

Topic 2

Applying Machine Learning (ML) using Tools

ST1511 (AIML) AI & MACHINE LEARNING

Learning Outcomes

□ Use of Machine Learning Tools

- Describe some online/cloud based machine learning tools
- Configure and setup machine learning in Orange DM

□ Understand principles of data representation and feature engineering

- Understanding representation of categorical variables
- Explain binning, discretization
- Understand feature selection
- Understand how to utilize expert knowledge

Use of Cloud-based Machine Learning Tools



Why Use Cloud Computing?



**Data is growing exponentially, we
created 44×10^{12} GB data**

**Cloud Machine Learning help you
make sense of it**

Ready AI/ML Cloud Services

IBM Watson

- Forecasting models
- Natural language processing
- Language translation
- Speech recognition



Ready AI/ML Cloud Services

Amazon AWS

- AWS ML: linear/logistic regression (classification and prediction)
- Amazon Lex: chatbots
- Amazon Recognition: object detection
- Amazon Polly: Speech recognition



Ready AI/ML Cloud Services

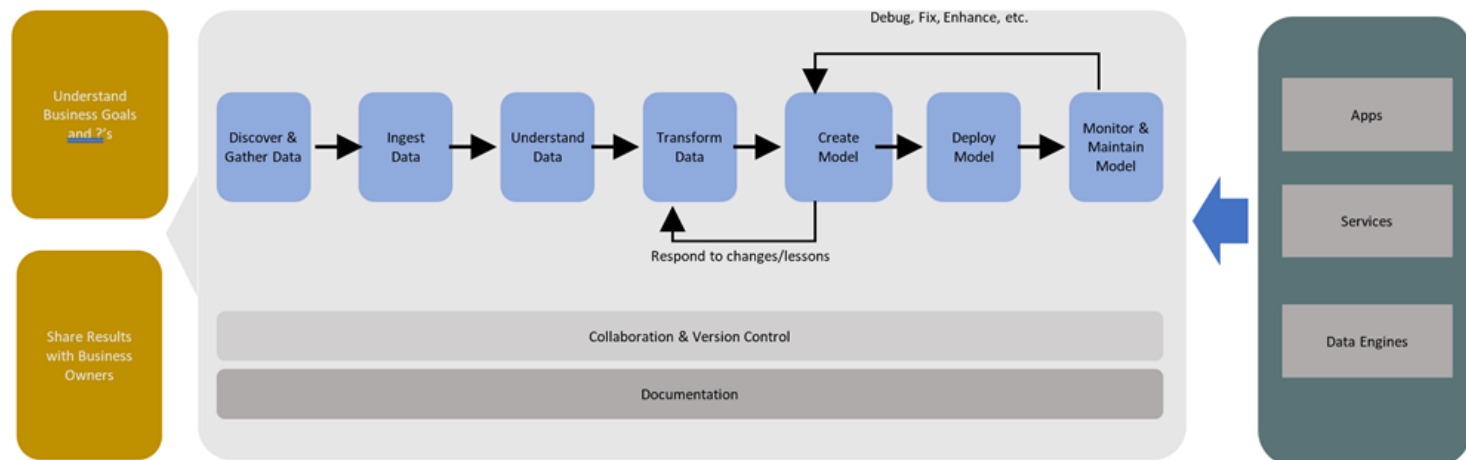
Google Cloud

- Vertex AI: Unified ML platform to build and deploy AI models
- **Auto ML**: automatically train ML models with minimal efforts
- Conversational AI: chatbots, speech recognition
- And many more...



Microsoft Azure ML

- ❑ Cloud based tool that covers the entire workflow from data source to deployment of machine learning algorithm as a web service
- ❑ Cloud based tool to perform all the steps needed to prepare the data, explore the data, create models, score and evaluate the models



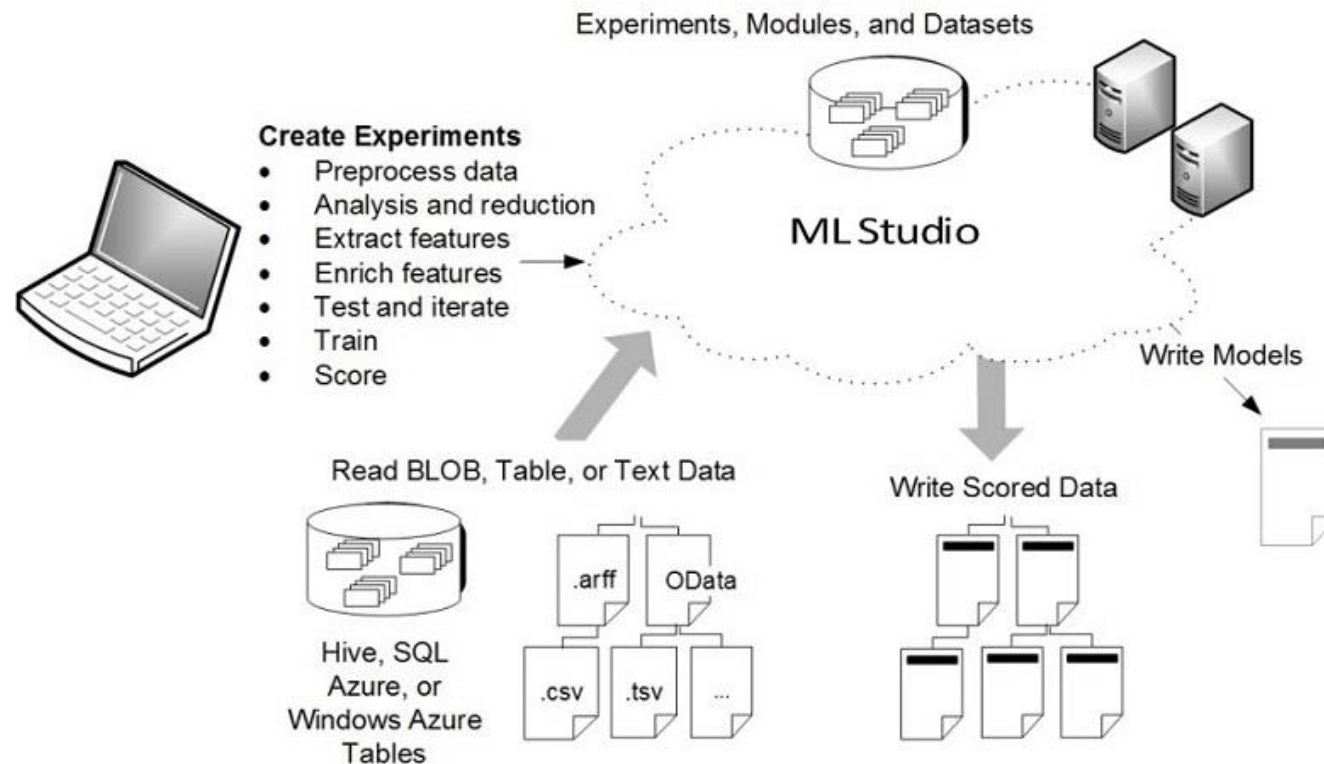
source <https://azure.microsoft.com/en-us/blog/diving-deep-into-what-s-new-with-azure-machine-learning/>

Azure Machine Learning

Microsoft Azure Machine Learning service is one of the main platforms for doing Machine Learning in a quick, easy, cloud-based way.

The service contains a set of tools and modules that help the data scientist setup and run the Machine Learning process. It is designed for applied machine learning meaning and is designed to be used by real world applications and developers.

What is Azure ML Studio?



source <https://docs.microsoft.com/en-us/azure/machine-learning/studio/what-is-ml-studio>

Benefits of Cloud Platform for ML

- **Cost Efficiency:** the cloud has pay-to-use machine learning models.
- **Easy to scale up:** companies can test and deploy model on the cloud, and then scale up as need and demand increases.



Will cloud platform and AutoML replace data scientist?

Definitely no! The data scientists who embrace AutoML will be able to expand deeper into the capabilities of machine learning and become more effective.

Orange Data Mining tools



Why Orange?



Introduction

- Orange is component based visual programming software for data mining.
 - Machine learning and data analysis.
 - Supports communication between data scientist and domain experts.
- **Open Source**
 - **No Programming**
 - **Data Visualization**
 - **Allows machine learning**
 - **Platform independent software**

Orange History

- 1996 – University of Ljubljana and Jozef Stefan Institute started development of ML*, a machine learning framework in C++.
- 1997 – Python integration layer.
- 2003 – GUI based on PyQt.
- 2013 – Orange Canvas 2.7 released – Major GUI redesign.

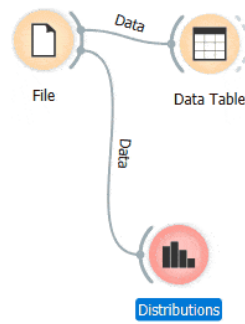
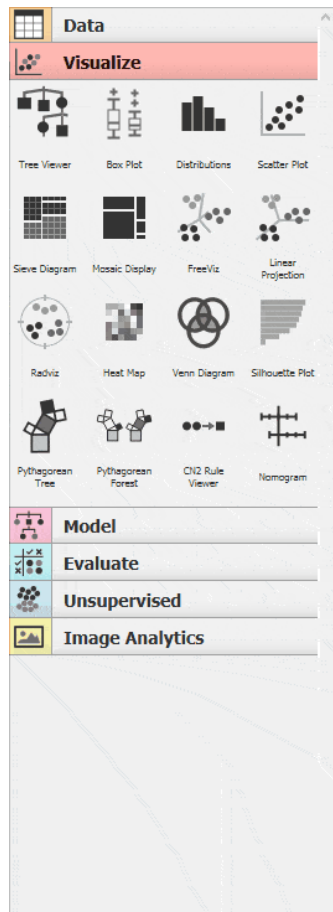
High-Level Architecture

Orange Canvas – Visual Programming

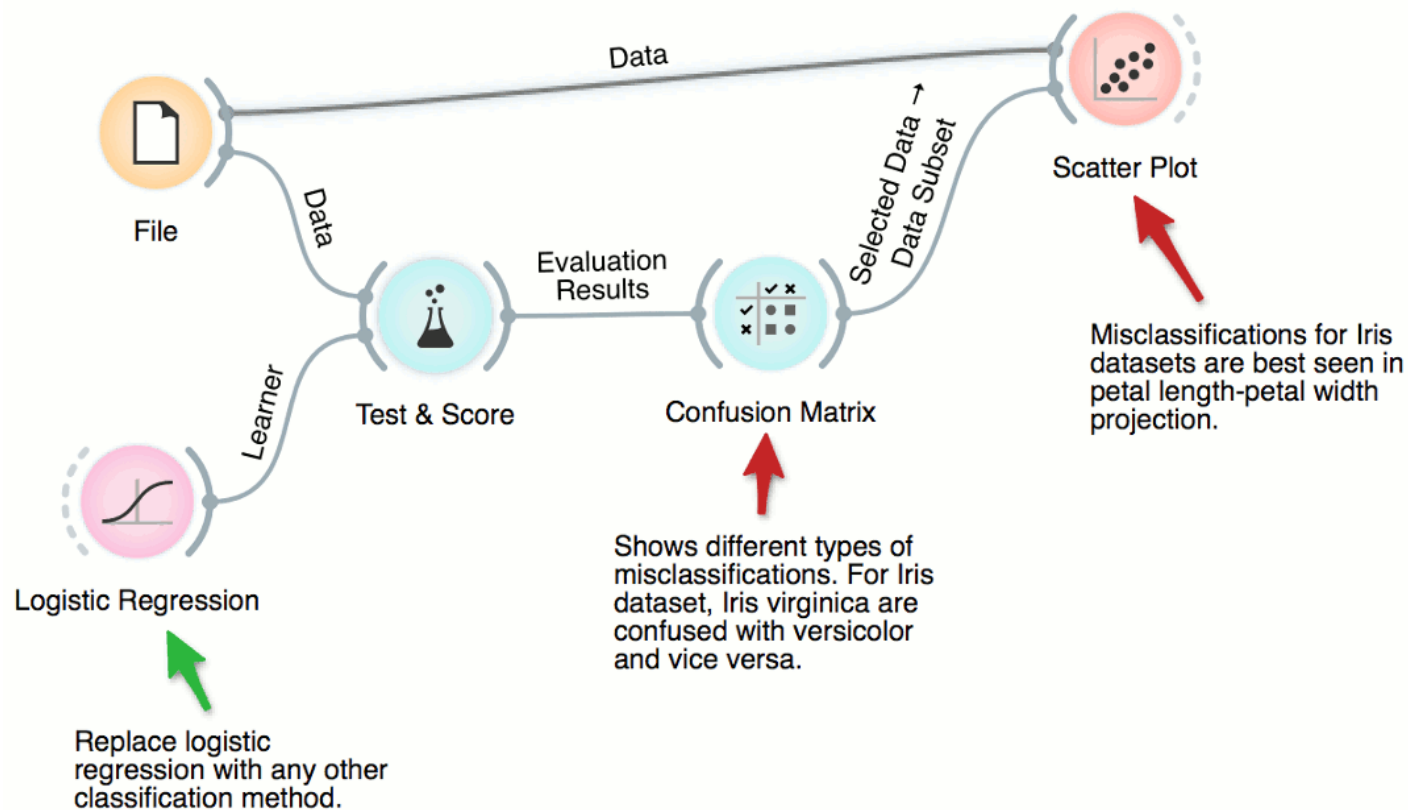
Python Integration layer (Python 3.x)

Algorithms written in C++

Simple Interface and various Widgets/Modules to use



Classification Setup Scheme

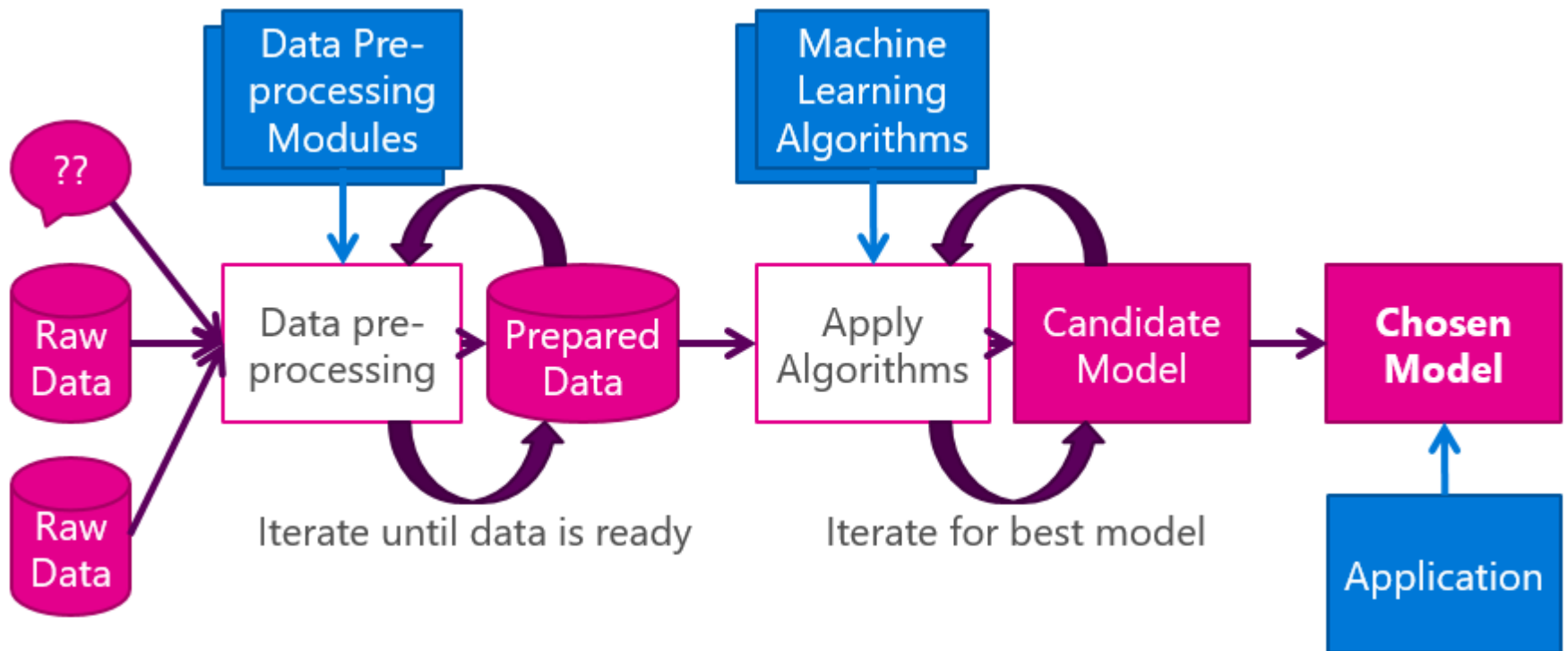


Exercise Time

(Download Orange and Play around)

<https://orangedatamining.com/download/#windows>

Machine Learning Process



source <https://blogs.msdn.microsoft.com/martinkearn/2016/03/01/machine-learning-is-for-muggles-too/>

Machine Learning Process

- **The primary goal of the process is to identify a 'Model'**. The Model is the main thing that applications can submit requests to in order to gain insight on new data. A person working as the role of a Data Scientist performs the Machine Learning process and will ultimately decide on the right model to use.
- **The process starts with a question**; what are you trying to learn from your Machine Learning experiment? For example, in the case of recommendations, the question might be *"identify most commonly sold products for each product in the inventory"*

Machine Learning Process

- **The next step is to provide 'prepared data'**. Prepared data is one or more data sets that have been pre-processed (formatted, cleaned and sampled) in readiness to apply Machine Learning algorithms to. Preparing the data means that the data is in the best shape to draw scientific conclusions from and is not skewed in any way.
- Once you have your prepared data, you apply one or more Machine Learning algorithms to it with a view to producing a **Model**. This is an iterative process and you may loop around testing various algorithms until you have a Model that sufficiently answers your question.
- Once you have produced your chosen model, it will typically be exposed via some kind of API.

Choosing a ML algorithm

- Every machine learning algorithm has its own style or *inductive bias*. For a specific problem, several algorithms may be appropriate and one algorithm may be a better fit than others. But it's not always possible to know beforehand which is the best fit. In cases like these, several algorithms are listed together in the cheat sheet. An appropriate strategy would be to try one algorithm, and if the results are not yet satisfactory, try the others.

Choosing a ML algorithm

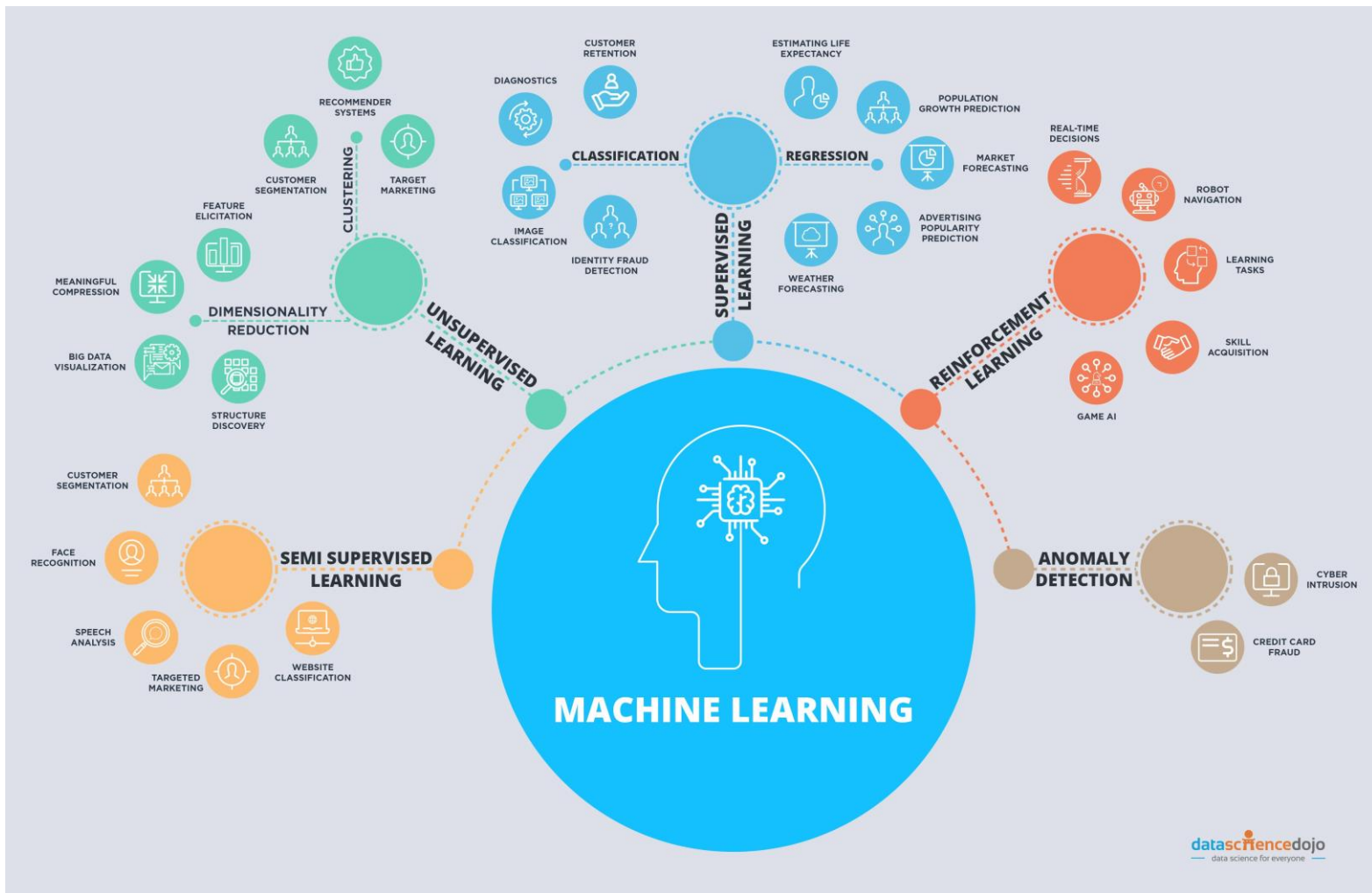
There are three main categories of machine learning: **supervised learning**, **unsupervised learning**, and **reinforcement learning**.

- In **supervised learning**, each data point is **labelled** or associated with a **category** or **value** of interest. An example of a categorical label is assigning an image as either a 'cat' or a 'dog'. An example of a value label is the sale price associated with a used car. The goal of supervised learning is to study many labelled examples like these, and then to be able to make predictions about future data points. For example, identifying new photos with the correct animal or assigning accurate sale prices to other used cars. This is a popular and useful type of machine learning. All of the modules in Azure Machine Learning are supervised learning algorithms except for K-Means Clustering.

Choosing a ML algorithm

- In **unsupervised learning**, data points have no labels associated with them. Instead, the goal of an unsupervised learning algorithm is to organize the data in some way or to describe its structure. This can mean grouping it into clusters, as K-means does, or finding different ways of looking at complex data so that it appears simpler.
- In **reinforcement learning**, the algorithm gets to choose an action in response to each data point. It is a common approach in robotics, where the set of sensor readings at one point in time is a data point, and the algorithm must choose the robot's next action. It's also a natural fit for Internet of Things applications. The learning algorithm also receives a reward signal a short time later, indicating how good the decision was. Based on this, the algorithm modifies its strategy in order to achieve the highest reward. Currently there are no reinforcement learning algorithm modules in Azure ML.

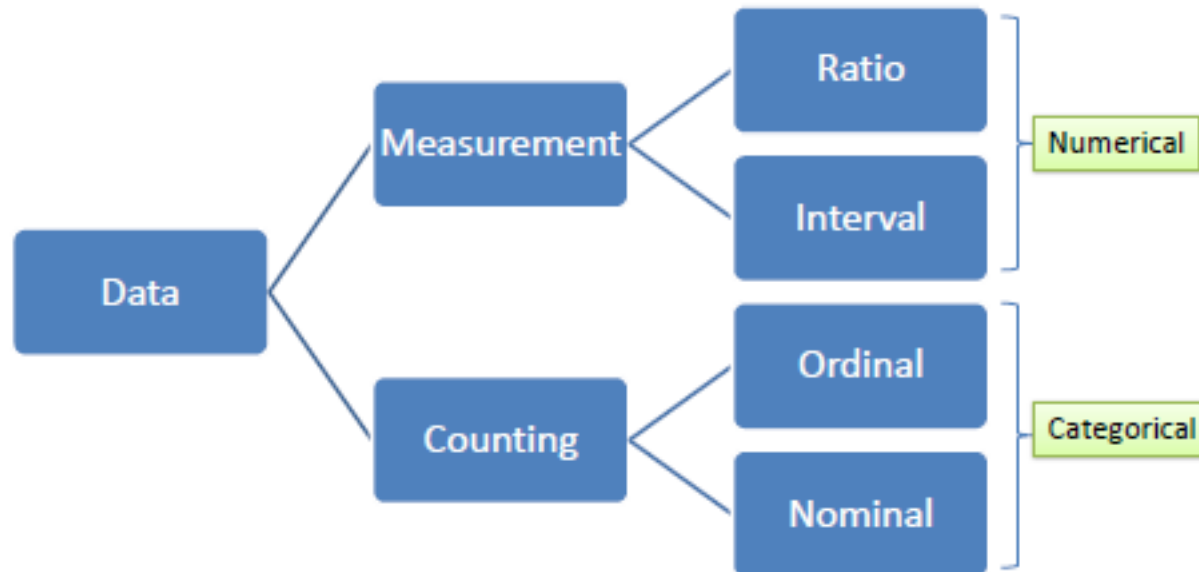
Choosing a ML algorithm



source <https://www.r-bloggers.com/2019/07/101-machine-learning-algorithms-for-data-science-with-cheat-sheets/>

Understand Principles of Data Representation and Feature Engineering

Data Types



source http://www.saedsayad.com/data_preparation.htm

Data Types

Data is information typically the results of measurement (numerical) or counting (categorical). Variables serve as placeholders for data. There are two types of variables, numerical and categorical.

- A **numerical** or **continuous variable** is one that can accept any value within a finite or infinite interval (e.g., height, weight, temperature, blood glucose, ...). There are two types of numerical data, **interval** and **ratio**. Data on an interval scale can be added and subtracted but cannot be meaningfully multiplied or divided because there is no true zero. For example, we cannot say that one day is twice as hot as another day. On the other hand, data on a ratio scale has true zero and can be added, subtracted, multiplied or divided (e.g., weight).
- A **categorical** or **discrete variable** is one that can accept two or more values (categories). There are two types of categorical data, **nominal** and **ordinal**. Nominal data does not have an intrinsic ordering in the categories. For example, "gender" with two categories, male and female. In contrast, ordinal data does have an intrinsic ordering in the categories. For example, "level of energy" with three orderly categories (low, medium and high).

ML algorithms require the inputs variables to be numerical

- No problem for data that is already a real number or integer
- Categorical variables need to be converted or encoded into a number

Encoding categorical variables

- Use of dummy variables encoding for categorical variables

Original values (Direction)	I0	I1	I2
NORTH	1	0	0
EAST	0	1	0
SOUTH	0	0	1
WEST	0	0	0

Number of dummy/indicator variables = number of values – 1
In the case 4 possible values – 1 = 3 (I0,I1,I2 dummy variables)
The original column is replaced by 3 new columns

Encoding categorical variables

- Use of one-hot encoding for categorical variables

Original values (Direction)	I0	I1	I2	I3
NORTH	1	0	0	0
EAST	0	1	0	0
SOUTH	0	0	1	0
WEST	0	0	0	1

One-hot encoding of variables = number of values
In the case 4 possible values = 4 (I0,I1,I2,I3 variables)
The original column is replaced by 4 new columns

Why not just use a value encoding (e.g. LabelEncoder)?

The problem with value encoding or LabelEncoder is that it leads to undesirable side effects if the ML algorithm uses the numerical values. For example, Average of East and West = $(1+3)/2 = 2$ (South).

Direction	Value
NORTH	0
EAST	1
SOUTH	2
WEST	3

This is nonsense as the average of East and West has no meaning.

Discretization/Binning

Discretization (otherwise known as quantization or binning) provides a way to partition continuous features into discrete values.

Certain datasets with continuous features may benefit from discretization, because discretization can transform the dataset of continuous attributes to one with only nominal attributes.

Time	TimeOfDay
00:00	0
07:30	1
08:00	1
12:00	2
13:00	2
14:00	2
17:00	2

We convert the 24H into 4 time period bins
00:00-06:00 is the bin 0 for early morning

Dataset

Dataset is a collection of data, usually presented in a tabular form. Each **column** represents a particular **variable (attribute)**, and each **row** corresponds to a given **member (observation)** of the data.

In predictive modeling, **predictors** or **attributes** are the **input variables** and **target** or **class attribute** is the **output variable** whose value is determined by the values of the predictors and function of the predictive model.

The diagram shows a table with 14 rows and 6 columns. An orange bracket at the top labeled 'Columns' spans the header row. An orange bracket on the left labeled 'Rows' spans the first column. Red dashed lines on the right labeled 'Values' point to specific data cells in the 'Humidity' and 'Windy' columns.

ID	Outlook	Temp	Humidity	Windy	Play Golf
1	Rainy	85	92	False	No
2	Rainy	80	88	True	No
3	Overcast	83	86	False	Yes
4	Sunny	70	80	False	Yes
5	Sunny	68	?	False	Yes
6	Sunny	65	58	True	No
7	Overcast	64	62	True	Yes
8	Rainy	72	95	?	No
9	Rainy	?	70	False	Yes
10	Sunny	75	72	False	Yes
11	Rainy	75	74	True	Yes
12	?	72	78	True	Yes
13	Overcast	81	66	False	Yes
14	Sunny	71	79	True	No

Dataset

There are some alternatives terms for columns, rows and values.

- Columns, Fields, Attributes, Variables
- Rows, Records, Objects, Cases, Instances, Examples, Vectors
- Values, Data

Feature Engineering

A **feature** is an **attribute** or **property** shared by all of the independent units on which analysis or prediction is to be done. Any attribute could be a feature, as long as it is useful to the model.

Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work.

- **feature engineering**: This process attempts to create additional relevant features from the existing raw features in the data, and to increase the predictive power of the learning algorithm.
- **feature selection**: This process selects the key subset of original data features in an attempt to reduce the dimensionality of the training problem.

Normally **feature engineering** is applied first to generate additional features, and then the **feature selection** step is performed to eliminate irrelevant, redundant, or highly correlated features.

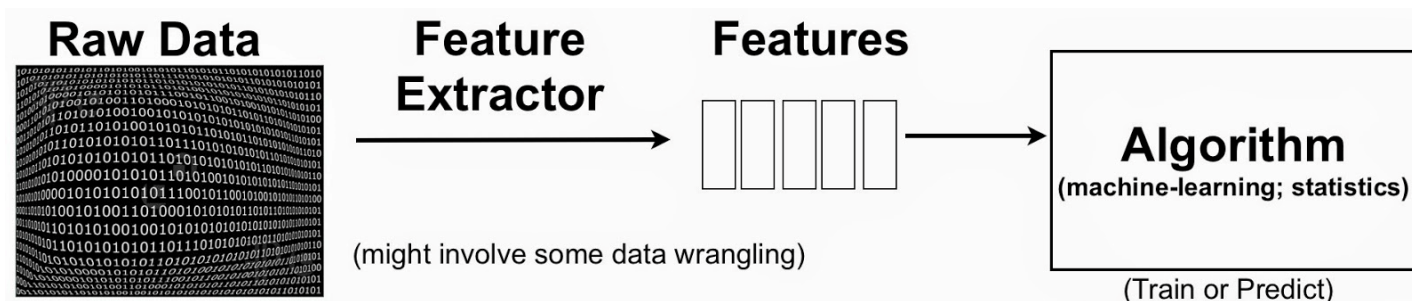
Feature engineering is fundamental to the application of machine learning, and is both difficult and expensive.

source <https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/create-features>

Feature Engineering

The **training data** used in machine learning can often be enhanced by **extraction of features** from the raw data collected. An example of an engineered feature in the context of learning how to classify the images of handwritten characters is creation of a bit density map constructed from the raw bit distribution data. This map can help locate the edges of the characters more efficiently than simply using the raw distribution directly.

What kind of features should be created to enhance the dataset when training a model? Engineered features that enhance the training provide information that better differentiates the patterns in the data. The new features are expected to provide additional information that is not clearly captured or easily apparent in the original or existing feature set. But **this process is something of an art**. Sound and productive decisions often require some domain expertise.



Feature Engineering

- Engineered and selected features increase the efficiency of the training process, which attempts to extract the key information contained in the data.
- They also improve the power of these models to classify the input data accurately and to predict outcomes of interest more robustly.
- Feature engineering and selection can also combine to make the learning more computationally tractable. It does so by enhancing and then reducing the number of features needed to calibrate or train a model. Mathematically speaking, the features selected to train the model are a minimal set of independent variables that explain the patterns in the data and then predict outcomes successfully.

Feature Selection vs Feature Extraction

In **feature selection** we try to find the best subset of the input feature set

In **feature extraction** we create new features based on the transformation or combination of the original feature set

Summary

We have learnt that:

- Azure Machine Learning Studio is a cloud based tool for using Machine Learning (ML) algorithm without any coding required
- Data may need to be transformed/encoded before it can be fed to a machine algorithm
- Categorical variables can be encoded as dummy variables
- Sometimes it is useful to discretize numeric values
- Feature engineering is the part that needs real intelligence