Leveraging Online Resources for Python Analytics

GETTING STARTED WITH PYTHON ANALYTICS

Janani Ravi
CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Python for data analysts

Explore commonly used online resources for Python analysts

Classic analytics workflow

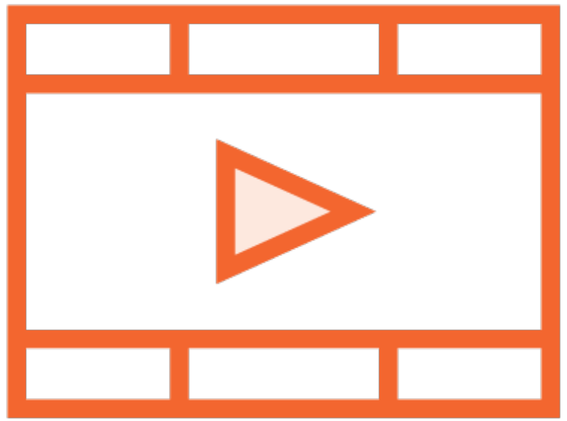
Very similar to machine learning workflow

Prototype models on Jupyter notebooks

Productionize models using a Python script

Prerequisites and Course Outline

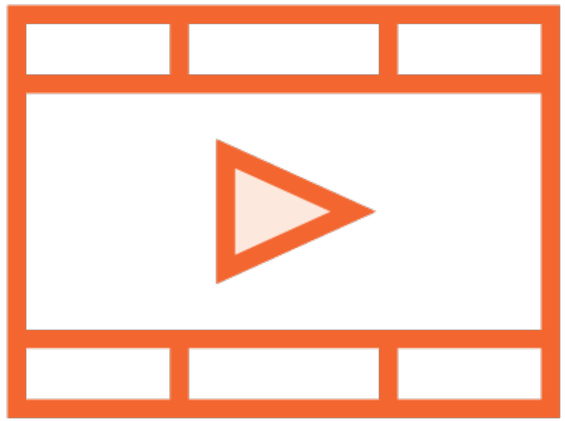
Prerequisites



Basic Python programming

Built and trained simple machine learning models

Prerequisites



Python Fundamentals

Understanding Machine Learning with Python

Building Your First scikit-learn Solution

Course Outline

Getting started with Python analytics

Leveraging online resources for Python analytics with BigML

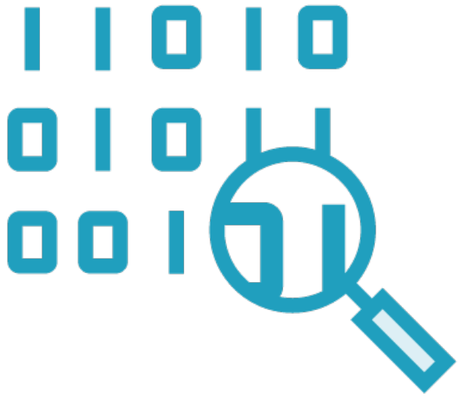
Working with interactive environments using Google Colab

Python for Data Analysts

“My mind is made up. Don’t confuse me with the facts.”

Some powerful person

Thoughtful, Fact-based Point of View



Fact-based

Built with painstakingly collected data

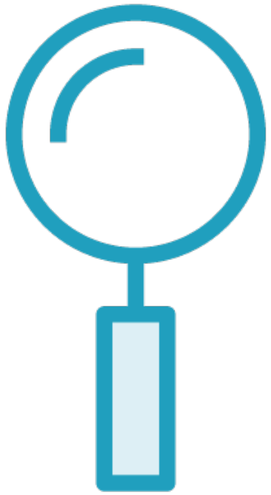
Thoughtful

Balanced, weighing pros and cons

Point of View

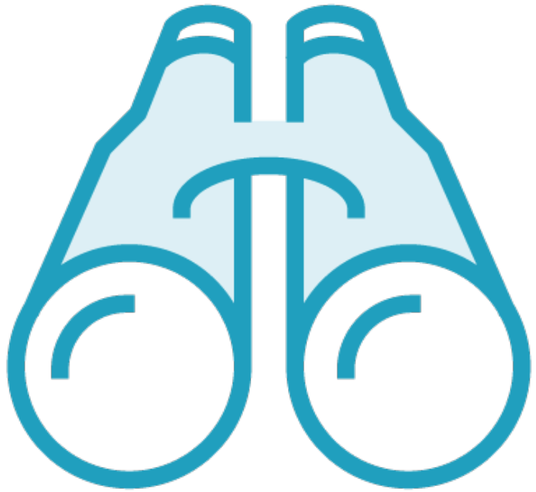
Prediction, recommendation, call to action

Two Sets of Statistical Tools



Descriptive Statistics

Identify important elements in a dataset



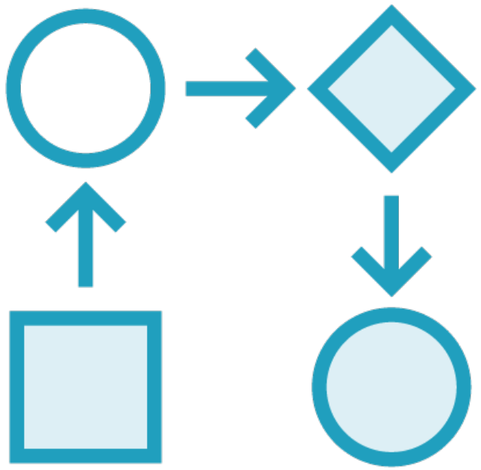
Inferential Statistics

Explain those elements via relationships with other elements

Two Hats of a Data Professional

Find the Dots

Identify important elements in a dataset



Connect the Dots

Explain those elements via relationships with other elements

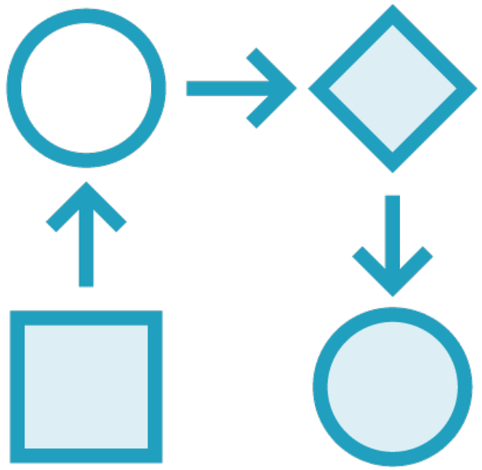
Finding the Dots

Data is more and more plentiful

However careful handling is needed

- Missing values
- Outliers
    - Genuine outliers
    - Erroneously measured points

Connecting the Dots



Spreadsheets

Programming languages

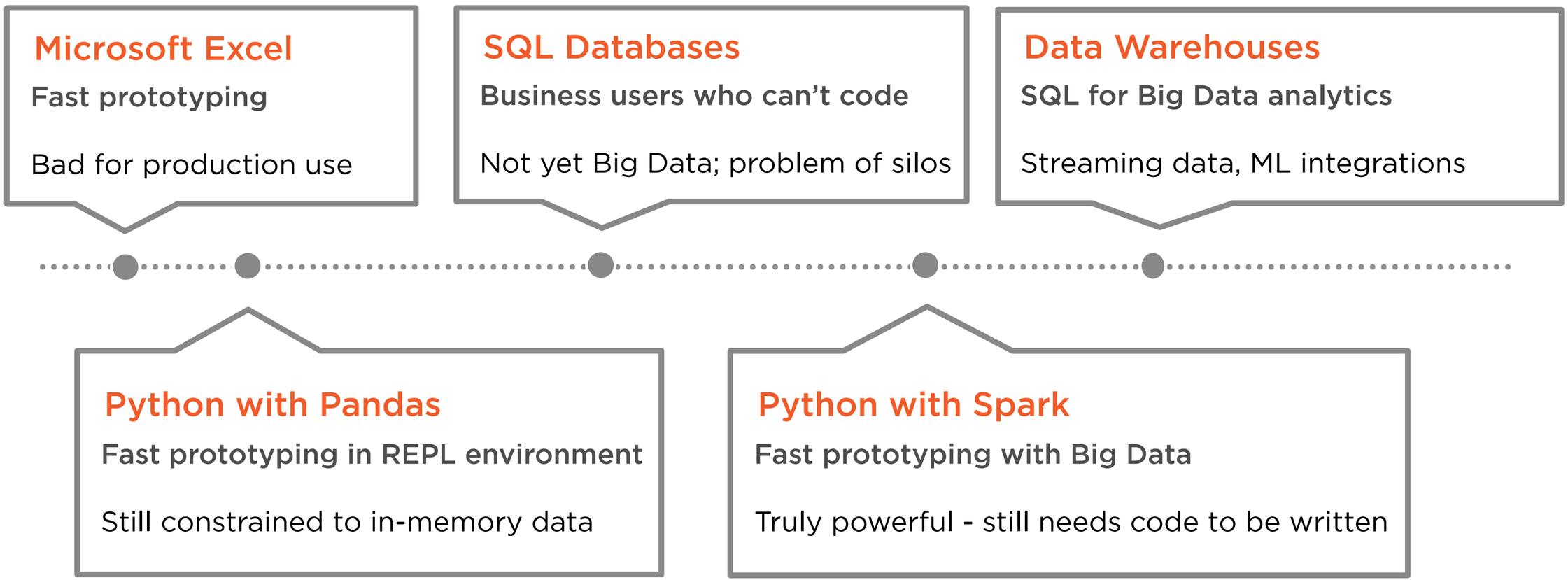
- In-memory processing
- Distributed processing

SQL

- Relational databases
- Data warehouses

Python has truly democratized data analysis more than any technology since Microsoft Excel

Choices of Technology



Essential Analytical Building Blocks

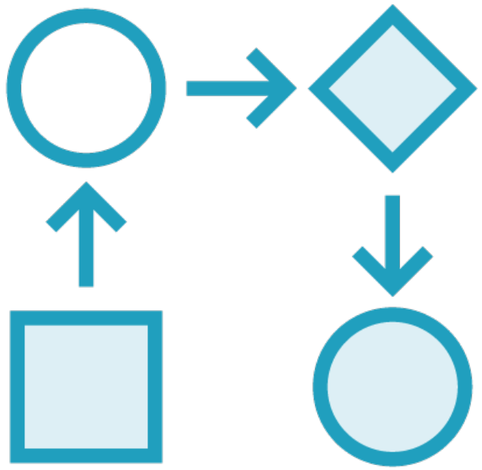
Conditional Execution

Interconnected Calculations

Repeated Execution (Iteration)

Re-use of Logic (Composition)

Python for Analytics



Programming languages offer full support for analytical operations

Conditionals: If-else

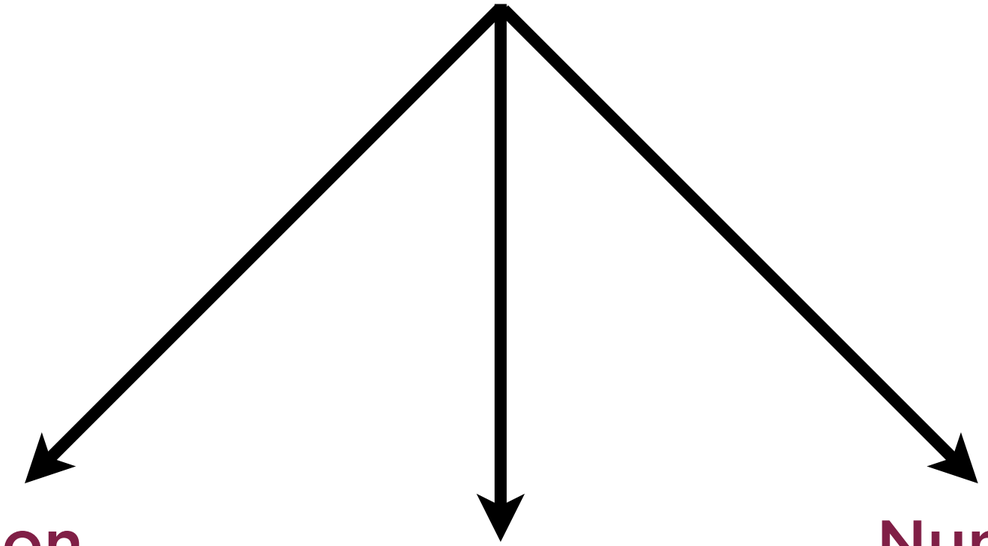
Iteration: For and while loops

Composition: Functions

Python combines Excel's ease-of-prototyping with SQL's simple syntax

Python Resources for Analytics

Analytics in Python

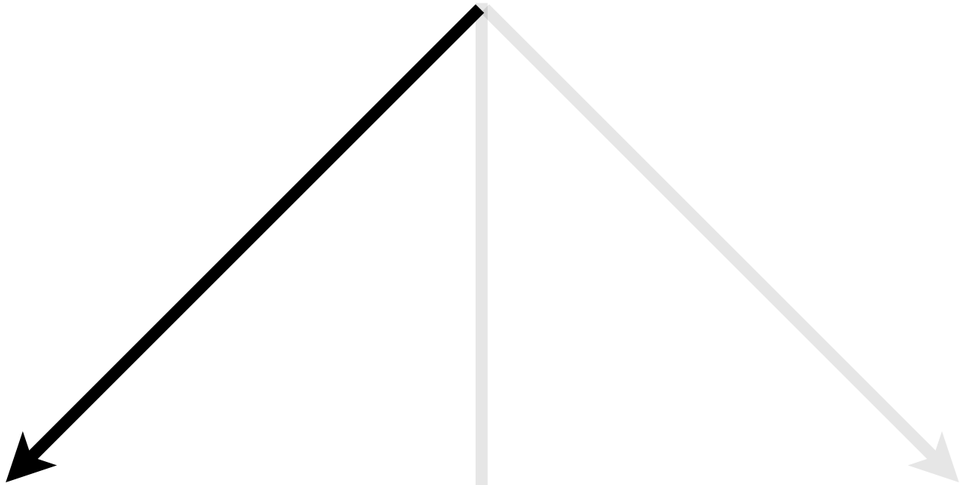


Visualization Libraries

ML Frameworks

Number-crunching Libraries

Analytics in Python



Visualization Libraries

ML Frameworks

Number-crunching Libraries

Visualization Libraries in Python

Matplotlib

Seaborn

Bokeh

Plotly.py

Many Libraries, Many Niches

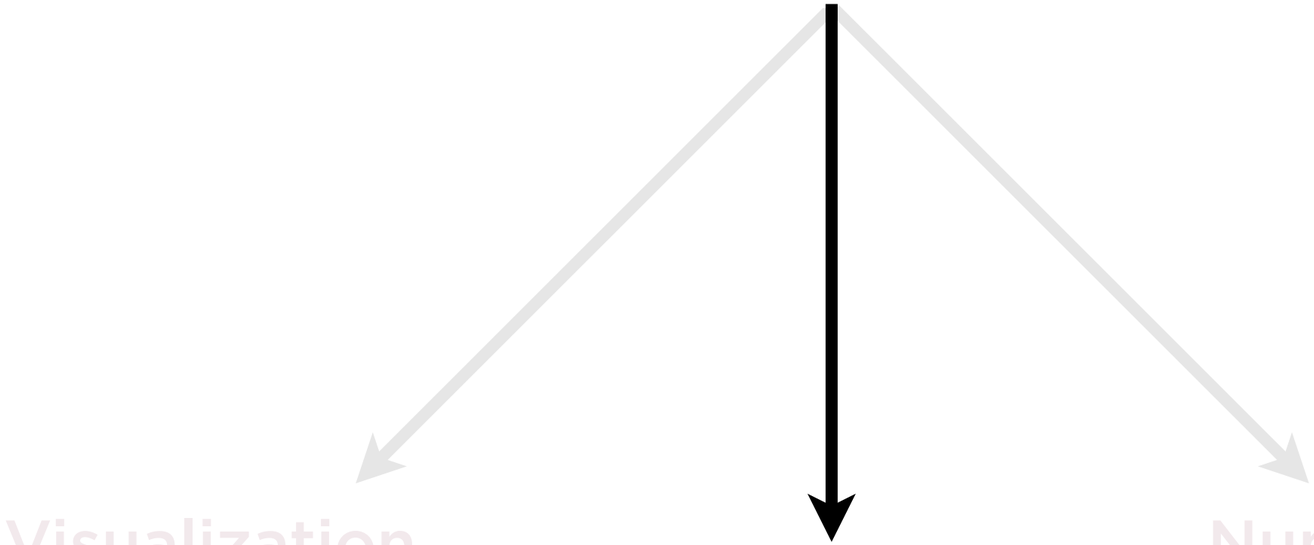
Matplotlib is powerful

Seaborn is easy-to-use

Bokeh for interactivity

Plotly.py for collaboration

Analytics in Python



Visualization Libraries

ML Frameworks

Number-crunching Libraries

scikit-learn

Easy-to-use, very comprehensive and efficient Python library for traditional ML models

PyTorch

A deep learning framework for fast, flexible experimentation.

https://pytorch.org/

TensorFlow

TensorFlow is an end-to-end open source platform for machine learning. A comprehensive, flexible ecosystem of tools, libraries and community resources to easily build and deploy ML powered applications.

https://tensorflow.org/

Keras

A high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. However, multi-backend Keras is superseded by tf.keras.

https://keras.io/

Other Popular ML Frameworks

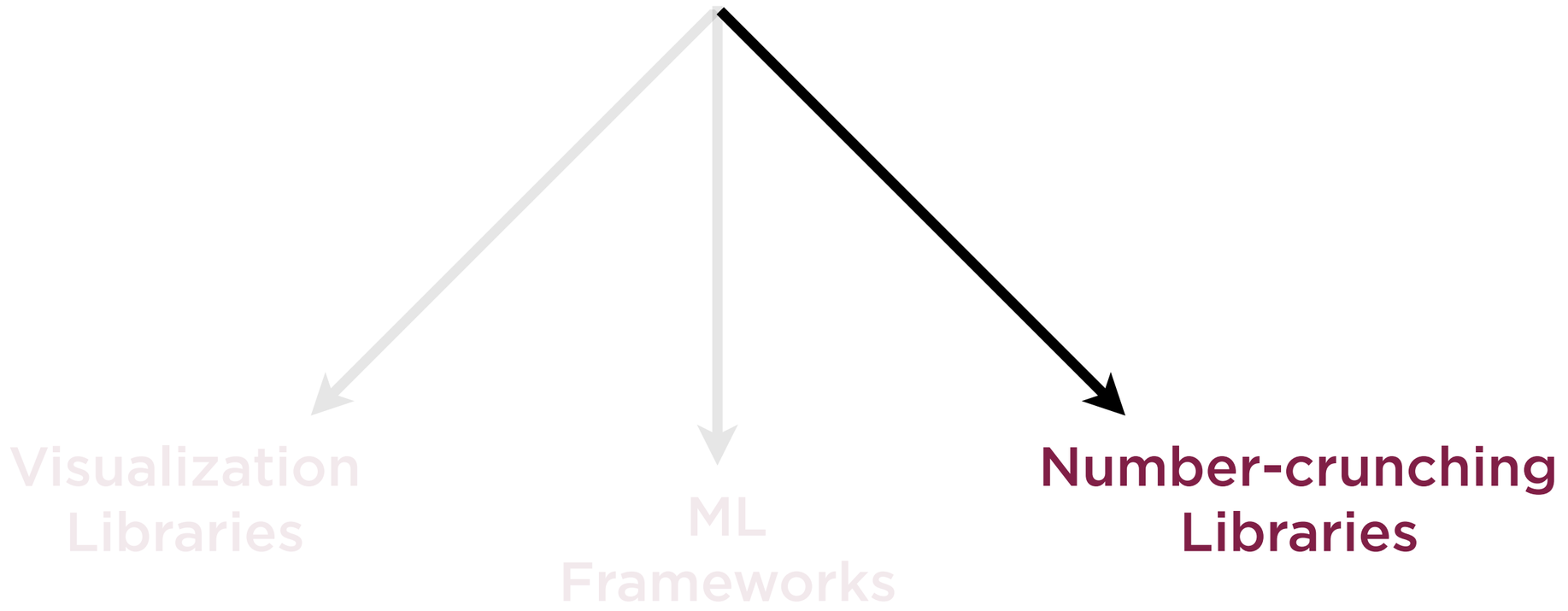
Apache MXNet

Microsoft CNTK

XGBoost

Theano

Analytics in Python



Number-crunching in Python

numpy

Perform operations on multidimensional arrays

pandas

Data analysis and manipulation

statsmodel

Estimate statistical models, and perform tests

scikit-image

Collection of algorithms for image processing

Demo

Exploring common online resources for data analysts

Workflows in Data Analytics

CRISP-DM

Standard six-step process used to perform data mining. Proposed in 1999 and still widely used.

CRISP-DM

Business understanding

Data understanding

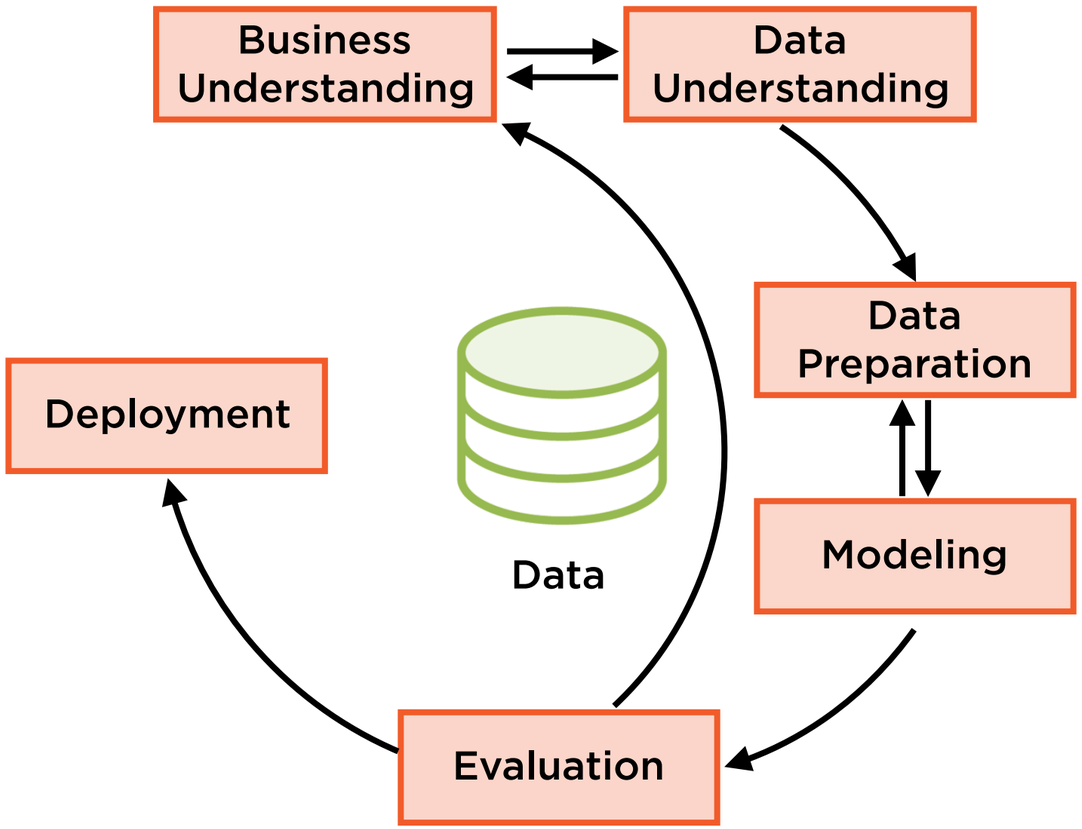
Data preparation

Modeling

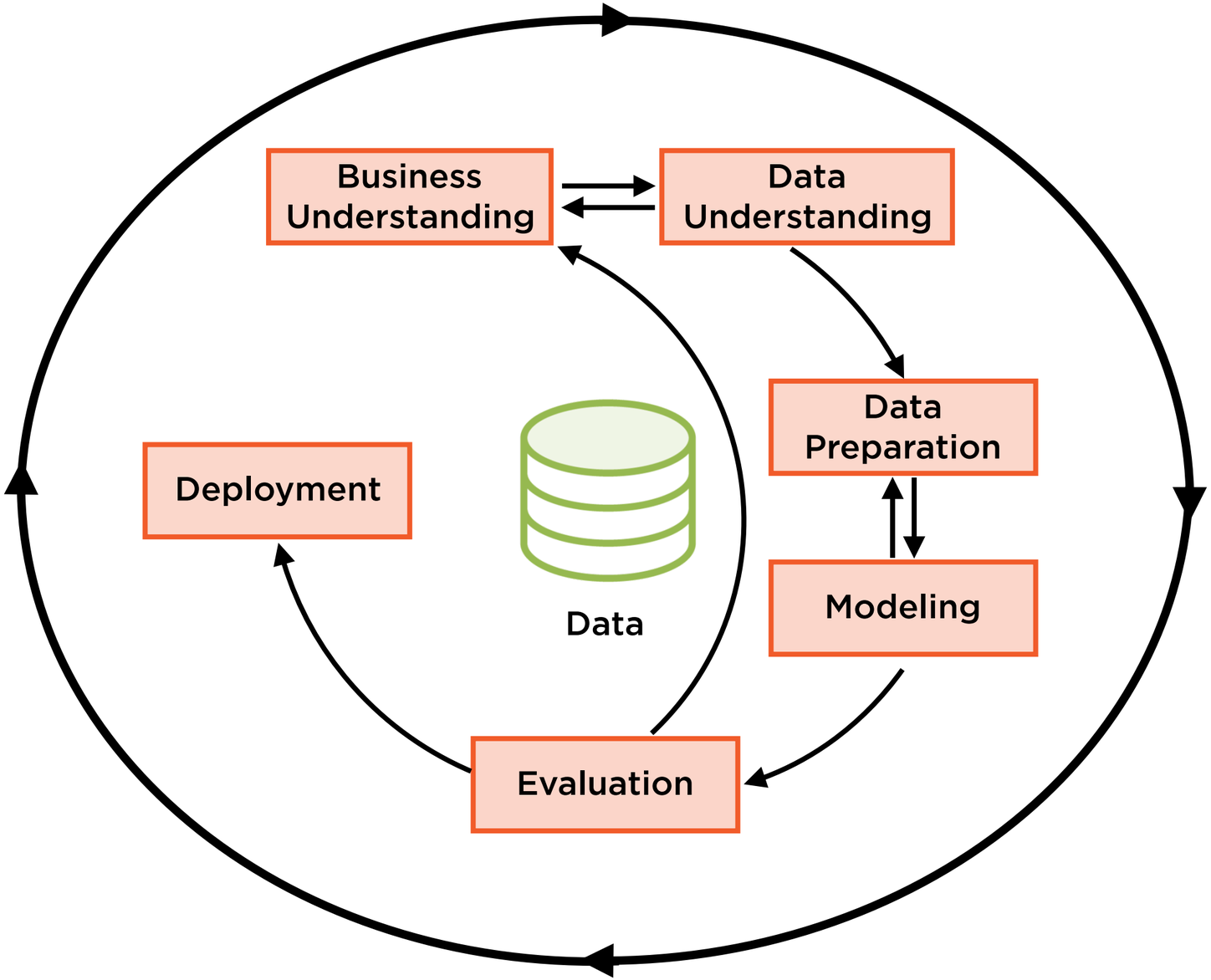
Evaluation

Deployment

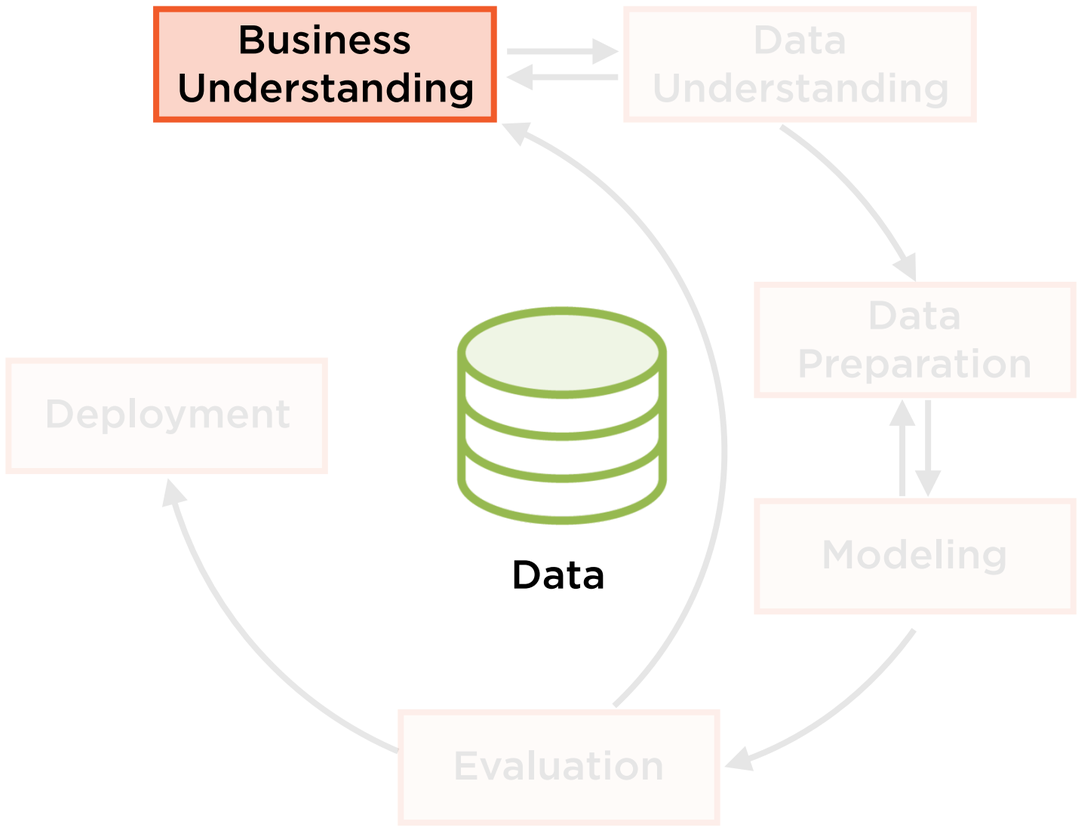
CRISP-DM


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data

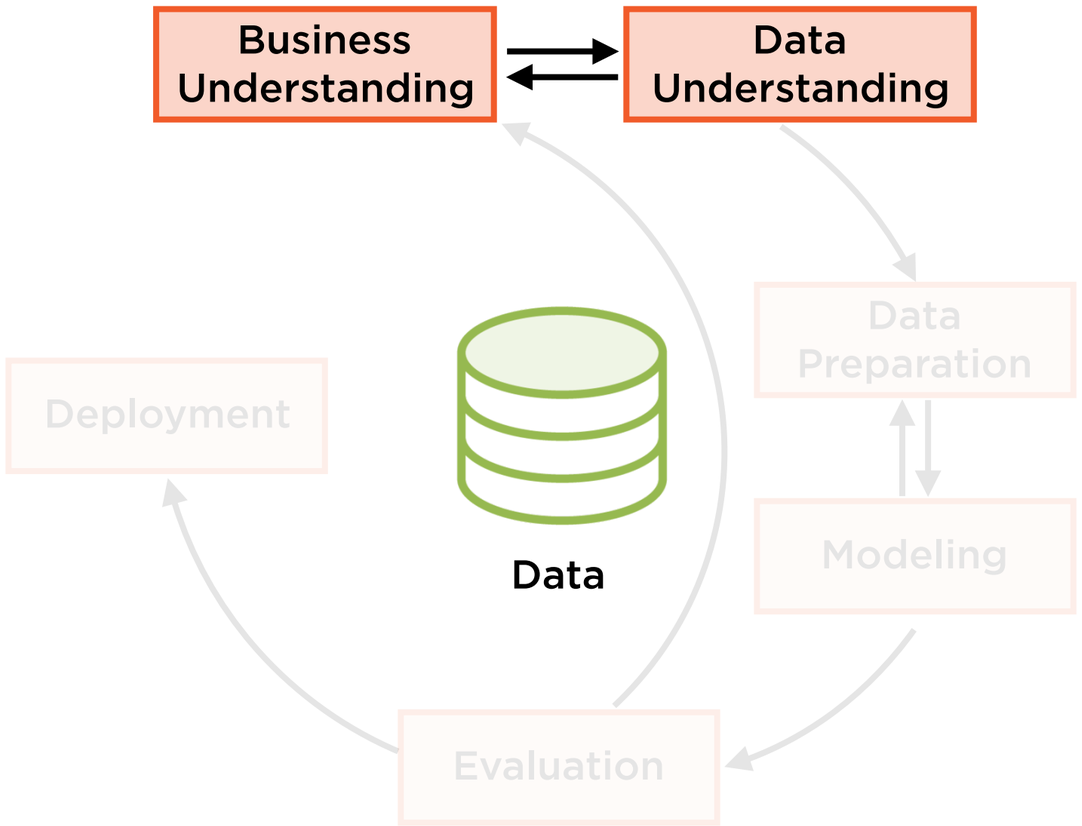
Data Mining: An Iterative Cycle



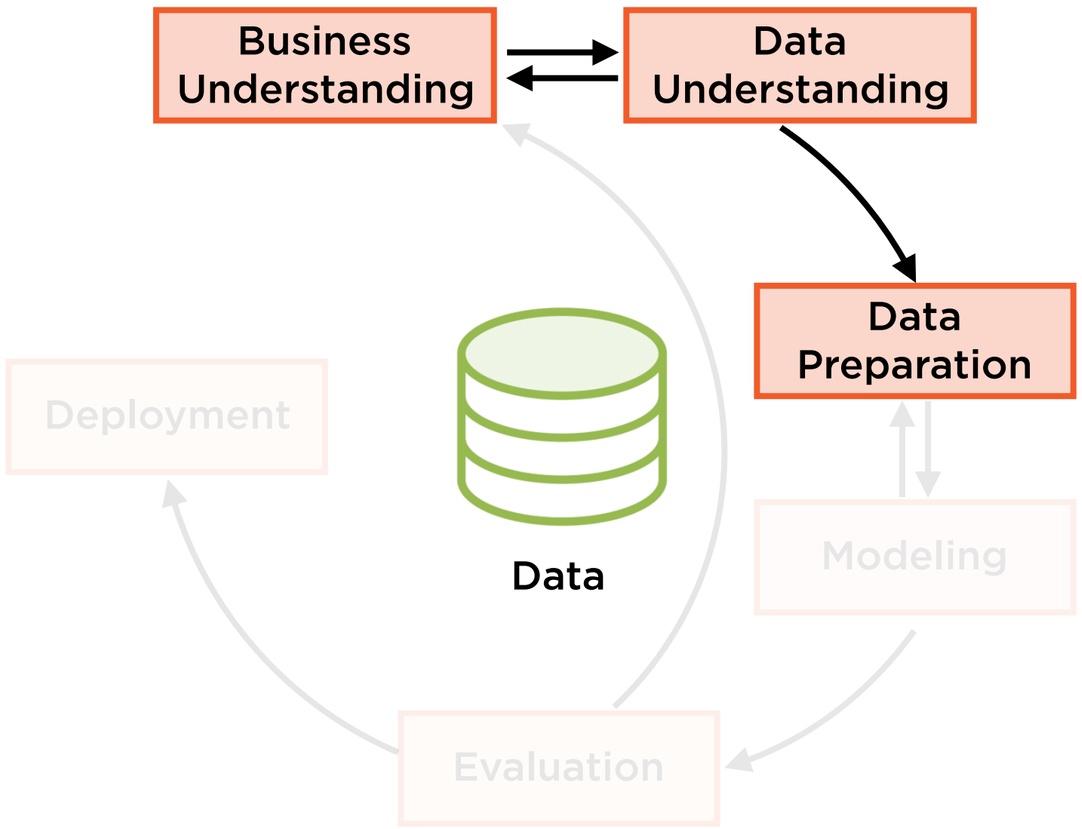
Know What You Want to Model



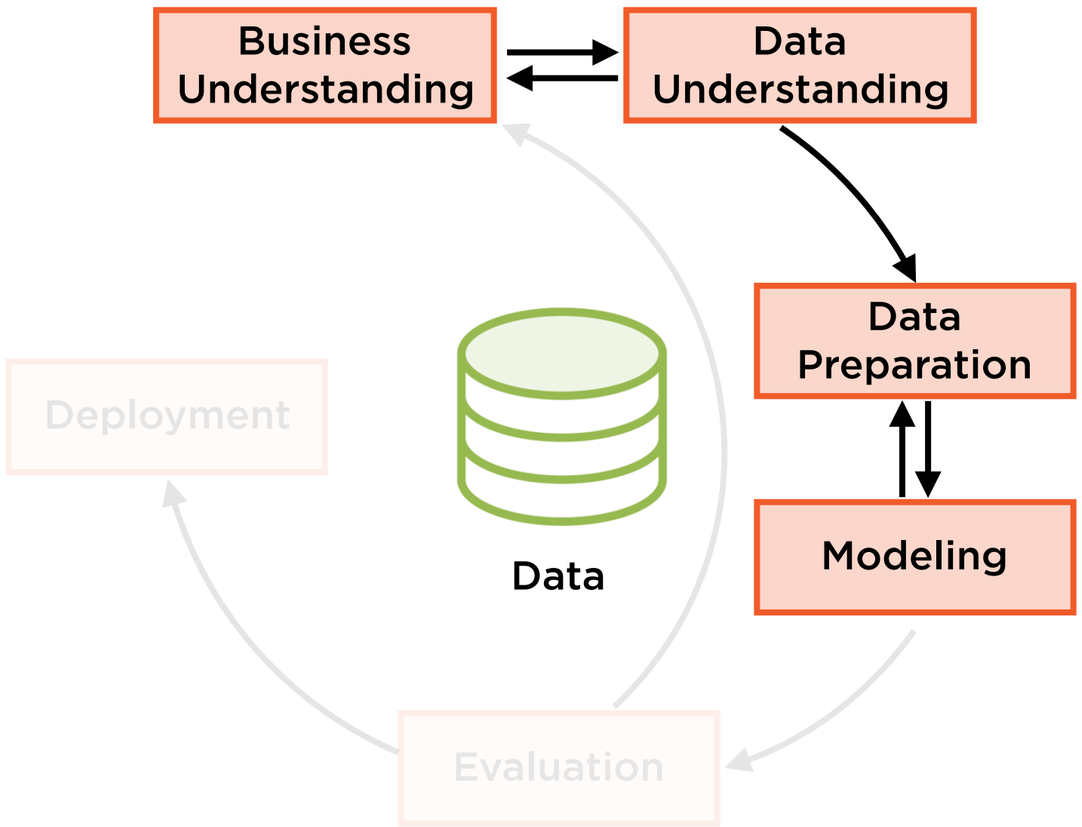
Understand the Data You Have to Work With



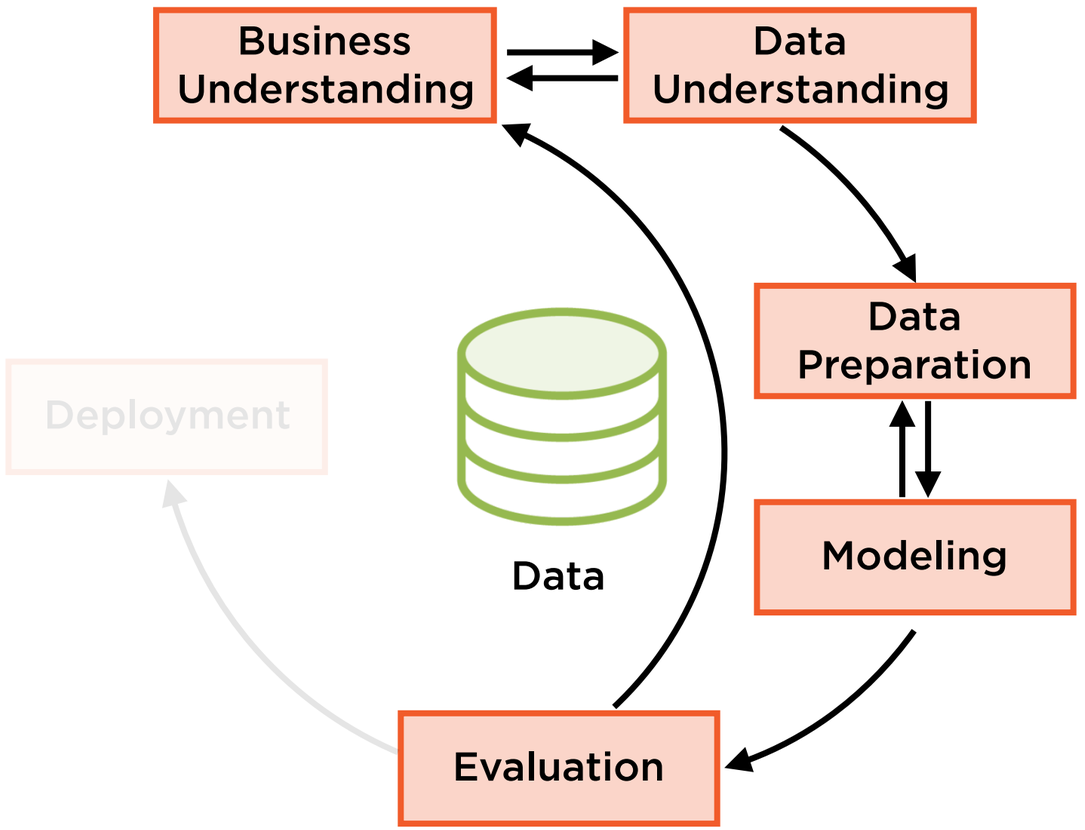
Prepare and Clean the Data



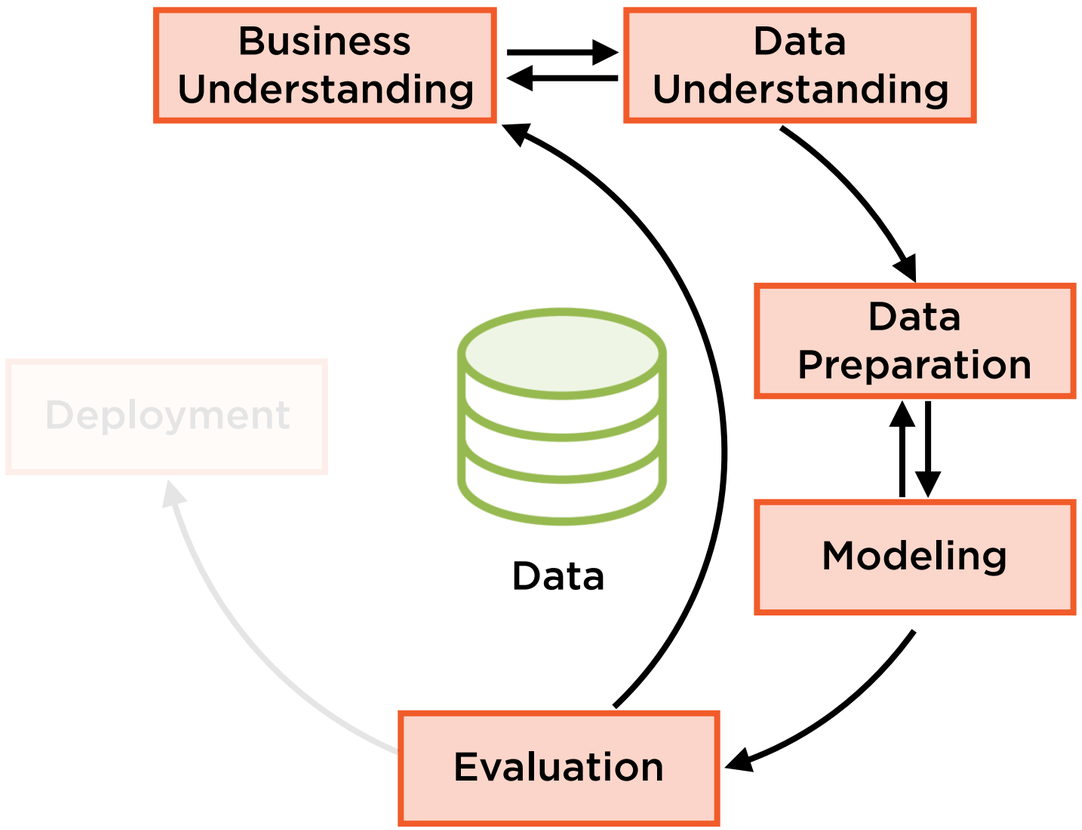
Build Predictive Models



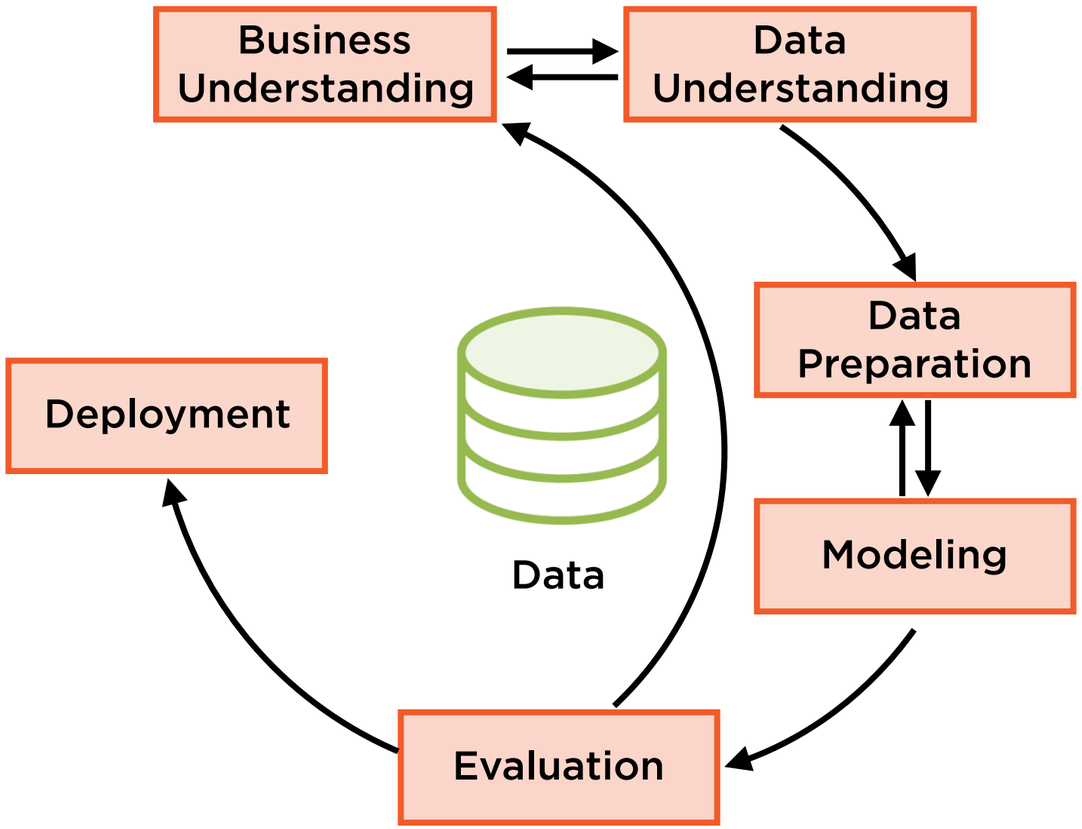
Apply Correct Evaluation Techniques



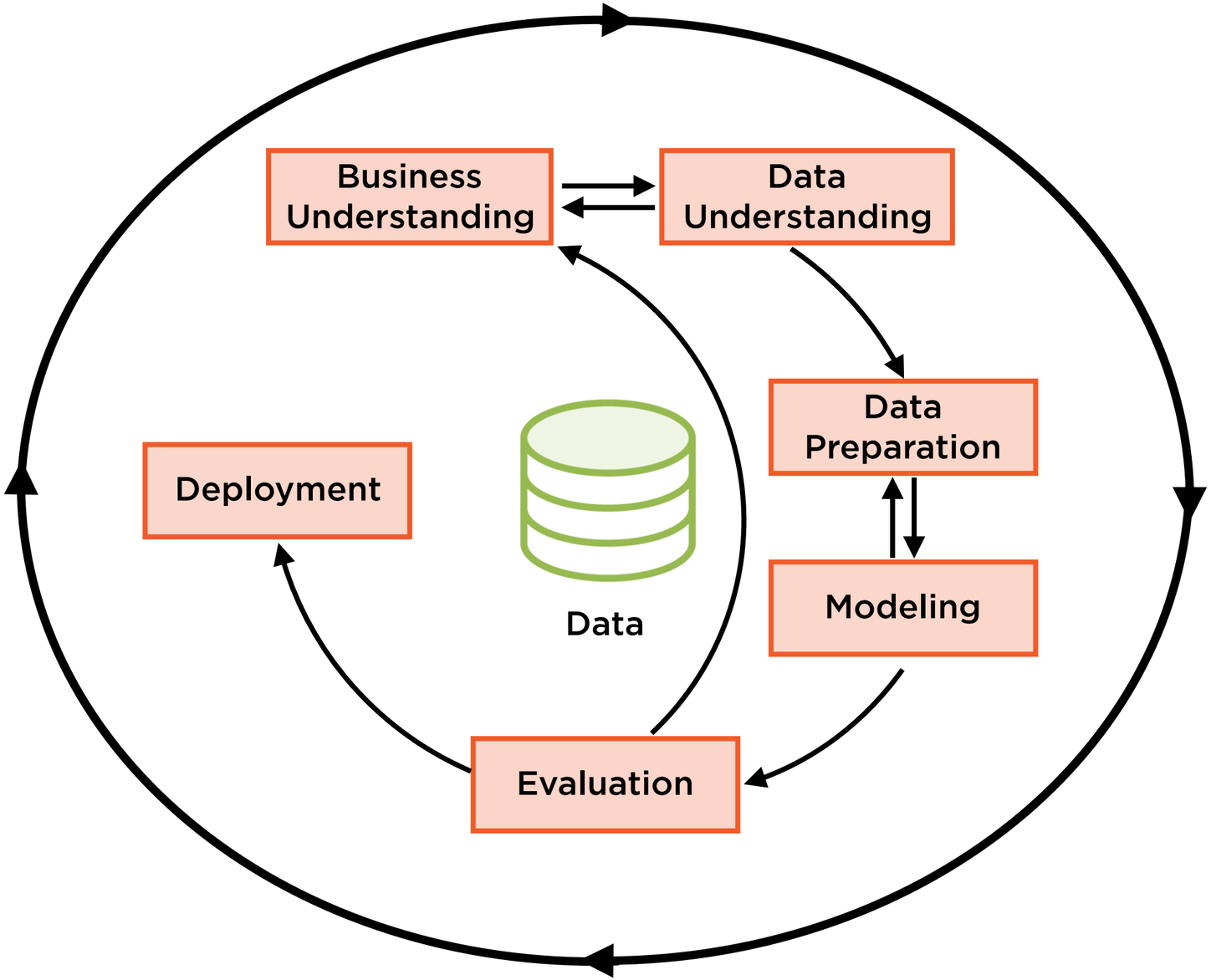
Requires Business Understanding



Deploy to Production

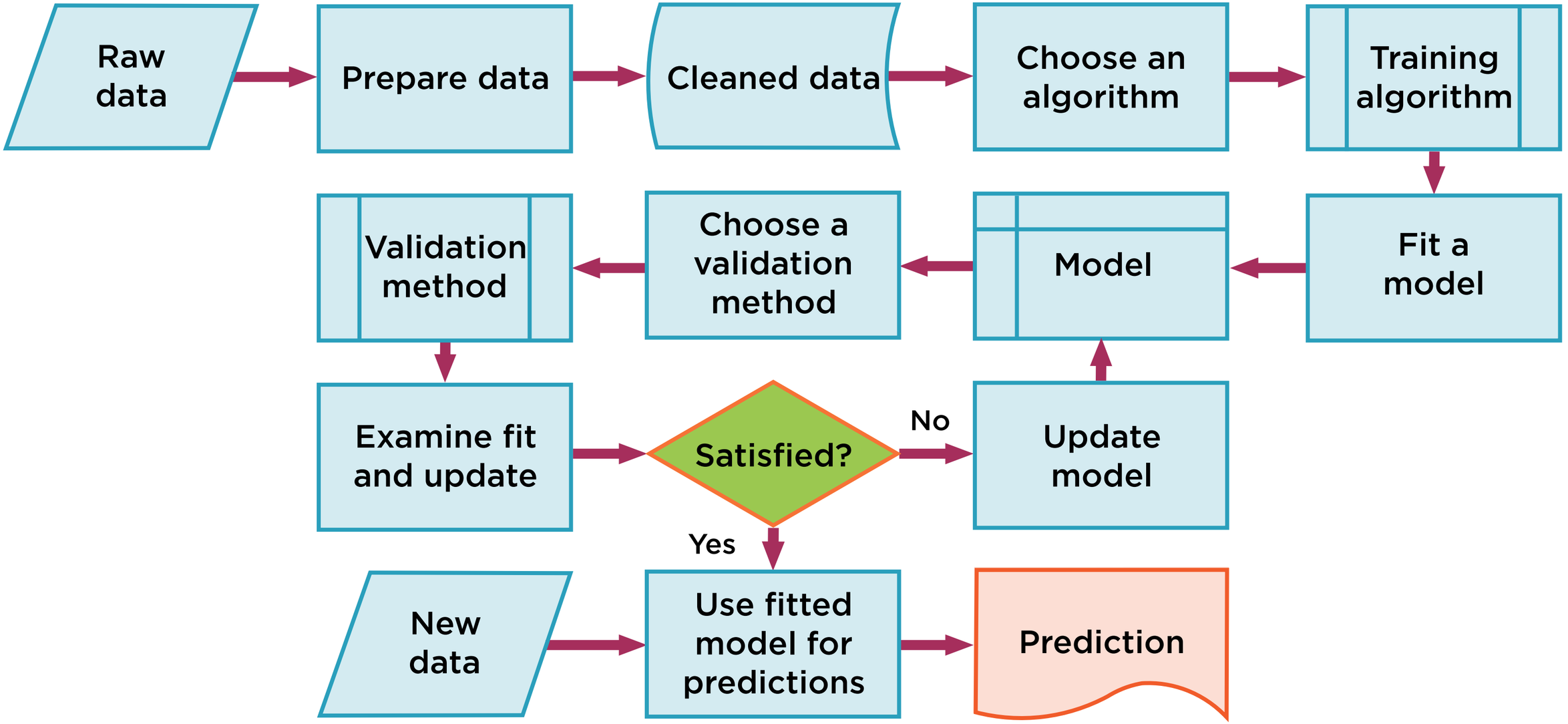


CRISP-DM

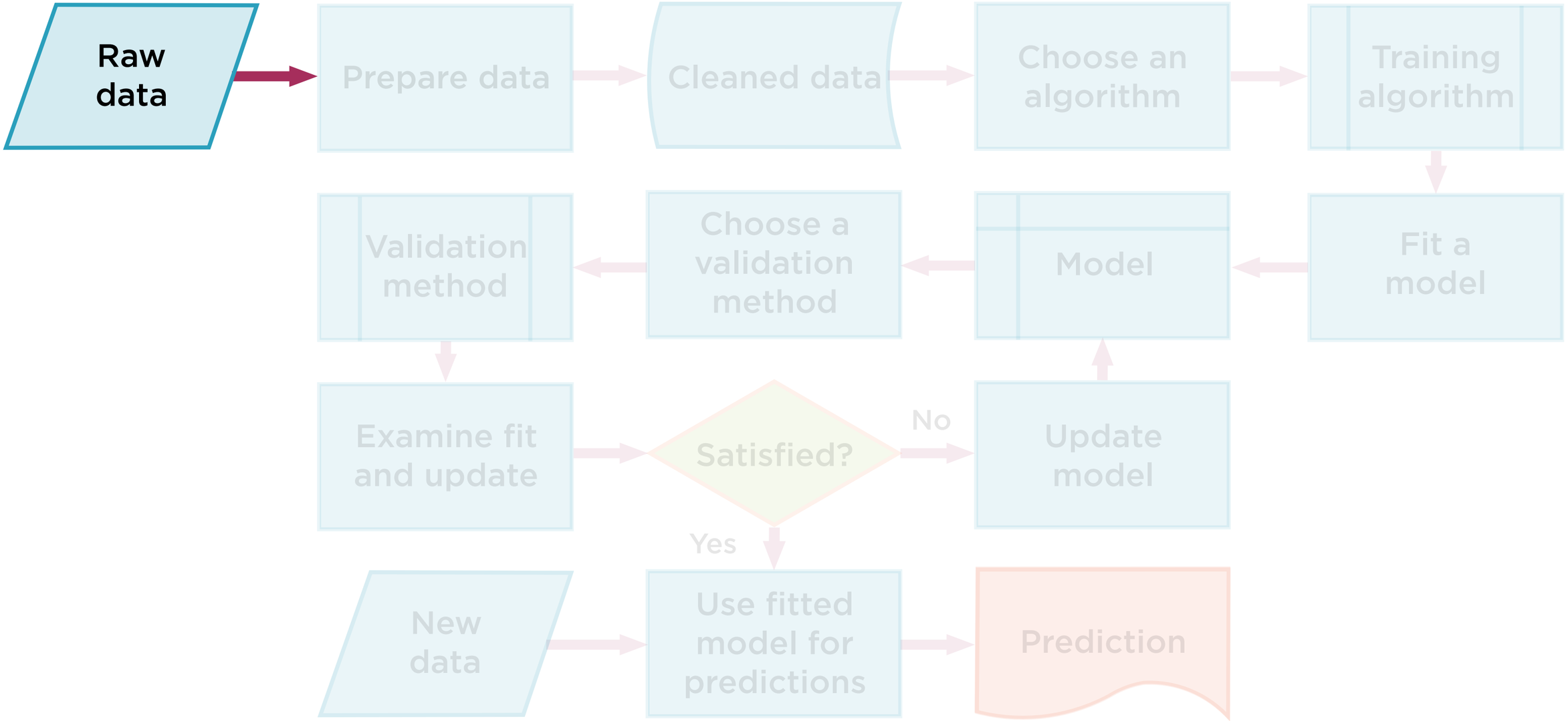

Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data

The CRISP-DM methodology closely matches the classic ML workflow in use today

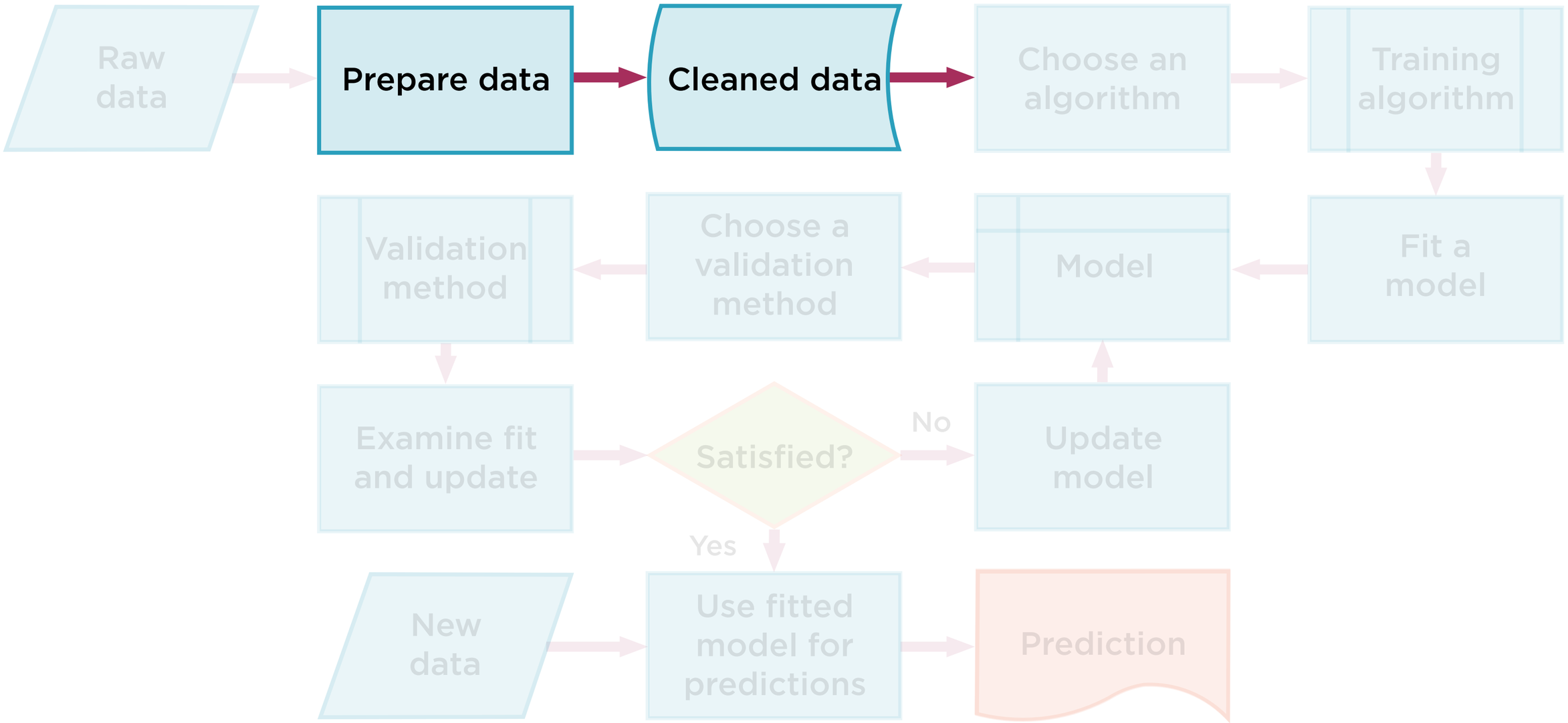
Basic Machine Learning Workflow



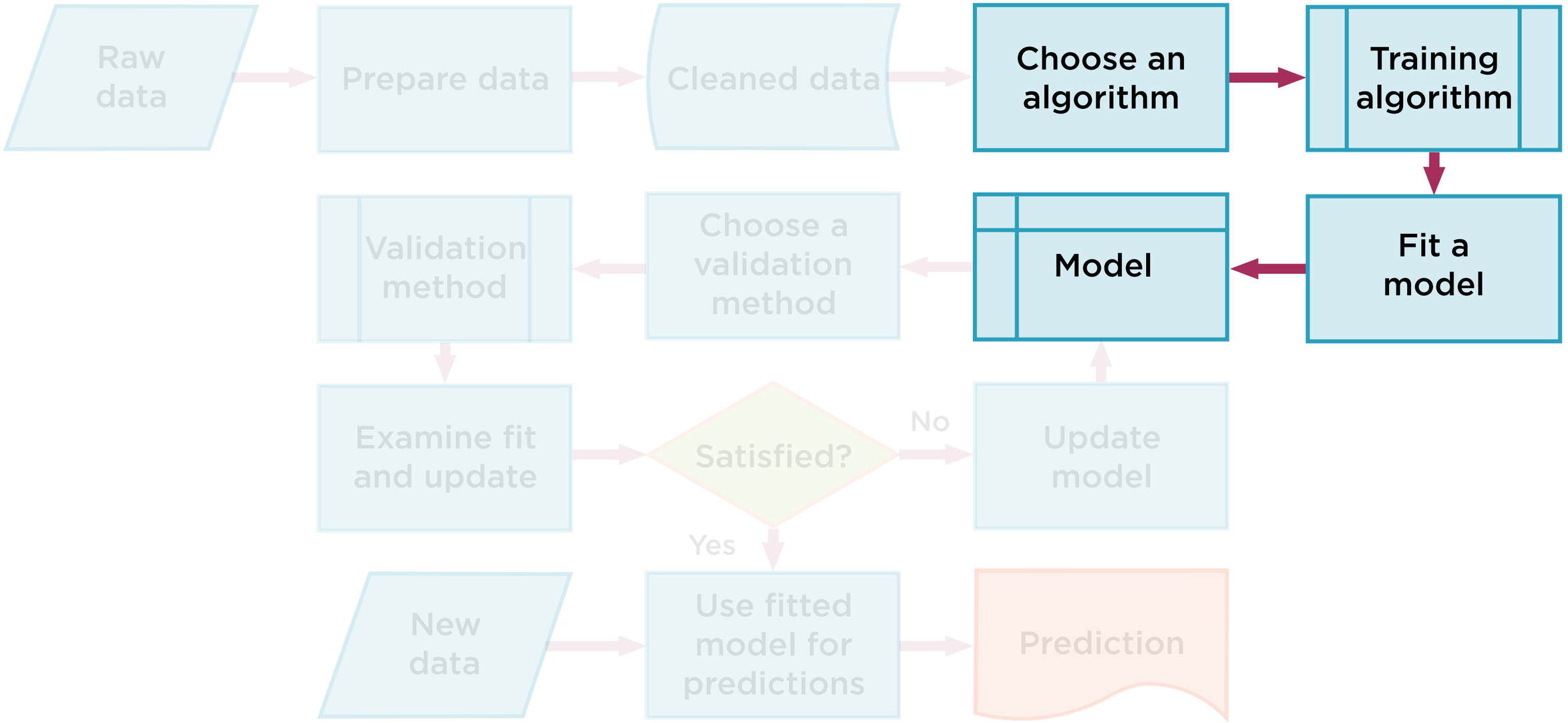
What Data Do You Have to Work With?



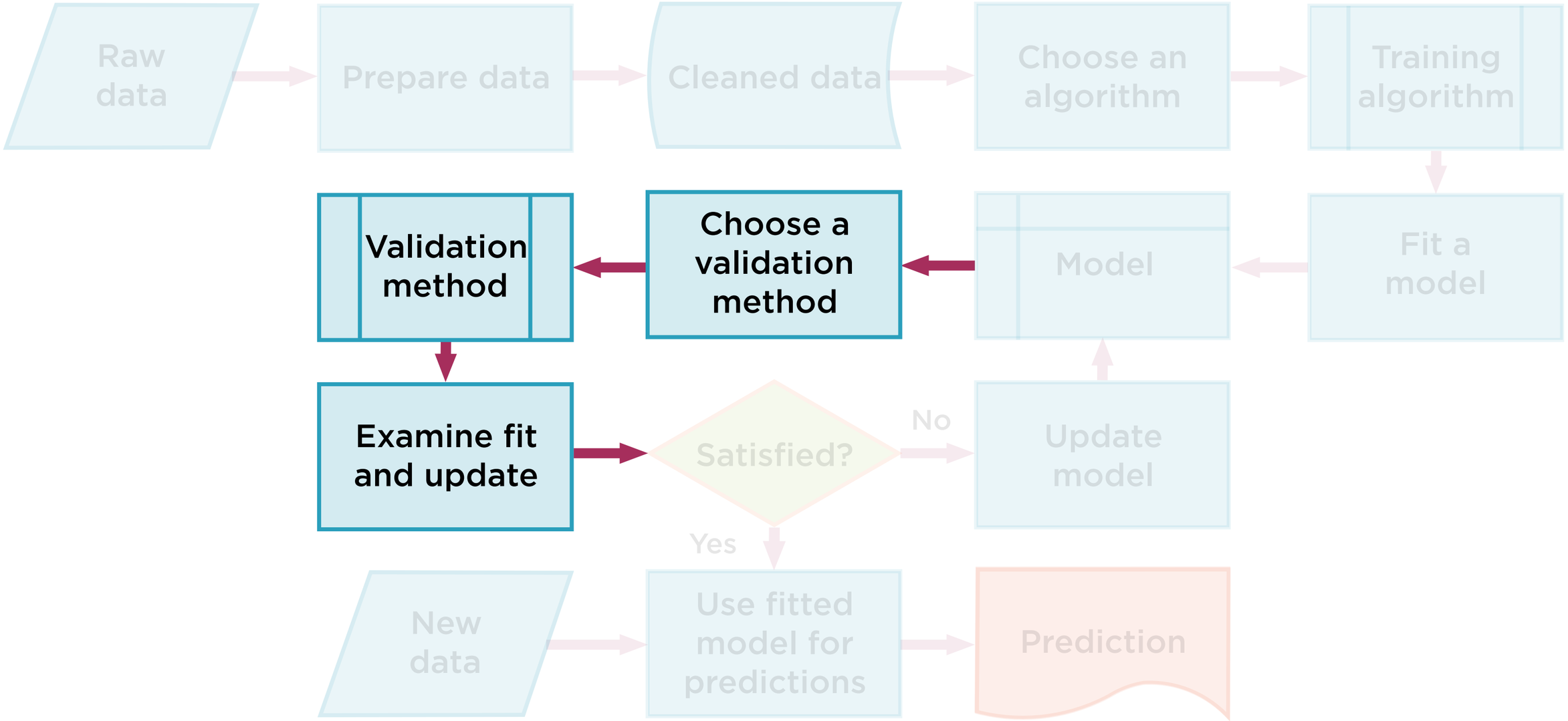
Data Preprocessing



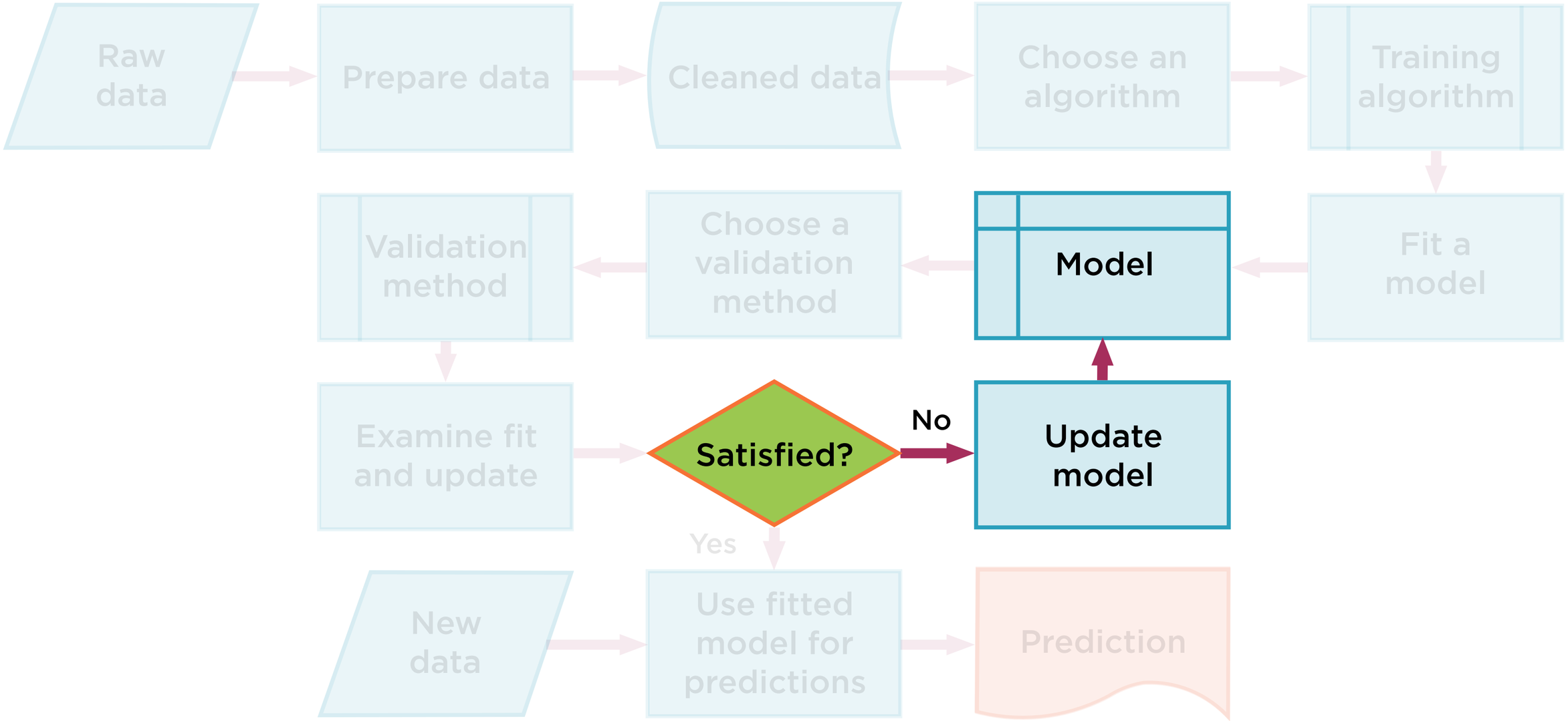
Build the Right Model



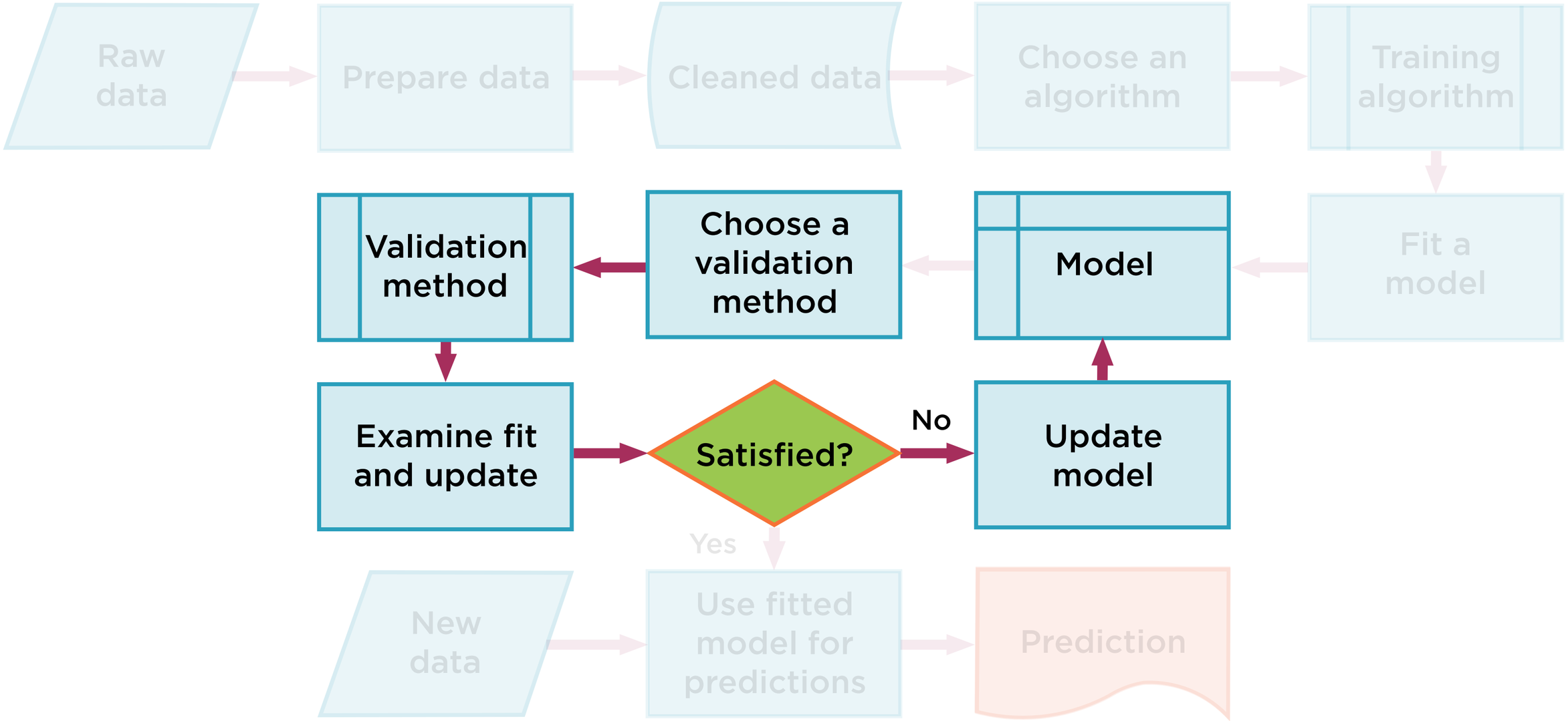
Evaluate and Score the Model



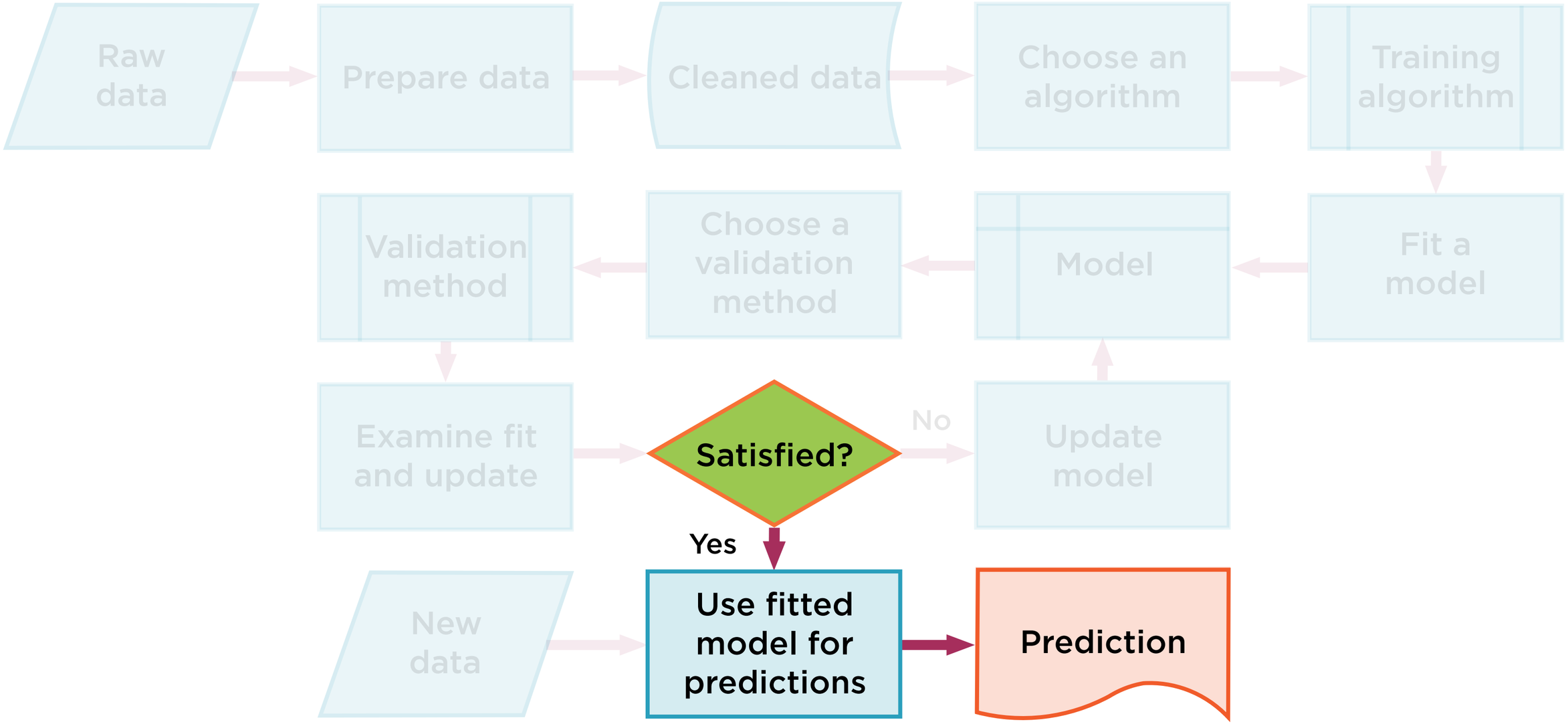
Different Algorithm, More Data, More Training?



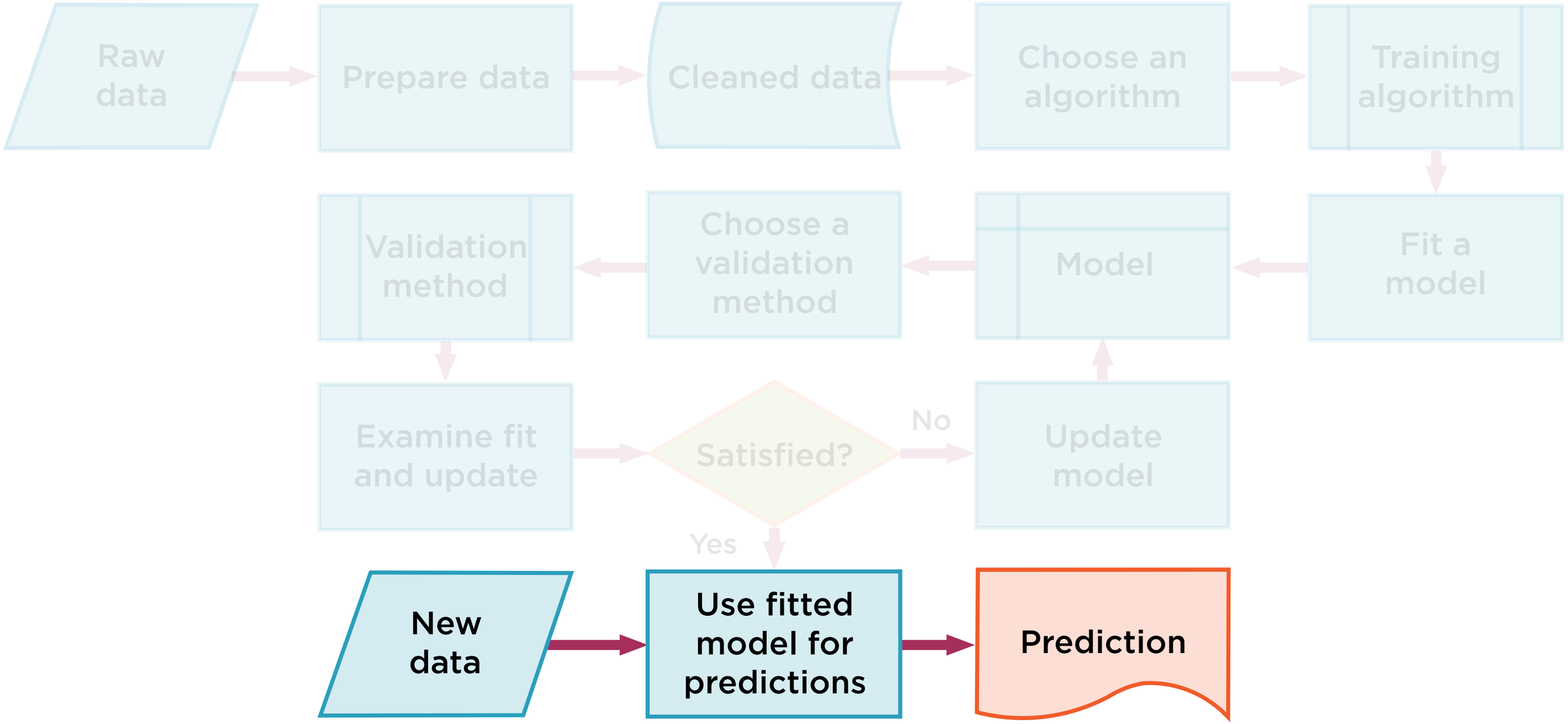
Iterate Till Model Finalized



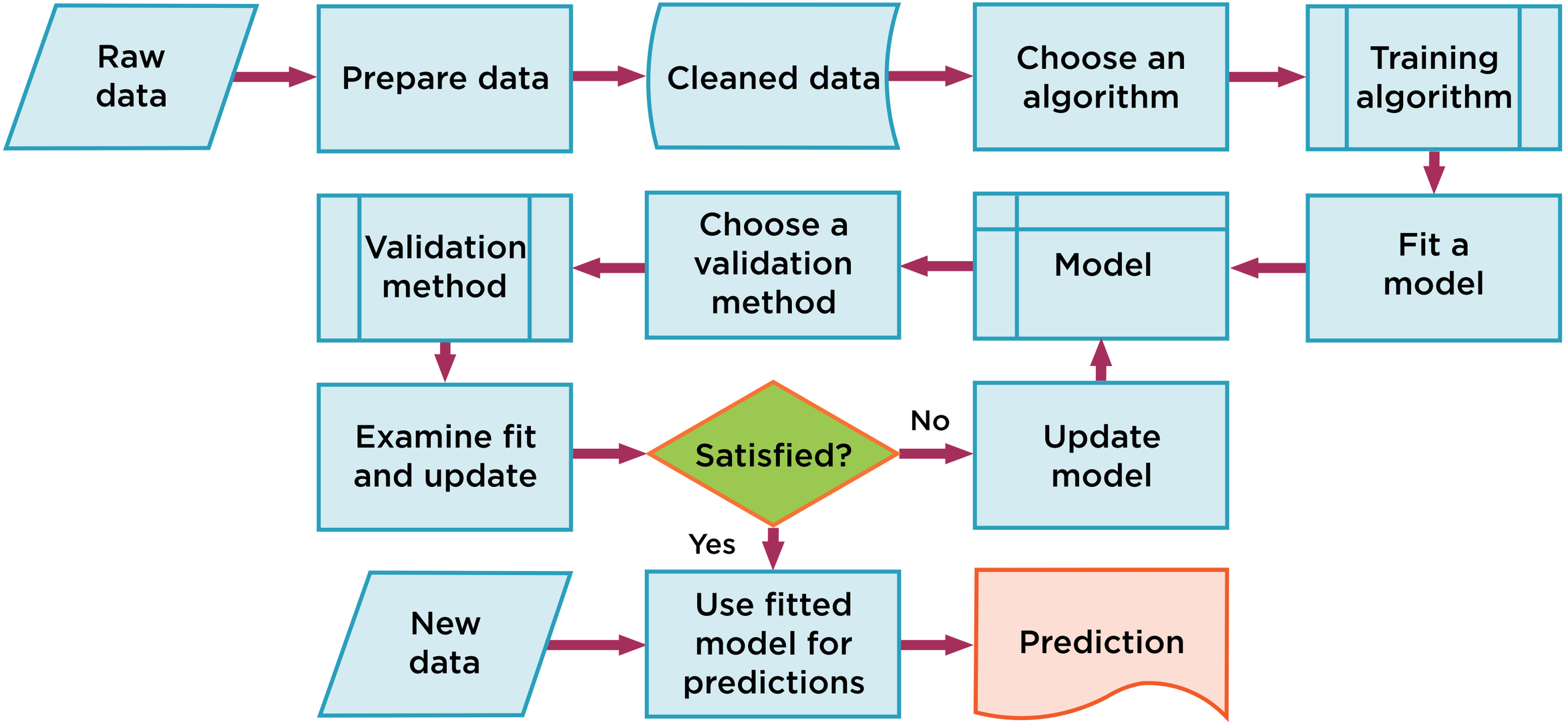
Model Used for Predictions



Retrained Using New Data



Basic Machine Learning Workflow



Demo

Cleaning, preparing and visualizing data

Demo

Prototyping a classification model

Demo

Creating a Python script to productionize a model

Summary

- Python for data analysts
- Explore commonly used online resources for Python analysts
- Classic analytics workflow
- Very similar to machine learning workflow
- Prototype models on Jupyter notebooks
- Productionize models using a Python script