

# Introduction to Elyra: AI-centric extensions to JupyterLab



Edward Leardi  
Saishruthi Swaminathan  
Yiwen Li

# About us



IBM – Center for Open Source Data and AI Technologies (CODAIT)

---



Yiwen Li  
Data Scientist



Edward Leardi  
Data Scientist



Saishruthi Swaminathan  
Developer Advocate & Data Scientist

---



[Yiwen.Li@ibm.com](mailto:Yiwen.Li@ibm.com)



[github.com/yil532](https://github.com/yil532)



[linkedin.com/in/yiwenli](https://linkedin.com/in/yiwenli)

[edward@ibm.com](mailto:edward@ibm.com)

[github.com/edwardleardi](https://github.com/edwardleardi)

[linkedin.com/in/edwardleardi](https://linkedin.com/in/edwardleardi)

[saishruthi.tn@ibm.com](mailto:saishruthi.tn@ibm.com)

<https://github.com/SSaishruthi>

<https://www.linkedin.com/in/saishruthi-swaminathan/>

---

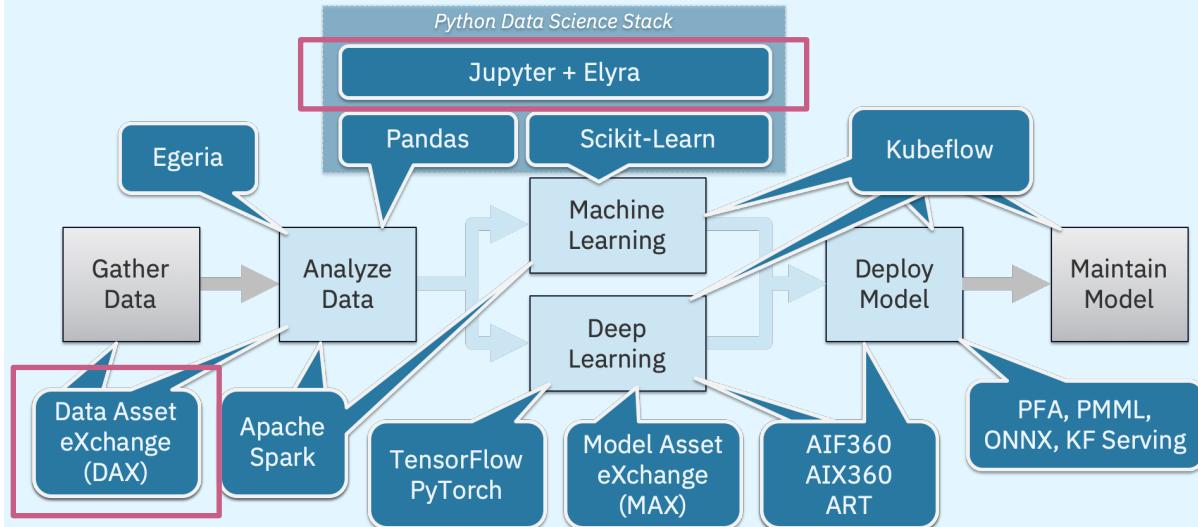
# Overview

- CODAIT
- Elyra overview
- Data Asset eXchange overview
- DAX pipeline live demo
- Get involved
- Other CODAIT talks



- CODAIT aims to make AI solutions dramatically easier to create, deploy, and manage in the enterprise.
- We contribute to and advocate for the open-source technologies that are foundational to IBM's AI offerings.
- 30+ open-source developers!

#### Improving the Enterprise AI Lifecycle in Open Source



# Elyra

Elyra is a set of AI centric extensions to JupyterLab. It aims to help data scientists, machine learning engineers and AI developer's through the model development life cycle complexities.

<http://bit.ly/elyra-dax>

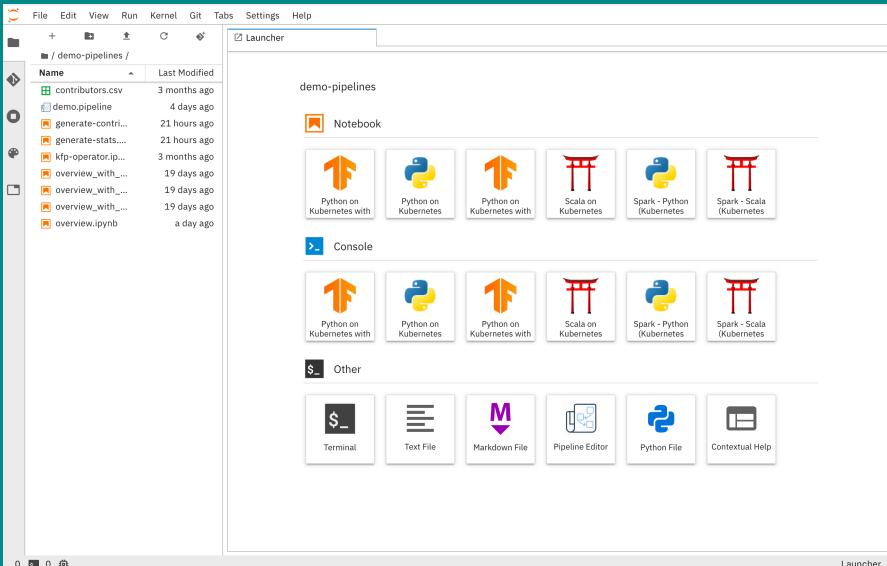


## Elyra at GitHub

<https://github.com/elyra-ai/elyra>

## Elyra Documentation

<https://elyra.readthedocs.io/en/latest/>





## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## JupyterLab Extensions

The screenshot shows the JupyterLab interface with the Elyra extensions installed. The top navigation bar includes File, Edit, View, Run, Kernel, Git, Tabs, Settings, and Help. A 'Launcher' tab is selected. The left sidebar displays a file tree with the following contents:

Name	Last Modified
community-stats	a month ago
debug	a month ago
demo	14 days ago
demo-pipelines	24 days ago
elyra-demo	3 months ago
elyra-pipelines	14 days ago
elyra-python	a month ago
kubernetes	4 months ago
python	5 months ago
r	a year ago
sample-notebooks	a month ago
sample-python	14 days ago
scala	3 months ago
tensorflow	a month ago
Tutorials	3 minutes ago

The main area is divided into three sections: 'Notebook' (with icons for R, Python, Python with Kubernetes, Python with Kubernetes with, Scala, Scala with, Spark-Python, Spark-R, and Spark-Scala), 'Console' (with icons for R, Python, Python with Kubernetes, Python with Kubernetes with, Scala, Scala with, Spark-Python, Spark-R, and Spark-Scala), and 'Other' (with icons for Terminal, Text File, Markdown File, Pipeline Editor, Python File, and Show Contextual Help). At the bottom right is a 'Launcher' button.



## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

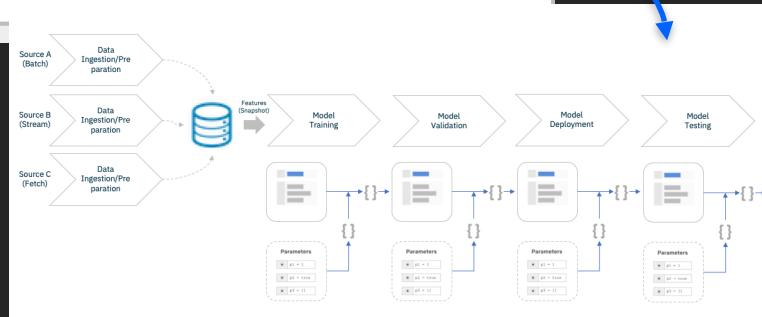
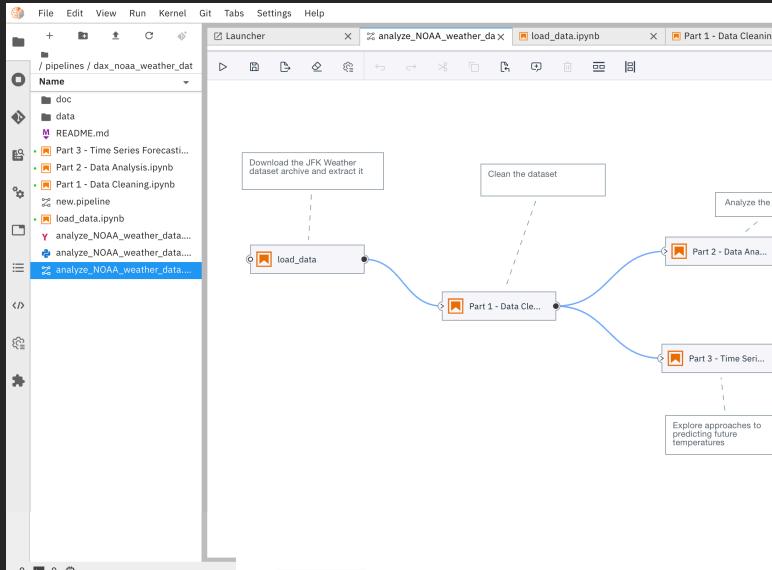
## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## Notebook Pipelines





## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

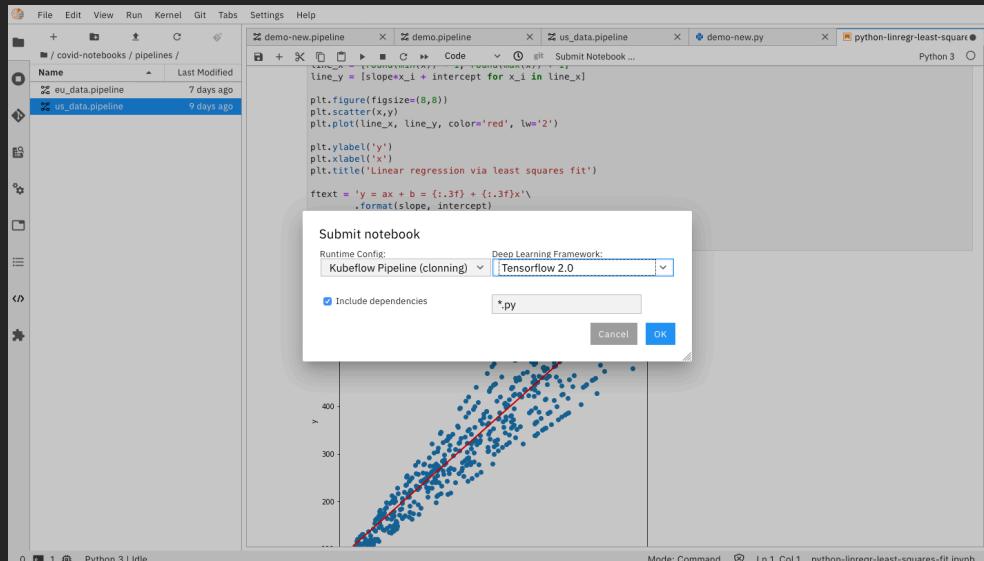
## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## Notebook as batch jobs





## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## Code Snippets

The screenshot shows the Elyra Notebook Pipelines editor interface. On the left, there's a file tree for a project named 'dax\_noaa\_weather\_data'. The tree includes files like 'README.md', 'Part 3 - Time Series Forecast.ipynb', 'Part 2 - Data Analysis.ipynb', 'Part 1 - Data Cleaning.ipynb' (which is currently selected), 'load\_data.ipynb', and several 'analyze\_NOAA\_weather\_data...' files. In the center, there's a code editor window with a 'Table of Contents' sidebar. The table of contents lists steps such as 'Read the Raw Data', 'Clean the Data', and 'Save the Cleaned Data'. Below the table of contents, there's a section for 'Import required modules' with a note about importing PyGithub and pandas. The main code area shows Python code for reading a CSV file and setting pandas display options. At the bottom, there's a terminal-like interface showing command-line output for saving the file.

```
File Edit View Run Kernel Git Tabs Settings Help
Launcher x analyze_NOAA_weather_data.ipynb x load_data.ipynb x Part 1 - Data Cleaning.ipynb x Part 2 - Data Analysis.ipynb x Part 3 - Time Series Forecast.ipynb x

Table of Contents:
1. Read the Raw Data
2. Clean the Data
  2.1 Select data columns
  2.2 Clean up precipitation column
  2.3 Convert columns to numerical types
  2.4 Reformat and process data
  2.5 Create a fixed interval dataset
  2.6 Feature encoding
  2.7 Rename columns
3. Save the Cleaned Data
Authors

Import required modules
Import and configure the required modules.

(1): pip install PyGithub pandas > /dev/null 2>&1

(2): # Define required imports
import pandas as pd
import numpy as np
import sys
import re
# These set pandas max column and row display in the notebook
pd.set_option('display.max_columns', 50)
pd.set_option('display.max_rows', 50)

1. Read the Raw Data
We start by reading in the raw dataset, displaying the first few rows of the dataframe, and taking a look at the columns and column types present.

(3): raw_data = pd.read_csv('data/noaa-weather-data-jfk-airport/jfk_weather.csv', parse_dates=['DATE'])
raw_data.head()

/opt/anaconda3/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3058: DtypeWarning: Columns (8,10,11,12,13,14,15,16,17,20,22,23,25,40,63,9,64,85,86) have mixed types. Specify dtype option on import or set low_memory=False.
interactivity=interactive, compiler=compiler, result=result
  STATION STATION_NAME ELEVATION LATITUDE LONGITUDE DATE REPORTTYPE HOURLYSKYCONDITIONS HOURLYVISIBILITY HOURLYPRSENTWEATHERTYPE HOURLYD
```



## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## Git integration

The screenshot shows a Jupyter Notebook interface integrated with Elyra's Git and Python execution features. On the left, a sidebar displays a file tree for a repository named 'sample-notebooks'. The 'Changes' section shows one staged file, 'generate-contributions.ipynb'. The main area contains two code cells:

```
In [ ]: pip install PyGitHub pandas >/dev/null 2>&1
```

```
In [ ]: import os  
import datetime  
import pandas as pd  
from github import Github  
github = Github(os.environ['GITHUB_TOKEN'])
```

Below the code cells, a 'Summary (required)' field is present. On the right, a panel titled 'Jupyter Enterprise Gateway Contribution Stats' displays a snippet of Python code for generating contributions data:

```
github_jip_org = github.get_organization('jupyter')  
github_eg_repo = github.org.get_repo('enterprise_gateway')  
datetime_start = datetime.datetime.now() +  
    datetime.timedelta(-180) #datetime.datetime(2018,7,1)  
(...)
```

```
contributions_df.to_csv('community_contributions.csv', index=False)
```

At the bottom right, a 'Launcher' tab is visible.



## Notebook Pipelines editor

Visual editor for building notebook-based AI pipelines, enabling the conversion of multiple notebooks into batch jobs or workflows.

## Notebook as batch jobs

Elyra extends the notebook UI to simplify the submission of notebooks as a batch job for model training

## Code Snippets

Easy creation and insertion of reusable code snippets for the various languages

## Git integration

Track project changes and share among teammates

## Python script execution

Edit and execute python scripts against local or cloud-based resources

## Python Script editor

The screenshot shows the Elyra Python Script editor interface. On the left is a code editor window titled 'PANDA.PY' containing the following Python code:

```
PANDA.PY
import io
def delay(seconds)
def df_from_url(url)
```

To the right of the code editor is a 'Launcher' window showing a terminal session with the command 'python3'. The terminal output displays the following data frame:

```
Launcher > python3
1 # Add sample panda code to manipulate the generated df
2 import io
3 import requests
4 import pandas as pd
5 import time
6
7 def delay(seconds):
8     time.sleep(seconds)
9
10 def df_from_url(url):
11     data = requests.get(url).content
12     df = pd.read_csv(io.StringIO(data.decode('utf-8')))
13     return df
14
15 # Uncomment the lines below to sleep for a bit
16 # useful to demonstrate kernel startup on container environments
17 # delay(3)
18
19 # Sample panda code to manipulate the generated data frame
20 # and calculate mean price per zipcode
21 df = df_from_url('http://samplecsvs.s3.amazonaws.com/SacramentoRealEstateTransactions.csv')
22 df.groupby('zip')['price'].mean()

Python Console Output >
[ 1: zip
  95603 405890.800000
  95608 295684.750000
  95610 226435.285714
  95614 300833.000000
  95619 216033.000000
  ...
  95838 149461.351351
  95841 213806.142857
  95842 143285.772727
  95843 221496.333339
  95864 364400.000000
Name: price, Length: 68, dtype: float64
```

# Data Asset eXchange

Data Asset Exchange offers high-quality datasets with clearly-defined open data licenses in standardized formats, according to IBM.

- Vetted data.
- Exclusive access to IBM Research datasets that have been used in creating popular AI products like [Debater System](#), Entity Recognition, and so on.
- Datasets with open data licenses for both business applications and advancing core science.
- Packaged with tutorials that shows how to read and analyze data. As well as, train machine or deep learning models on IBM Cloud using IBM Cloud AI services as well as multi-cloud AI open-sourced tools.

[ibm.biz/data-exchange](http://ibm.biz/data-exchange)

## Data Asset eXchange

Explore useful and relevant data sets for enterprise data science

[Learn More](#)

[What's New](#)



[Get Involved](#)



Dataset | CSV

NOAA Weather Data -  
JFK Airport

September 12, 2019

Dataset | IOB format

Groningen Meaning  
Bank - Modified

May 14, 2020

Dataset | CSV

Fashion-MNIST

September 12, 2019

Dataset | JPG, JSON

PubLayNet

October 25, 2019

Dataset | WAV

TensorFlow Speech  
Commands

March 17, 2020

Dataset | PNG, JSON

PubTabNet

November 11, 2019

# Data Preview and Data Glossary

Dataset Preview	
Notebook Preview	
Run Notebook in Watson Studio	
Dataset Homepage	
PubLayNet	
Dataset Preview	
Dataset Metadata	
Dataset Glossary	
Feature	Description
images	JSON field containing a list of images and their metadata (size, ID, name)
annotations	Each object instance annotation contains a series of fields, including the category id and segmentation mask of the object.
annotations -> segmentations	Contains the polygon coordinates for the segmentation mask for the specific class instance (table, list, text etc)
annotations -> bbox	Contains the bounding box coordinates for the specific class instance (table, list, text etc).
annotations -> is_crowd	This field indicates whether the class instance is a single object ( <code>is_crowd=0</code> ) or multiple objects ( <code>is_crowd=1</code> ). In this dataset we only have single objects so this field is always set to 0.
annotations -> category_id	The class label for the current class instance. This indicates what the current bbox/segmentation mask encapsulates (table, list, text etc).
categories	JSON field containing a list of classes and their metadata (ID, name) This dataset has 5 categories (w/ corresponding "ids") - text ("1"), title ("2"), list ("3"), table ("4"), figure ("5").

# Access notebook in Watson Studio

IBM Cloud Pak for Data

Log In

Sign Up

Gallery / DAX Weather Project / 

[← Back](#)

**DAX Weather Project**

Tags	Required Services	Modified
<a href="#">Environment</a> <a href="#">Transportation</a>	0	May 22, 2020

This project includes the NOAA Weather Dataset - JFK Airport (New York) from the Data Asset Exchange and supporting notebooks. The notebooks teach the user to extract, clean and analyze sample weather data and predict weather trends to help airports schedule better flight times. This sample project contains 3 notebooks and 1 CSV file. Please run the notebooks in sequential order of their part numbers using a Python 3.6 runtime.

**Images** [Assets](#) [Info](#)

# Access from Cloud Pak for Data

The screenshot shows the IBM Cloud Pak for Data product hub interface. At the top, there's a navigation bar with the IBM logo, a search bar, and links for 'What's new', 'Community', and 'Get support'. Below the header, a 'Table of contents' sidebar is visible, showing sections like 'Overview', 'Use cases', 'Planning', 'Installing', 'Services and integrations' (which is expanded to show 'Services in the catalog' and 'Services outside the catalog'), and 'External data sets' (which is also expanded to show 'Industry accelerators', 'Integrations', 'Administering', 'Analytics projects', 'Accessing data', 'Governing and curating data', 'Integrating and preparing data', 'Analyzing data', 'AI solutions', 'Developer resources', and 'Troubleshooting'). The main content area is titled 'External data sets' and discusses the benefits of using external data sets for business analysis. It highlights a partnership with The Weather Company for historical weather data. A table provides details about this offering, including the provider ('The Weather Company®'), the data source ('Weather Company Data Limited Edition'), and the fact that it's included with Cloud Pak for Data. The table also includes sections for 'About this offering', 'Use cases', 'Industry accelerators', and a 'Get started' section.

Data offering	Provided by	Pricing	Learn more
Weather Company Data Limited Edition	The Weather Company®	Included with Cloud Pak for Data	<p><b>About this offering</b></p> <p>90-day access to cloud-based APIs that enable you to obtain historical weather data, current conditions, and forecast conditions.</p> <p><b>Use cases</b></p> <p>You can use weather data to optimize operations, reduce overhead costs, increase safety, and uncover new revenue opportunities. For example, you can:</p> <ul style="list-style-type: none"><li>Predict power outages with greater accuracy so that you can restore power to customers faster</li><li>Reduce utility costs with smarter vegetation management</li><li>Improve flight safety, efficiency and performance</li><li>Keep policyholders safe while reducing insurance claims and fraud</li><li>Improve supply chain visibility and minimize weather-related disruptions</li><li>Transport people and goods more safely</li></ul> <p><b>Industry accelerators</b></p> <p>The following industry accelerators can help you get started with this data set:</p> <ul style="list-style-type: none"><li><a href="#">Manufacturing Analytics with Weather</a></li><li><a href="#">Retail Predictive Analytics with Weather</a></li><li><a href="#">Sales Prediction using The Weather Company Data</a></li></ul> <p><b>Get started</b></p> <p>For details, see <a href="https://www.ibm.com/weather">https://www.ibm.com/weather</a>.</p>

[https://www.ibm.com/support/producthub/icpdata/docs/content/SSQNUZ\\_current/svc-nav/data-sets.html](https://www.ibm.com/support/producthub/icpdata/docs/content/SSQNUZ_current/svc-nav/data-sets.html)

<http://bit.ly/elyra-dax>

# Industrial Accelerator - Cloud Pak for Data

Cloud Pak for Data

View Only

Group Home    Blogs 0    Members 3

## Effective Farming - Monitor Crop Growth

28 days ago

The accelerator is created using Data Asset eXchange data to support effective farming by monitoring crop growth using crop guide and provide timely alert to farmers about weather change, possible development of crop disease, evaporation of fungicide, and efficient use of solar panels (agrvoltaics support).

### What's included?

- A structured business glossary of 90 business terms.
- Sample data science assets

### How does it work?

The glossary provides the information architecture that you need to understand weather related business measures. Your data scientists can use the sample notebooks, predictive models and dashboards to accelerate data preparation, machine learning modeling, and data reporting. Moreover, the data scientists may modify the sample notebooks for other business use cases and corresponding datasets.

Timely alert to farmers can save crop life and bring in more cost savings.

When you import the accelerator:

- The terms are added to your business glossary under the Effective Farming - Monitor Crop Growth category in the Industry Accelerators category.
- The data science assets are added to a new analytics project.

Statistics

0 Favorited  
17 Views  
0 Files  
0 Shares  
0 Downloads

<https://community.ibm.com/community/user/cloudpakfordata/viewdocument/effective-farming-monitor-crop-gr>

# Data Science Process

Data Extraction

Data Cleaning

Data Exploration

Model Development

Result Interpretation



# Get involved

## Getting started with Elyra

[https://elyra.readthedocs.io/en/latest/getting\\_started/installation.html](https://elyra.readthedocs.io/en/latest/getting_started/installation.html)



## Elyra's Github

<https://github.com/elyra-ai/elyra>

## Data Asset eXchange

<https://developer.ibm.com/exchanges/data/>

## DAX notebooks Github

[https://github.com/elyra-ai/examples/tree/master/pipelines/dax\\_noaa\\_weather\\_data](https://github.com/elyra-ai/examples/tree/master/pipelines/dax_noaa_weather_data)

## Contributing to these projects

- Bug reports
- Enhancement requests
- Code reviews



[gitter.im/elyra-ai/community](https://gitter.im/elyra-ai/community)

# IBM CODAIT talks in JupyterCon 2020



[ibm.biz/codait-at-jupytercon2020](http://ibm.biz/codait-at-jupytercon2020)

- Debugging notebooks and python scripts in JupyterLab
- Using and creating JupyterLab extensions
- A Generic Metadata-Store for JupyterLab extensions
- Intro to Elyra - an AI centric extension for JupyterLab
- What's new on Elyra - A set of AI centric JupyterLab extensions
- Building AI Pipelines with Elyra - A deep dive using COVID-19 Analytics Scenario
- Explore and Extend AI Pipeline Runtimes with Elyra and JupyterLab

