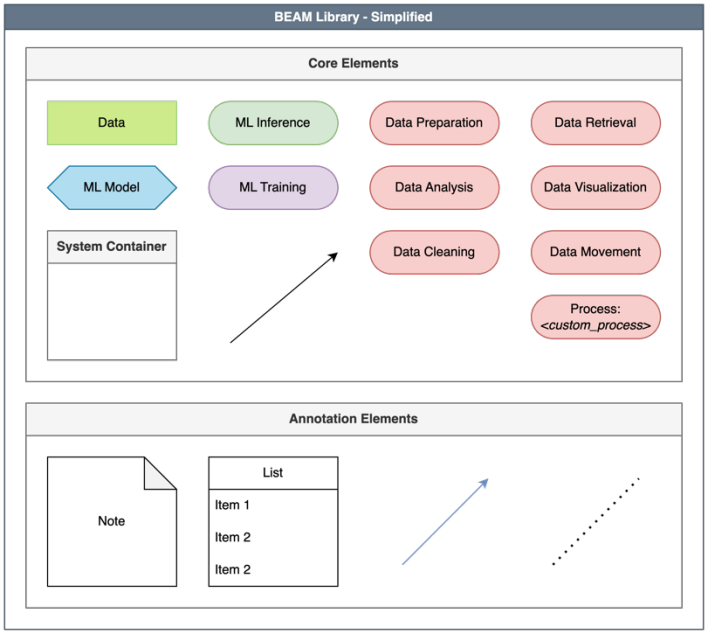
## Setup

The boxology modeler is based on draw.io, a popular diagramming tool. Here are the necessary steps to prepare for modeling your solution:

1. **Install or access draw.io**
   To use draw.io, you have 2 options:
   a. Download and install draw.io Desktop from https://www.drawio.com/
   b. Go to https://draw.io and open it in your browser
2. **Import the extended boxology library** that contains all the graphical elements [1].
   a. Download the beam_lib_v2 into your PC
   b. For both desktop and online version: File > open library > [location of your beam_lib_v2]
3. **Start drawing**
   a. Access the beam_lib_v2 on the top-left of your drawio application
   b. Start with loading the "Legend" component by clicking it

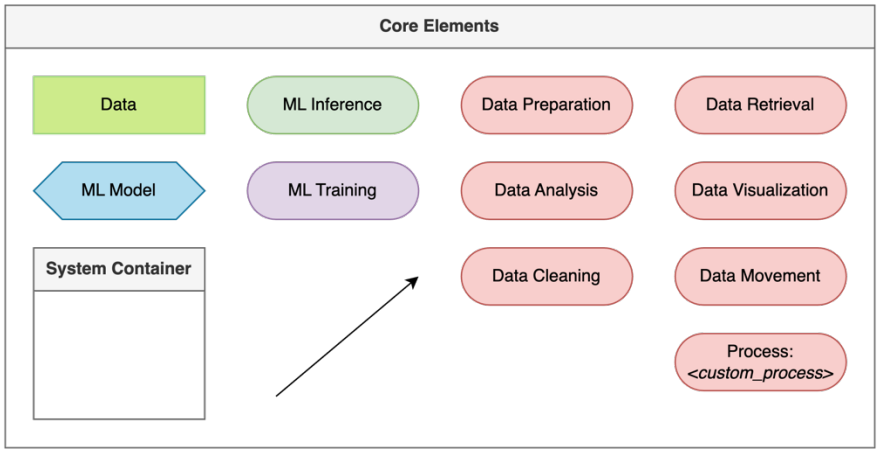



   c. You can now draw from scratch by copy pasting elements from the legends or by clicking the needed components from the shapes-bar (top-left of draw-io UI)
4. **Save and submit**
   a. Save your file (*.drawio)
   b. Submit your file via Canvas

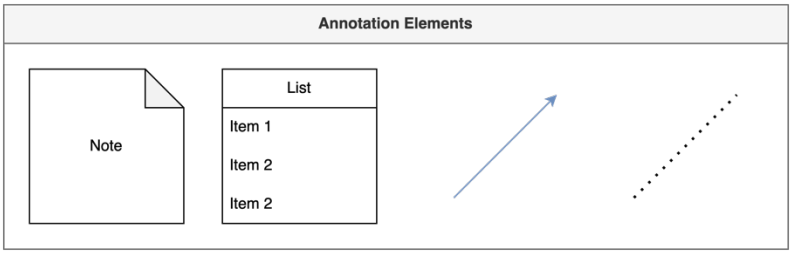
## Modeling

In the following, we will (briefly) explain the drawing elements provided in the library (the grouping of the components is shown in the “Legend” box). It consists of

1. **Core drawing elements** used to represent the *building blocks* of your Data Science pipeline**,** including
    a. **Input/Output Data** (green box),
    b. **ML Model** (blue hexagon),
    c. **ML Training** (green rounded rectangle),
    d. **ML Inference** (purple rounded rectangle), and
    e. Several **processing components** (pink squashed rectangle).
    f. **Solid black arrow** to draw the connection/flow between core-components.



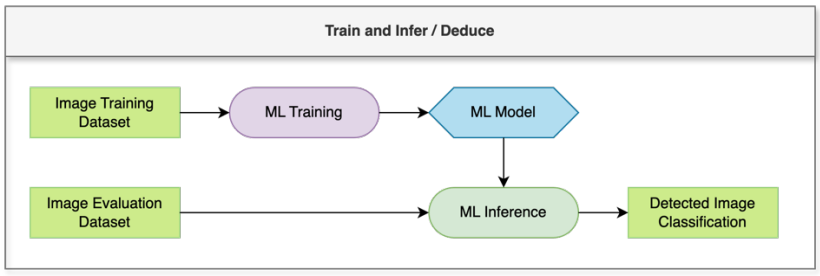

2. **Annotation drawing elements** used to annotating your Data Science pipeline building block**,** including
    a. **Note** to represent free-text annotations,
    b. **List** to represent key-value pair annotations,
    c. **Blue pointed arrow** to represent relation between a single core component and a group of annotation components,
    d. **Dotted arrow** to represent relation between a single core component and a single annotation



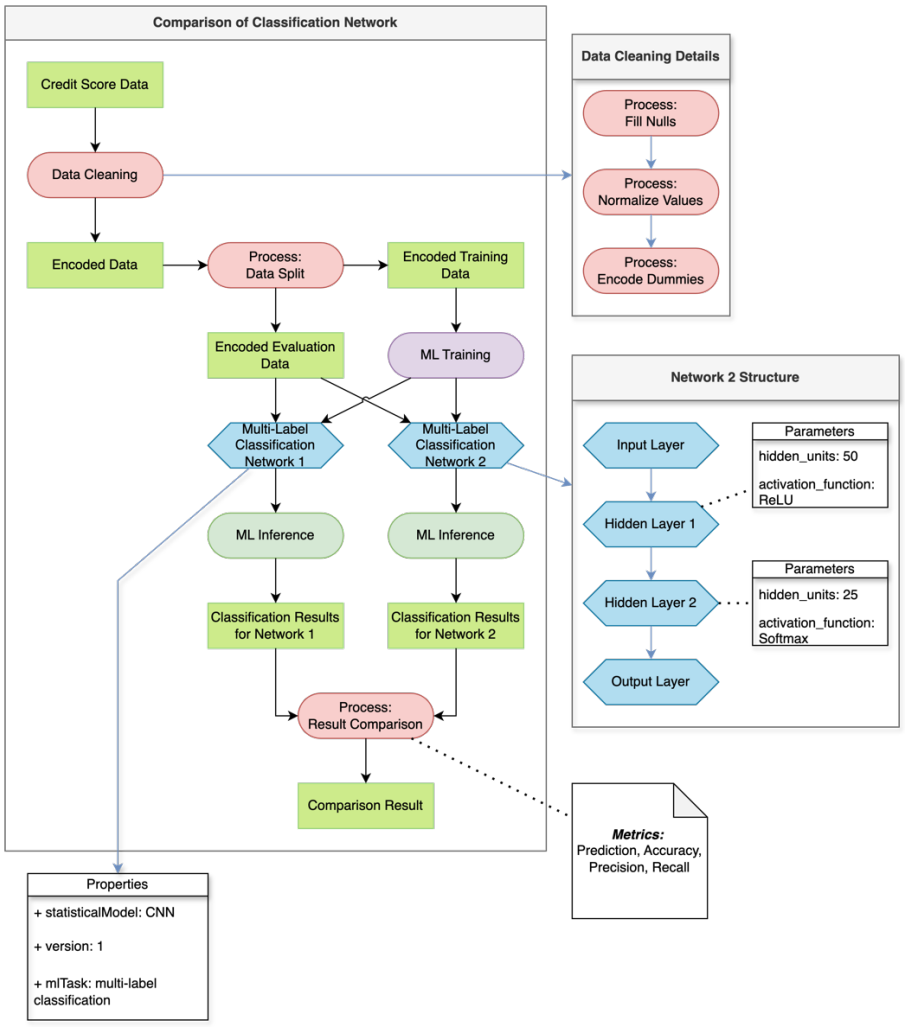
## Example

We provided two examples of the DS process representation: (i) Simple train and infer/deduce model for image recognition, and (ii) Comparison of Neural Networks approaches for multi-modal classification

1. Simple train and deduce model for image recognition (without annotation) [2]




2. Comparison of Neural Networks approaches for multi-modal classification (with example annotation) [3]



## References

[*] A more comprehensive documentation is available in the following link (*note that we exclude some elements explained in this full documentation for simplification – in case you want to ask questions, please let us know*):
https://github.com/semanticsystems/beam_tutorial/blob/main/beam_tutorial.md

[1] https://github.com/semanticsystems/beam_tutorial/blob/main/beam_lib_v2.xml

[2] https://github.com/semanticsystems/beam_tutorial/blob/main/example_1.drawio

[3] https://github.com/semanticsystems/beam_tutorial/blob/main/example_2.drawio