and deploying Pipelines
- Creating Snowpark-Java UDF, and using it to score result on test dataset writing back to Snowflake
- Use cloning and time travel for test environment

## Setting up Snowflake

Duration: 5

- If you haven't already, register for a [Snowflake free 30-day trial](#)

- **Region** - Kindly choose which is physically closest to you
- **Snowflake edition** - Select the **Enterprise edition** so you can leverage some advanced capabilities that are not available in the Standard Edition.

# START YOUR 30-DAY FREE TRIAL

- Gain immediate access to the Data Cloud
- Enable your most critical data workloads
- Scale instantly, elastically, and near-infinitely across public clouds
- Snowflake is HIPAA, PCI DSS, SOC 1 and SOC 2 Type 2 compliant, and FedRAMP Authorized



Start your 30-day free Snowflake trial which includes \$400 worth of free usage

Choose your Snowflake edition\*

- Standard**  
A strong balance between features, level of support, and cost.
- Enterprise**  
Standard plus 90-day time travel, multi-cluster warehouses, and materialized views.
- Business Critical**  
Enterprise plus enhanced security, data protection, and database failover/fallback.

Choose your cloud provider\*



US West (Oregon)

Check here to indicate that you have read and agree to the terms of the [Snowflake Self Service On Demand Terms](#).

**GET STARTED**

Snowflake Trial

signup.snowflake.com

Home - The Lift at... Home - Workday Snowflake Compu... checklist\_SE/PE PE\_links Partner\_Document Training Dataiku certification Learning Other Bookmarks Reading List

# START YOUR 30-DAY FREE TRIAL

Start your 30-day free Snowflake trial which includes \$400 worth of free usage

**YOU'RE NOW SIGNED UP!**

An email to activate your account has been sent to [REDACTED] (it may take a few minutes to arrive).

Why did you signup for a Snowflake account today?

Company is considering Snowflake  
 Virtual hands-on lab or demo  
 Training or certification  
 Attending an in-person event  
 Personal learning and development  
 Other Tell us about your use case

**Submit**

Privacy | Terms  
© 2022 Snowflake Inc. All Rights Reserved

Privacy - Terms

- After registering, you will receive an **email** with an **activation** link and your Snowflake account URL. Kindly activate the account.

The screenshot shows a Gmail inbox with a single email from "Snowflake Computing <no-reply@snowflake.net>" titled "Activate Your Snowflake Account!". The email body contains the following text:

Hi Prash,

Congratulations on taking the first step to become a data-driven organization by signing up for Snowflake. Click the button below to activate your account.

[CLICK TO ACTIVATE](#)

Please note, your activation link is temporary and will expire in 72 hours. Once you activate your account, you can access it at <https://FEA74065.snowflakecomputing.com/console/login>.

Be sure to bookmark your login link to easily access your account going forward. If you experience any problems logging into your account or you forgot your username or password, please contact [support@snowflake.com](mailto:support@snowflake.com).

Best regards.

- After activation, you will create a **user name** and **password**. Write down these credentials



## Welcome to Snowflake!

**Prash Medirattaa**, please choose a username and password to get started

**Username**

A text input field containing the text "prash".

Username can contain only letters and numbers.

**Password**

A password input field showing six dots to represent the password characters.

Your password must be 8 - 256 characters and contain at least 1 number(s), 0 special character(s), 1 uppercase and 1 lowercase letter(s).

**Confirm password**

A password input field showing six dots to represent the password characters.

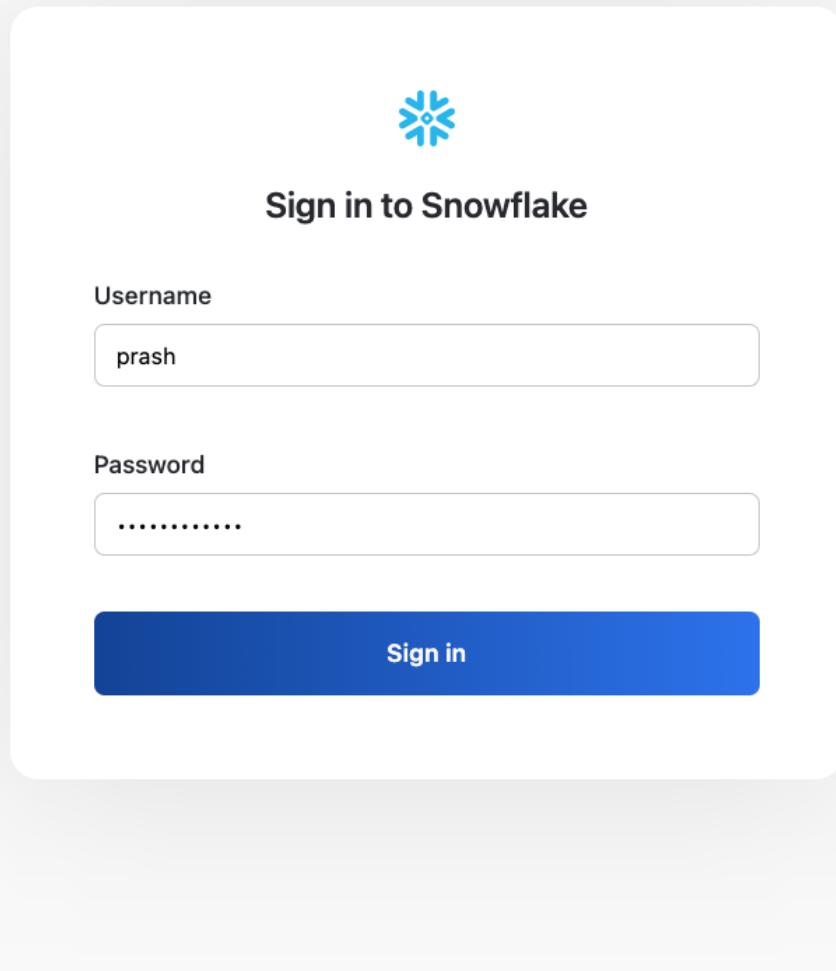
**Get started**

## Logging in Snowflake

Duration: 2

### Step 1

Log in with your credentials



### Bookmark this URL for easy, future access

Resize your browser window, so that you can view this guide and your web browser side-by-side and follow the lab instructions. If possible, use a secondary display dedicated to the lab guide.

### Step 2

Log into your Snowflake account. By default it will open up

The screenshot shows the Snowflake Learn interface. On the left, there's a sidebar with a navigation menu including 'Worksheets' (which is highlighted with a red box), 'Dashboards', 'Data', 'Marketplace', 'Activity', 'Admin', 'Help & Support', and 'Learn'. Below the menu is a 'Classic Console' link and a trial status message: '51 days left in trial' with a 'Upgrade' button. The main content area is titled 'Learn' and contains several sections: 'Snowflake Trial' (with a blue icon), 'Snowflake Data Marketplace' (with a blue icon), 'Foundations' (with a blue icon), 'Snowflake 101' (with a blue icon), 'Loading Data' (with a blue icon), and 'Snowflake 101' (with a blue icon). Each section has a brief description and a small icon.

If you have just created a free trial account, feel free to minimize or close and hint boxes that are looking to help guide you. These will not be needed for this lab and most of the hints will be covered throughout the remainder of this exercise

### Step 3

#### Create Worksheet

The screenshot shows the Snowflake Learn interface. The 'Worksheets' menu item in the sidebar is highlighted with a red box and a red arrow points to it from the bottom left. The main content area is titled 'Learn' and contains several sections: 'Snowflake Trial' (with a blue icon), 'Snowflake Data Marketplace' (with a blue icon), 'Foundations' (with a blue icon), 'Snowflake 101' (with a blue icon), and 'Snowflake 101' (with a blue icon). Each section has a brief description and a small icon.

### Step 4

## Adding a Worksheet

The screenshot shows the 'Worksheets' page in the Snowflake web interface. On the left, there's a sidebar with a user profile (PM, Prash Mediratta), navigation links like 'Dashboards', 'Data', 'Marketplace', etc., and a trial status message ('51 days left in trial'). The main area is titled 'Worksheets' and shows a list of recent worksheets. The list includes:

TITLE	VIEWS	UPDATED	ROLE
Tutorial 1: Sample queries on ...	1 hour ago	ACCOUNTADMIN	
Tutorial 2: Sample queries on ...	1 hour ago	ACCOUNTADMIN	
Tutorial 3: TPC-DS 10TB Com...	1 hour ago	ACCOUNTADMIN	
Tutorial 4: TPC-DS 100TB Co...	1 hour ago	ACCOUNTADMIN	
Tutorials	1 hour ago	ACCOUNTADMIN	

A red arrow points to the '+ Worksheet' button in the top right corner of the main area.

## Step 5

- Creating a new **Worksheet** and **Renaming** it to **Data Loading**

The screenshot shows the '2022-05-09 2:32pm - Snowflake' worksheet in the Snowflake web interface. The top navigation bar shows the URL 'app.snowflake.com/us-west-2/fza49556/wKgZPiELVmX/query'. The main area has a date/time dropdown ('2022-05-09 2:32pm') highlighted with a red box and a red arrow pointing to it. Below the dropdown is a search bar and a 'Tutorials' folder. The main workspace shows a single query:

```
1 select :datebucket(created), count(1) from table group by 1
```

At the bottom are 'Objects' and 'Query' tabs.



## Load data in Snowflake

Download the following .sql file that contains a series of SQL commands we will execute throughout this lab. You can either execute cell by cell commands from the sql file or copy the below code blocks and follow.

[Snowflake\\_Dataiku\\_ML.sql](#)

### Part 1 : Step 1 – Step 4

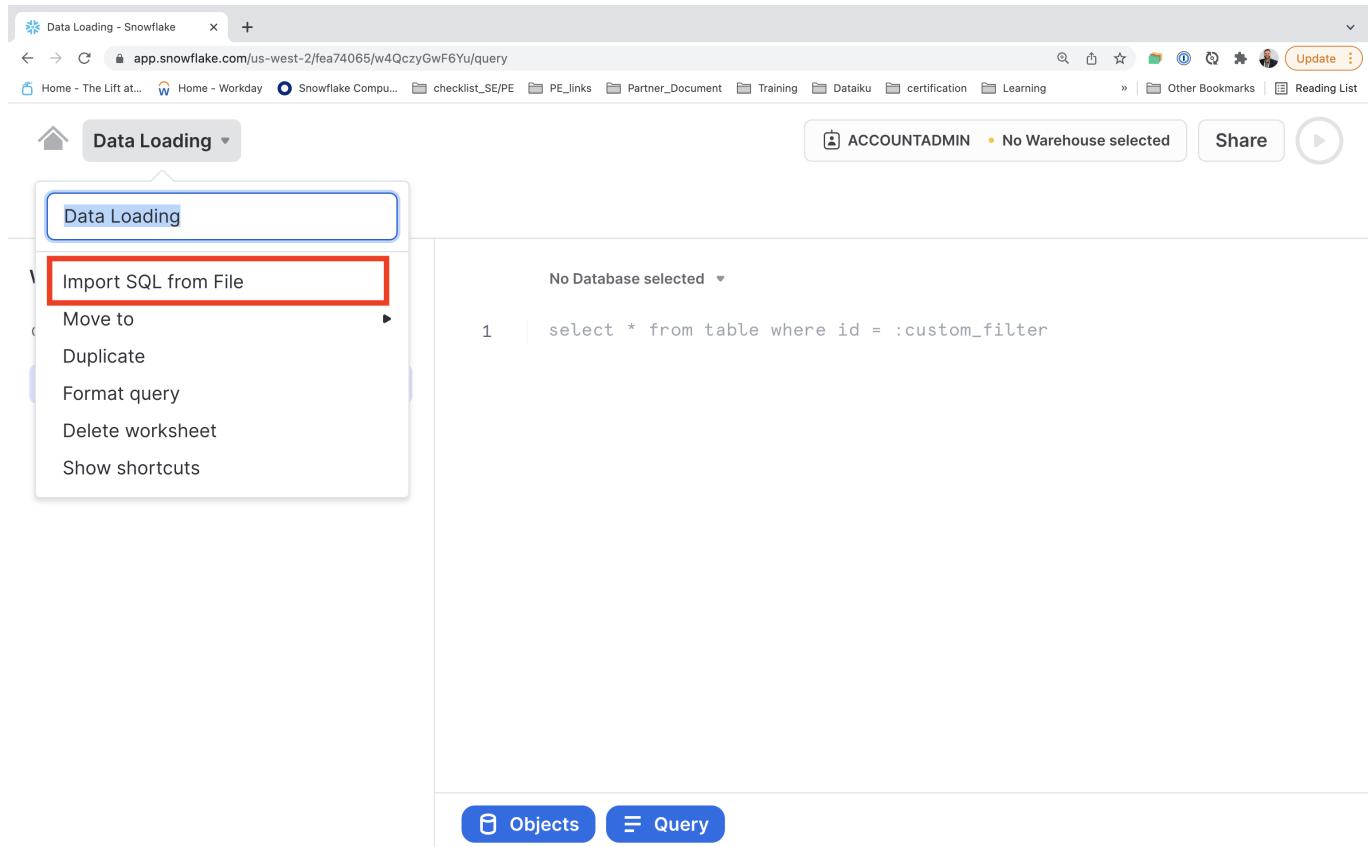
Creating database, Warehouse, loading dataset

### Part 2 : Step 5 – Step 8

Tapping Snowflake Marketplace dataset

After creating the worksheet in the last step we can import the sql file provided .

Importing **Sql to Worksheet** To ingest our script in the Snowflake UI, Import SQL from File.



The screenshot shows the Data Loading interface in the Snowflake web application. On the left, a context menu is open over a worksheet titled 'Data Loading'. The menu items include 'Import SQL from File' (which is highlighted with a red box), 'Move to', 'Duplicate', 'Format query', 'Delete worksheet', and 'Show shortcuts'. The main workspace shows a single line of SQL code: 'select \* from table where id = :custom\_filter'. At the bottom of the workspace, there are two tabs: 'Objects' (highlighted in blue) and 'Query'.

## Data Loading : Steps

Each step throughout the guide has an associated SQL command to perform the work we are looking to execute, and so feel free to step through each action running the code line by line as we walk through the lab. If you wish to run the code at once **Part 1 : Step 1 – Step 4** need to run first and then additional **Steps** are required before executing **Part 2 : Step 5 – Step 8** can be executed.

To execute this code, all we need to do is place our cursor on the line we wish to run and then either hit the "run" button at the top left of the worksheet or press **Cmd/Ctrl + Enter**

**Step 1 :** Virtual warehouse that we will use to compute with the **SYSADMIN** role, and then grant all privileges to the **ML\_ROLE**.

```
USE ROLE SYSADMIN;

CREATE OR REPLACE WAREHOUSE ML_WH

WITH WAREHOUSE_SIZE = 'XSMALL'

AUTO_SUSPEND = 120

AUTO_RESUME = true

INITIALLY_SUSPENDED = TRUE;
```

**Step 2 : Create Loan\_data table in the database**

```
USE WAREHOUSE ML_WH;

CREATE DATABASE IF NOT EXISTS ML_DB;

USE DATABASE ML_DB;

CREATE OR REPLACE TABLE loan_data (
    LOAN_ID NUMBER(38,0),
    LOAN_AMNT FLOAT,
    FUNDED_AMNT FLOAT,
    TERM VARCHAR(4194304),
    INT_RATE VARCHAR(4194304),
    INSTALLMENT FLOAT,
    GRADE VARCHAR(4194304),
    SUB_GRADE VARCHAR(4194304),
    EMP_TITLE VARCHAR(4194304),
    EMP_LENGTH_YEARS NUMBER(38,0),
    HOME_OWNERSHIP VARCHAR(4194304),
    ANNUAL_INC FLOAT,
    VERIFICATION_STATUS VARCHAR(4194304),
    ISSUE_DATE_PARSED TIMESTAMP_TZ(9),
    LOAN_STATUS VARCHAR(4194304),
    PYMNT_PLAN BOOLEAN,
    PURPOSE VARCHAR(4194304),
    TITLE VARCHAR(4194304),
    ZIP_CODE VARCHAR(4194304),
    ADDR_STATE VARCHAR(4194304),
    DTI FLOAT,
```

```

DELINQ_2YRS FLOAT,
EARLIEST_CR_LINE VARCHAR(4194304),
INQ_LAST_6MTHS FLOAT,
MTHS_SINCE_LAST_DELINQ FLOAT,
MTHS_SINCE_LAST_RECORD FLOAT,
OPEN_ACC FLOAT,
REVOL_BAL FLOAT,
REVOL_UTIL FLOAT,
TOTAL_ACC FLOAT,
TOTAL_PYMNT FLOAT,
MTHS_SINCE_LAST_MAJOR_DEROG FLOAT,
TOT_CUR_BAL FLOAT,
ISSUE_MONTH NUMBER(38,0),
ISSUE_YEAR NUMBER(38,0)
);

```

After running the cell above, we have successfully created a **loan data** table.

The screenshot shows a Dataiku DSS interface with the following details:

- Top Bar:** Includes 'Data Loading' dropdown, 'ML\_ROLE' and 'ML\_WH' buttons, 'Share' button, and a 'Run' button.
- Left Sidebar:** Shows 'Worksheets' and 'Databases' sections. 'Data Loading' is selected and highlighted with a blue background.
- Central Area:** Shows a code editor with the SQL query for creating the 'loan\_data' table. The code includes columns like LOAN\_ID, LOAN\_AMNT, FUNDING\_AMNT, TERM, INT\_RATE, and INSTALLMENT, all defined with the FLOAT data type.
- Bottom Status Bar:** Displays the message "Table LOAN\_DATA successfully created." followed by a timestamp "Updated 36 seconds ago".
- Bottom Right:** Includes 'Query Details' section showing "Query duration 183ms".

**Step 3 :**Creating a external stage to load the lab data into the table. This is done from a public S3 bucket to simplify the workshop. Typically an external stage will be using various secure integrations as described in

this [link](#).

```
CREATE OR REPLACE STAGE LOAN_DATA
url='s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/';
----- List the files in the stage
list @LOAN_DATA;
```

## Screen shot again after moving to new s3 folder

The screenshot shows the Snowflake Data Loading interface. On the left, there's a sidebar with 'Data Loading' selected. The main area has a query editor with the following code:

```
CREATE OR REPLACE STAGE LOAN_DATA
url='s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/';
----- List the files in the stage
list @LOAN_DATA;
```

Below the query editor is a results table with the following data:

	name	size	md5	last_modified
1	s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/cardholder_info.csv	2,564,528	0690be2f1c10820114b14e6041869285	Sun, 27 Mar 2
2	s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/loans_history_enriched.csv	6,532,878	24c38b16c0395c45841188350735944d	Thu, 5 May 2022
3	s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/merchant_info.csv	3,396,144	7b7c32dc68d161f149c2e09198d46f21	Sun, 27 Mar 2
4	s3://snowflake-corp-se-workshop/Summit_Snowflake_Dataiku/data/transactions.csv	24,450,797	46a54eab3c021cd468c655192b0f1d35-2	Sun, 27 Mar 2

On the right, there's a 'Query Details' panel showing a duration of 543ms and 4 rows.

## Step 4 :Cloning the data in the database

```
COPY INTO loan_data FROM @LOAN_DATA/loans_data.csv
FILE_FORMAT = (TYPE = 'CSV' field_optionally_enclosed_by='''',SKIP_HEADER =
1);
SELECT * FROM loan_data LIMIT 100;
```

Below is the snapshot of the data and it represents aggregation from various internal systems for lender information and loans. We can have a quick look and see the various attributes in it.

The screenshot shows the Data Loading interface. On the left, there's a sidebar with 'Data Loading' selected. The main area has tabs for 'Objects', 'Query' (which is selected), and 'Results'. Below these tabs is a table with four rows of loan data. To the right of the table is a 'Query Details' panel showing 'Query duration: 1.0s' and 'Rows: 100'. At the top right, there are buttons for 'ML\_ROLE', 'ML\_WH', 'Share', and a refresh icon.

LOAN_ID	LOAN_AMNT	FUNDED_AMNT	TERM	INT_RATE	INSTALLMENT
145509846	9,600	9,600	36	23.4	373.6
145616468	7,000	7,000	36	11.31	230.2
145521194	35,000	35,000	60	19.92	925.2
145439831	30,000	30,000	60	26.31	903.2

We have successfully loaded the data from **external stage** to snowflake.

----- End of Part 1 -----

## Step 5 : Time to switch to get **Konema Employment Data** from Snowflake Market place

We can now look at additional data in the Snowflake Marketplace that can be helpful for improving ML models. It may be good to look at employment data in the region when analyzing loan defaults. Let's look in the Snowflake Data Marketplace and see what external data is available from the data providers.

Lets go to home screen

This screenshot is identical to the one above, showing the Data Loading interface with a successful query execution. The table and 'Query Details' panel are the same, and the top right corner shows the same status indicators.

## Imp Note

1. Click Market place tab
2. Make Sure ACCOUNTADMIN role is selected
3. In search bar **Labor Data Atlas**

Snowflake Data Marketplace

Search: Labor Data Atlas

Results for Labor Data Atlas

Knoema  
Labor Data Atlas  
300+ public labor market datasets from the Eurostat, ILO, OECD, BLS, DOL, and other authoritative sources.

Knoema  
Civilian Labor Force in US  
US labor force by gender, age, race and ethnicity

Knoema  
Global Labor Data Pack  
ENTERPRISE PACK: 340M time series on national and subnational Labor related indicators from 150+ sources.

Personalized

51 days left in trial

Upgrade

Click on the tile with **Labor Data Atlas**.

**Labor Data Atlas**

Knoema Commerce Daily

The Labor Data Atlas is a curated collection of the most important, used, and high-quality datasets on the labor market and human resources on national and sub-national levels from a dozen of sources. It covers a breadth of topics such as labor force statistics, productivity, labor costs, labor mobility, and migration, as well as labor market forecasts. In addition to the main indicators such as unemployment and employment rates, labor force, and productivity, you will also find more detailed statistics such as labor market transitions, employee stock options, number of strikes and lockouts, and more.

Topics covered:

- Labor force
- Labor cost
- Productivity
- Labor mobility
- Labor market forecasts

Show More ▾

**Free**

Unlimited queries

Get Data

**K**  
Knoema

Knoema is the most comprehensive source of global decision-making data in the world. Our data technology solutions compress the complex data discovery to insight workflow from countless hours to...

Next click on the **Get Data** button. This will provide a pop up window in which you can create a database in your account that will provide the data from the data provider.

### Important : Steps

1. Change the name of the database to **KNOEMA\_LABOR\_DATA\_ATLAS**

2. Select additional roles drop down **PUBLIC**

3. Click **Get Data**

Labor Data Atlas  
300+ public labor market datasets from the Eurostat, ILO, OECD, BLS, DOL, and other authoritative sources.

**Get it for Free**

- ✓ Ready to query
- ✓ No additional storage cost
- ✓ Data updated daily

**Create Database**  
Query this data instantly in a new database. Takes up no storage in your account.

KNOEMA\_LABOR\_DATA\_ATLAS

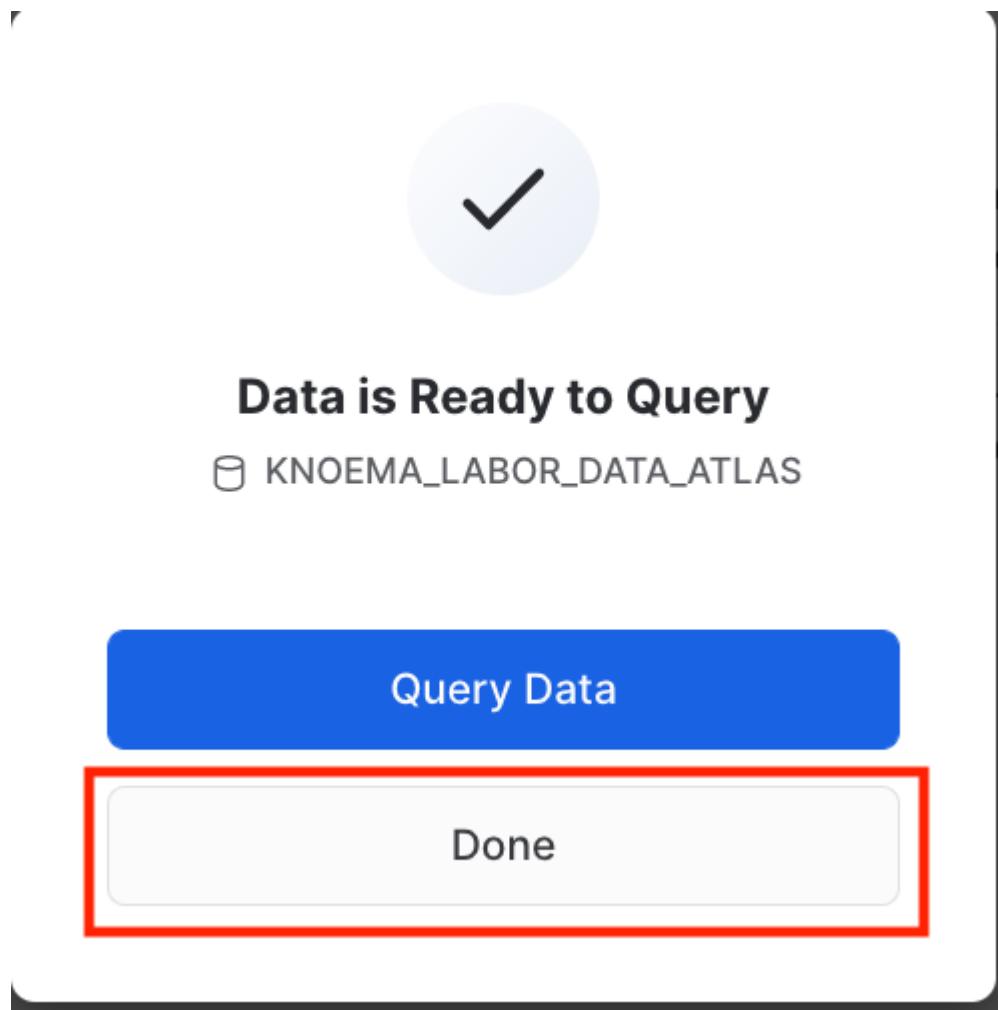
Add Roles ↗

Which roles, in addition to ACCOUNTADMIN, can access this database?

PUBLIC

Get Data

When the confirmation is provided click on done and then you can close the browser tab with the Preview App.



Other advantage of using Snowflake Data Marketplace does not require any additional work and will show up as a database in your account. A further benefit is that the data will automatically update as soon as the data provider does any updates to the data on their account. After done just to confirm the datasets are properly configured.

Click on Data tab **Database**

You should see **KNOEMA\_LABOR\_DATA\_ATLAS** and **ML\_DB**

NAME ↑	SOURCE	OWNER	CREATED	...
KNOEMA_LABOR_DAT...	Share	ACCOUNTAD...	2 minutes a...	...
ML_DB	Local	ML_ROLE	1 hour ago	...
SNOWFLAKE	Share	—	3 hours ago	...
SNOWFLAKE_SAMPLE...	Share	ACCOUNTAD...	3 hours ago	...

After confirming **Databases**. Lets go to **Worksheets tab** and open the **Data Loading** worksheet

The screenshot shows the Dataiku interface with the 'Worksheets' tab selected. The left sidebar includes options like Dashboards, Data, Marketplace, Activity, Admin, and Help & Support. The main area displays a list of worksheets with columns for Title, Viewed, Updated, and Role. A red arrow points to the 'Data Loading' worksheet.

TITLE	VIEWED	UPDATED	ROLE
Data Loading	57 minutes ago	52 minutes ago	ML_ROLE
Tutorial 1: Sample queries on...	3 hours ago	ACCOUNTADMIN	PM
Tutorial 2: Sample queries o...	3 hours ago	ACCOUNTADMIN	PM
Tutorial 3: TPC-DS 10TB Co...	3 hours ago	ACCOUNTADMIN	PM
Tutorial 4: TPC-DS 100TB C...	3 hours ago	ACCOUNTADMIN	PM
Tutorials	3 hours ago	ACCOUNTADMIN	PM

## Step 6 :Querying the Market Place dataset for some basic analysis

There are multiple datasets in **Labor Atlas dataset**. Lets try to find unemployment dataset in US to narrow down our search.

```
USE WAREHOUSE ML_WH;

USE DATABASE KNOEMA_LABOR_DATA_ATLAS;

SELECT *
FROM "LABOR"."DATASETS"
WHERE "DatasetName" ILIKE '%unemployment%'
AND "DatasetName" ILIKE '%U.S%';
```

The screenshot shows the results of the query in the Dataiku interface. The left sidebar shows the 'Data Loading' worksheet selected. The main area displays the query results in a table. The table has columns for DatasetId, DatasetName, Frequencies, FirstDate, and LastDate. The results show several datasets related to employment and unemployment rates.

DatasetId	DatasetName	Frequencies	FirstDate	LastDate
1	NCESEUR2019	A	1975-01-01	2021-01-01
2	BLSAV2	A, M	1976-01-01	2022-03-01
3	BSLA	A, M	1976-01-01	2020-10-01
4	BLS_LA	A, M	1976-01-01	2022-03-01
5	USUID2021MA	W	1967-01-02	2022-05-02
6	USUID2017Sep	W	1967-01-02	2021-03-08
7	USUNEMP2019	M	1948-01-01	2022-04-01

Amazing! isn't we have successfully tapped into live data collection of the most important, used, and high-quality datasets on the labor market and human resources on national and sub-national levels from a dozen of sources.

We can find answers such as what is the number of initial claims for unemployment insurance in the US over time?

```
SELECT * FROM "LABOR"."USUID2017Sep" WHERE "Region Name" = 'United States' AND
"Indicator Name" = 'Initial Claims' AND "Measure Name" = 'Value' AND
"Seasonal Adjustment Name" = 'Seasonally Adjusted' ORDER BY "Date";
```

The screenshot shows a Dataiku DSS interface. On the left, there's a sidebar with 'Data Loading' selected. The main area has a title 'KNOEMA\_LABOR\_DATA\_ATLAS'. A code editor window displays the following SQL query:

```
186
187 SELECT * FROM "LABOR"."USUID2017Sep" WHERE "Region Name" = 'United States' AND
188 "Indicator Name" = 'Initial Claims' AND "Measure Name" = 'Value' AND
189 "Seasonal Adjustment Name" = 'Seasonally Adjusted' ORDER BY "Date";
190
191
```

Below the code editor is a results table with the following data:

	Region	Region Name	Region Note	RegionId	Indicator	Indicator Name	Indicator Unit	Indicator Note
1	US	United States	null	US	KN.001	Initial Claims	Number	null
2	US	United States	null	US	KN.001	Initial Claims	Number	null
3	US	United States	null	US	KN.001	Initial Claims	Number	null
4	US	United States	null	US	KN.001	Initial Claims	Number	null
5	US	United States	null	US	KN.001	Initial Claims	Number	null

Now for this exercise we are going to **Enrich** the **Loan dataset** we created earlier using **BLSLA** dataset

**Step 7 :Creating a KNOEMA\_EMPLOYMENT\_DATA marketplace data view to pivot the data for the different employment metrics to columns for easier consumption.**

```
USE DATABASE ML_DB;

CREATE OR REPLACE VIEW KNOEMA_EMPLOYMENT_DATA AS (
SELECT *
FROM (SELECT "Measure Name" MeasureName, "Date",
"RegionId" State,
AVG("Value") Value
FROM "KNOEMA_LABOR_DATA_ATLAS"."LABOR"."BLSLA" WHERE "RegionId" is
not null
and "Date" >= '2018-01-01' AND "Date" < '2018-12-31' GROUP BY
"RegionId", "Measure Name", "Date")
PIVOT(AVG(Value) FOR MeasureName
IN ('civilian noninstitutional population', 'employment', 'employment-
population ratio',
'labor force', 'labor force participation rate', 'unemployment',
'unemployment rate')) AS
p (Date, State, civilian_noninstitutional_population, employment,
employment_population_ratio,
labor_force, labor_force_participation_rate, unemployment,
unemployment_rate)
);
```

```
SELECT * FROM KNOEMA_EMPLOYMENT_DATA LIMIT 100;
```

The screenshot shows a Data Loading interface with a query editor and a results table.

**Query Editor:**

```

1 USE DATABASE ML_DB;
2 CREATE OR REPLACE VIEW KNOEMA_EMPLOYMENT_DATA AS (
3   SELECT *
4     FROM (SELECT "Measure Name" MeasureName, "Date",
5            "RegionId" State,
6            AVG("Value") Value
7           FROM "KNOEMA LABOR DATA ATLAS"."LABOR"."BLSLA" WHERE "RegionId" is not null
8         and "Date" >= '2018-01-01' AND "Date" < '2018-12-31' GROUP BY "RegionId", "Measure Name", "Date")
9   PIVOT(AVG(Value) FOR MeasureName
10    IN ('civilian noninstitutional population', 'employment', 'employment-population ratio',
11        'labor force', 'labor force participation rate', 'unemployment', 'unemployment rate')) AS
12      p (Date, State, civilian_noninstitutional_population, employment, employment_population_ratio,
13          labor_force, labor_force_participation_rate, unemployment, unemployment_rate)
14  );
15  | SELECT * FROM KNOEMA_EMPLOYMENT_DATA LIMIT 100;
16

```

**Results Table:**

	DATE	STATE	CIVILIAN_NONINSTITUTIONAL_POPULATION	EMPLOYMENT	EMPLOYMENT_POPULATION_RATIO
1	2018-08-01	US_SC	4,009,500	113,851,612244898	56.45
2	2018-08-01	US_ME	1,104,768	6,666,68245614	61.65
3	2018-03-01	US_WV	1,457,161	39,762,797752809	50.45

**Query Details:**

- Query duration: 1.4s
- Rows: 100

**Step 8 :** Create a new table **UNEMPLOYMENT DATA** using the geography and time periods. This will provide us with unemployment data in the region associated with the specific loan.

```

CREATE OR REPLACE TABLE UNEMPLOYMENT_DATA AS

SELECT l.LOAN_ID, e.CIVILIAN_NONINSTITUTIONAL_POPULATION,
       e.EMPLOYMENT, e.EMPLOYMENT_POPULATION_RATIO, e.LABOR_FORCE,
       e.LABOR_FORCE_PARTICIPATION_RATE, e.UNEMPLOYMENT,
       e.UNEMPLOYMENT_RATE

  FROM LOAN_DATA l LEFT JOIN KNOEMA_EMPLOYMENT_DATA e

  on l.ADDR_STATE = right(e.state,2) and l.issue_month = month(e.date) and
  l.issue_year = year(e.date);

SELECT * FROM UNEMPLOYMENT_DATA LIMIT 100;

```

The screenshot shows the Dataiku Data Loading interface. At the top, there are tabs for Worksheets, Databases, and a search bar. Below the search bar, there are links for Tutorials and Data Loading, with Data Loading being the active tab. The main area displays a query result table titled "ML\_DB.PUBLIC \*". The table has columns: LOAN\_ID, CIVILIAN\_NONINSTITUTIONAL\_POPULATION, EMPLOYMENT, EMPLOYMENT\_POPULATION\_RATIO, and LABOR\_I. The results show data for six rows, with the first row highlighted in blue. To the right of the table, there is a sidebar with filters for LOAN\_ID, CIVILIAN\_NONINSTITUTIONAL\_P..., and EMPLOYMENT. The bottom of the interface shows a footer with "Row 1 / 100" and a "Chart" button.

LOAN_ID	CIVILIAN_NONINSTITUTIONAL_POPULATION	EMPLOYMENT	EMPLOYMENT_POPULATION_RATIO	LABOR_I
1	1,451,673	40,437,516,853,933	51.5	42,439,5955
2	21,749,117	148,069,787,934,186	61.7	153,587,7294
3	8,140,196	100,250,757,936,508	60.1	104,171,809
4	11,215,534	428,282,078,048,781	58.125	445,490,7365
5	4,621,564	101,020,748,344,371	65.1	103,901,0132
6	21,749,117	148,069,787,934,186	61.7	153,587,7294

----- End of Part 2 -----

**IMPORTANT: Database for Machine learning consumption will be created after connecting Snowflake with Dataiku using partner connect.**

## Connect Dataiku with Snowflake

Duration: 8

Verify that your user is operating under the Account Admin role.

To do this:

- Click your account name in the upper left-hand corner (if you are using the Classic Console this is top-right)
- Choose **Switch Role** from the drop-down list
- Click **ACCOUNTADMIN**

Screenshot of the Snowflake Worksheets interface. The left sidebar shows navigation options: Worksheets (selected), Dashboards, Data, Marketplace, Activity, Admin (highlighted with a red box and arrow), and Help & Support. Below the sidebar is a promotional banner for a 21-day trial, with an 'Upgrade' button. The main area is titled 'Worksheets' and features a 'Get started' section with a 'Import Worksheets' button.

Screenshot of the Snowflake Partner Connect interface. The left sidebar shows navigation options: Worksheets, Dashboards, Data, Marketplace, Activity, Admin, Usage, Warehouses, Resource Monitors, Users & Roles, Security, Billing, Contacts, Accounts (highlighted with a red box and arrow), and Help & Support. The main area is titled 'Partner Connect' and includes a search bar with the text 'dataiku'. A 'Data Science & Machine Learning' section displays a 'dataiku' tile with a brief description: 'End-to-end AI platform from data prep to AutoML and MLOps leveraging the power of Snowflake'.

- Click on the **Dataiku** tile. This will launch the following window, which will automatically create the **connection parameters** required for Dataiku to connect to Snowflake.

Snowflake will create a dedicated database, warehouse, system user, system password and system role, with the intention of those being used by the Dataiku account.

## Connect to Dataiku

Dataiku requires the following information to create your new trial account: first name, last name, and email address.

In order to configure the connection with Snowflake, the following objects will be created in your Snowflake account:

Database	<b>PC_DATAIKU_DB</b>
Warehouse	<b>PC_DATAIKU_WH (X-Small)</b>
System User	<b>PC_DATAIKU_USER</b>
System Password	Autogenerated & Randomized
System Role	<b>PC_DATAIKU_ROLE</b> Role PUBLIC will be granted to the PC_DATAIKU_ROLE Role PC_DATAIKU_ROLE will be granted to the SYSADMIN role

Optional Grant ▾

[Close](#)

[Connect](#)



We'd like to use the **PC\_DATAIKU\_USER** to connect from Dataiku to Snowflake, and use the **PC\_DATAIKU\_WH** when performing activities within Dataiku that are pushed down into Snowflake.

Note that the user password (which is autogenerated by Snowflake and never displayed), along with all of the other Snowflake connection parameters, are passed to the Dataiku server so that they will automatically be used for the Dataiku connection. **DO NOT CHANGE THE PC\_DATAIKU\_USER password**, otherwise Dataiku will not be able to connect to the Snowflake database.

Click on **Connect**. You may be asked to provide your first and last name. If so, add them and click Connect. Your partner account has been created. Click on **Activate** to get it activated.

## Your partner account has been created

Dataiku has successfully created your new account. Click Activate to go to Dataiku's website to finish the activation process and start loading data.

[Finish Later](#)

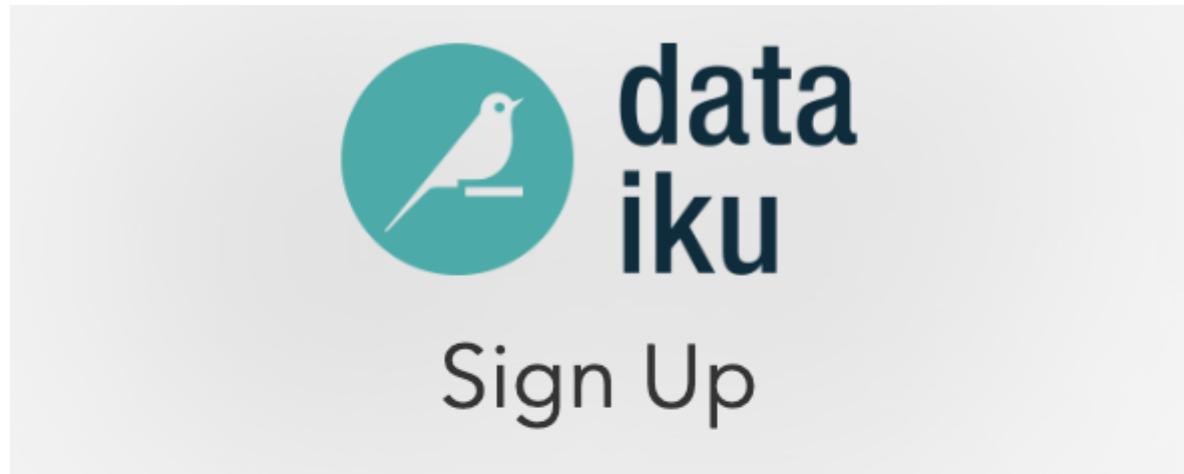
[Activate](#)

This will launch a new page that will redirect you to a launch page from Dataiku. Here, you will have two options:

1. Login with an existing Dataiku username
2. Sign up for a new Dataiku account

We assume that you're new to **Dataiku**, so ensure the "Sign Up" box is selected, and sign up with either GitHub, Google or your email address and your new password.

Click sign up. <<NOTE: ADD INSTRUCTION FOR EXISTING IN ADDITION>>



**Sign up with GitHub**



**Sign up with Google**

or



**dataikulab@gmail.com**



**your password**

By signing up, you agree to our [terms of service](#) and [privacy policy](#).

**SIGN UP >**

When using your email address, ensure your password fits the following criteria:

1. **At least 8 characters in length**
2. **Should contain: Lower case letters (a-z)**

**Upper case letters (A-Z)**

**Numbers (i.e. 0-9)**

Upon clicking on the activation link, please briefly review the Terms of Service of Dataiku Cloud. In order to do so, please scroll down to the bottom of the page. Click on **I AGREE**



## Welcome to Dataiku Online

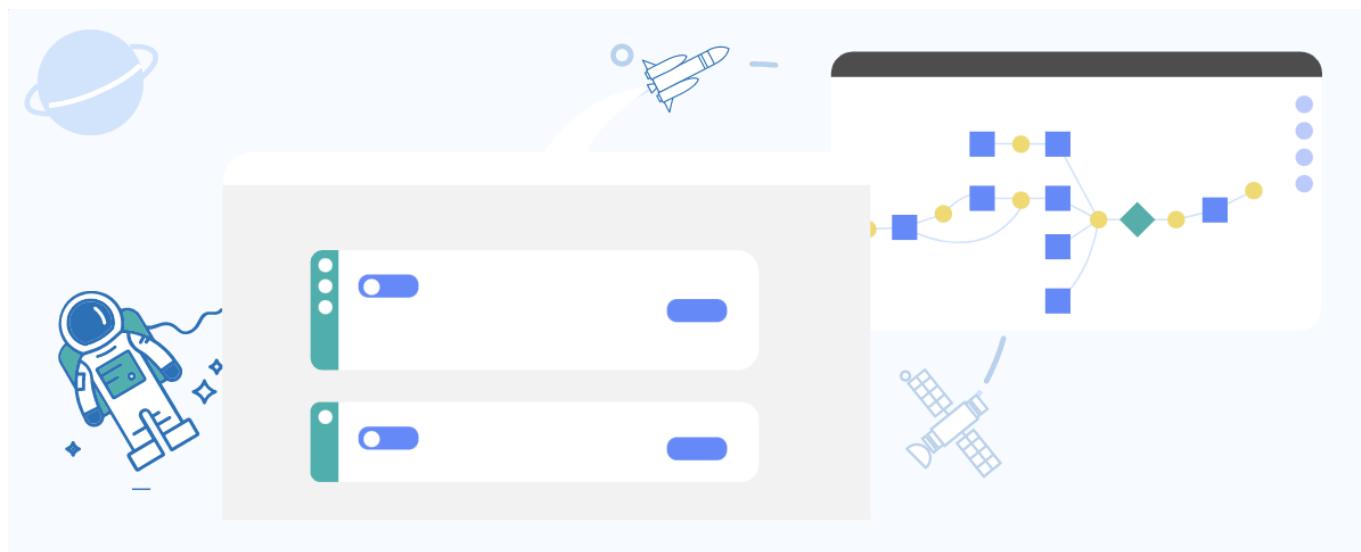
To continue using the service, scroll down and accept the terms of service.

Expand Contract	<a href="#">Download</a>
<input checked="" type="checkbox"/> I understand and agree to Terms of Service	

NEXT

Next, you'll need to complete your sign up information then click on **Start**.

You will be redirected to the Dataiku Cloud Launchpad site. Click **GOT IT!** to continue.



## Here is your Launchpad!

**This is where you can:**

Launch your **Dataiku DSS**

Invite **users**

Customize with **features**

Access your **subscription** information

**GOT IT!**

A screenshot of the Dataiku DSS interface. On the left is a vertical sidebar with various icons. The main area shows a project titled "FORTUITOUS VEGETATION". At the top right, there is a message: "15 days left before end of trial". Below the title, it says "Snowflake Partner Connect" and "Version: 10.0.5-stw-15". It shows "1 user" and "3 features". The status is "Running" with a green play button icon. There is a blue button labeled "OPEN DATAIKU DSS" and a three-dot menu icon. At the bottom, there is a button labeled "+ ADD A DATAIKU DSS".

You've now successfully set up your Dataiku trial account via Snowflake's Partner Connect. We are now ready to continue with the lab. For this, move back to your Snowflake browser.

### Database for Machine Learning

After connecting **Snowflake** to **Dataiku** via partner connect. We will clone the table created in **ML\_DB** to **PC\_DATAIKU\_DB** for the Dataiku consumption. Snowflake provides a very unique feature called **Zero Copy**

**Cloning** that will create a new copy of the data by **only making a copy of the metadata of the objects**.

This drastically speeds up creation of copies and also drastically reduces the storage space needed for data copies.

NAME ↑	SOURCE	OWNER	CREATED
KNOEMA_LABOR_DAT...	Share	ACCOUNTAD...	2 hours ago
ML_DB	Local	ML_ROLE	3 hours ago
PC_DATAIKU_DB	Local	ACCOUNTAD...	1 hour ago
SNOWFLAKE	Share	—	5 hours ago
SNOWFLAKE_SAMPL...	Share	ACCOUNTAD...	5 hours ago

You should see three database now **PC\_DATAIKU\_DB** is the system generated database created. Go back to **Worksheet** you are working and run below commands.

### Granting Privileges of **ML\_DB** to **PC\_Dataiku\_role**

```
grant all privileges on database ML_DB to role PC_Dataiku_role;
grant usage on all schemas in database ML_DB to role PC_Dataiku_role;
grant select on all tables in schema ML_DB.public to role PC_Dataiku_role;
grant select on all views in schema ML_DB.public to role PC_Dataiku_role;
```

### Cloning tables to DATAIKU Database before consuming it for Dataiku DSS

```
USE ROLE PC_DATAIKU_ROLE;
USE DATABASE PC_DATAIKU_DB;
USE WAREHOUSE PC_DATAIKU_WH;

--- cloning

CREATE OR REPLACE TABLE LOANS_ENRICHED CLONE ML_DB.PUBLIC.LOAN_DATA;
CREATE OR REPLACE TABLE UNEMPLOYMENT_DATA CLONE
ML_DB.PUBLIC.UNEMPLOYMENT_DATA;

SELECT * FROM LOANS_ENRICHED LIMIT 10;
```

After running above commands, we have created clones for the tables to be used for analysis. Kindly check **PC\_DATAIKU\_DB** you should have two datasets **LOANS\_ENRICHED** and **UNEMPLOYMENT\_DATA**

The screenshot shows the Dataiku DSS interface. On the left, a sidebar menu includes options like Worksheets, Dashboards, Data, Databases (selected), Marketplace, Activity, Admin, Help & Support, and Classic Console. The main area displays a tree view of databases: KNOEMA\_LABOR\_DATA\_ATLAS, ML\_DB, PC\_DATAIKU\_DB (selected), INFORMATION\_SCHEMA, PUBLIC, LOANS\_ENRICHED, UNEMPLOYMENT\_DATA, Views, Stages, Data Pipelines, Functions, Procedures, and SNOWFLAKE\_SAMPLE\_DATA.

**PC\_DATAIKU\_DB**

Database Details

**Privileges**

- SYSADMIN (Current Role) - CREATE SCHEMA, USAGE
- ACCOUNTADMIN - OWNERSHIP
- PC\_DATAIKU\_ROLE - CREATE SCHEMA, USAGE

Now lets move to Dataiku console for feature engineering, model building, Scoring and deployment.

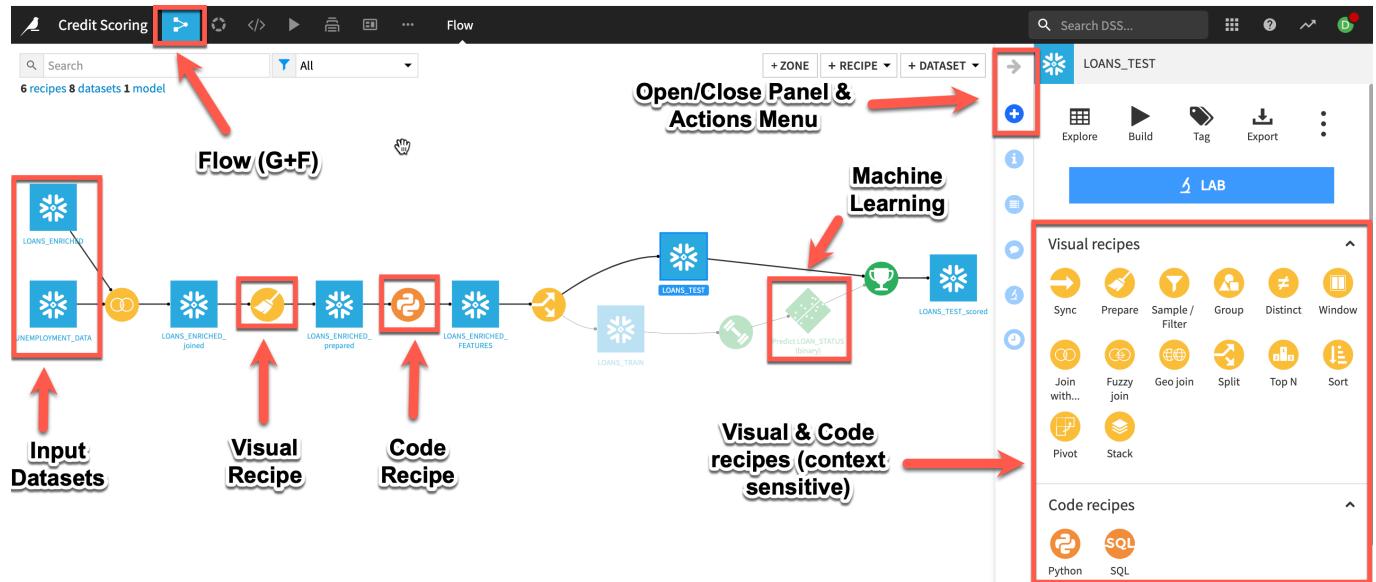
## Getting Started with a Dataiku Project

Duration: 2

Return to Dataiku Online and if you haven't already click on **OPEN DATAIKU DSS** from the Launchpad to start your instance of Dataiku DSS

The screenshot shows the Dataiku Launchpad. A vertical sidebar on the left contains icons for Home, Projects, DSS, Data, Machine Learning, and Settings. The main area displays a project titled "FORTUITOUS VEGETATION". It shows a "Snowflake Partner Connect" component, which is "Running". The component details are: Version: 10.0.5-stw-15, 1 user, 3 features. There are buttons for "OPEN DATAIKU DSS" and "ADD A DATAIKU DSS". A message at the top right says "15 days left before end of trial".

At the end of the lab, the project Flow will look like this:



- A **dataset** is represented by a blue square with a symbol that depicts the dataset type or connection. The initial datasets (also known as input datasets) are found on the left of the Flow. In this project, the input datasets will be the ones we just created in Snowflake.
- A **recipe** in Dataiku DSS (represented by a circle icon with a symbol that depicts its function) can be either visual or code-based, and it contains the processing logic for transforming datasets.
- **Machine learning processes** are represented by green icons.
- The **Actions Menu** is shown on the right pane and is context sensitive.
- Whatever screen you are currently in you can always return to the main **Flow** by clicking the **Flow** symbol from the top menu (also clicking the project name will take you back to the main Project page).

**Input dataset:** *In the interests of time we have performed some initial steps of the data pipeline such as cleansing and transformations on the loans dataset. These steps can be created in Dataiku from the raw datasets from the Lending Club to form a complete pipeline with the data and execution happening in Snowflake*

## How We'll Build The Project

Our goal is to build an optimized machine learning model that can be used to predict the risk of default on loans for customers and advise them on how to reduce their risk. To do this, we'll join the input datasets, perform transformations & feature engineering so that they are ready to use for building a binary classification model.

## Creating a Dataiku Project

Once you've logged in, **click on +NEW PROJECT** and **select Blank project** to create a new project.

The screenshot shows the Dataiku DSS interface. At the top, there's a search bar and a navigation bar with icons for help, tasks, and user profile. Below the navigation bar, there's a list of recent projects:

- Credit Scoring**: dataikulab@gmail.com's analysis of LOANS ENRICHED prepared
- LOANS ENRICHED joined**
- LOANS ENRICH**

In the center, there are two project cards:

- Covid-19**: The project Covid was created by stephen.frank on Nov 07th 2020.
- Flight Delay**: The project Flight Delay was created by lpkronek on Nov 07th 2020.

A red box highlights the **+ NEW PROJECT** button in the top right corner of the screen. A dropdown menu is open, listing several options:

- Blank project**: Create a new blank project
- DSS tutorials**: Create a tutorial project
- Sample projects**: Create a project from a sample
- Industry solutions**: Create a project from an industry solution
- Import project**: Import a project into the system

At the bottom left, a message says "don't have any workspace yet."

## Data Import, Analysis & Join

Duration: 5

After creating our project let's add our datasets from Snowflake to the Flow.

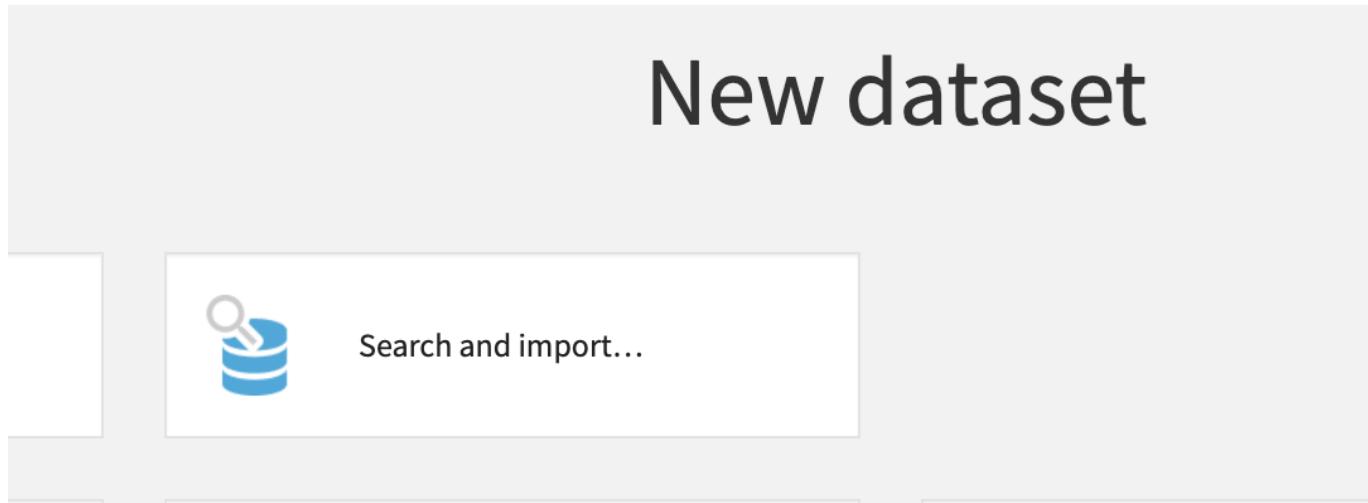
- From the Flow click **+ Import Your First Dataset** in the centre of the screen.

The screenshot shows the **Credit Scoring** project page. The header includes a back arrow, a refresh icon, and a search bar. To the right, there are buttons for **WATCH** (with 1 item), **STAR** (0 items), and **TODAY**.

The main content area has a green sidebar on the left with the project name and a summary of datasets, recipes, and models. The main area contains tabs for **Flow**, **Lab**, **Dashboards**, **Wiki**, and **Tasks**. The **Flow** tab is active, showing counts for datasets (0), recipes (0), and models (0). A red box highlights the **+ IMPORT YOUR FIRST DATASET** button in the center of the Flow section.

Below the tabs, there's a section for **Your new project's Todo** (1/4 done) with a delete icon.

- Select the **Search and import option**



- Select the **PC\_DATAIKU\_DB** connection from the dropdown then click the refresh icon next to the database or schema dropdowns to populate these options.
- Select the database and schema as below then click on **LIST TABLES**

**Connection to browse**

**Restrict to database**

**Restrict to schema**

**LIST TABLES**

- Select the **Loans\_Enriched** and **Unemployment\_Data** datasets and click **CREATE 2 DATASETS**

## CREATE 2 DATASETS

### Dataset name



UNEMPLOYMENT\_DATA



LOANS\_ENRICHED



The screenshot shows the Dataiku DSS interface. At the top, there's a navigation bar with icons for Home, Flow, Project, Datasets, and Help. Below the navigation bar, there's a search bar and filters for Last modified, Tags, and Favorites. The main area displays two datasets:

- LOANS\_ENRICHED**: Originated from Snowflake and was modified just now. It has a yellow "origin:sql\_import" tag and a purple "PC\_DATAIKU\_DB" tag.
- UNEMPLOYMENT\_DATA**: Originated from Snowflake and was modified just now. It has a yellow "origin:sql\_import" tag and a purple "PC\_DATAIKU\_DB" tag.

- Navigate back to the Flow from the left-most menu in the top navigation bar (or use the keyboard shortcut G+F).

The screenshot shows the Dataiku DSS Flow interface. At the top, there's a navigation bar with icons for Credit Scoring, Back, Forward, Refresh, and Save. A search bar says 'Search' and a dropdown says 'All'. On the right, there are buttons for '+ ZONE', '+ RECIPE', and '+ DATASET'. Below the navigation bar, it says '2 datasets'. Two dataset cards are displayed side-by-side. Each card has a blue square icon with a white snowflake-like symbol. The left card is labeled 'LOANS\_ENRICHED' and the right card is labeled 'UNEMPLOYMENT\_DATA'.

Now we have all of the raw data needed for this lab. Let's explore what's inside these datasets.

- From the Flow, double click on the `loans_enriched` dataset to open it.
- You can analyze column metrics to better understand your data: Either click on the column name and `Select Analyze` or, if you wish for a quick overview of columns key statistics, `Select Quick Column Stats` button on the top-right.

The screenshot shows the Dataiku DSS Datasets view for the 'LOANS\_ENRICHED' dataset. The top navigation bar includes 'Credit Scoring', 'Datasets', 'Explore', 'Charts', 'Statistics', 'Status', 'History', 'Settings', and 'ACTIONS'. The 'ACTIONS' menu has a red box around its 'More' icon. The main area shows a table with 10,964 rows and 35 columns. A context menu is open over the 'EMP\_LENGTH\_YE...' column, with the 'Select Analyze...' option highlighted and surrounded by a red box. Other options in the menu include 'Edit column schema...', 'Filter', 'Sort', 'Color column by value', 'Create Prediction model...', and 'RENT' and 'MORTGAGE' buttons.

## Join the Data

So far, your Flow only contains datasets. To take action on datasets, you need to apply recipes. The **LOANS\_ENRICHED** and **UNEMPLOYMENT\_DATA** datasets both contain a column of Loan IDs. Let's join these two datasets together using a visual recipe.

- Select the **LOANS\_ENRICHED** dataset from the Flow by `single clicking` on it.
- Choose `Join With...` from the `Visual recipes` section of the Actions sidebar near the top right of the screen (note: click the `Open Panel` arrow if it is minimized and notice there are three different types of join recipe, we want `Join With...`).
- Choose **UNEMPLOYMENT\_DATA** as the second input dataset.

## ∞ New join recipe

The screenshot shows the 'Input datasets' section with two dropdown menus: 'LOANS\_ENRICHED' and 'UNEMPLOYMENT\_DATA'. The 'Output dataset' section shows 'Name' set to 'LOANS\_ENRICHED\_joined' and 'Store into' set to 'PC\_DATAIKU\_DB'. A note at the bottom left says 'Additional inputs can be added after the recipe creation.' A button at the bottom right says 'NEW DATASET | USE EXISTING DATASET'.

- Leave the default option of **PC\_DATAIKU\_DB** for “**Store into**” and **Create** the recipe.
- On the Join step you can **click on Left Join** to observe the selected join type and conditions.

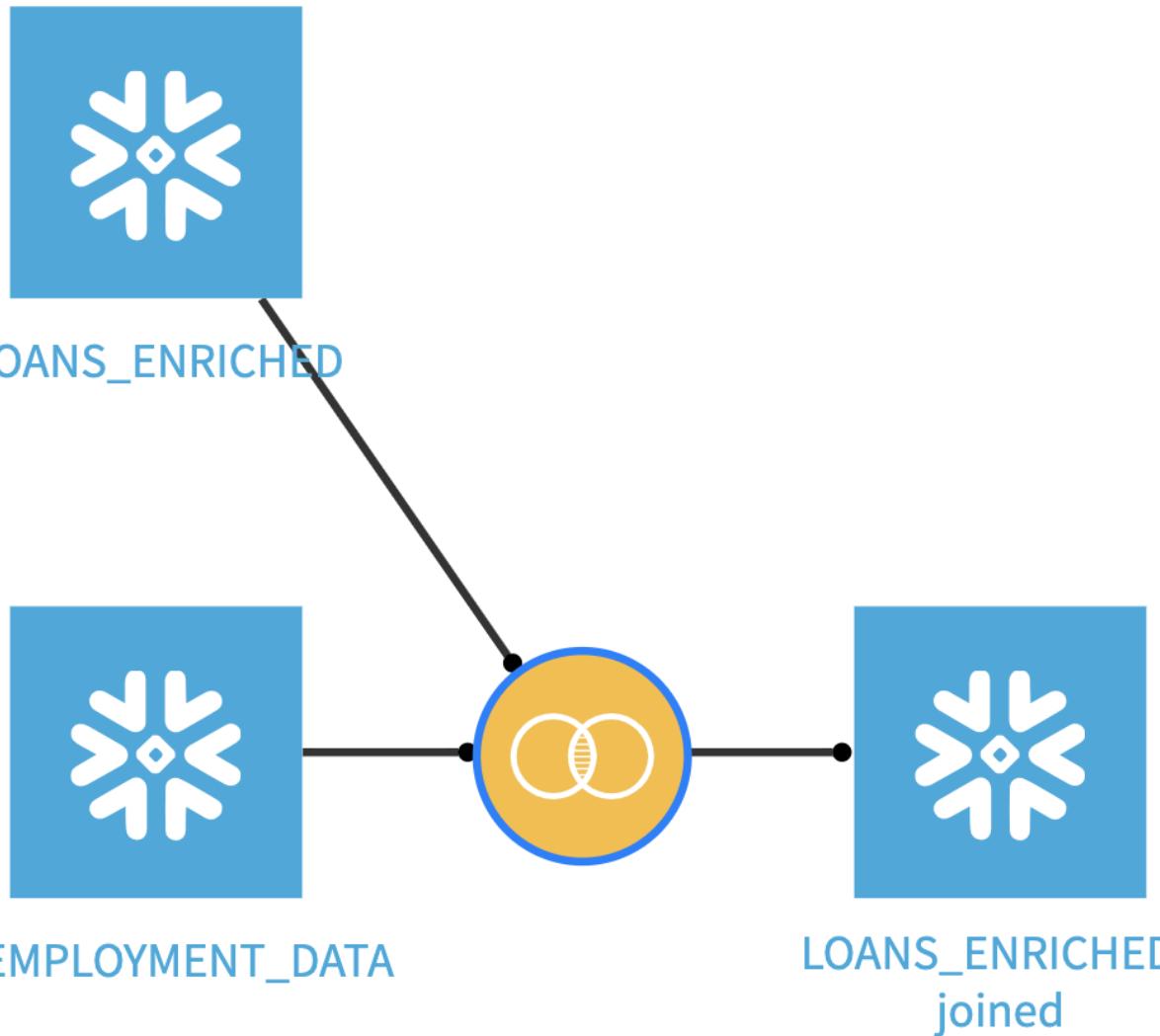
## ∞ Join conditions

The screenshot shows a join condition where 'LOAN\_ID' from 'LOANS\_ENRICHED' is compared to 'LOAN\_ID' from 'UNEMPLOYMENT\_DATA' using an equals operator (=). Both columns have their values set to '145509846'. A 'CLEAR ALL' button is visible in the top right.

- On the Selected columns step you can leave the defaults
- On the Output step, note the output column names
- Before running it, **Save the recipe**
- Ensure that **In-database (SQL)** is selected as the engine. You can view this underneath the **Run button** (Bottom left). If it is set to a different engine **click on the three cogs** to change it
- Click **RUN** and **Update Schema** if prompted, then return to the Flow

Note: You can view the SQL query as well as the execution plan generated by selecting **VIEW QUERY** on the **Output** screen.

Your flow should now look like this



## Prepare the Data

Duration: 5

Data cleaning and preparation is typically one of the most time-consuming tasks for anyone working with data. In our lab, in order to save some of that time, our main lending dataset has already been largely cleaned. In the real world this would be done by other colleagues, say, from the data analytics team collaborating on this project and you would see their work as steps in our projects flow.

Let's take a brief look at the [Prepare recipe](#), the workhorse of the visual recipes in Dataiku, and perform some final investigations and transformations.

- [Single click](#) on the output dataset of our join from the flow and [select Prepare](#) from the visual recipes in the [Actions Panel](#). (If the automatically generated output dataset name is starting to get unwieldy feel free to change it)

In a Prepare recipe you assemble a series of steps to transform your data from a library of ~100 processors. There are a couple of ways you can select these processors to build your script. Firstly you can select these processors directly by using the [+ADD A NEW STEP](#) button on the left. Secondly because Dataiku DSS infers meanings for each column, it suggests relevant actions in many cases. In the example below although

the column is stored in Snowflake as a String Dataiku DSS recognises it as a date format so infers a **Date (unparsed)** meaning and suggests the **Parse Date** processor, by selecting the **More actions** menu item further suggestions are made.

The screenshot shows the Dataiku DSS interface with a dataset named "compute\_LOANS\_ENRICHED\_prepared". The "Script" tab is selected. On the left, there's a sidebar with instructions for adding steps. At the bottom left, there are buttons for "VIEW QUERY", "RUN", and "In-database (SQL)". A red box highlights the "+ ADD A NEW STEP" button. In the main area, a context menu is open over the "EARLIEST\_CR\_LINE" column header, with "Parse date..." highlighted. Another red box highlights this menu. The dataset sample shows various columns like DTI, DELINQ\_2..., INQ\_LAST\_6M..., MTHS\_SINCE\_LAST\_DELI..., and MTHS\_SINCE\_LAST.

Let's try using processors with both methods, firstly via the suggested actions:

- Click on the **EARLIEST\_CR\_LINE** column header and from the dropdown, select **Parse date**
- In **Add a custom format** set the format to **d-MMM-yyyy** and click on **USE DATE FORMAT**
- A step is generated on the left. Change the **Locale** to **en\_US**

EARLIEST\_CR\_LINE      INQ\_LAST\_6M...

- Delete
- Rename
- Move
- Analyze...
- Edit column details...
- Parse date...

More actions 

Filter

Sort

Smart Date for EARLIEST\_CR\_LINE 

Detected format	Example	Parses
yy-MMMZ	13-Apr+0000	<div style="width: 50%;"><div style="width: 100%; background-color: #ccc;"></div></div>
<input checked="" type="checkbox"/> d-MMM-yyyy		<div style="width: 100%; background-color: #00c000;"></div>

Sample input      Output preview

01-Mar-2008	Sat, 01 Mar 2008 00:00:00 +0000
01-Mar-1995	Wed, 01 Mar 1995 00:00:00 +0000
01-Oct-2008	Wed, 01 Oct 2008 00:00:00 +0000
01-Jul-2006	Sat, 01 Jul 2006 00:00:00 +0000
...	...

Working with date formats

Date	Format
2001-07-04 12:08:56	yyyy-MM-dd HH:mm:ss
4 July 2001 12:08:56 PM PDT	d MMM yyyy HH:mm:ss a z
Wed, 4 Jul 2001 12:08:56 -0700	EEE, dd MMM yyyy HH:mm:ss Z

Advanced help

CANCEL  USE DATE FORMAT

**Parse date in EARLIEST\_CR\_LINE**

[\*\*10000\*\*](#) eye **power** **trash** ...

Column [single](#) | [multiple](#) | [pattern](#) | [all](#) mouse cursor icon

**EARLIEST\_CR\_LINE**

Output column (Leave empty for in place) **EARLIEST\_CR\_LINE\_parsed**

Input date format(s) [Find with Smart Date](#)

**d-MMM-yyyy** **trash**

**+ ADD FORMAT**

**Locale** en\_US

**Timezone** UTC

x<sup>2</sup> info **trash** **checkbox** **question**

**+ ADD A NEW STEP**

- Click on the newly created column (click outside the step to action the change) and select **Compute time difference**
- Change **Until** to **Another Date Column** and add **ISSUE\_DATE\_PARSED** as that column.
- Change the unit to **Years** and name the new column **since\_Earliest\_CR\_LINE\_years**

Compute time difference between  
EARLIEST\_CR\_LINE\_parsed and  
ISSUE\_DATE\_PARSED

10000 ...

Time since column

EARLIEST\_CR\_LINE\_parsed

until

Another date column ▾

Other column

ISSUE\_DATE\_PARSED

Output time unit

Years ▾

Output column

since\_EARLIEST\_CR\_LINE\_years

Reverse output

x<sup>2</sup>

Now we have our desired feature we can remove the two date columns.

- Click on EARLIEST\_CR\_LINE and select delete, do the same for EARLIEST\_CR\_LINE\_parsed

Your script steps should now look like this:

The screenshot shows a list of three script steps in a Dataiku interface:

- Parse date in EARLIEST\_CR\_LINE**: This step is currently selected, indicated by a blue edit icon labeled "10000". It includes icons for eye, power, delete, and more.
- Compute time difference between EARLIEST\_CR\_LINE\_parsed and ISSUE\_DATE\_PARSED**: This step also has a blue edit icon labeled "10000" and a yellow "..." button.
- Remove columns EARLIEST\_CR\_LINE\_parsed, EARLIEST\_CR\_LINE**: This step has a blue edit icon labeled "10000" and a yellow "..." button.

At the bottom of the list is a yellow button labeled "+ ADD A NEW STEP".

Optionally you can place the three date transformation script steps into their own group with comments to make it simple for a colleague to follow everything you have done Let's turn our attention to the `INT_RATE` column. The interest rate is likely to be a powerful predictive feature when modeling credit defaults but currently its stored as a string:

- Click on the `+ADD A NEW STEP` button at the bottom of your script steps.
- Select the `Find and Replace` processor either by looking in the `Strings` menu or using the search function.

The screenshot shows a Dataiku interface with a search bar containing 'replace'. Below the search bar, there is a list of processor categories: Filter data, Data cleansing, **Strings**, Math / Numbers, Natural Language, Code, and Misc. The 'Strings' category is highlighted with a red box. To the right of the search bar, there is a table with columns for ID, Value, and other metrics. A red box highlights the 'Find and replace' processor in the list.

1	139277661	12000.0	12000.0	36		17.97	433.1
2	140962137	35000.0	35000.0	60		17.97	888
3	138443666	8000.0	8000.0	36		6.67	245.1
1	127162918	20000.0	20000.0	60		17.09	498.1
3	138931648	10000.0	10000.0	60		10.08	212.1

**Find and replace**  
This processor performs string replacements either in a specified column or across multiple columns.  
**Matching modes**  
By setting the replacement mode, you can specify whether you want to replace:

- 'Complete value': replace complete cell values (For example, replacing '123' with '456')
- 'Substring': replace all occurrences of a string within the cell (For example, replacing '123abc' with '456abc')
- 'Regular expression': replace matches of a regular expression (For example, replacing 'abc' with 'def' in '123abc456')

**Normalization modes**

- Select INT\_RATE as the column then click +ADD REPLACEMENT and replace % with a blank value. Ensure the Matching Mode dropdown is set to Substring

**Replace % by " in INT\_RATE**

**9840**  **SQL**

Column [single](#) | [multiple](#) | [pattern](#) | [all](#)

**INT\_RATE**

Output column (empty for in-place)

**Replacements**

<input type="checkbox"/>	%	→	No value	<input type="button"/>
<b>+ ADD REPLACEMENT</b>				

**Raw text edit**

**Matching mode**

Substring	<input type="button"/>
-----------	------------------------

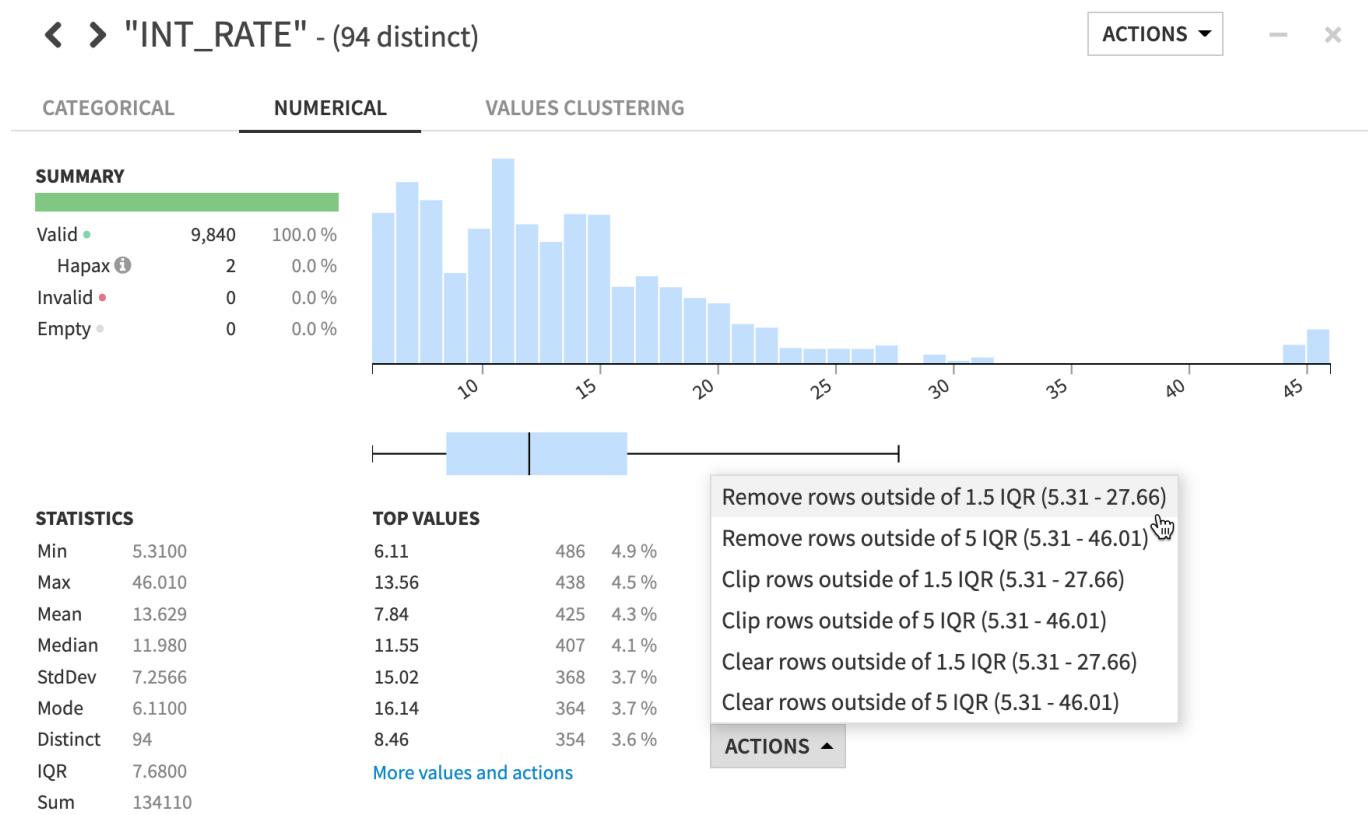
**Normalization mode**

Exact	<input type="button"/>
-------	------------------------

Only perform the first matching

Our `INT_RATE` column has some suspiciously high values. Let's use the Analyze tool again and see how it can be used to take certain actions in a Prepare recipe

- From the **INT\_RATE** column header dropdown, select **Analyze**.
  - In the Outliers section, choose **Remove rows outside 1.5 IQR** from the menu.



As before you can optionally group and comment on these int\_rate transformation steps.

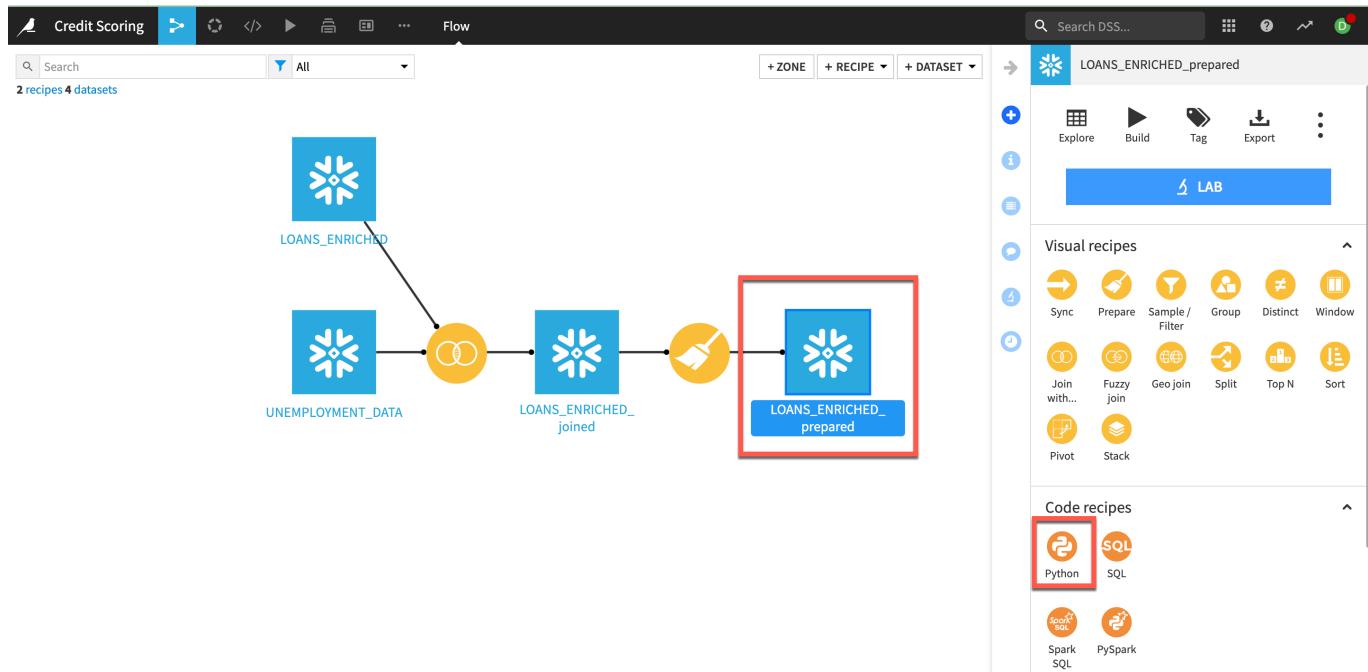
- ensure **In-database (SQL)** engine is selected and then click **RUN**

## Feature Engineering with Code

Duration: 5

Till now we've used visual tools but let's see how users who prefer to code can collaborate alongside their low/no code colleagues

- Return to the Flow.
- Click on the output dataset of the prepare recipe (in this case **LOANS\_ENRICHED\_prepared** but you may have renamed your output)
- Once selected click on the **Python Code recipe** from the **Actions** panel



- Now Add a new output dataset and click CREATE RECIPE

Dataiku DSS generates some starter code for us, we can also use code samples our colleagues have created and tagged and, if we prefer, work from Jupyter notebooks or a range of IDE's. For this lab we will stick with the standard code editor.

- To save some typing let's change our dataframe name to `df` on line 8
- Remove the to-do starter code on lines 11 – 15
- Replace with the following lines to generate new features

```

df['DEBT_AMNT'] = [d*df.INSTALLMENT.values[idx]/100.0 for idx,d in
enumerate(df.DTI.values)]

df["DEBT_AMNT_NORM"] = (df.DEBT_AMNT.values -
np.mean(df.DEBT_AMNT.values))/np.std(df.DEBT_AMNT.values)

df["INSTALL_NORM"] = (df.INSTALLMENT.values -
np.mean(df.INSTALLMENT.values))/np.std(df.INSTALLMENT.values)

```

- Ensure you replace the name of the dataframe in the final line (`.write_with_schema(your_dataframe_name)`) with `df`.

Your code should now look like this

```

1 # -*- coding: utf-8 -*-
2 import dataiku
3 import pandas as pd, numpy as np
4 from dataiku import pandasutils as pdu
5
6 # Read recipe inputs
7 LOANS_ENRICHED_prepared = dataiku.Dataset("LOANS_ENRICHED_prepared")
8 df = LOANS_ENRICHED_prepared.get_dataframe()
9
10
11 # Compute recipe outputs from inputs
12 # TODO: Replace this part by your actual code that computes the output, as a Pandas dataframe
13 # NB: DSS also supports other kinds of APIs for reading and writing data. Please see doc.
14
15 df['DEBT_AMNT'] = [d*df.INSTALLMENT.values[idx]/100.0 for idx,d in enumerate(df.DTI.values)]
16
17 df["DEBT_AMNT_NORM"] = (df.DEBT_AMNT.values - np.mean(df.DEBT_AMNT.values))/np.std(df.DEBT_AMNT.values)
18
19 df["INSTALL_NORM"] = (df.INSTALLMENT.values - np.mean(df.INSTALLMENT.values))/np.std(df.INSTALLMENT.values)
20
21
22 # Write recipe outputs
23 LOANS_ENRICHED_FEATURES = dataiku.Dataset("LOANS_ENRICHED_FEATURES")
24 LOANS_ENRICHED_FEATURES.write_with_schema(df)
25

```

Validation successful

VALIDATE

- click RUN

Dataiku DSS allows you to create an arbitrary number of **Code environments** to address managing dependencies and versions when writing code in R and Python. Code environments in Dataiku DSS are similar to the Python virtual environments. In each location where you can run Python or R code (e.g., code recipes, notebooks, and when performing visual machine learning/deep learning) in your project, you can select which code environment to use.

## Training

Duration: 5

Having sufficiently explored and prepared the loans and employment data, the next stage of the AI lifecycle is to experiment with machine learning models.

This experimentation stage encompasses two key phases: model building and model assessment.

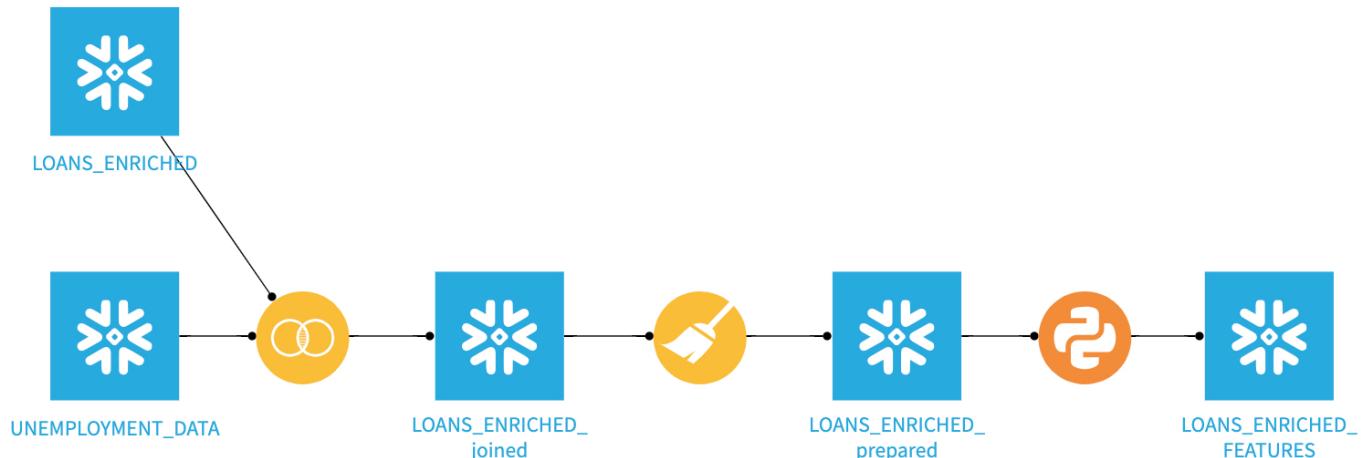
**Model building:** Users have full control over the choice and design of a model — its features, algorithms, hyperparameters and more.

**Model assessment:** Tools such as visualizations and statistical summaries allow users to compare model performance.

These two phases work in tandem to realize the idea of Responsible AI. Either through a visual interface or code, building models with DSS can be transparently done in an automated fashion. At the same time, the model assessment tools provide a window into ensuring the model is not a black box.

Before building our model first we will split our output dataset from our python step.

This is how your flow should look like before splitting



- Return to the flow and select the output dataset of the python recipe and the **Split** recipe from the **Actions** menu.
- Add two datasets named **Test** and **Train** and click **CREATE RECIPE**
- Choose **Dispatch Percentiles** as the splitting strategy and have 80% go to Train and 20% to Test.
- Choose **ISSUE\_DATE\_PARSED** to sort by and then click **RUN**

Input dataset	Output
LOANS_ENRICHED_FEATURES DATASET - View	LOANS_TRAIN (Managed) LOANS_TEST (Managed) <b>+ ADD</b>

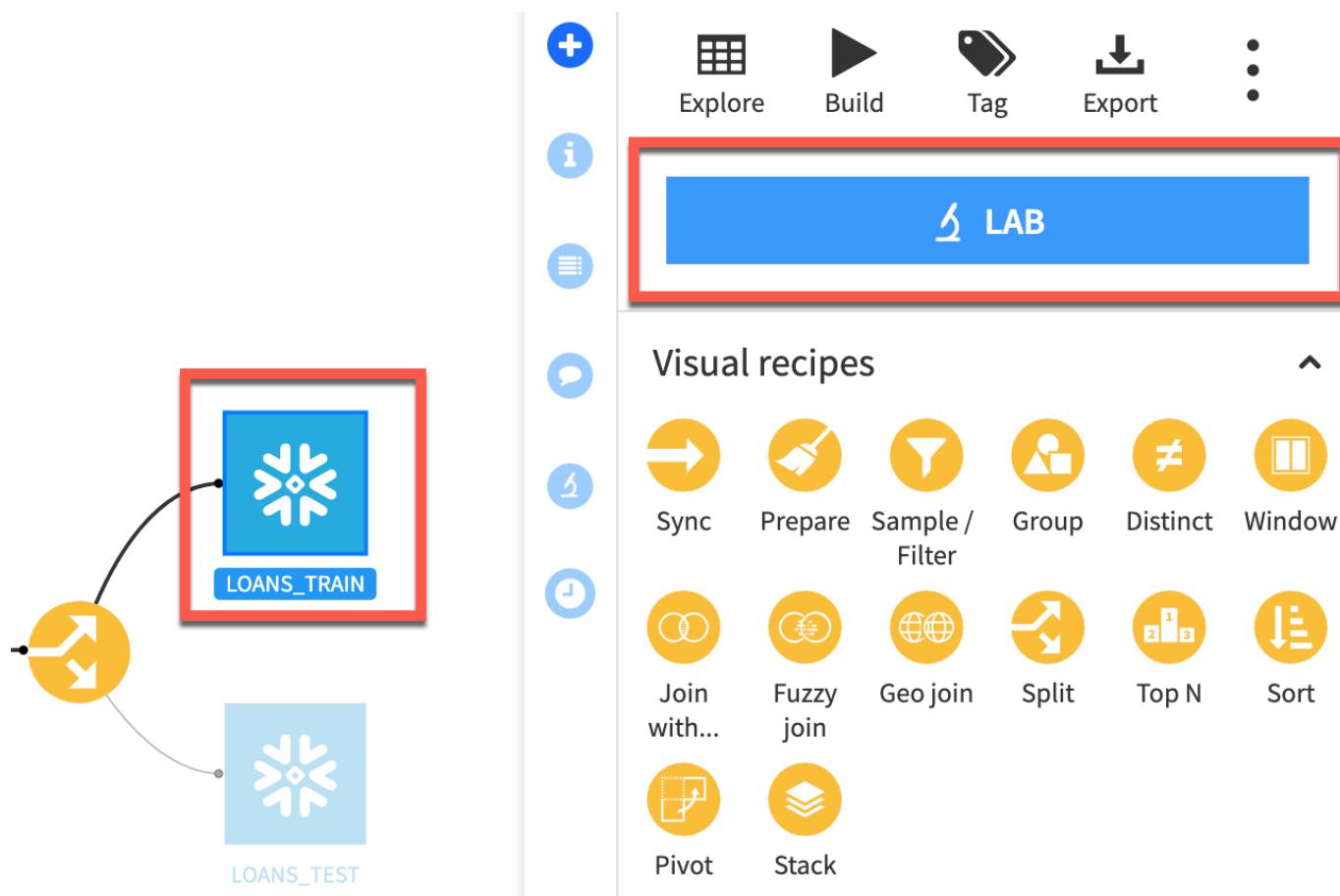
Dispatch percentiles of sorted data on output datasets CHANGE

Sort according to... ISSUE\_DATE\_PARSED date Select column

**Ratio**  
% 80 Remaining 20% → LOANS\_TRAIN Remaining 20% → LOANS\_TEST

+ ADD RATIO

- Return to the flow and select the **Train** dataset and click the **LAB** button in the Actions menu
- Select **AutoML Prediction** and set **LOAN\_STATUS** as the target and leave the default template of **Quick Prototypes** then click **CREATE**



**Create prediction model on [LOAN\\_STATUS](#)**

**AutoML** Let Dataiku create your models.

- Quick Prototypes** Get some models, generic and quick. 
- In-memory**
- Interpretable Models for Business Analysts** Produce decision tree and simple linear models. 
- High Performance Models** Be patient and get even more accurate models. 

**Expert** Have full control over the creation of your models.

- Deep Learning** Create the architecture of your deep learning models and train them. 
- Choose Algorithms** Select the algorithms and the hyper parameters to use in cross-validation. 
- Write Your Own Estimator** Train your own Python or Scala models. 

Name your analysis  **CREATE**

When building a visual model, users can choose a template instructing DSS to prioritize considerations like speed, performance, and interpretability. Having decided on the basic type of machine learning task, you retain full freedom to adjust the default settings chosen by DSS before training any models. These options include the metric for which to optimize, what features to include, and what algorithms should be tested.

Feel free to try some experiments of your own in the **Design** tab. Here are some suggestions to try. Don't forget to click :

- Run with the employment features **off/on** to see if the Marketplace enrichment data makes a difference to our models accuracy. Click **SAVE** and then **TRAIN** in the top right after you've made your changes in **DESIGN**

**BASIC**

- Target
- Train / Test Set
- Metrics
- Debugging

**FEATURES**

- Features handling** (highlighted with a red box)
- Feature generation
- Feature reduction

**MODELING**

- Algorithms
- Hyperparameters

**ADVANCED**

- Runtime environment
- Weighting strategy
- Probability calibration

**Features Handling**

**Handling of "UNEMPLOYMENT\_**

**Distribution**

Minimum	Maximum	Mean	StdDev	Median
2.2455	10.123			

Histogram of UNEMPLOYMENT RATE distribution:

- While in the **Features handling** menu look at **MTHS\_SINCE\_LAST\_DELINQ**, **MTHS\_SINCE\_LAST\_RECORD** and **MTHS\_SINCE\_LAST\_MAJOR\_DEROG**. In the Distribution table we can see most cells are empty. We have various techniques available to us in the **Missing Values** dropdown but given that there are so few values in these columns lets just turn reject the features.

- You may notice on the **RESULT** screen that ML Diagnostics are flagged against a model. These identify and help troubleshoot potential problems and suggest possible improvements at different stages of training and building machine learning models.
- Hover your cursor over **Diagnostics** to see the potential issues

In this example I can see we have an imbalanced dataset, let's fix that.

- Go to the **DESIGN** page and the **TRAIN/TEST SET** menu. Here you can rebalance your dataset.

**BASIC**

- Target
- Train / Test Set** (highlighted with a red box)
- Metrics
- Debugging

**FEATURES**

- Features handling
- Feature generation
- Feature reduction

**MODELING**

- Algorithms
- Hyperparameters

**ADVANCED**

- Runtime environment
- Weighting strategy
- Probability calibration

**DESIGN**

**RESULT**

**Train / test set for final evaluation**

Policy: Split the dataset

**Time ordering**: Enabled (OFF)

**Sampling & Splitting**: If your dataset does not fit in your RAM, you may want to subsample the set on which splitting will be performed.

Sampling method	Class rebalance (approx. ratio)
% to use	33
Column	LOAN_STATUS

**Split**: Randomness (For more advanced splitting, use a split recipe, and then use "Explicit extracts from two datasets" policy)

- K-fold cross-test:  Gives error margins on metrics, but strongly increases training time
- Train ratio: 0.8 (Approximate proportion of the sample that goes to the train set. The rest goes to the test set)
- Random seed: 1337 (Using a fixed random seed allows for reproducible result)

**Sampling & splitting**: Class rebalance (approximately 33%) & 0.8 train ratio

**Sampling**: [progress bar]

**Evaluation**: [progress bar]

The metrics used to rank models obtained by different algorithms are computed on the **test set**. The final model is trained on the **train set**.

**Hyperparameters**: [EDIT](#)

- Try different **Algorithms**
- In **Runtime Environment** choose Select a **container configuration** from the drop down for **Containerized execution** and run with a larger container
- You can directly compare models from different experiments by selecting them via the **checkbox** and then selecting **Compare** from the **ACTIONS** menu.

**Credit Scoring**

**Quick modeling of LOAN\_STATUS on Train**

**Predict LOAN\_STATUS (Binary classification)**

**DESIGN**

**RESULT**

**ACTIONS** (highlighted with a red box) ▾

- Delete
- Star
- Create ensemble model
- Compare** (highlighted with a red box)

**SESSION 6** Started Today at 14:29, ended Today at 14:30 2 models 32/

ROC AUC score

0.950  
0.900  
0.850  
0.800  
0.750

0s 5s 10s 15s 20s

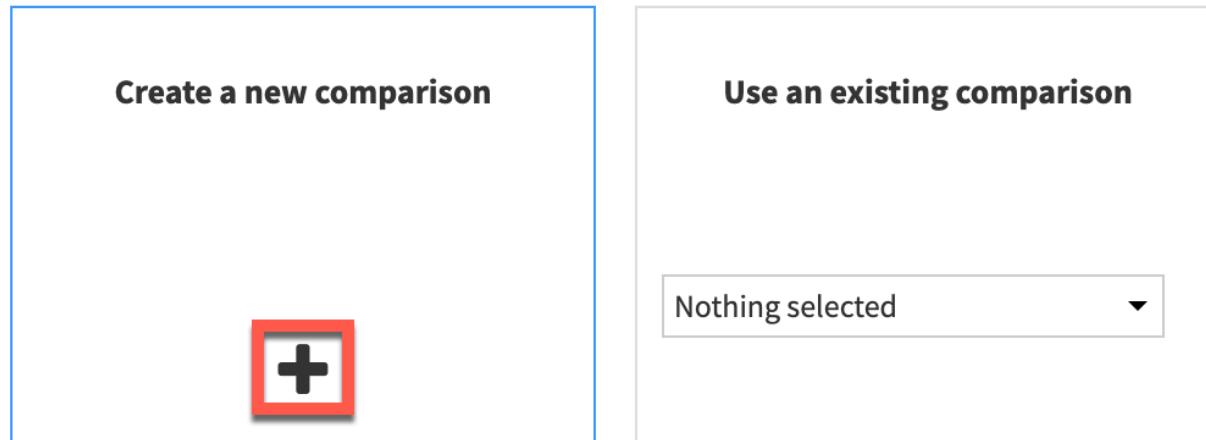
**Random forest (s6)** 0.828 ✓ Done 2 minu

Most important variables

TOTAL\_PYMNT

- Select **Create a new comparison** and then click **compare**

## Where to compare the models ?



Name your comparison Compare 4 models from Predict LOA **COMPARE**

After having trained as many models as desired, DSS offers tools for full training management to track and compare model performance across different algorithms. DSS also makes it easy to update models as new data becomes available and to monitor performance across sessions over time.

In the **Result** pane of any machine learning task, DSS provides a single interface to compare performance in terms of sessions or models, making it easy to find the best performing model in terms of the chosen metric.

In the example below we see an improvement between session 4 and 5 when the employment data is added and then a further minor improvement when using the LightGBM algo

Previously trained

SESSION	Model	Metric	Value	Star
SESSION 7	Logistic Regression (s7)		0.802	★
SESSION 7	LightGBM (s7)	trophy	0.803	★

<input type="checkbox"/>	<input checked="" type="radio"/> XGBoost (s7)	0.766	
<input type="checkbox"/>	<input checked="" type="radio"/> Decision Tree (s7)	0.660	

<input type="checkbox"/>	<b>SESSION 5</b>		
<input type="checkbox"/>	<input checked="" type="radio"/> Random forest (s5)	0.727	
<input type="checkbox"/>	<input checked="" type="radio"/> Logistic Regression (s5)	0.802	

<input type="checkbox"/>	<b>SESSION 4</b>		
<input type="checkbox"/>	<input checked="" type="radio"/> Random forest (s4)	0.647	
<input type="checkbox"/>	<input checked="" type="radio"/> Logistic Regression (s4)	0.671	

Clicking on any model produces a full report of tables and visualizations of performance against a range of different possible metrics.

- **Click** on your best performing model
- Step through the various graphs and interactive charts to better understand your model.
- For example **Subpopulations Analysis** allows you to identify potential bias in your model by seeing how it performs across different sub-groups
- **Interactive Scoring** allows you to run real time “what-if” analysis to understand the impact of given features

**Summary**

**INTERPRETATION**

- Variables importance
- Partial dependence
- Subpopulation analysis**
- Individual explanations
- Interactive scoring

**PERFORMANCE**

- Confusion matrix
- Decision chart
- Lift charts
- Calibration curve
- ROC curve
- Density chart
- Metrics and assertions

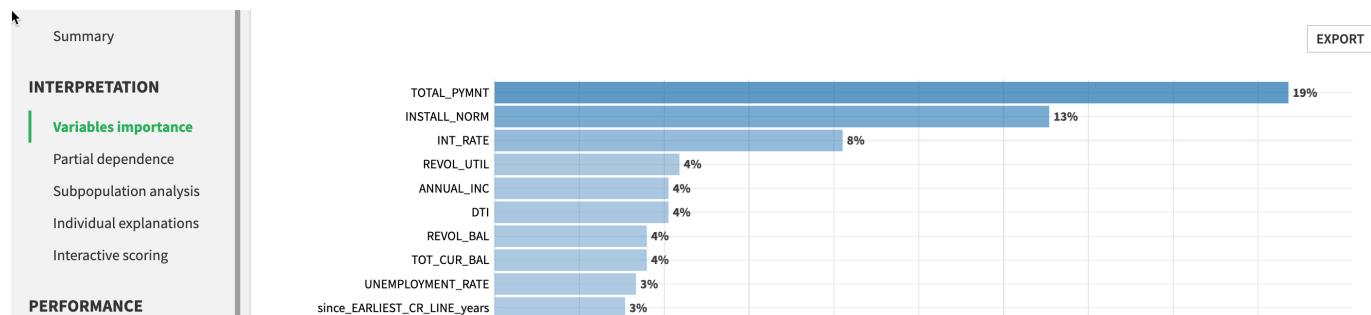
Select your variable: ADDR\_STATE

COMPUTE

Search...

Modality	Actually ok	Predicted ok	Metric: ROC AUC	Precision	Recall
	96 %	99.5 %			
100 %	96 %	99.5 %	0.803	0.964	1.0000
14 % — CA	94 % <span style="color:red">↓2%</span>	98.7 % <span style="color:red">↓0.8%</span>	0.777	0.951	1.0000
8 % — NY	96 %	100.0 % <span style="color:green">↑0.5%</span>	0.853	0.963	1.0000
8 % — TX	95 % <span style="color:red">↓1%</span>	100.0 % <span style="color:green">↑0.5%</span>	0.743	0.953	1.0000
7 % — FL	95 % <span style="color:red">↓1%</span>	99.1 % <span style="color:red">↓0.4%</span>	0.956	0.955	1.0000
5 % — IL	98 % <span style="color:green">↑2%</span>	98.8 % <span style="color:red">↓0.7%</span>	0.934	0.988	1.0000
4 % — PA	93 % <span style="color:red">↓3%</span>	98.3 % <span style="color:red">↓1.2%</span>	0.823	0.948	1.0000
3 % — OH	96 %	100.0 % <span style="color:green">↑0.5%</span>	0.864	0.965	1.0000

Here we can see Variable importance



## Deployment

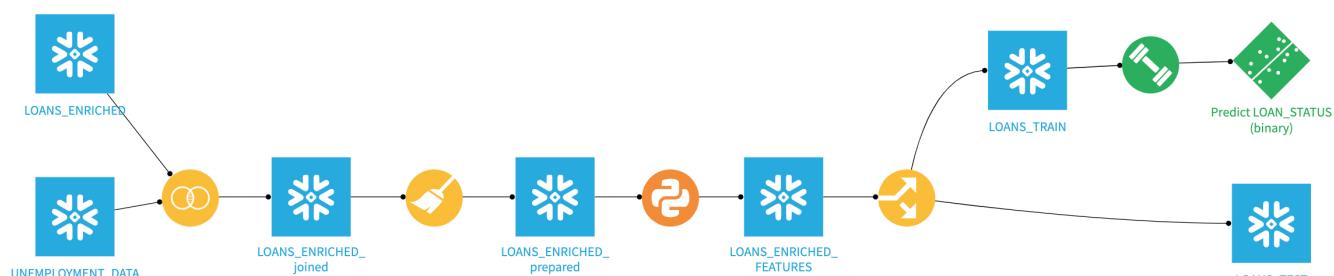
Duration: 2

After experimenting with a range of models built on historic training data, the next stage is to deploy our chosen model to score new, unseen records.

For many AI applications, batch scoring, where new data is collected over some period of time before being passed to the model, is the most effective scoring pattern. Deploying a model creates a "saved" model in the Flow, together with its lineage. A saved model is the output of a Training recipe which takes as input the original training data used while designing the model.

- Click on **DEPLOY**, accept the default model name and click **CREATE**

Your flow should now look like this:



## Scoring and Evaluation

Duration: 2

- Select the LOANS\_TEST dataset and the Score recipe from the actions menu
- Select your model
- Ensure In-Database (Snowflake native) is selected as the engine in order to use the Java UDF capability

Score a dataset

Input dataset

Output dataset

Input dataset

LOANS\_TEST  
DATASET - View

Prediction Model

Predict LOAN\_STATUS (binary)

Name

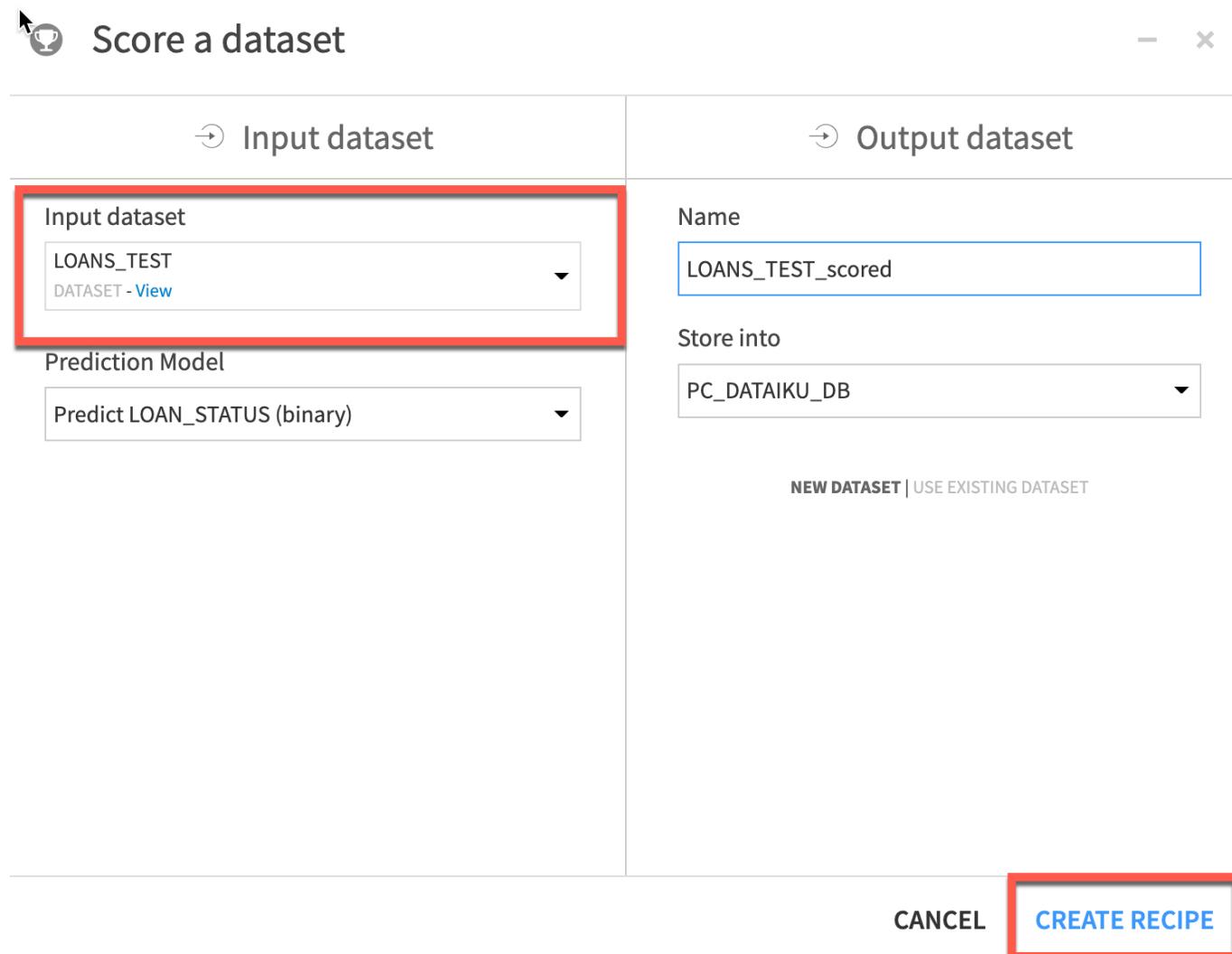
LOANS\_TEST\_scored

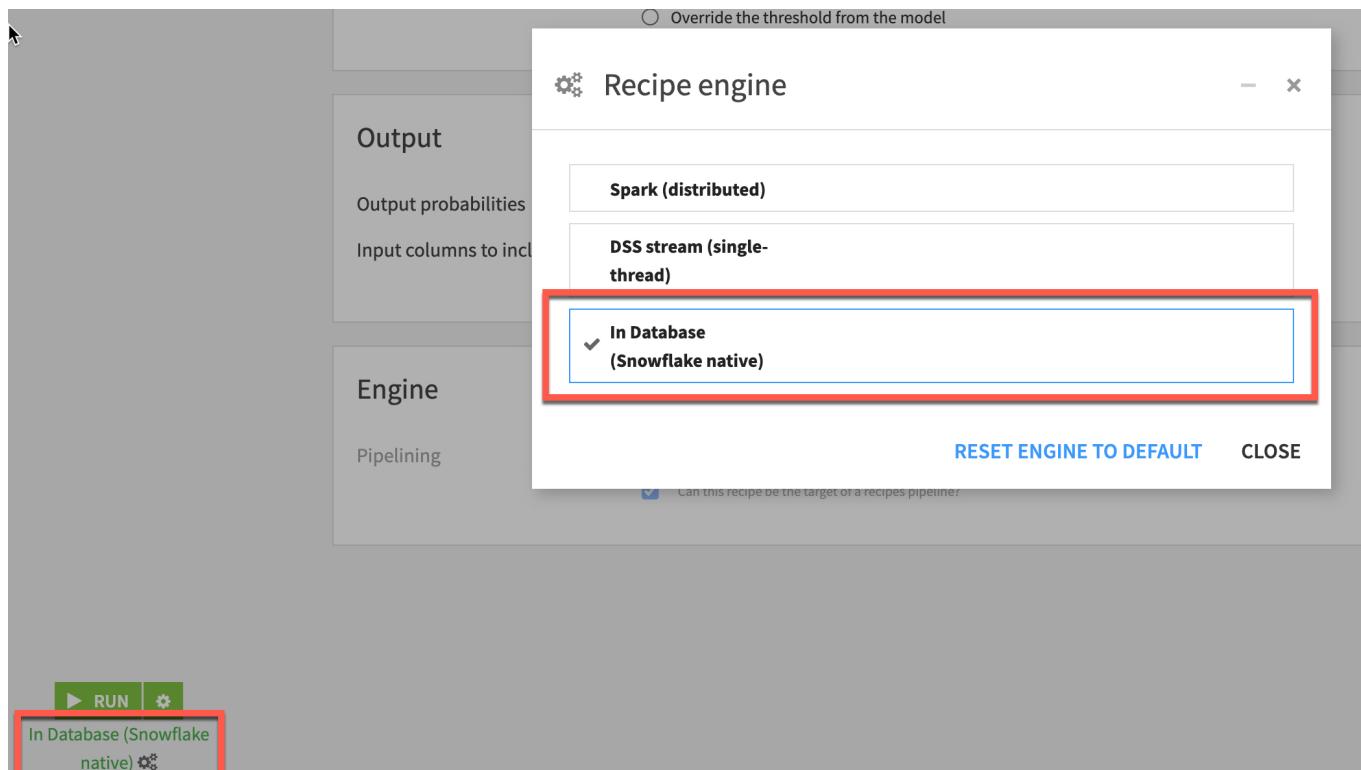
Store into

PC\_DATAIKU\_DB

NEW DATASET | USE EXISTING DATASET

CANCEL CREATE RECIPE





We can now see the results back on the Snowflake tab. If you hit the refresh icon near the top left of our screen by your databases, you should see the `CREDIT_SCORING_LOANS_TEST_SCORED` table that was created once we kicked off our prediction job.

**Preview Data** will give you glimpse of additional column added to the list.

```
USE ROLE SYSADMIN;
USE DATABASE PC_DATAIKU_DB;
USE WAREHOUSE PC_DATAIKU_WH;

SELECT *
FROM CREDIT_SCORING_LOANS_TEST_SCORED
LIMIT 10;
```

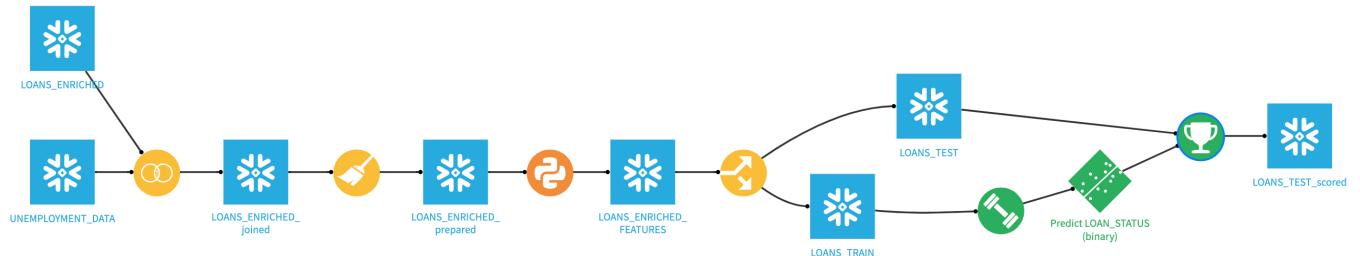
Additional info,

```
SELECT
    EMP_TITLE ,
    SUM(CASE WHEN "prediction" = 'ok' THEN 1 ELSE 0 END) AS
prediction_yes,
    SUM(CASE WHEN "prediction" = 'incident' THEN 1 ELSE 0 END) AS
prediction_no
        FROM CREDIT_SCORING_LOANS_TEST_SCORED
GROUP BY
    EMP_TITLE
order by prediction_yes DESC;
```

## Conclusion and Next Steps

Duration: 2

Congratulations you have now successfully built, deployed and scored your model results back to Snowflake. Your final flow should look like this.



## Bonus Material - Snowpark -Python

Duration: 5 To be added soon