



AI SUMMIT

CINCINNATI, OH • NOVEMBER 14-16, 2023

Intro to Jupyter



HEALTHCARE
PRODUCTS
COLLABORATIVE

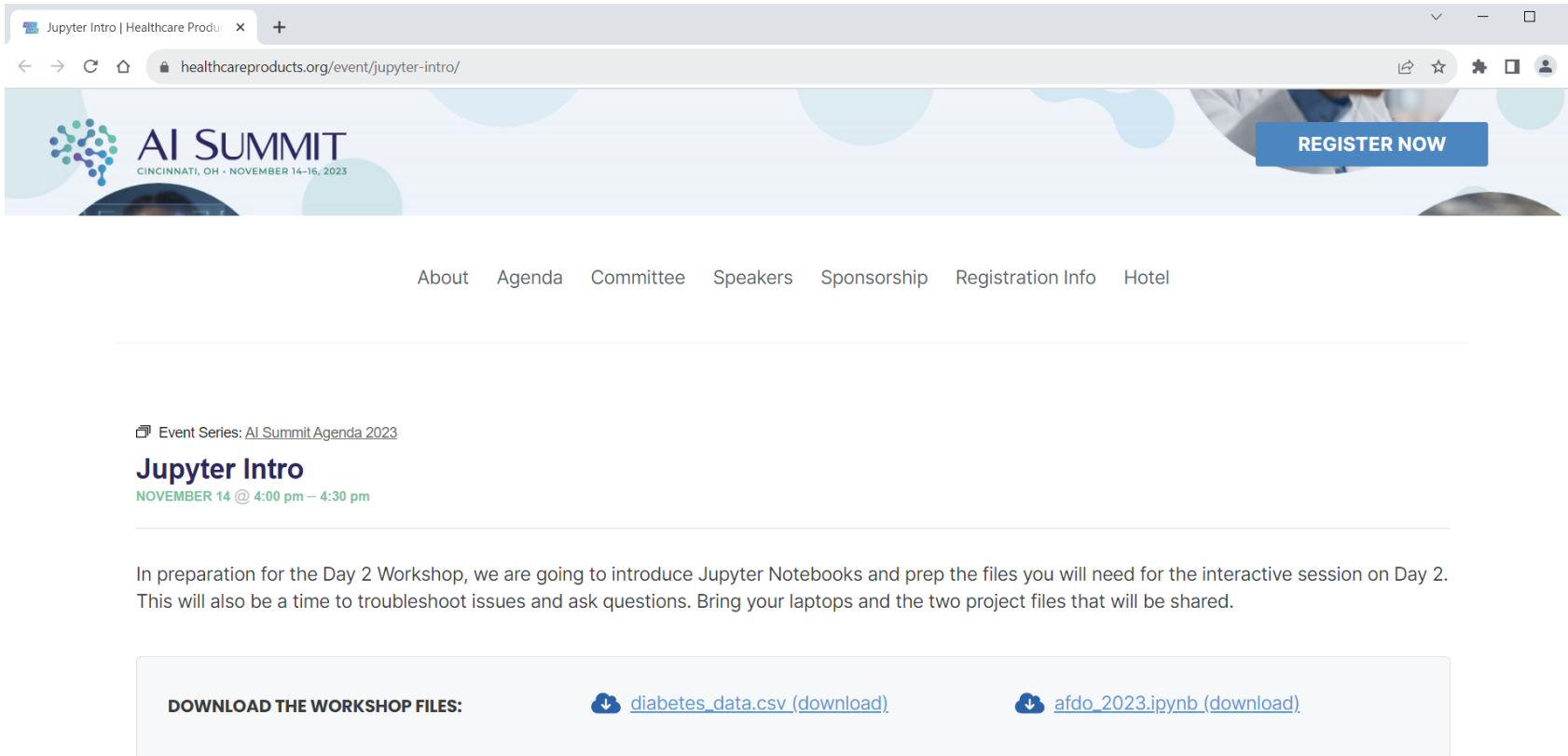


Jupyter

- Web-based development environment for notebooks, code, and data
- Can run Python



Download files from healthcareproducts.org/events/jupyter-intro



The screenshot shows a web browser window with the URL healthcareproducts.org/event/jupyter-intro/. The page features a header with the AI Summit logo and a "REGISTER NOW" button. Below the header is a navigation menu with links: About, Agenda, Committee, Speakers, Sponsorship, Registration Info, and Hotel. The main content area displays the event series "AI Summit Agenda 2023" and the event title "Jupyter Intro" scheduled for "NOVEMBER 14 @ 4:00 pm – 4:30 pm". A paragraph of text describes the event: "In preparation for the Day 2 Workshop, we are going to introduce Jupyter Notebooks and prep the files you will need for the interactive session on Day 2. This will also be a time to troubleshoot issues and ask questions. Bring your laptops and the two project files that will be shared." Below this text is a section titled "DOWNLOAD THE WORKSHOP FILES:" with two download links: [diabetes_data.csv \(download\)](#) and [afdo_2023.ipynb \(download\)](#).

Jupyter Intro | Healthcare Products Collaborative

healthcareproducts.org/event/jupyter-intro/

AI SUMMIT
CINCINNATI, OH • NOVEMBER 14–16, 2023

REGISTER NOW

About Agenda Committee Speakers Sponsorship Registration Info Hotel

Event Series: [AI Summit Agenda 2023](#)

Jupyter Intro
NOVEMBER 14 @ 4:00 pm – 4:30 pm

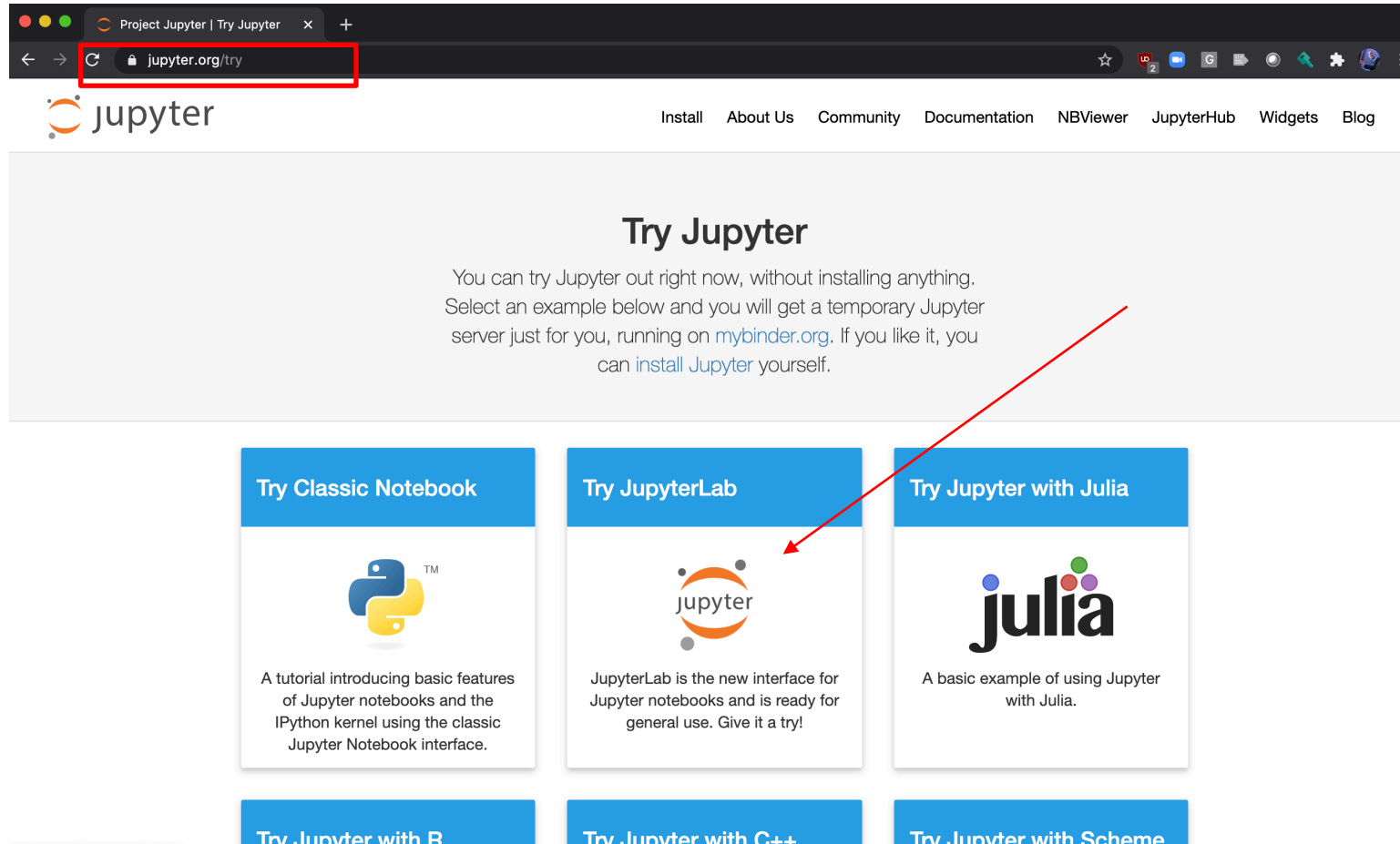
In preparation for the Day 2 Workshop, we are going to introduce Jupyter Notebooks and prep the files you will need for the interactive session on Day 2. This will also be a time to troubleshoot issues and ask questions. Bring your laptops and the two project files that will be shared.

DOWNLOAD THE WORKSHOP FILES:

[diabetes_data.csv \(download\)](#) [afdo_2023.ipynb \(download\)](#)

Preparing the environment

Go to jupyter.org and click 'Try' JupyterLab




The screenshot shows a web browser window with the address bar displaying jupyter.org/try. The page title is "Project Jupyter | Try Jupyter". The main heading is "Try Jupyter". Below the heading, the text reads: "You can try Jupyter out right now, without installing anything. Select an example below and you will get a temporary Jupyter server just for you, running on mybinder.org. If you like it, you can [install Jupyter](#) yourself."

There are six buttons arranged in a 2x3 grid, each with a logo and a description:

- Try Classic Notebook**: A tutorial introducing basic features of Jupyter notebooks and the IPython kernel using the classic Jupyter Notebook interface. (Python logo)
- Try JupyterLab**: JupyterLab is the new interface for Jupyter notebooks and is ready for general use. Give it a try! (Jupyter logo)
- Try Jupyter with Julia**: A basic example of using Jupyter with Julia. (Julia logo)
- Try Jupyter with R**: (R logo)
- Try Jupyter with C++**: (C++ logo)
- Try Jupyter with Scheme**: (Scheme logo)

A red arrow points from the top right towards the "Try JupyterLab" button.

Rename the download

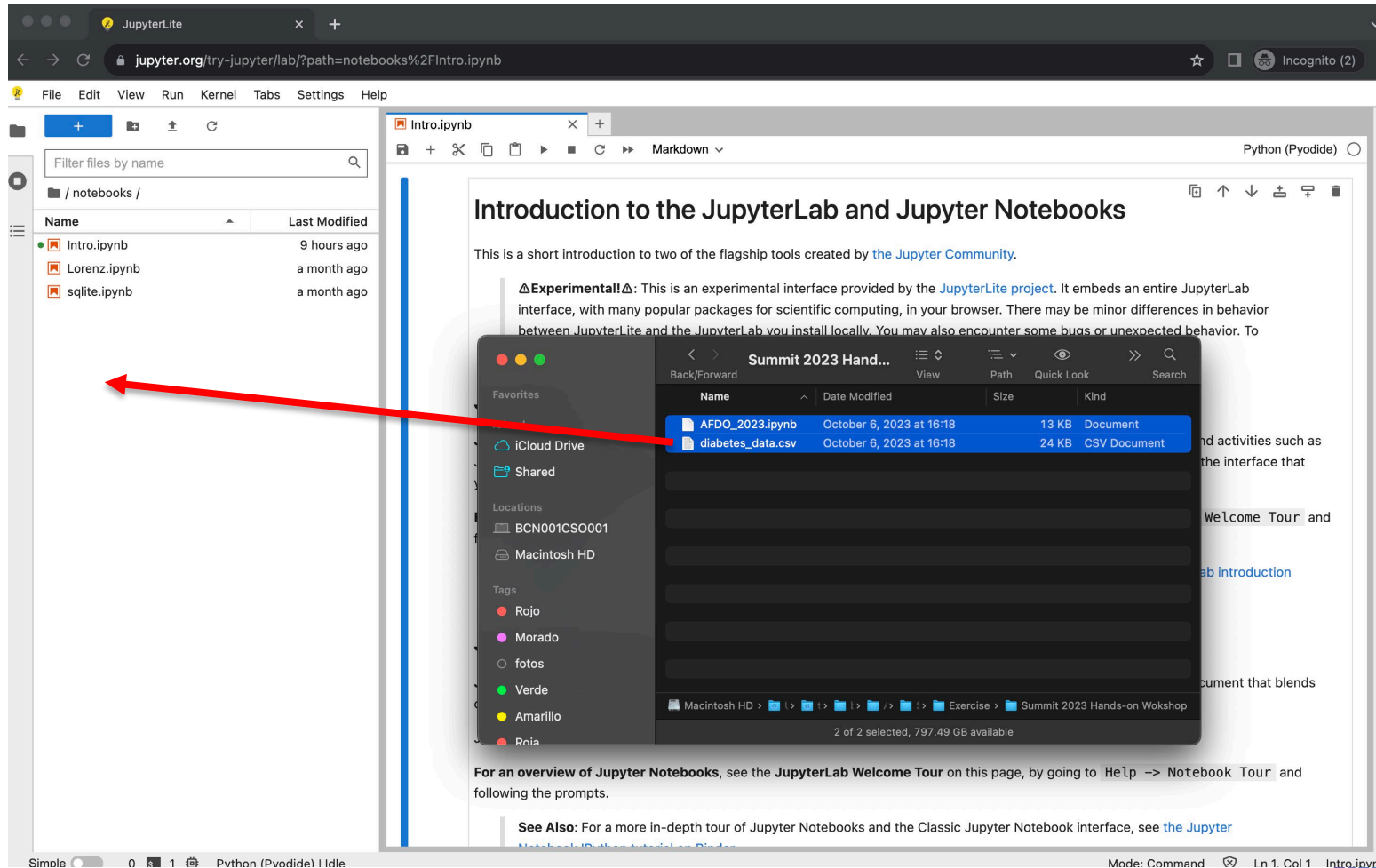
 https___www.healthcareproducts.org_wp-content/uploads_2023_11_diabetes_data



 [diabetes_data](#)

Preparing the environment

Drag the two files to the data folder



The screenshot shows the JupyterLab interface in a web browser. The left sidebar displays the file explorer with a list of files in the 'notebooks' directory:

Name	Last Modified
Intro.ipynb	9 hours ago
Lorenz.ipynb	a month ago
sqlite.ipynb	a month ago

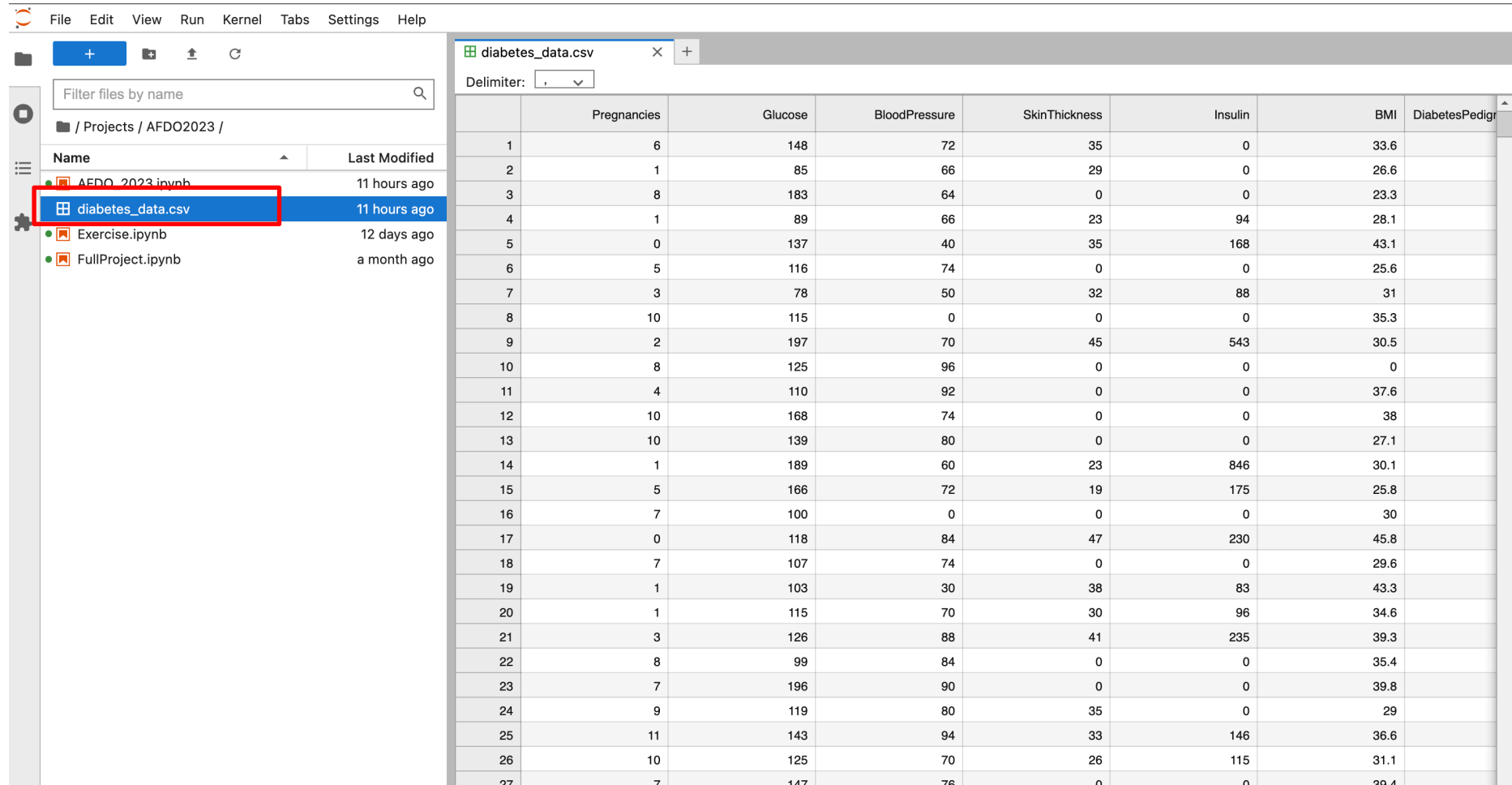
A red arrow points from the 'Intro.ipynb' file in the sidebar to a file browser overlay window. The overlay window, titled 'Summit 2023 Hand...', shows a table of files:

Name	Date Modified	Size	Kind
AFDO_2023.ipynb	October 6, 2023 at 16:18	13 KB	Document
diabetes_data.csv	October 6, 2023 at 16:18	24 KB	CSV Document

The overlay window also shows a sidebar with 'Favorites' and 'Locations' (BCN001CSO001, Macintosh HD) and a 'Tags' section with various color-coded tags. The main JupyterLab window displays the 'Intro.ipynb' notebook content, which includes an introduction to JupyterLab and Jupyter Notebooks.

Preparing the environment

Confirm the data upload

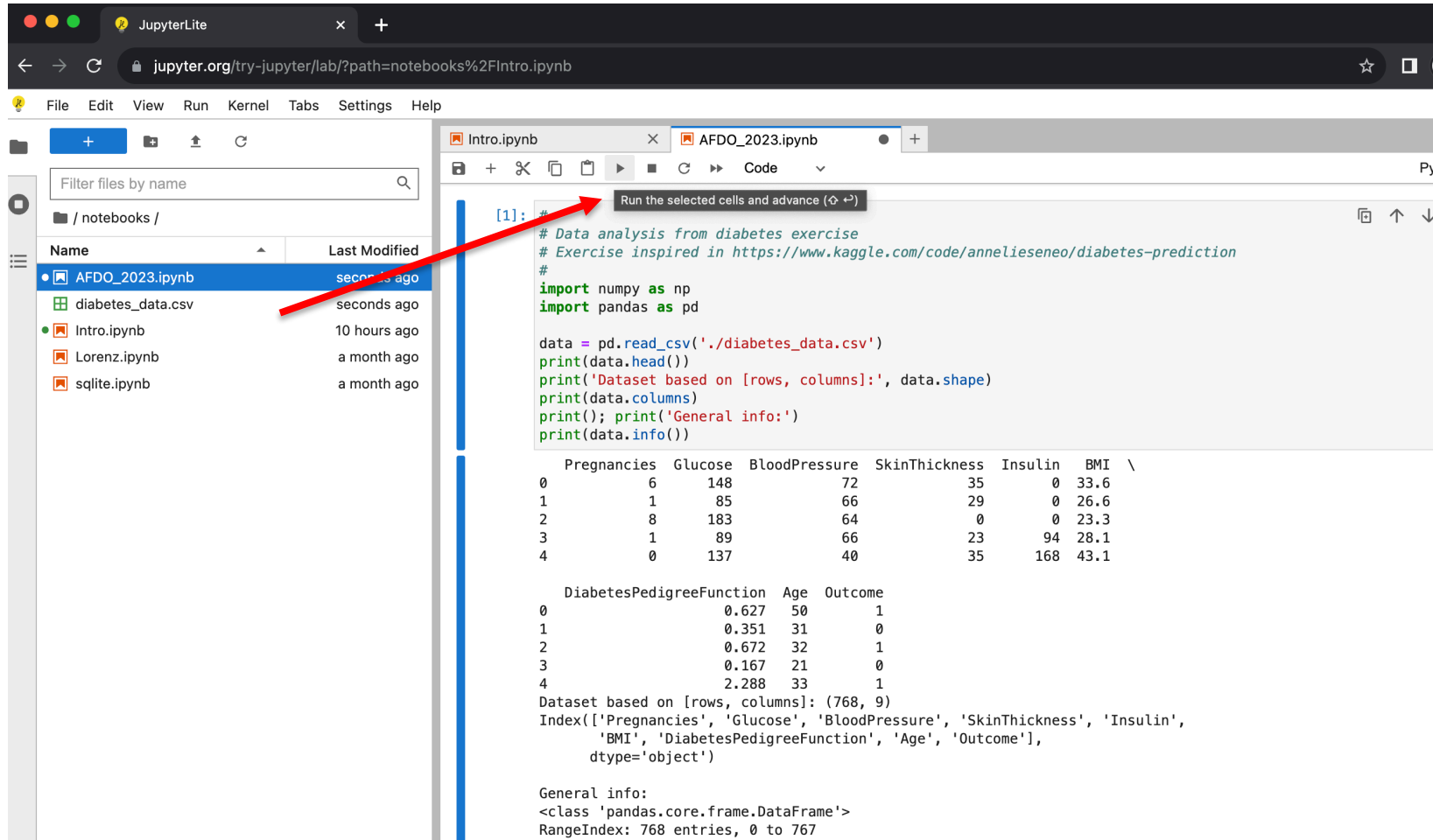


The screenshot shows a JupyterLab environment. On the left, the file explorer displays the project structure under '/ Projects / AFDO2023 /'. The file 'diabetes_data.csv' is highlighted with a red box. The main editor area shows the 'diabetes_data.csv' file open, displaying a table of data. The table has 27 rows and 7 columns: Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, and DiabetesPedigree. The data is as follows:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigree
1	6	148	72	35	0	33.6	
2	1	85	66	29	0	26.6	
3	8	183	64	0	0	23.3	
4	1	89	66	23	94	28.1	
5	0	137	40	35	168	43.1	
6	5	116	74	0	0	25.6	
7	3	78	50	32	88	31	
8	10	115	0	0	0	35.3	
9	2	197	70	45	543	30.5	
10	8	125	96	0	0	0	
11	4	110	92	0	0	37.6	
12	10	168	74	0	0	38	
13	10	139	80	0	0	27.1	
14	1	189	60	23	846	30.1	
15	5	166	72	19	175	25.8	
16	7	100	0	0	0	30	
17	0	118	84	47	230	45.8	
18	7	107	74	0	0	29.6	
19	1	103	30	38	83	43.3	
20	1	115	70	30	96	34.6	
21	3	126	88	41	235	39.3	
22	8	99	84	0	0	35.4	
23	7	196	90	0	0	39.8	
24	9	119	80	35	0	29	
25	11	143	94	33	146	36.6	
26	10	125	70	26	115	31.1	
27	7	147	76	0	0	30.4	

Preparing the environment

Confirm the code upload



The screenshot shows the JupyterLab interface with a file browser on the left and a code editor on the right. A red arrow points from the file browser to the code editor, indicating the upload of the file.

File Browser (Left):

- Filter files by name
- / notebooks /
- Name | Last Modified
- AFDO_2023.ipynb | seconds ago
- diabetes_data.csv | seconds ago
- Intro.ipynb | 10 hours ago
- Lorenz.ipynb | a month ago
- sqlite.ipynb | a month ago

Code Editor (Right):

Run the selected cells and advance (⇧↵)

```
[1]: #
# Data analysis from diabetes exercise
# Exercise inspired in https://www.kaggle.com/code/annelieseneo/diabetes-prediction
#
import numpy as np
import pandas as pd

data = pd.read_csv('./diabetes_data.csv')
print(data.head())
print('Dataset based on [rows, columns]:', data.shape)
print(data.columns)
print(); print('General info:')
print(data.info())
```

Output:

```
Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin  BMI \
0           6       148             72             35      0  33.6
1           1        85             66             29      0  26.6
2           8       183             64              0      0  23.3
3           1        89             66             23     94  28.1
4           0       137             40             35     168  43.1

DiabetesPedigreeFunction  Age  Outcome
0           0.627      50         1
1           0.351      31         0
2           0.672      32         1
3           0.167      21         0
4           2.288      33         1
Dataset based on [rows, columns]: (768, 9)
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')
```

General info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
dtypes: object(9)

Questions?

We will be here if you need help.

Networking reception will start at 5:00 in the Fountain Room (where lunch was) and end at 6:30.



Workshop

AI SUMMIT

CINCINNATI, OH • NOVEMBER 14-16, 2023

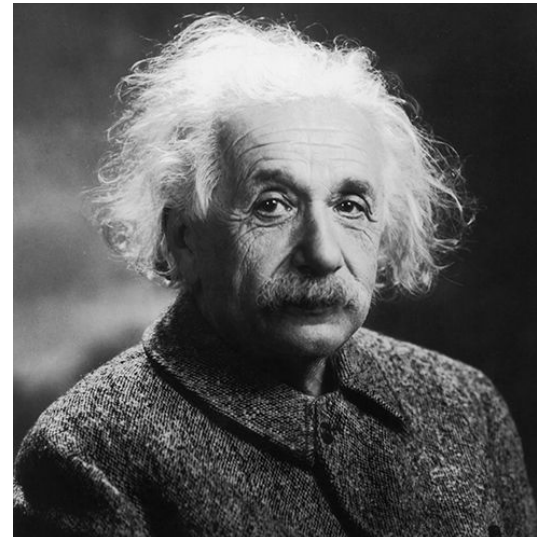
How do I use AI?

- Avoid the downfalls:
 1. Executives lack a clear vision for advanced analytics
 2. Ignore Lean Principles
 3. Goals are driven by tools and not business problems



Defining Problems

“If I were given an hour in which to do a problem upon which my life depended, I would spend 40 minutes studying it, 15 minutes reviewing it, and 5 minutes solving it.”



Defining Problems

1. Establish the need for a solution
 - What is the basic need?
 - What is the desired outcome?
2. Justify the need
 - Does this follow the company's strategy?
 - What are the benefits, and how will they be measured?
3. Contextualize the problem
 - What approaches have been tried?
 - Have other organizations tried?
4. Write the problem statement

Good vs Bad

We are behind schedule.

The lack of clean water in developing countries is leading to increased incidence of water-borne diseases and limiting socio-economic development.

We need to use ChatGPT.

Employee turnover of 25% in Company X is negatively impacting productivity, morale, and profitability.

Using AI

- Is there a clear objective?
- Is good quality data available?
- Have traditional statistics failed?
- Do you have a skilled team?
- Are your operations ready for AI?

Business Problem

1. Establish the need for a solution
 - The disease progression for diabetic patients on a certain treatment cannot be determined until 1 year after the treatment started. Knowing the progression sooner will allow earlier adjustments to the treatment.
2. Justify the need
 - Adjusting the treatments earlier will improve the chances of a positive progression improving the lives of our patients. The benefits will be a savings in unnecessary treatment and extended life for patients.
3. Contextualize the problem
 - The dataset was reviewed, but there was no apparent correlation between the measurable attributes and the outcome. Other organizations have used AI models to predict the outcomes.

Business Problem

4. Write the problem statement.

- The disease progression for diabetic patients on a certain treatment cannot be determined until one year after the treatment started. Knowing the progression sooner will allow earlier adjustments to the treatment, improving the chances of a positive. The benefits will be savings in unnecessary treatment and extended life for patients. The dataset was reviewed, but there was no apparent correlation between the measurable attributes and the outcome. Other organizations have used AI models to predict the outcomes.

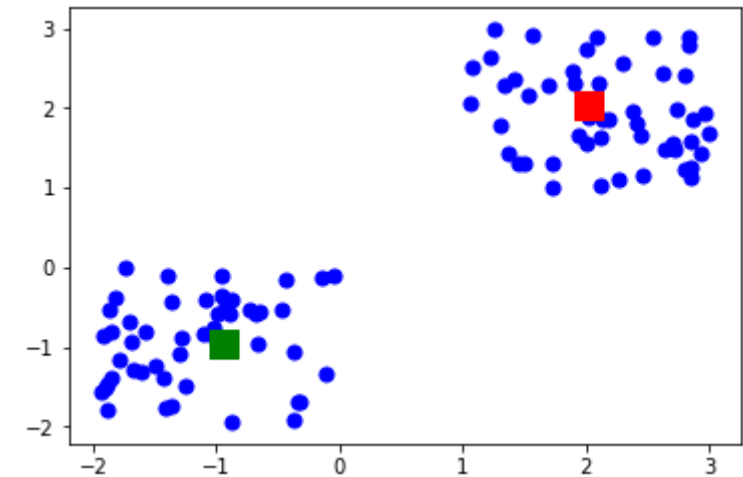
K-Means Clustering

Unsupervised approach used to group similar objects into clusters

Clusters so that the sum of the squared distances between the objects and their cluster mean is minimized.

Input: Columns of features with numerical values

Output: Objects placed into similar clusters

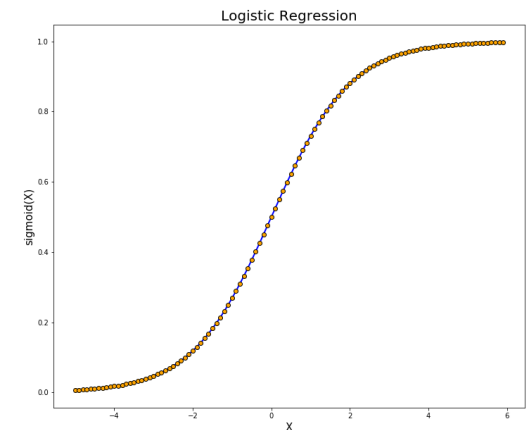


Logistic Regression

Supervised approach used to determine the probability of a discrete outcome (y) given input variables (x)

Input: Columns of numerical inputs, column of categorical output

Output: Predictive model predicting the outcome of future objects

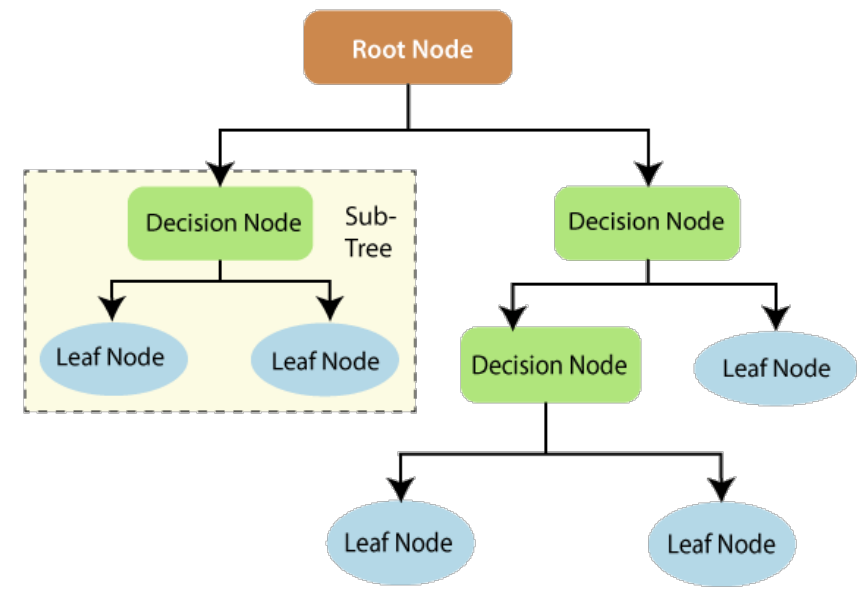


Random Forest

Supervised approach using decision trees to solve regression and classification problems

Input: Columns of numerical inputs, column of numerical output

Output: Predictive model to categorize future objects



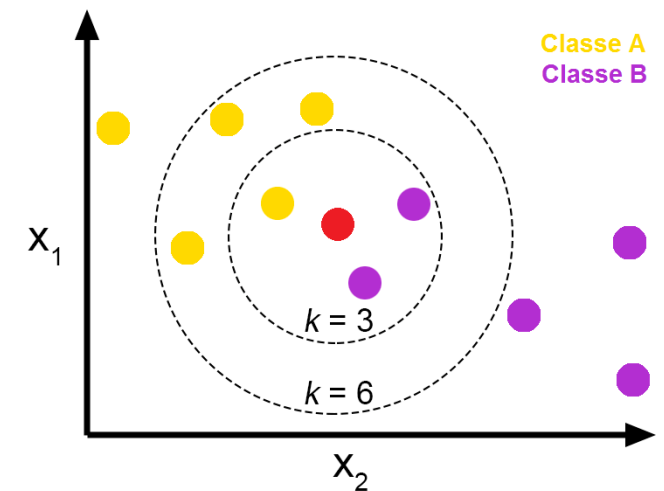
K-Nearest Neighbors Algorithm

Supervised approach used to group similar objects into clusters

Uses distance to locate the closest neighbors of an object

Input: Columns of features with numerical values, column of numerical categorization

Output: Predictive model to categorize future objects





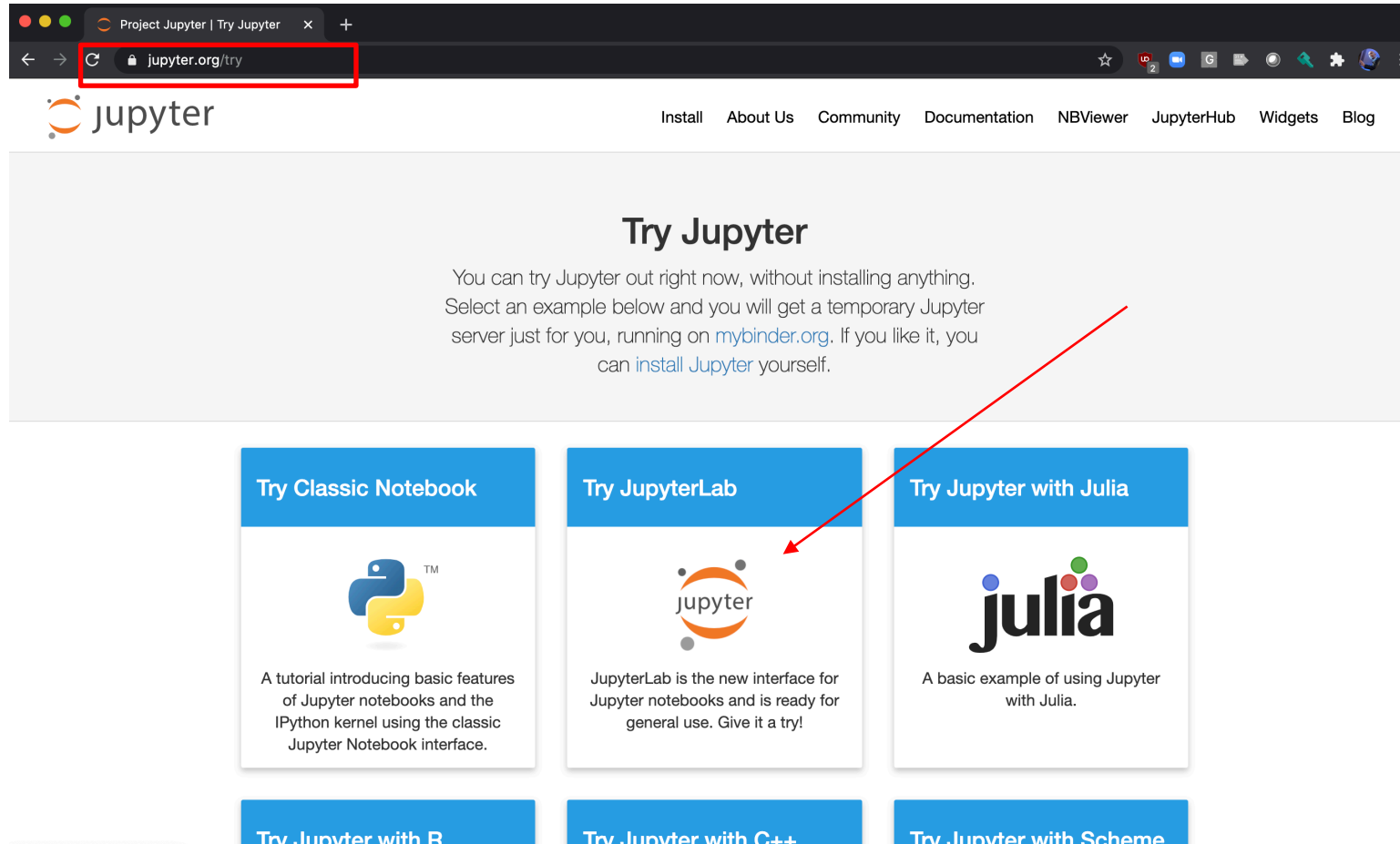
Hand—on application

Analyzing a disease



Preparing the environment

Go to jupyter.org and click 'Try' JupyterLab



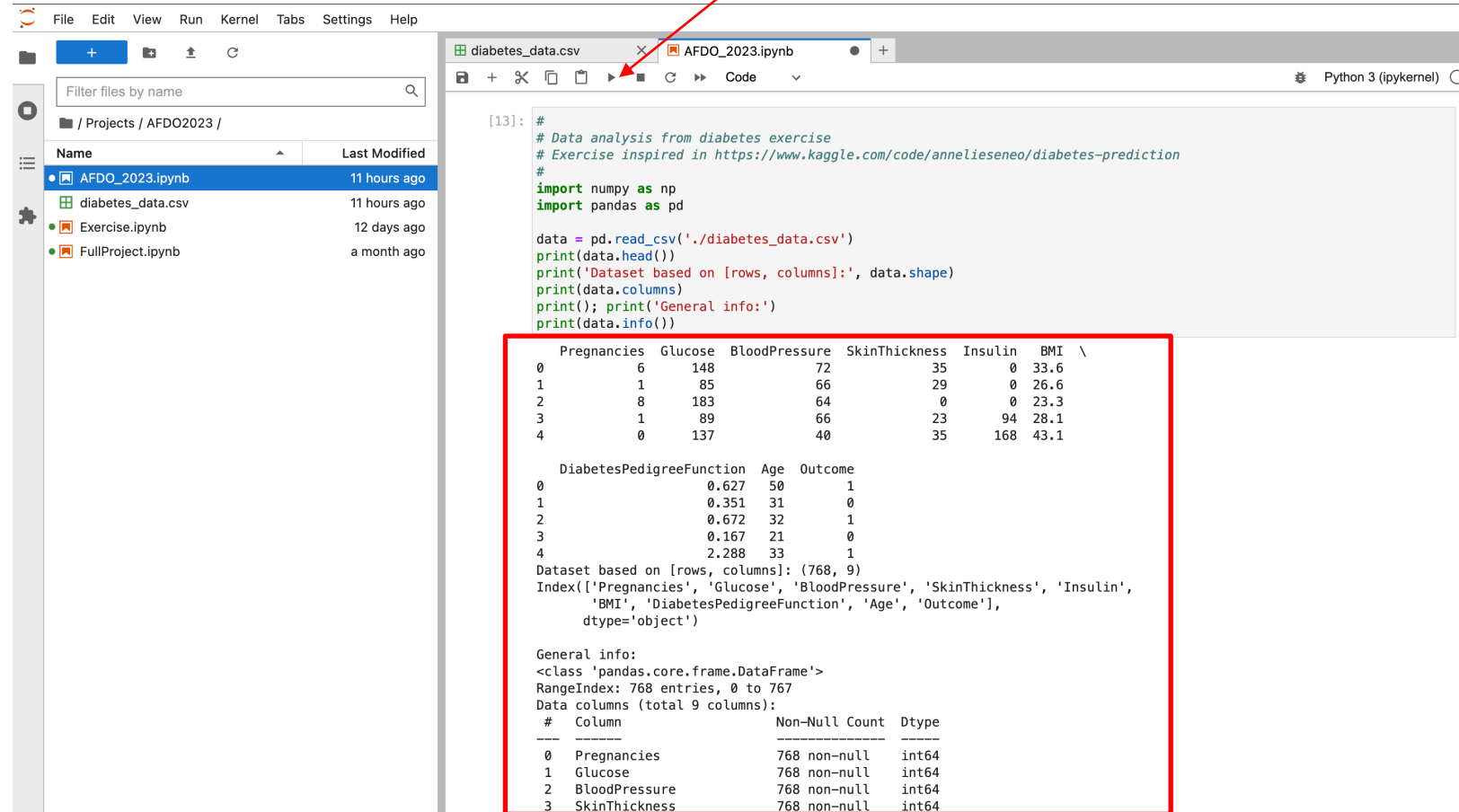
The screenshot shows a web browser window with the address bar displaying jupyter.org/try. The page title is "Project Jupyter | Try Jupyter". The main heading is "Try Jupyter", followed by the text: "You can try Jupyter out right now, without installing anything. Select an example below and you will get a temporary Jupyter server just for you, running on mybinder.org. If you like it, you can [install Jupyter](#) yourself."

Below the text are six buttons for trying different Jupyter environments:

- Try Classic Notebook
- Try JupyterLab (indicated by a red arrow)
- Try Jupyter with Julia
- Try Jupyter with R
- Try Jupyter with C++
- Try Jupyter with Scheme

Starting with the data

Execute the code and look at the data



The screenshot shows a Jupyter Notebook interface with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'AFDO2023' with files 'AFDO_2023.ipynb', 'diabetes_data.csv', 'Exercise.ipynb', and 'FullProject.ipynb'. The code editor shows the following code:

```
[13]: #
# Data analysis from diabetes exercise
# Exercise inspired in https://www.kaggle.com/code/annelieseneo/diabetes-prediction
#
import numpy as np
import pandas as pd

data = pd.read_csv('./diabetes_data.csv')
print(data.head())
print('Dataset based on [rows, columns]:', data.shape)
print(data.columns)
print(); print('General info:')
print(data.info())
```

The output of the code is displayed below the code cell. A red box highlights the output, which includes a table of pregnancy data and a summary of the dataset.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI \
0	6	148	72	35	0	33.6
1	1	85	66	29	0	26.6
2	8	183	64	0	0	23.3
3	1	89	66	23	94	28.1
4	0	137	40	35	168	43.1

DiabetesPedigreeFunction Age Outcome

	DiabetesPedigreeFunction	Age	Outcome
0	0.627	50	1
1	0.351	31	0
2	0.672	32	1
3	0.167	21	0
4	2.288	33	1

Dataset based on [rows, columns]: (768, 9)
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'], dtype='object')

General info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):

#	Column	Non-Null Count	Dtype
0	Pregnancies	768 non-null	int64
1	Glucose	768 non-null	int64
2	BloodPressure	768 non-null	int64
3	SkinThickness	768 non-null	int64

Looking at the data

Making decisions around data

```
#  
# Values ••like glucose,bloodpressure or BMI can not be 0. We have to fix the problem.  
# looking for empty data, lost data  
  
print(); print('Null values:')  
data.isna().sum()  
print(); print('Empty values:')  
data.eq(0).sum()
```

Null values:

Empty values:

Pregnancies	111
Glucose	5
BloodPressure	35
SkinThickness	227
Insulin	374
BMI	11
DiabetesPedigreeFunction	0
Age	0
Outcome	500
dtype:	int64

Looking at the data

Interpolating values to not lose data for columns with value = 0

First option

```
#Missing Data Imputation Using Regression

def ImputeZeroValuesWithRegression(dataset):

    columnsToBeImputed = ['BloodPressure', 'Glucose', 'Insulin', 'SkinThickness', 'BMI']
    for column in columnsToBeImputed:

        test_df = dataset[dataset[column]!=0]

        y_train= dataset[column]
        x_train= dataset.drop(column,axis=1)

        X_test = test_df.drop(column, axis=1)

        lr=LinearRegression()
        lr.fit(x_train,y_train)
        y_pred=lr.predict(X_test)

        dataset.loc[dataset[column]==0,column] = y_pred

    return dataset

# Interppolating blank values for columns that do not make sense to have empty values
df=ImputeZeroValuesWithRegression(dataset=data)
print(); print('Empty values:')
df.eq(0).sum()
```

Empty values:

Pregnancies	111
Glucose	0
BloodPressure	0
SkinThickness	0
Insulin	0
BMI	0
DiabetesPedigreeFunction	0
Age	0
Outcome	500
dtype:	int64

Looking at the data

Second option

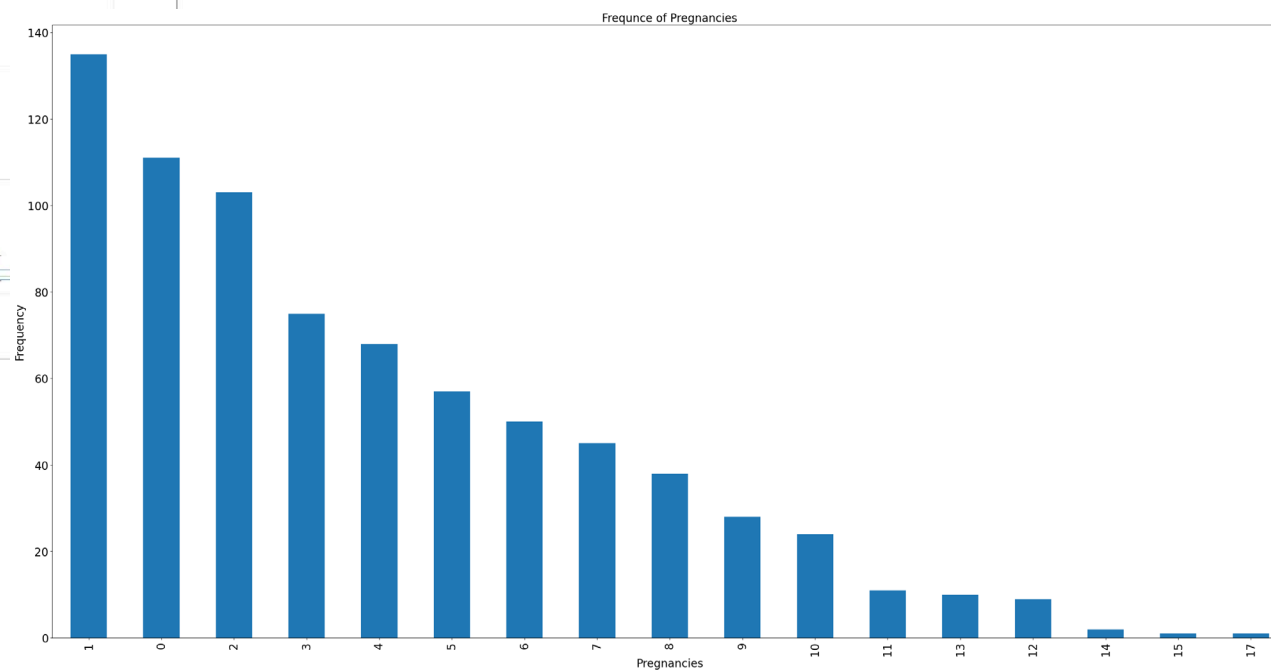
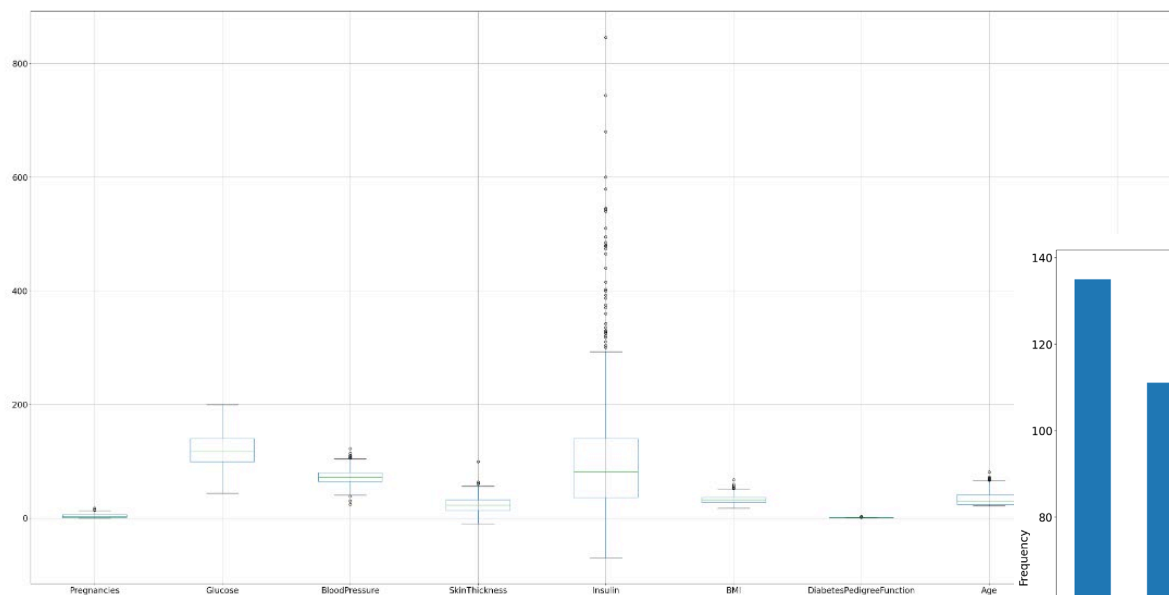
Changing values = 0 with the mean

```
# Changing first the empty values of the table.
# NaN instead of 0
data[["Pregnancies","Glucose","BloodPressure","SkinThickness","Insulin","BMI","DiabetesPedigreeFunction","Age"]] = data[["Pregnancies","Glucose","BloodPressure","SkinThickness","Insulin","BMI","DiabetesPedigreeFunction","Age"]].replace(0,np.NaN)

#filling in the missing values
data.fillna(data.mean(),inplace=True)

#I filled the empty spaces with average.
data.head()
```

Exploring the Data

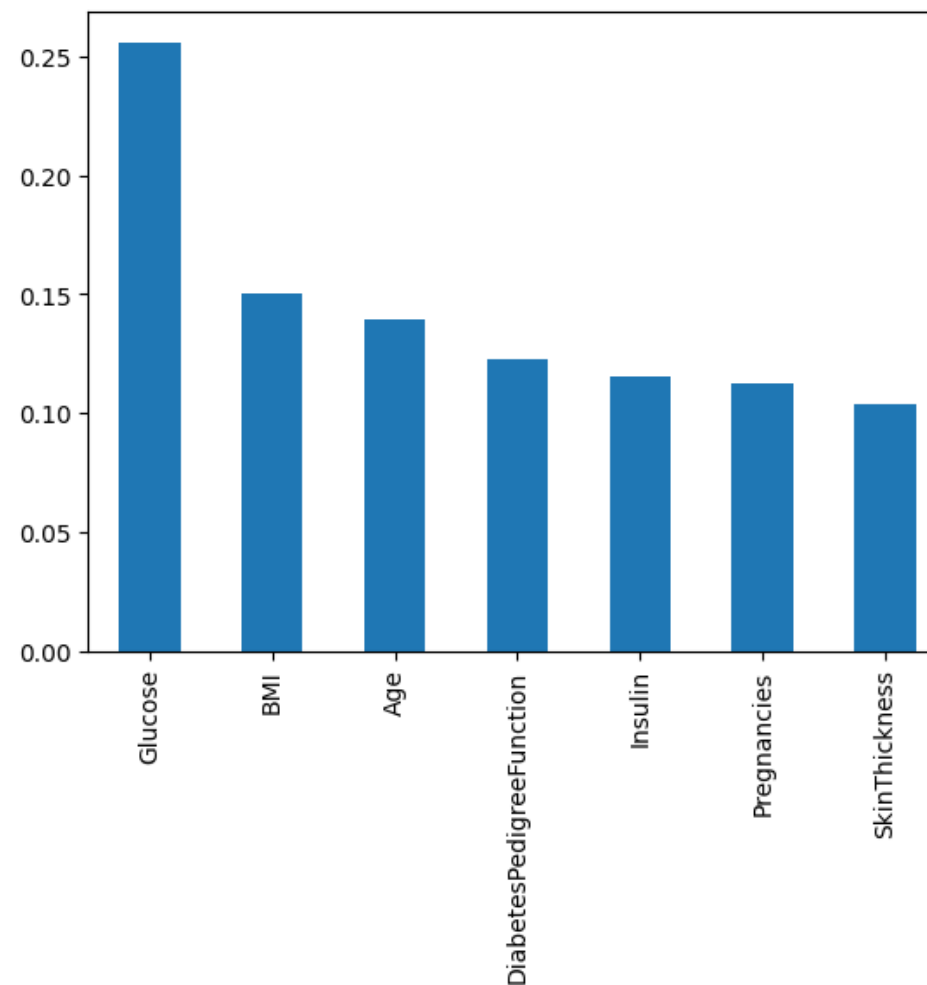




Order of Importance

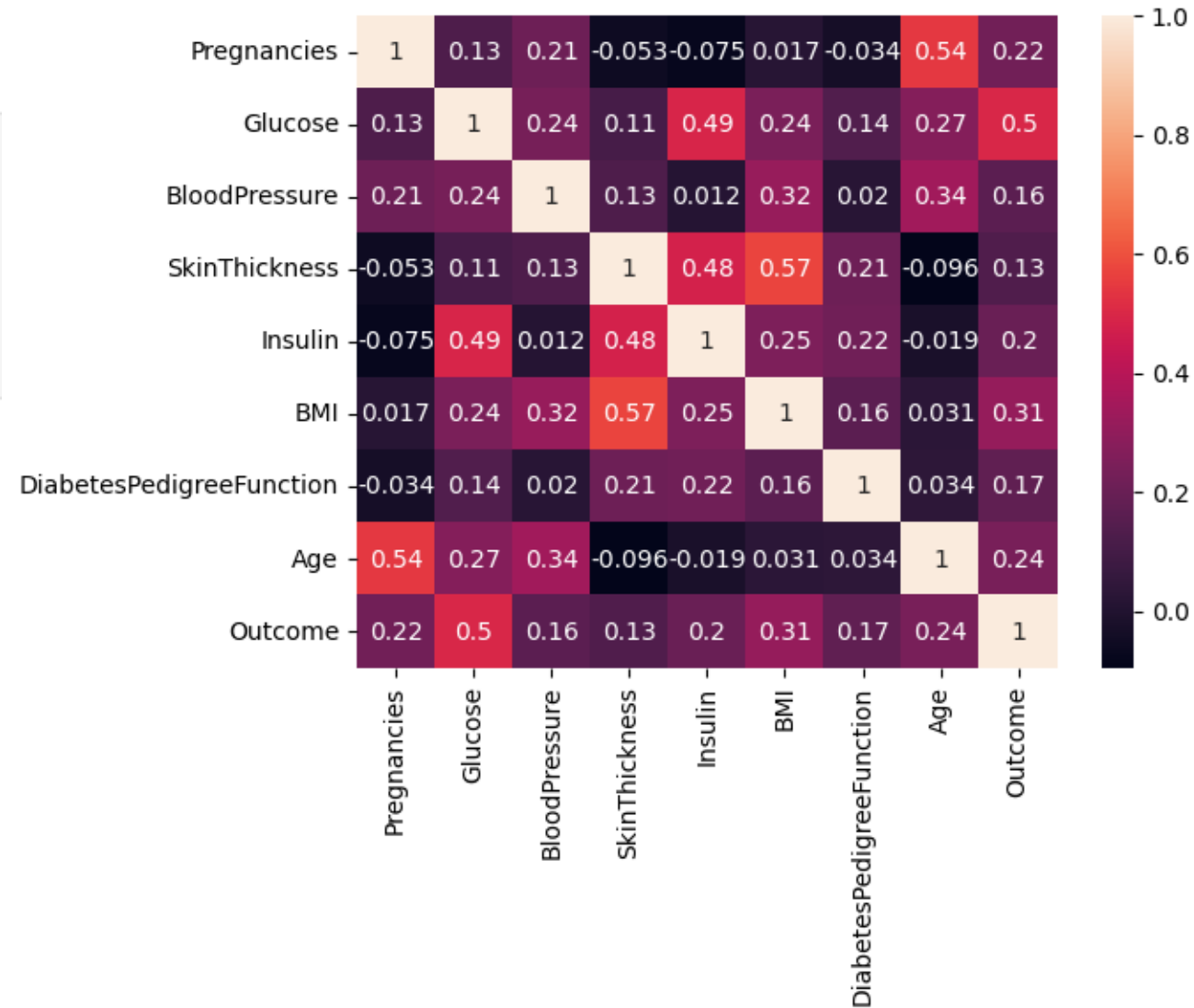
```
#
# Order of importance
#
x=data[['Glucose', 'BMI', 'Age', 'Pregnancies', 'SkinThickness',
        'Insulin', 'DiabetesPedigreeFunction']]
y=data.iloc[:,8]

model = ExtraTreesClassifier()
model.fit(x,y)
print(model.feature_importances_)
#plot graph of feature importances for better visualization
feat_importances = pd.Series(model.feature_importances_, index=x.columns)
feat_importances.nlargest(20).plot(kind='bar')
plt.show()
```



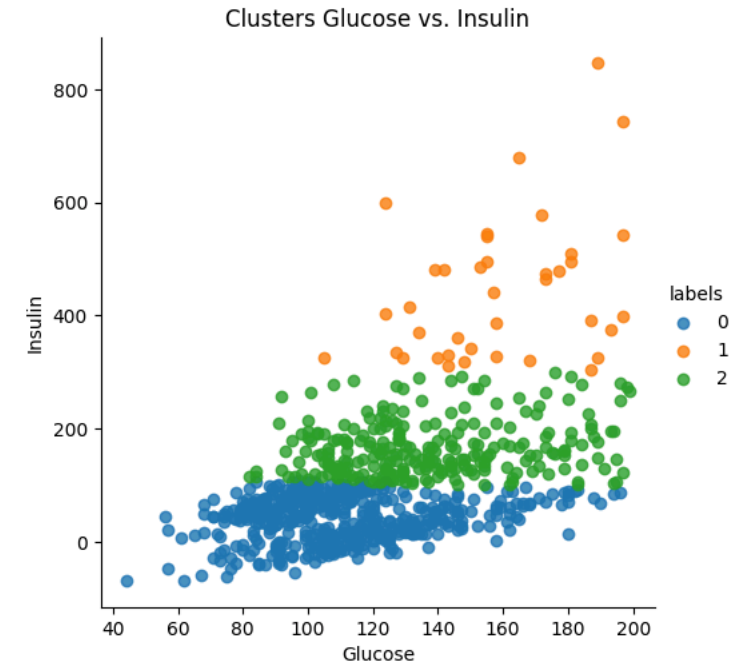
Classic Correlations

```
#
# Classic Correlation
#
#The measure of the relationship between variables.
print(data.corr())
sns.heatmap(data.corr(),annot=True)
```



Creating AI Models

- Clustering data by means k-means
- Classification Task: Logistic Regression
- Classification Task: Random Forest
- Classification Task: KNN Classifier



Solve the problem

Which model would you implement? Why?

Explore the outputs and discuss the options

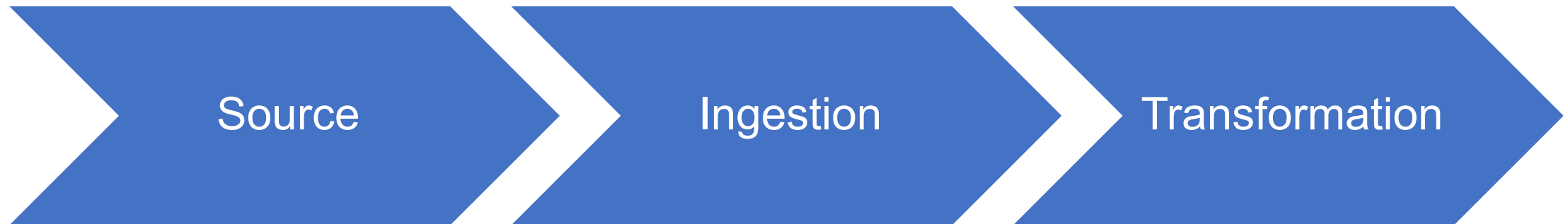
Lessons Learned

- Did you trust the...
 - Data
 - Algorithms
 - Models
- Next Steps:
 - How to measure the model drifting?
 - What causes drift? Can it be documented in a risk assessment?
 - What are the dangers of model drift?

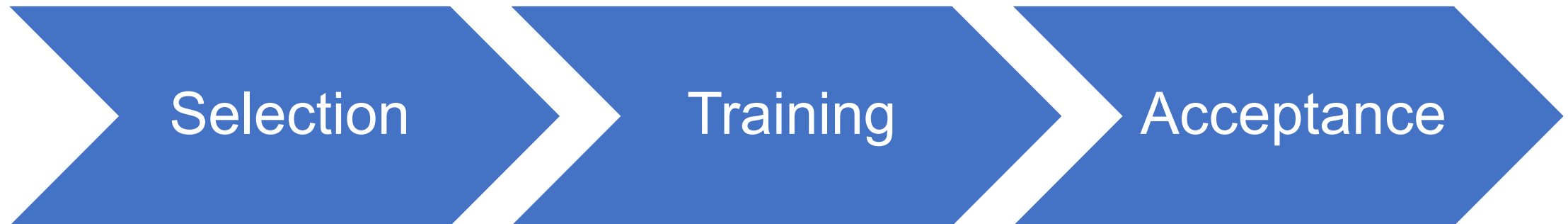
What if...

- The average BMI of the patients has decreased?
- The starting age of participants all increased?
- The precision of an instrument has increased? Decreased?
- Blanks are being introduced due to a change in the process?
- The disease mutates or becomes less understood?

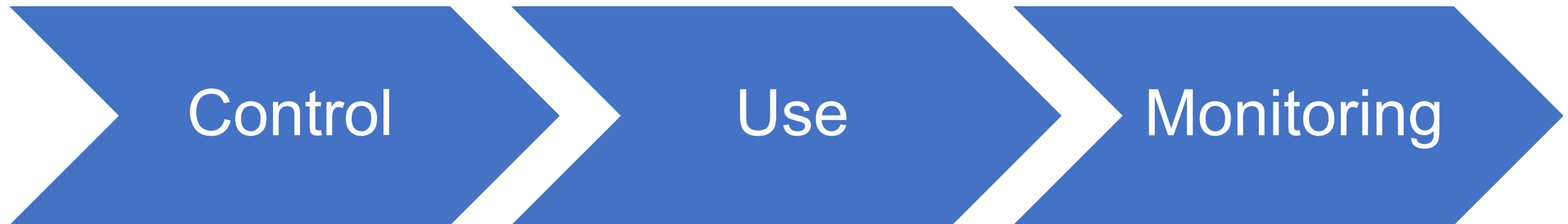
Validation – Data



Validation – Algorithm



Validation – Model



Q&A

