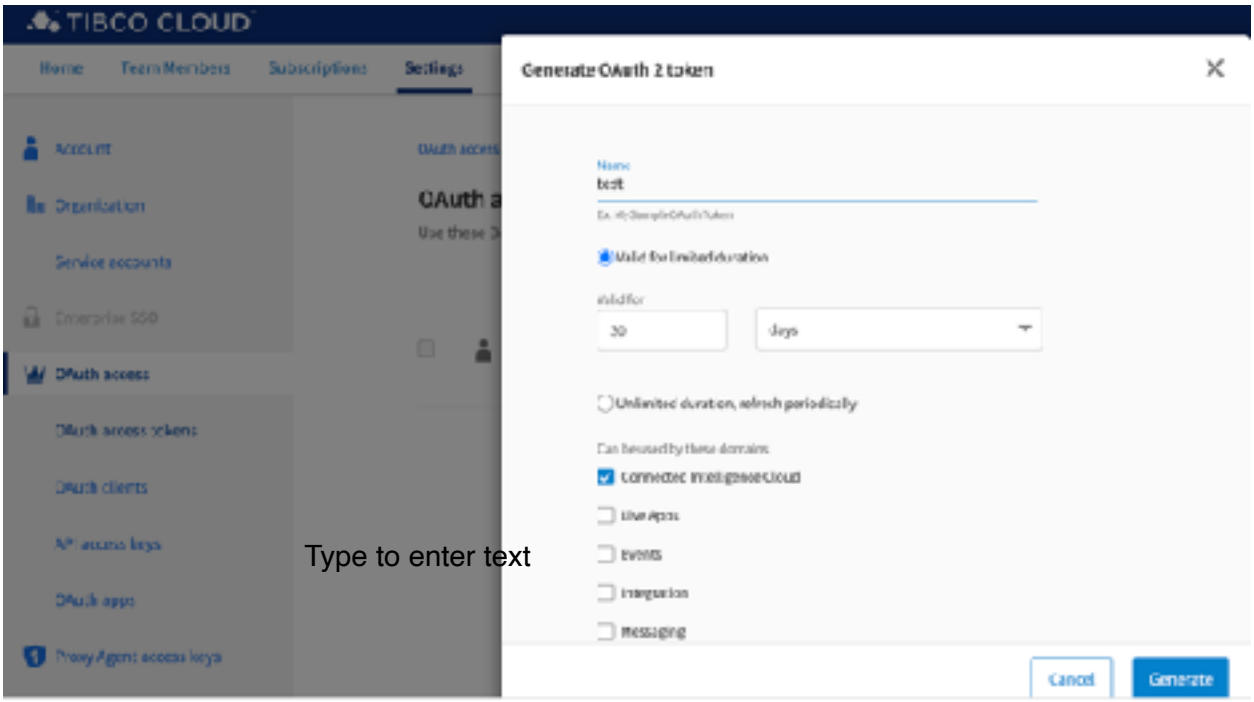
Setup

1. Register at cloud.tibco.com if you don't have an account already
2. Generate an OAuth Access Token with access to "Connect Intelligence Cloud"

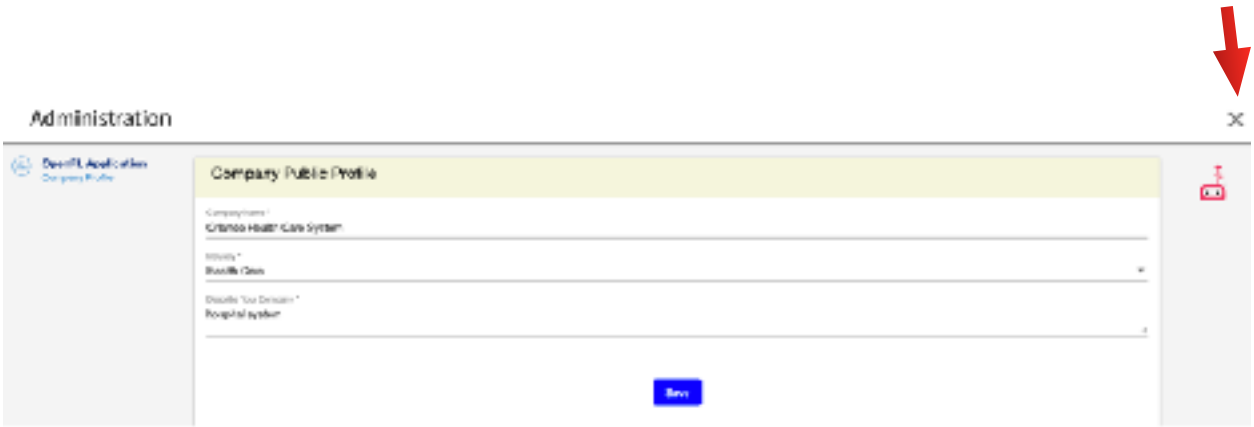


Type to enter text

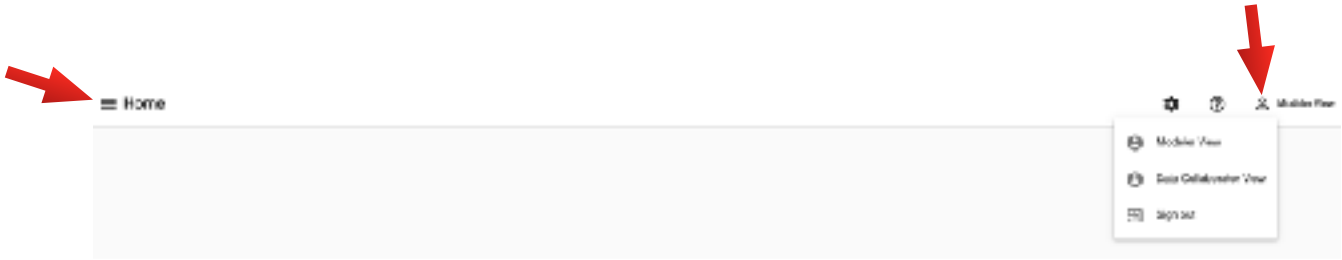
3. Go to OpenFL web console and login with your access token.

https://n8ah4g4d0j.execute-api.us-east-2.amazonaws.com/login

4. For the first time login, you will be directed to the profile page, enter your company info, and close by click on the X link



5. From the home page, you can select the view perspective, it is default to modeler view, for Synthetic Data Generation, select Data Collaborator View



6. Click on the Home menu button on the left corner to see list of available menus for Synthetic Data Generation

- **Define Dataset** :
  - Define the meta data about your dataset
- **View Datasets**:
  - View list of datasets defined
  - Create synthetic data generation tasks
    - Model driven
    - Rules driven
- **View Synthetic Data Results:**
  - View task status
  - Download task config json file
  - Use existing task as template to create new task
  - Create model-driven synthetic data generation task using trained models from a completed task

Tutorials

Tutorial 1 - Generate rule-driven synthetic data (Only single table is supported at present)

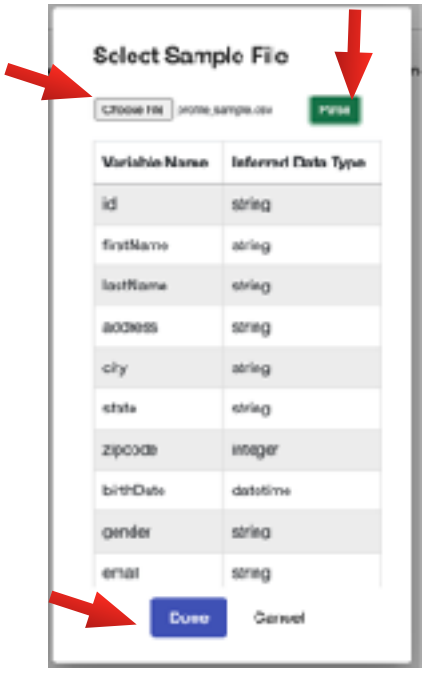
1 Define a profile single table dataset

1.1 Dataset General Info

1.2 Dataset Fields - Import variables from the sample file (sample_profile.csv) provided

Click on “Add New Variable” to manually add a new variable. Or you can import variables from a sample CSV File, it is expected to have a header row and at least one data row. Header name will become the variable name, and the data type is derived from sample data. Once it is imported, you can manually to edit each variable to add new meta information, such as variable type, and if it is a primary key field.

For rule-driven synthetic data generation, variable name, data type, variable type and primary key fields are required.



1.3 After import, update following fields

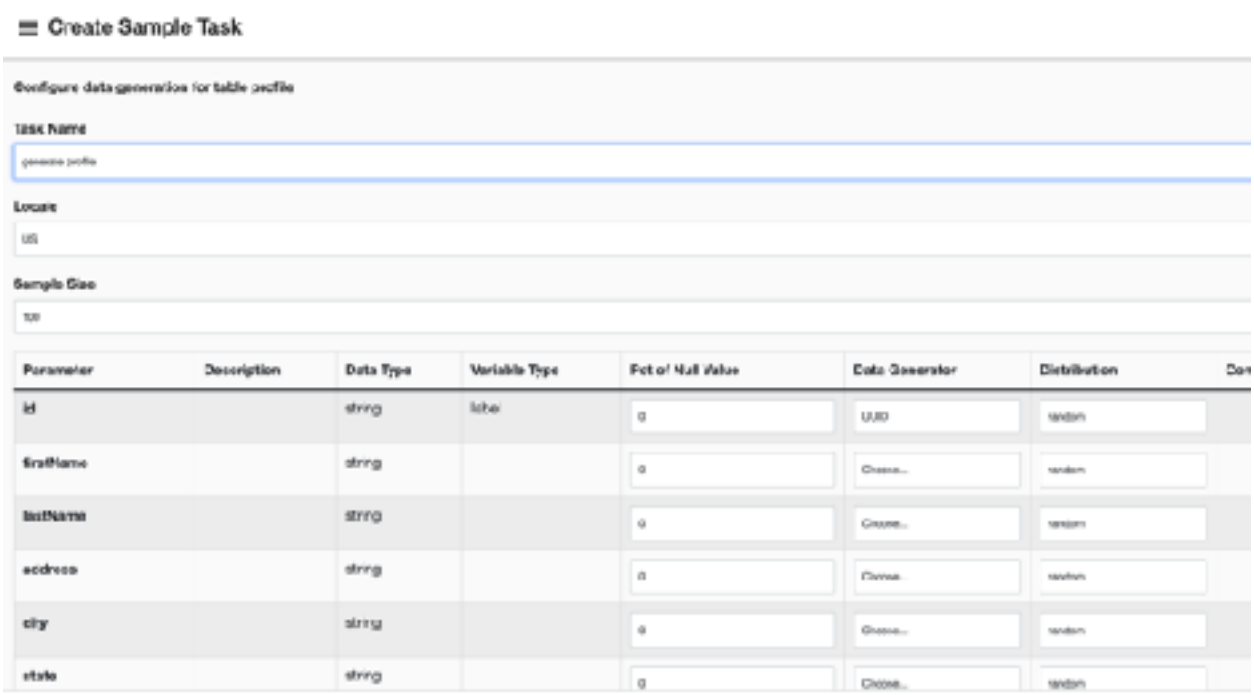
- id - check primary key field
- gender - change variable type to categorical
- zipcode - change data type to “text”, and variable type to “label”

1.4 Click Next and Next to “Review & Save” tab, save the dataset

2 Create Synthetic Data Generation Task

2.1 Go to Home -> View Datasets and click on the “graph” icon

2.2 Configure task



update generator and distribution for each field, and configure rules if necessary

- id - default to UUID generator, keep as is
- firstName - use “First Name” generator

- lastName - use “Last Name” generator
- address - use “Street Address” generator
- city - use “City” generator
- state - use “State” generator
- zipcode - use “Postal Code” generator
- birthDate - use “Birth Date” generator
  - use random distribution
  - Configure min and max age - e.g. 18 and 70
- gener - use “Choose from a list” generator
  - for categorical values, you can configure a list of codes to use
  - Add “M” , “F” codes
- email - use “Email” generator
- phone - use “Phone” generator
- employed -use ‘Boolean” generator
- income - use “Random Float” generator
  - use “decision rules” distribution
    - select dependent field “yearsEmployed”
    - Add rule 1
      - Operator: >= && <
      - Lower bound: 1
      - Upper bound: 10
      - Distribution: normal
      - Configure mean/std/lower/upper/decimals: 35000/1/25000/50000/2
    - Add rule 2
      - Operator: >= && <
      - Lower bound: 10
      - Upper bound: 20
      - Distribution: manual
      - Configure manual rules
        - 50000 - 100000, pct: 100, decimals: 2
    - Add rule 3
      - Operator: >=
      - Value: 20
      - Distribution: manual
      - Configure manual rules
        - 100000 - 125000, pct: 50, decimals: 2
        - 125000 - 150000, pct: 25, decimals: 2
        - 150000 - 500000, pct: 25, decimals: 2

- yearEmployed - use “Random Integer” generator
  - Change to “normal” distribution
  - configure mean/std/min/max values: e.g. 5/1/1/40

2.3 Save or Export

- Save for later use
- Export the task json file

2.4 Run the task

Task can be run manually or automatically, to run automatically, an agent needs to run on your local machine, for this tutorial, we will run it manually.

- Pull docker images
  - docker pull mweny88/project-synthetic-data-sample:1.0.0

- Run the task manually
  - Create a working directory
  - Copy the task ison file to this folder
  - docker run --rm -v <replace with your working dir>:/output mweny88/project-synthetic-data-sample:1.0.0 ./run.sh <replace your task_id uuid value>
  - Sample data will be created in your working directory under task_id sub-directory

Tutorial 2 - Model Driven Single Table Data Generation

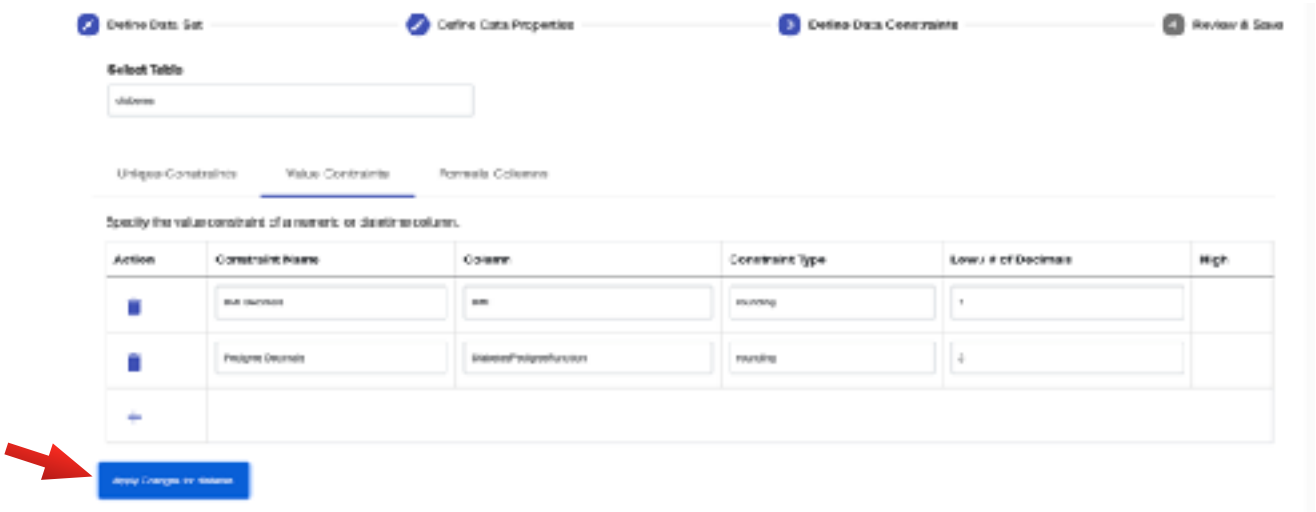
1 Define a diabetes single table dataset

1.1 Dataset General Info (see Tutorial 1)

1.2 Dataset Fields - Import variables from the sample file (diabetes.csv) provided

1.3 After import, modify following fields
- Outcome - check “Is Machine Learning Classifier Field”

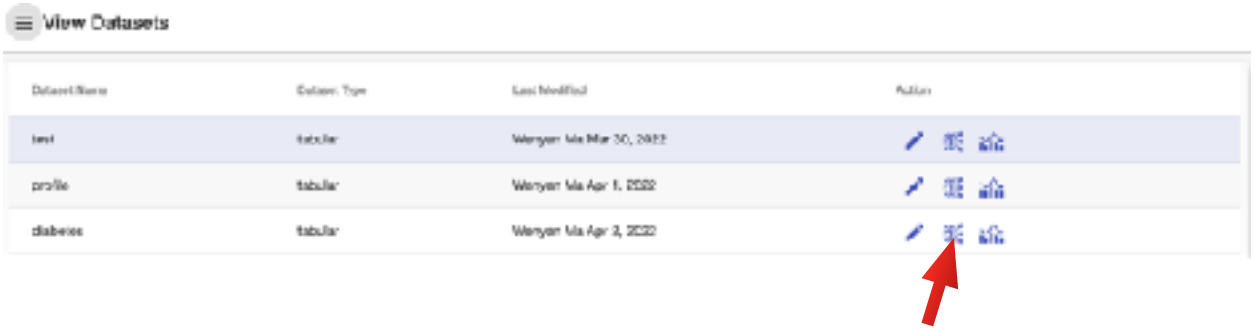
1.4 click Next to “Define Data Constraints” Step
- Select diabetes table
- Select “Value Constraints” tab
- Add two rounding constraints, click “Apply Changes for diabetes” button



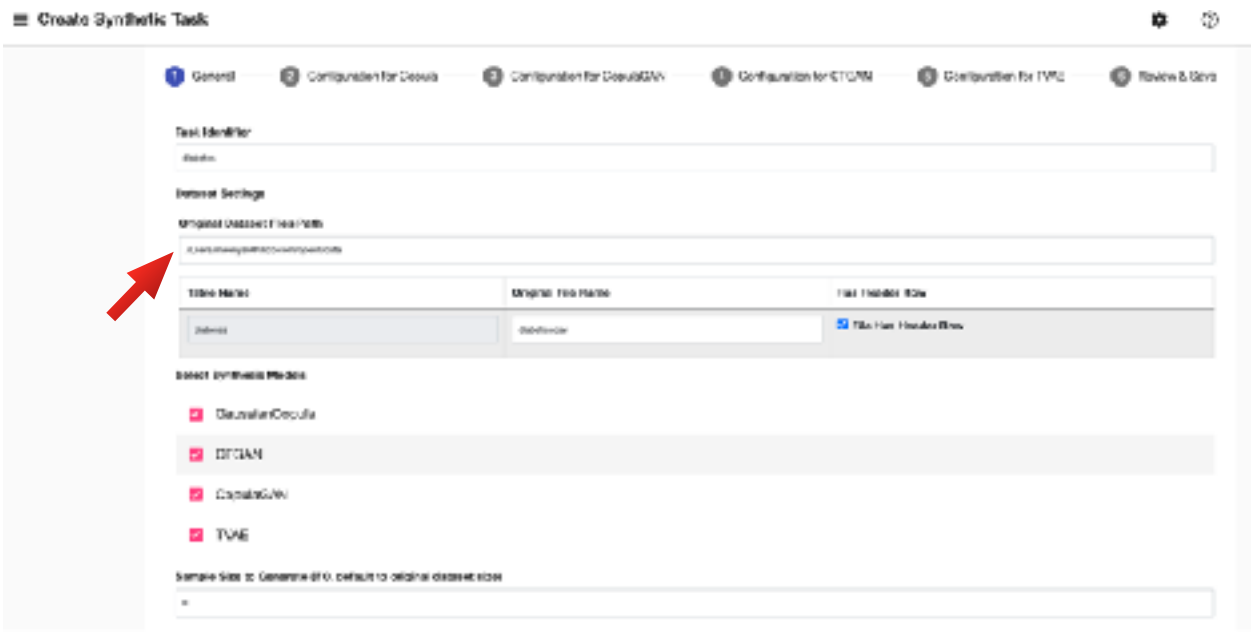
1.5 Click “Next” to review step, and Save

2 Create Synthetic Data Generation Task

2.1 Go to Home -> View Datasets and click on the “brain” icon



2.2 Create Synthetic Task



- Update “Original Dataset Files Path” to the directory where diabetes.csv is located
- Click Next -> Next -> Next -> Next to Review Step and “Submit Task”

2.3 Run the task

Task can be run manually or automatically, to run automatically, an agent needs to run on your local machine, for this tutorial, we will run it using the agent so we can view the result in the console.

- Pull docker images
  - docker pull mweny88/project-synthetic-model-sample:1.0.0

- Run the task using the agent
  - Create a working directory
  - Download the opendata-agent executeable (For MAC only)
  - Create config.properties file with following entries

    region = US
    oauthToken = your oath token
    sImage = mweny88/project-synthetic-data-sample:1.0.0
    mImage = mweny88/project-synthetic-data-model:1.0.0
    workDir = your working dir

  - Start up the agent
    - ./opendata-agent -c config.properties

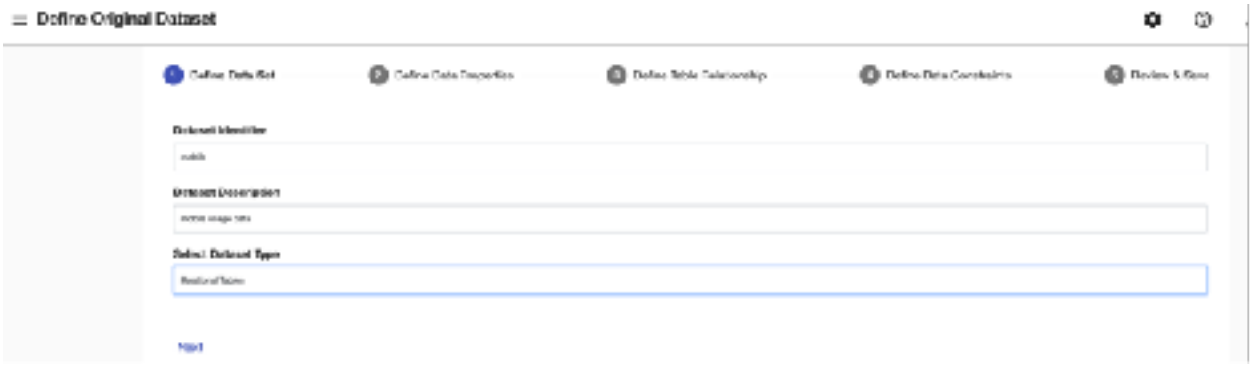
the agent will pull all tasks with “SUBMITTED” status, and run each sequentially, the results will be available in your working directory under <task_id> subdirectory. You can also view the metrics and histograms in the web console

Tutorial 3 - Model Driven Multi-Tables Data Generation

In this tutorial, we will learn and generate from 3 mobile usage tables: Users, Sessions and Transaction. One user can have multiple sessions, and within each session, there can be multiple transactions.

1 Define a diabetes single table dataset

1.1 Dataset General Info, select “Relation Tables” as Dataset Type



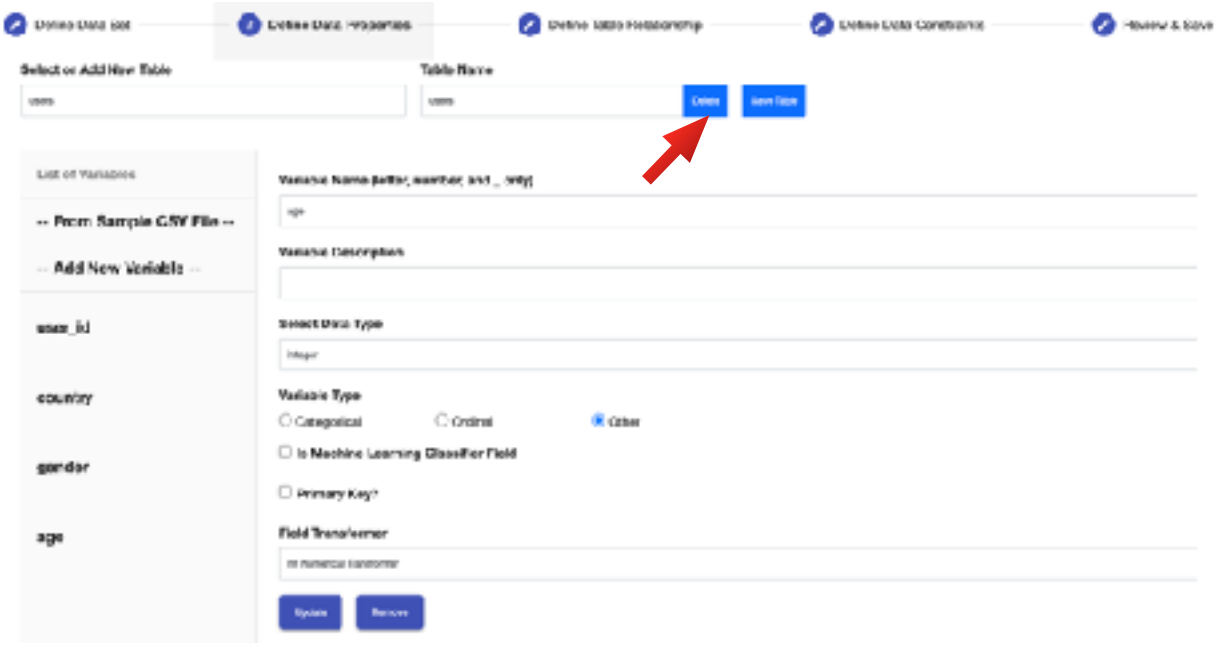
1.2 Dataset Tables and Fields

Add 3 tables: **users, sessions and transactions**

import from users.csv, sessios.csv and transactions.csv to create variables for each table respectively.

users table:
- user_id : check “Is Primary Key”
- country: change variable type to “categorical”
- gender: change variable type to “categorical”

After update the fields, click on “Save Table” button

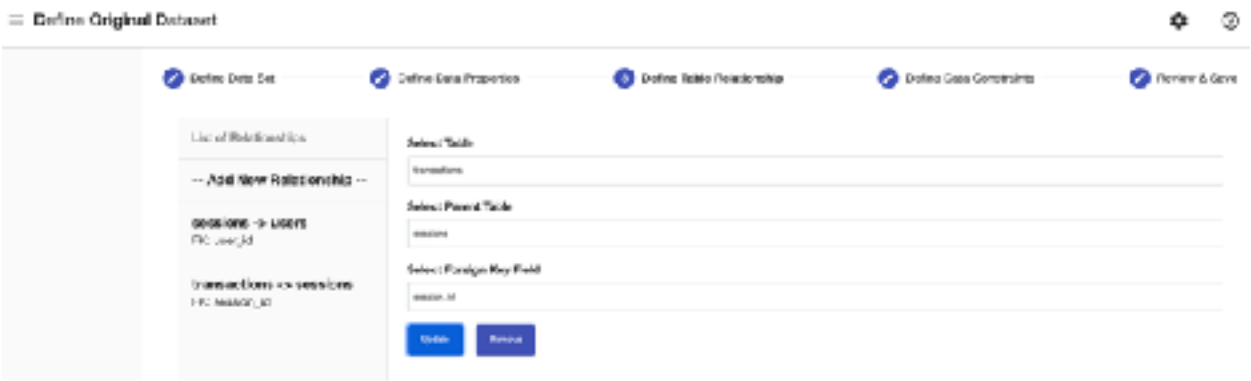


sessions table:
- session_id: check primary key
- device: change variable type to categorical
- os - change variable type to categorical

transactions table:
- transaction_id: check primary key
- timestamp: update Datetime Format to %Y-%m-%dT%H:%M
- amount: change data type to float

1.3 Define Table Relationships

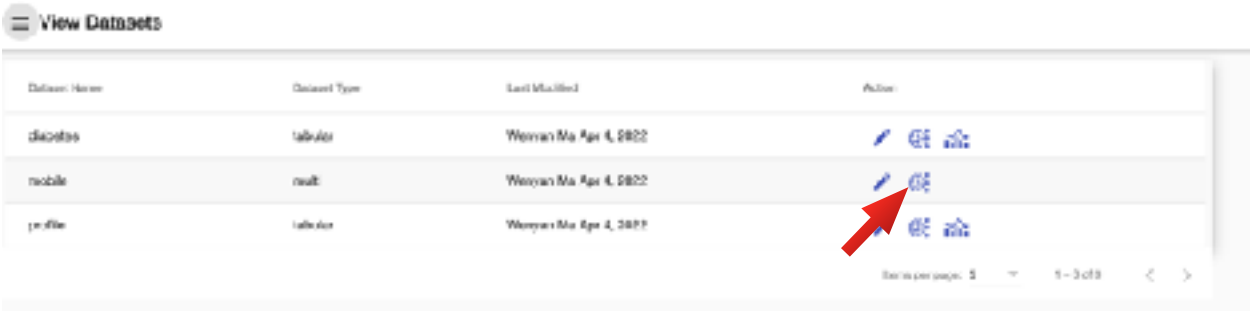
1.3.1 users - sessions one to many relationship
1.3.2 sessions - transaction one to many relationship



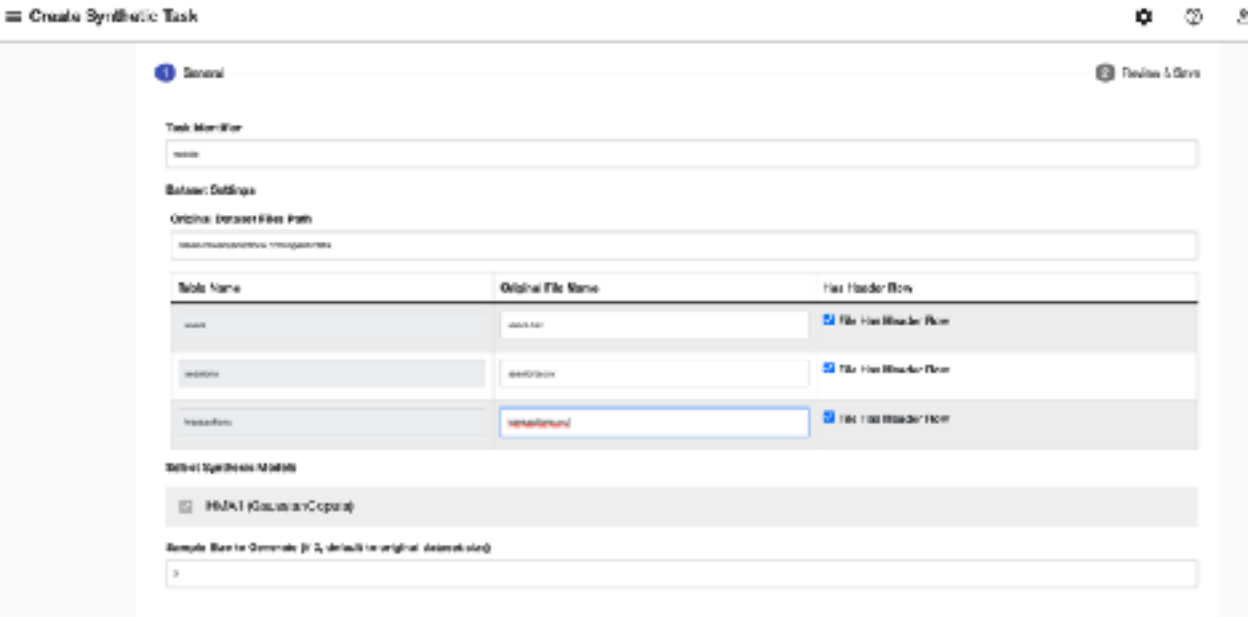
1.4 Review and save the dataset

2 Create Synthetic Data Generation Task

2.1 Go to Home -> View Datasets and click on the “brain” icon



2.2 Specify dataset file path and file names



2.3 Review and Submit task

3. Run the task

See tutorial 2

Tutorial 4 - Model Driven Time Series Data Generation

In this tutorial, we will learn and generate time series data

1 Define a diabetes single table dataset

1.1 Dataset General Info, select “Time Series” as Dataset Type

1.2 Define data fields, import from stocks.csv file

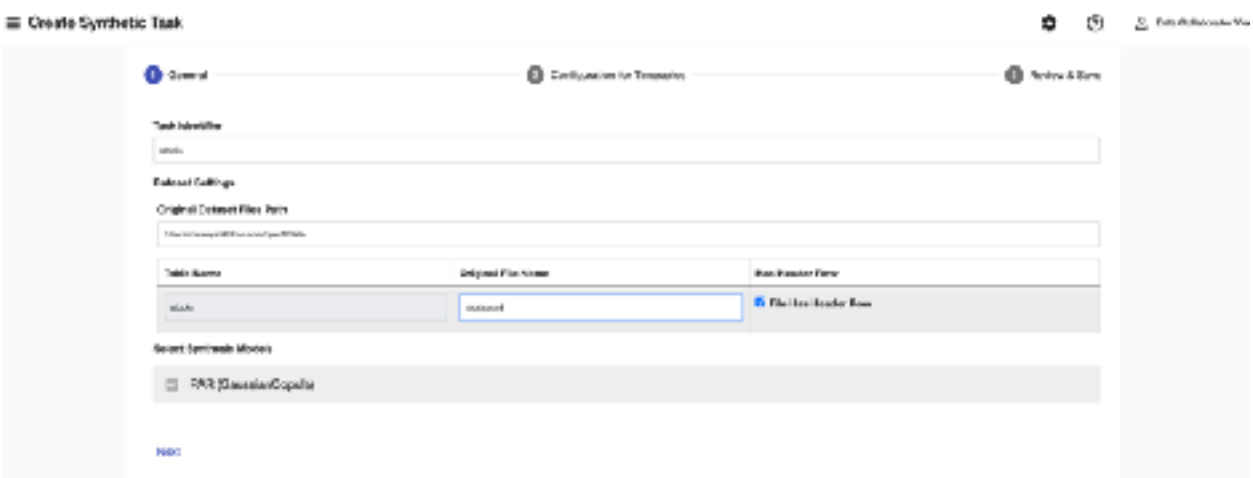
- Symbol: change variable type to “categorical”, and select “Entity Column”
- Date: select “Sequence Index Column”
- Open: select ‘Data Column”
- Close: select “Data Column”
- Volume: select “Data Column”
- MarketCap: select “Context Column”
- Sector: select “Context Column”
- Industry: select “Context Column”

1.3 Review and Save

2. Create Synthetic Task

2.1 Go to Home -> View Datasets and click on the “brain” icon

2.2 Specify dataset file path and file names



2.3 Configure parameters for time series task

You can either define regex for Entity Column - Symbol, or Specify specific list of values for Symbol and its associated context columns, we will use regex in this tutorial

2.3.1 check “Use Regular Expression to Generate Entity Column”
2.3.2 use [A-Z]{2,4} as the regex
2.3.3 generate 1 records for each sequence

2.3.4 Review and Submit task

3. Run the task

See tutorial 2