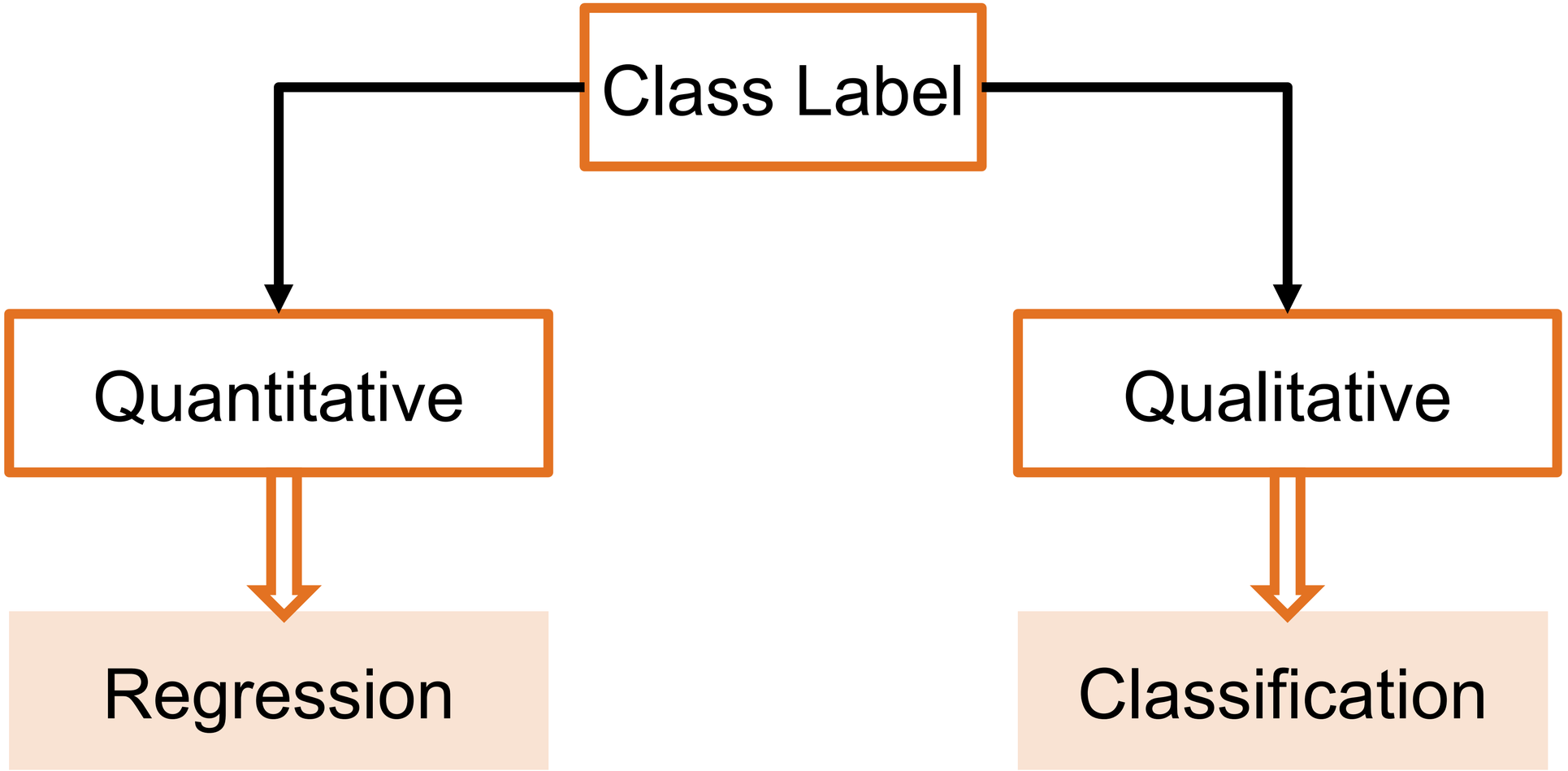
# Supervised Learning

Mirco Schönfeld
mirco.schoenfeld@uni-bayreuth.de

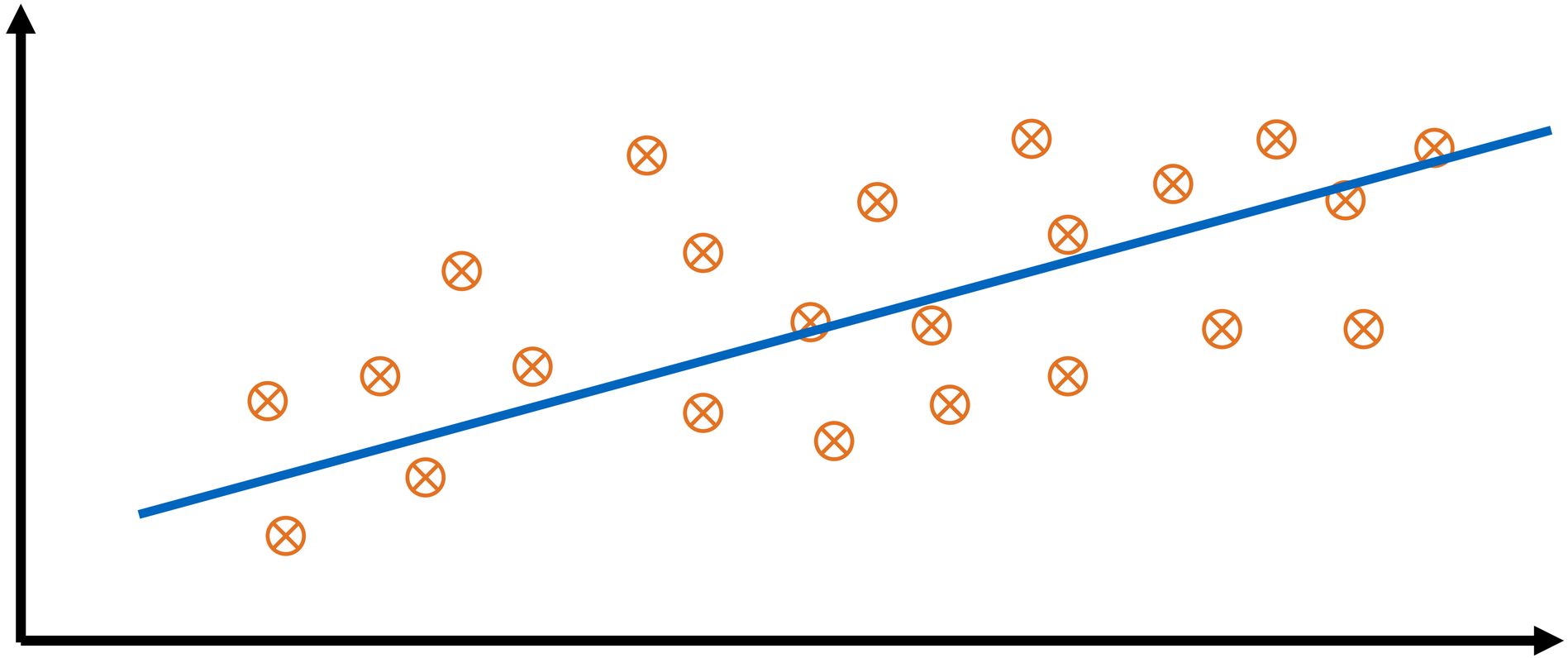
Supervised Machine Learning


https://exeter-data-analytics.github.io/MachineLearning/supervised-learning-1.html

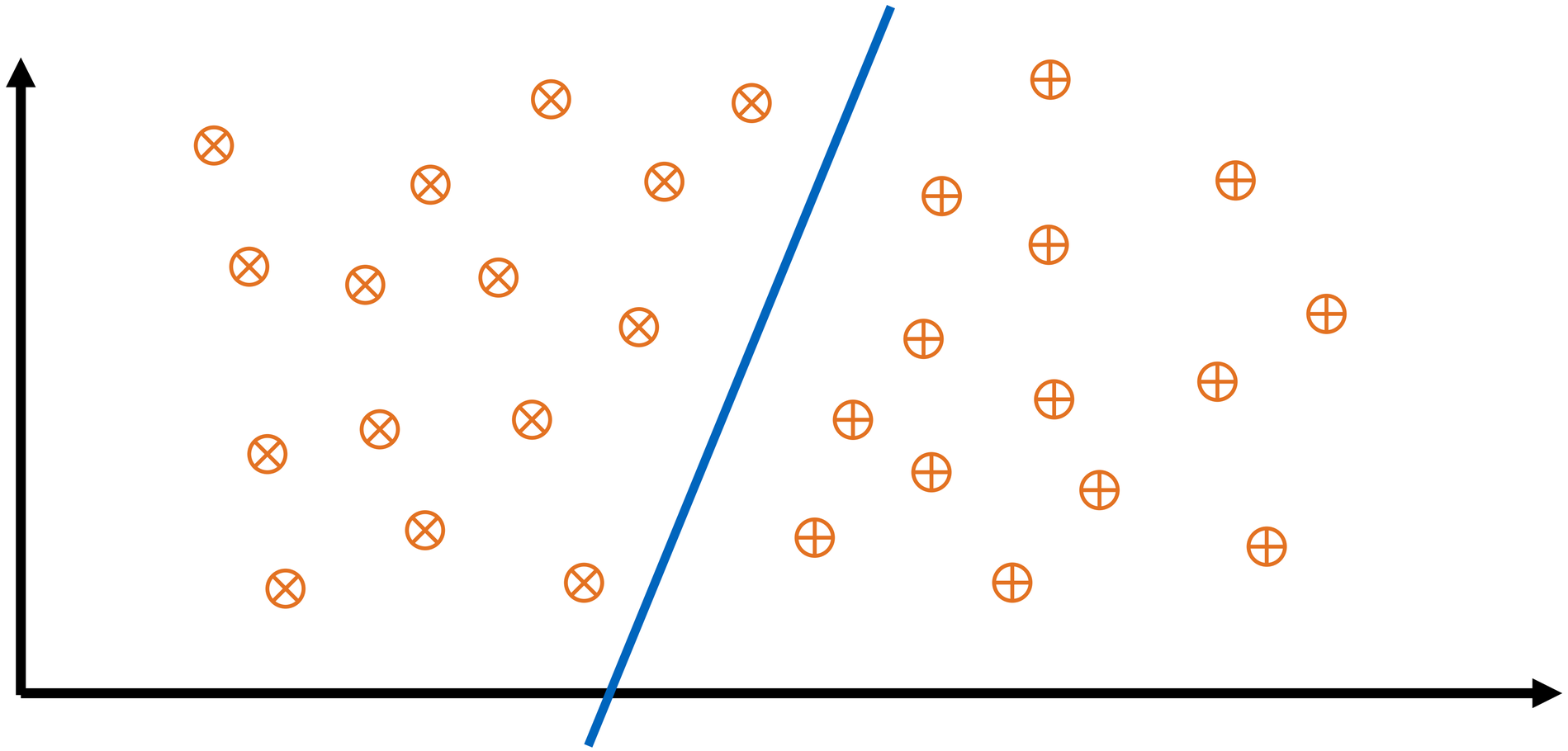
# Nature of the Class Label



# Regression



# Classification



## Supervised Learning

| $X_1$ | $X_2$ | $X_3$ | $X_p$ | Y |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Target

## Un-Supervised Learning

| $X_1$ | $X_2$ | $X_3$ | $X_p$ | Y |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

No Target

https://www.sharpsightlabs.com/blog/supervised-vs-unsupervised-learning/

# Supervised Machine Learning…in (other) words

Supervised Machine Learning aims at forecasting class labels for measured data

*Correct* class labels are known for training data

Training means to search for a good function mapping
measured artifacts to known class labels

What you need:

- Classificator
- Measured data
- Class labels

# Supervised Machine Learning… do you speak math?


Objective:

learn the *class label* $y$ for each value of $x$, the *feature vector* consisting of multiple *features* (categorical or numerical)

Result:

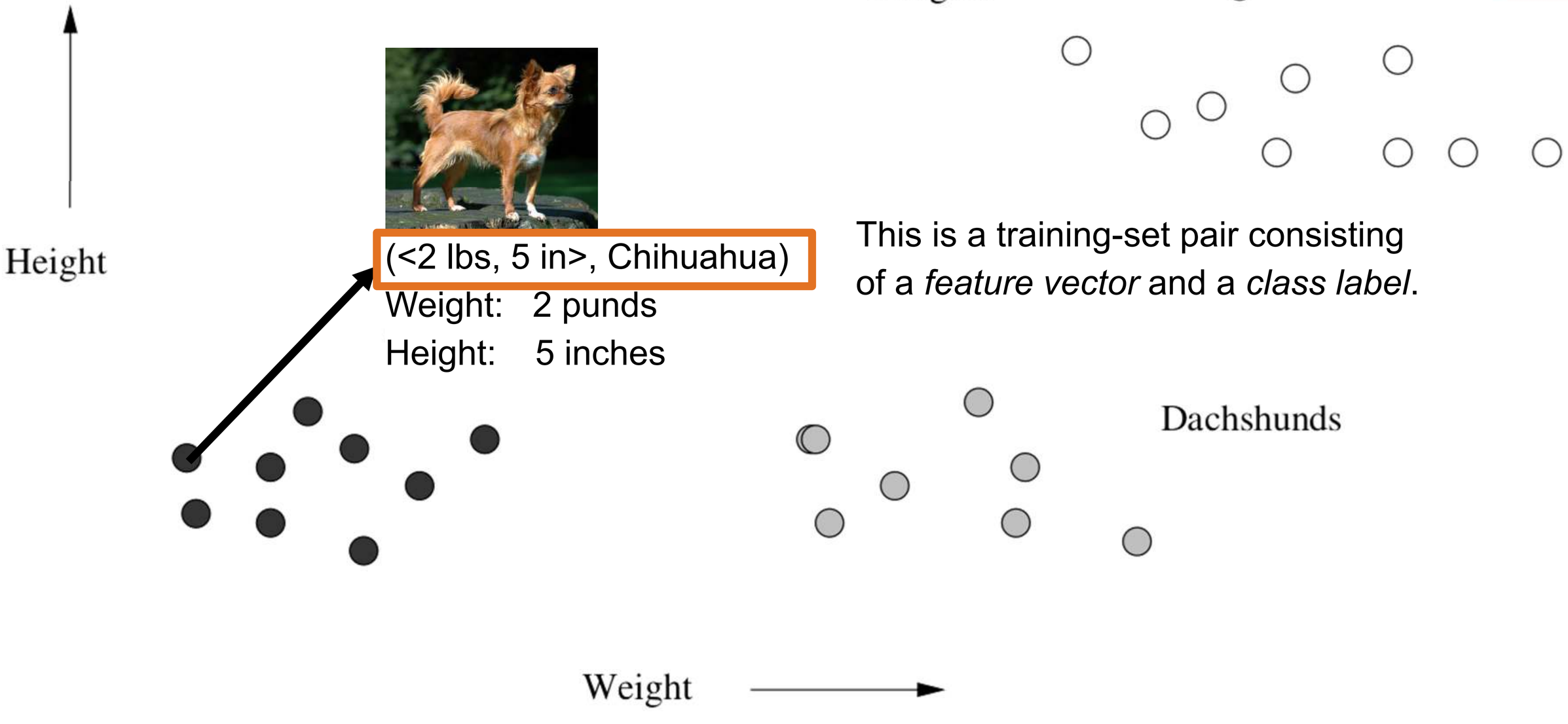
A function $f(x) = y$ that best predicts $y$ for each value of $x$

If $y$ is

- a real number, a regression model is learned
- a Boolean value (true/false, +1/-1), we speak of *binary classification*
- a nominal value of some finite set, it is a *multiclass classification*

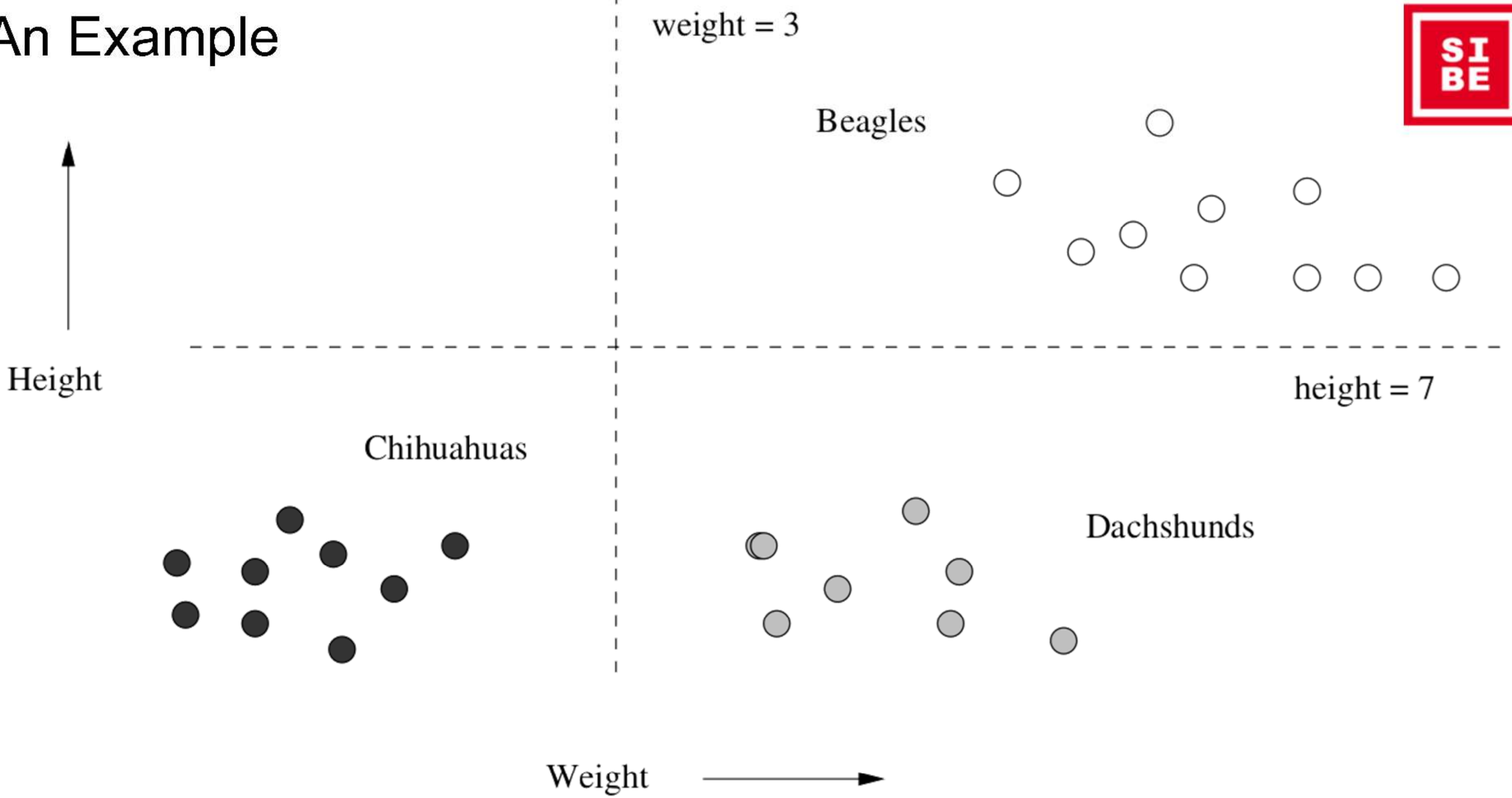
# An Example

# An Example

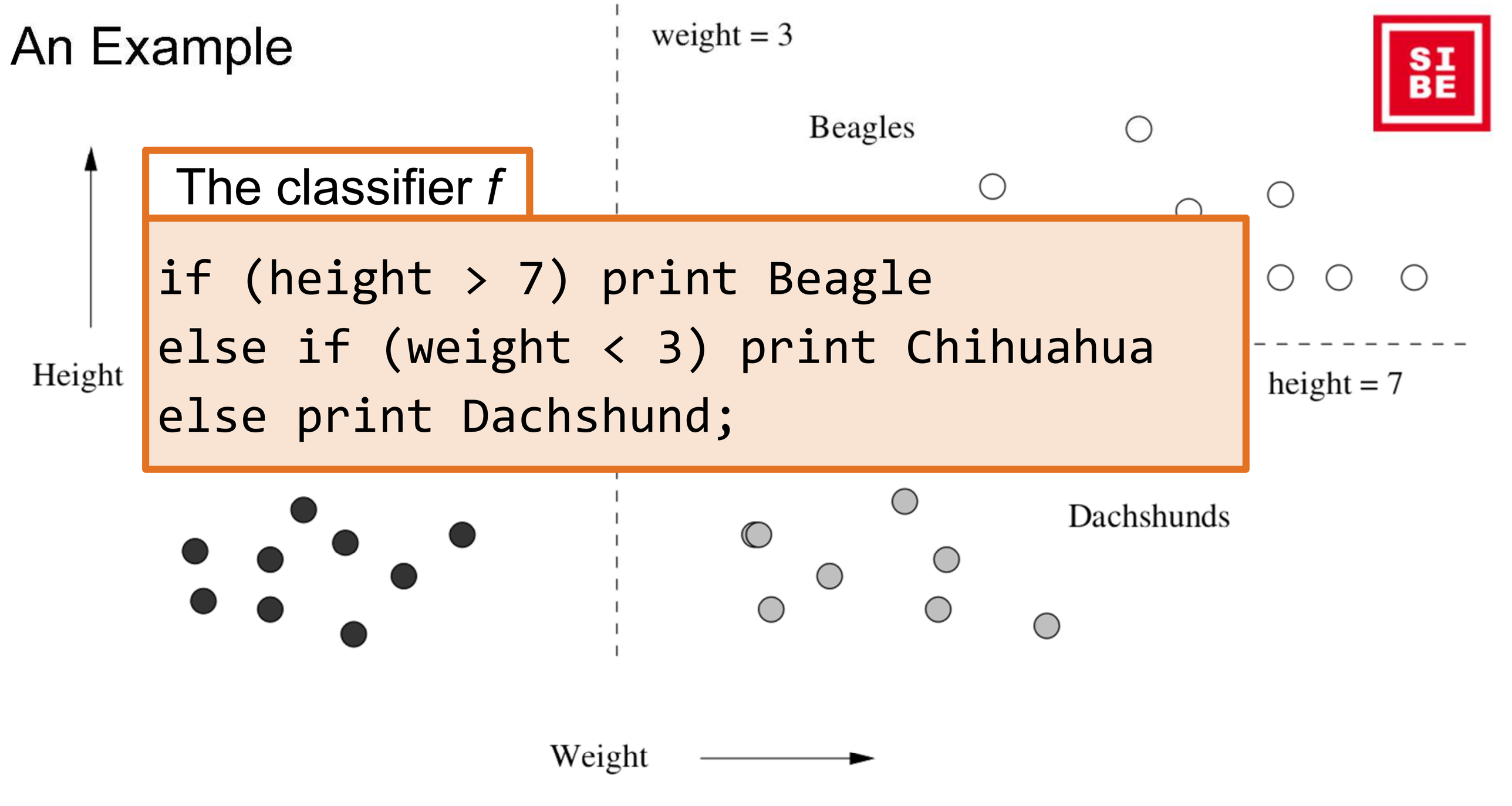



This is a training-set pair consisting of a *feature vector* and a *class label*.

# An Example


weight = 3
Beagles
Height
height = 7
Chihuahuas
Dachshunds
Weight

# An Example




## The classifier *f*

```
if (height > 7) print Beagle
else if (weight < 3) print Chihuahua
else print Dachshund;
```

# Relation to Clustering

Classification:

- Class label is discrete
- Enough training data for *each class*

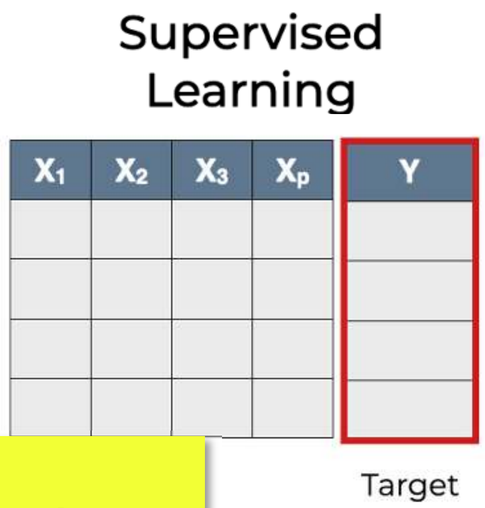
Regression:

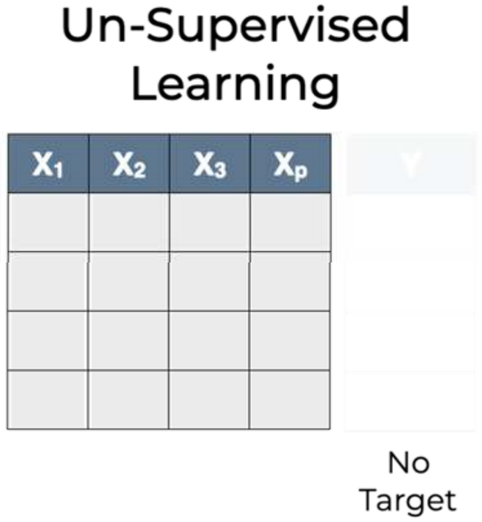
- Target variable is numeric

Clustering:

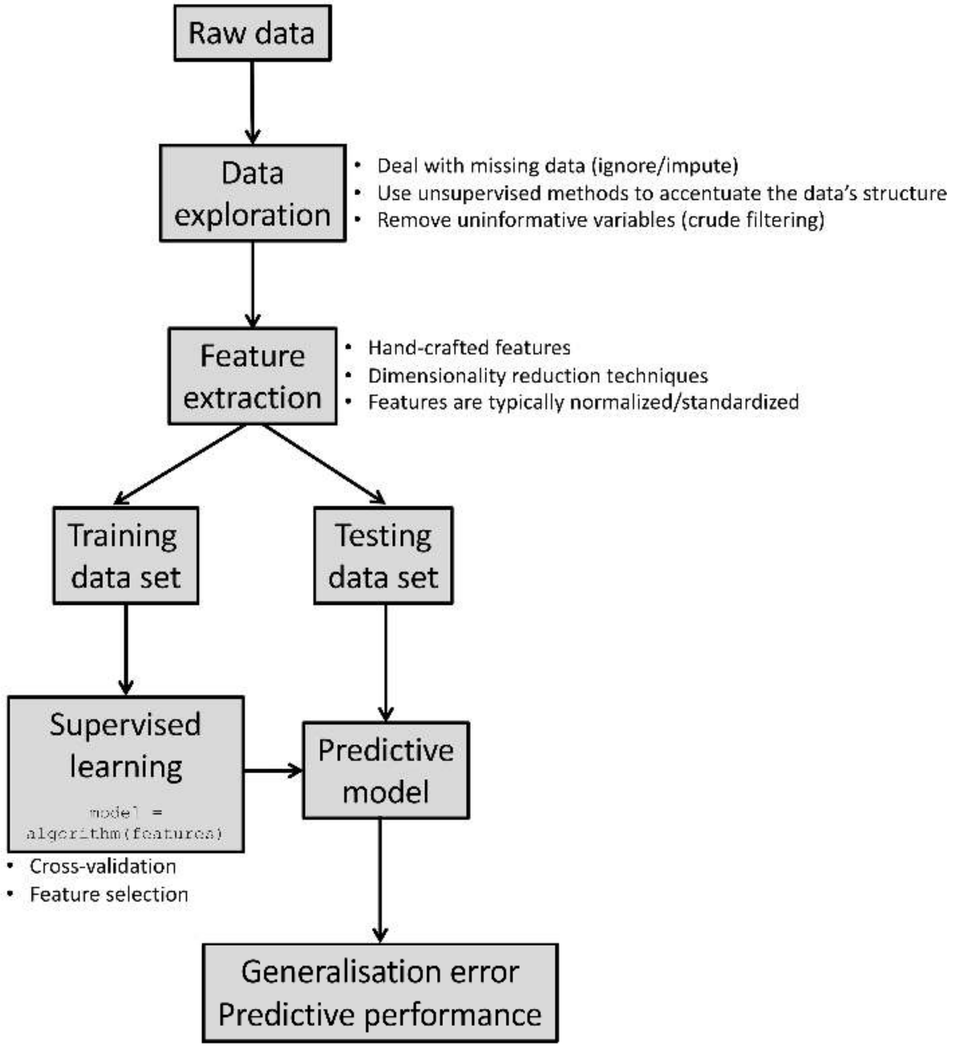
- no class label / target variable
- no training – input data without pre-defined classes
- produces partitioning of data

Clusters could be used to identify new classes



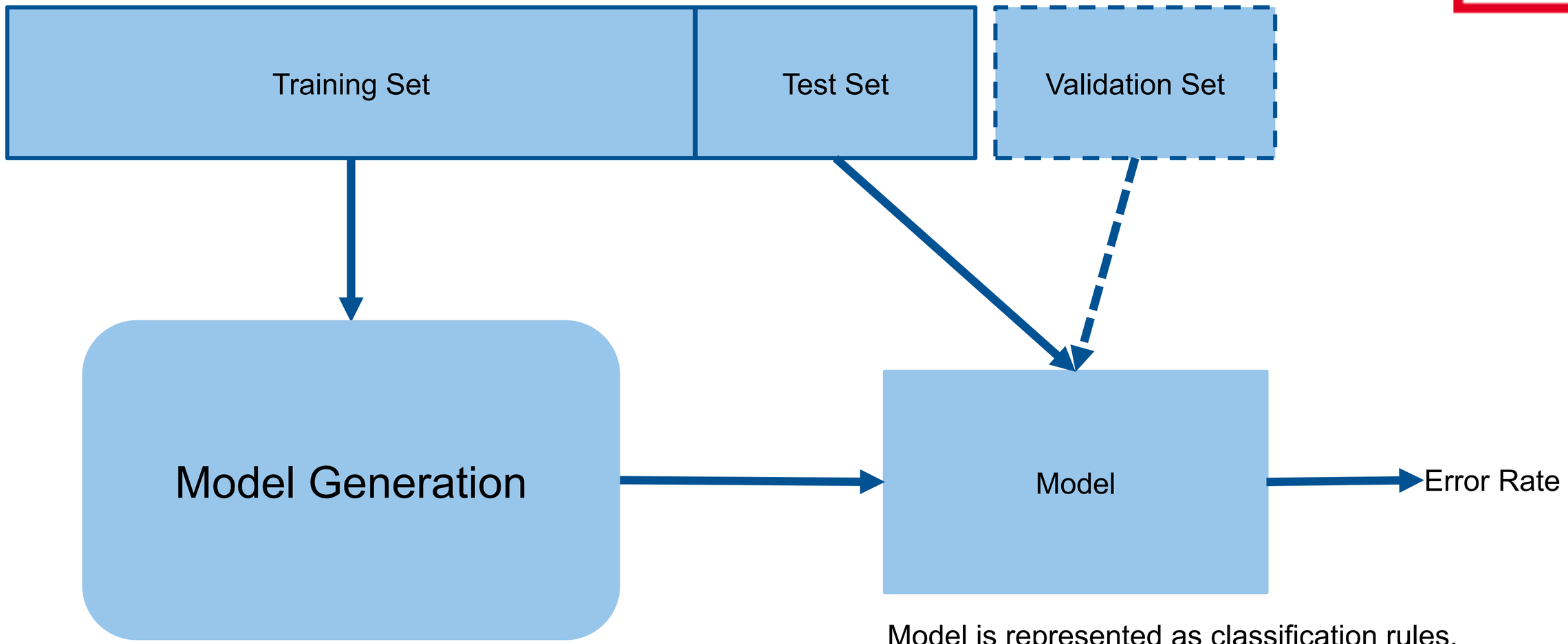



# Supervised Machine Learning: Bird View




https://exeter-data-analytics.github.io/MachineLearning/supervised-learning-1.html

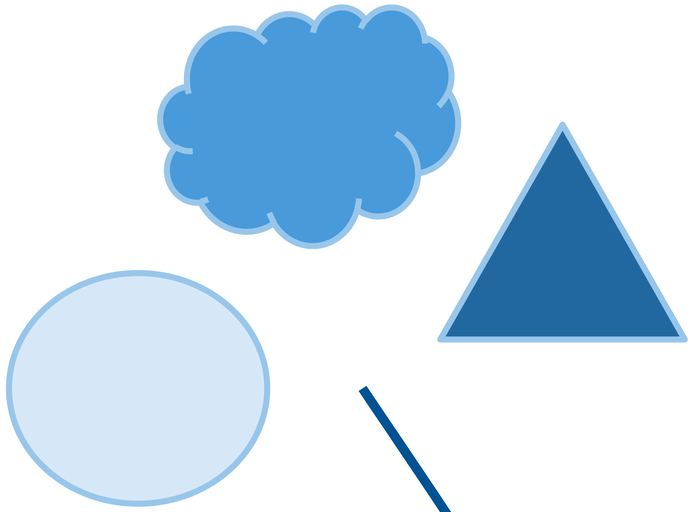
# Supervised Learning: Training and Testing

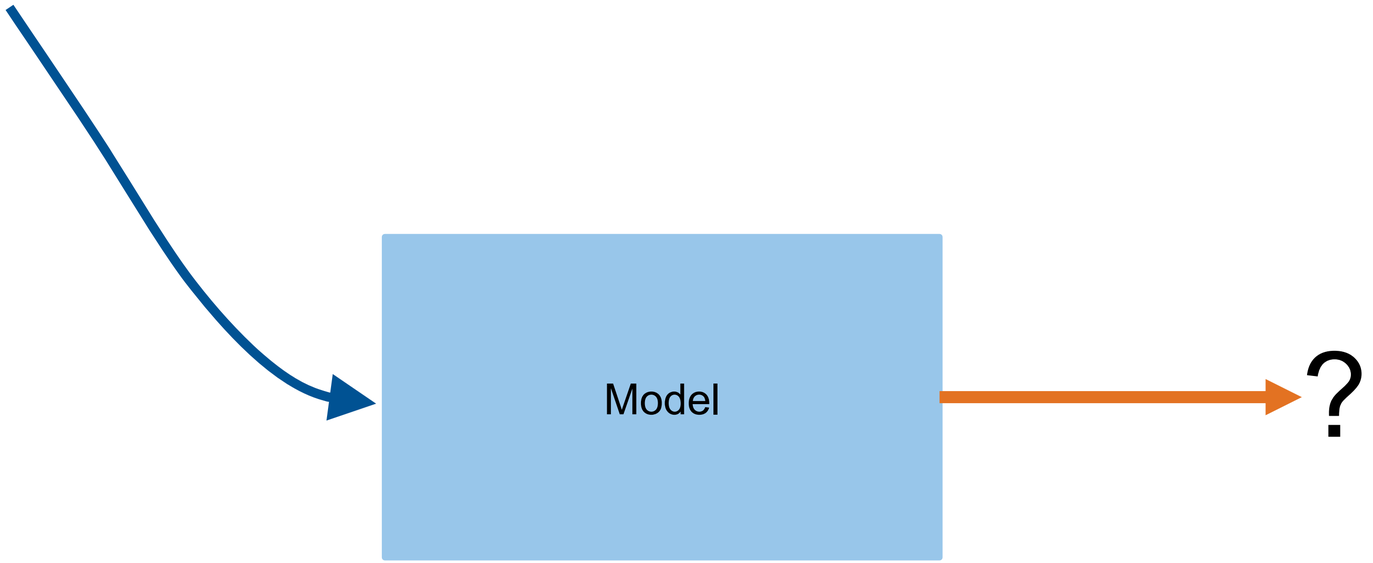



Model is represented as classification rules, decision trees, or mathemical formulas

Leskovec, J., Rajaraman, A. and Ullman, J.D., 2020. *Mining of massive data sets*. Cambridge university press.

# What the Model is for

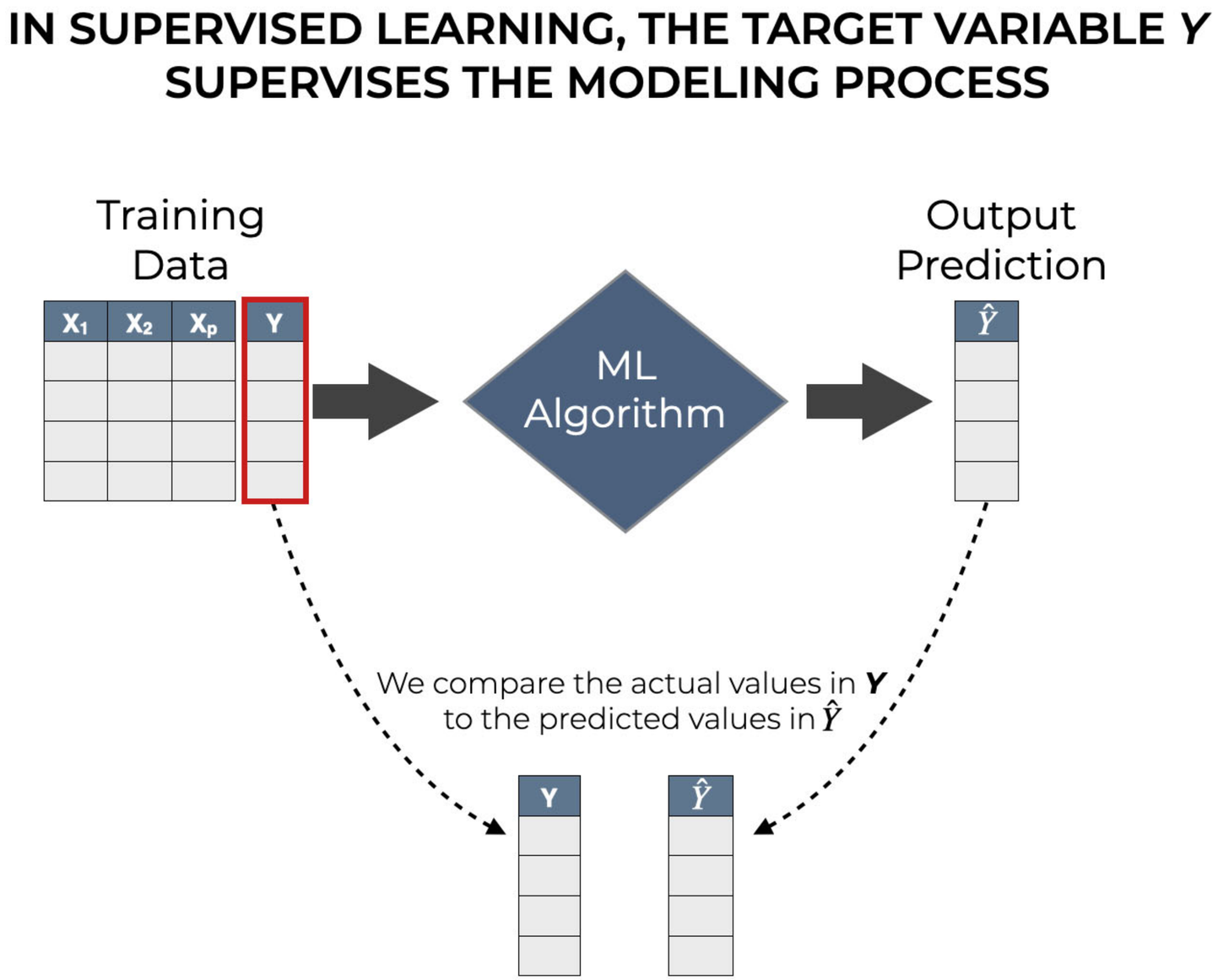





The model is used to classify the data whose class is *unknown*!

Its purpose is *generalization*!

# Models make mistakes…

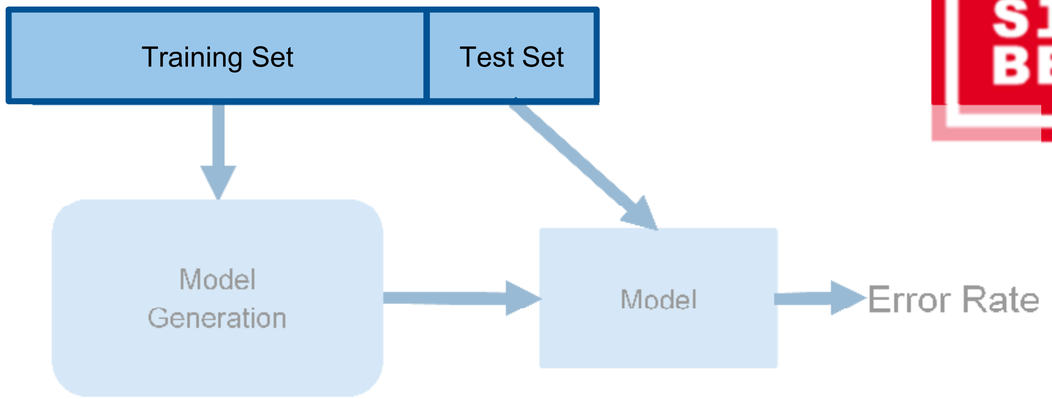


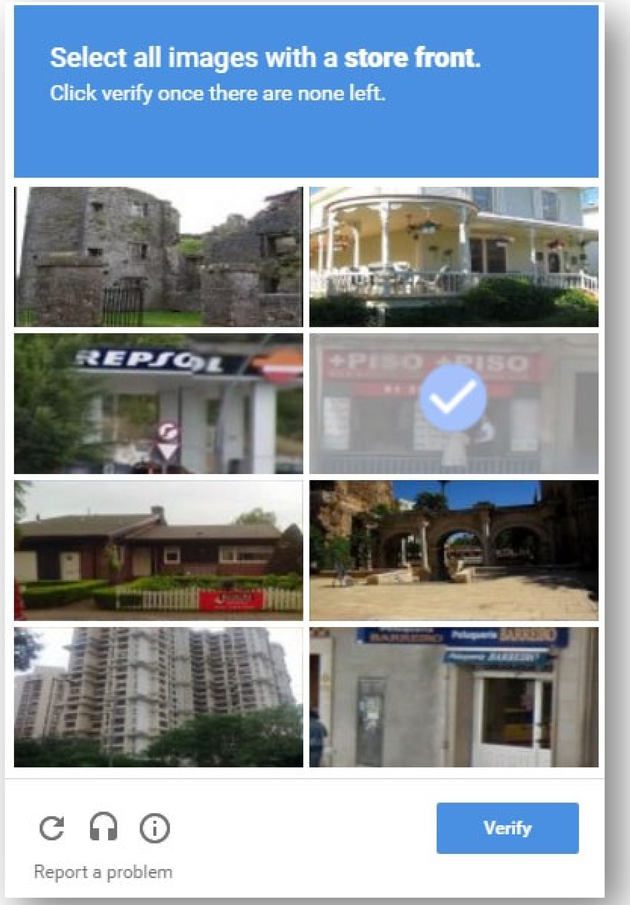

https://www.sharpsightlabs.com/blog/supervised-vs-unsupervised-learning/

# /kæp.tʃə/ or Creating a Training Set

## Label Data Manually?

Often time consuming and costly process for large training sets




*Active learning*:

kick start a classifier only with some training examples, but leave it primarily with unclassified data, which it must classify. If the classifier is unsure of the classification (e.g., the newly arrived example is very close to the boundary), then the classifier can ask for ground truth at some significant cost

# Major Classes of Classifiers


- Decision trees

  Suitable for binary and multiclass classification with a limited number of features
- Perceptrons

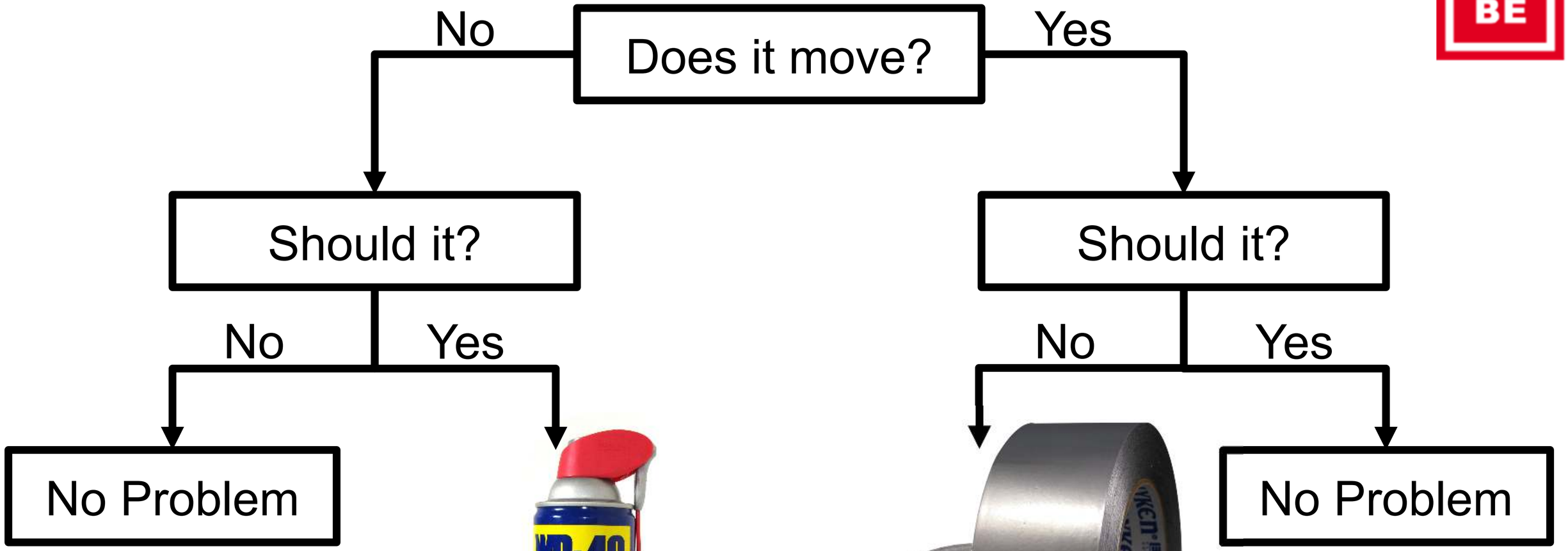
  Applies weights to components of vectors. Output +1 if sum exceeds a threshold, otherwise -1.
- Neural networks

  Acyclic networks of perceptrons
- Instance-based learning

  Compares instances of data to the entire training set. Example: k-nearest neighbor.
- Support-vector machines

  Maps training examples to points in space so as to maximise the gap between categories

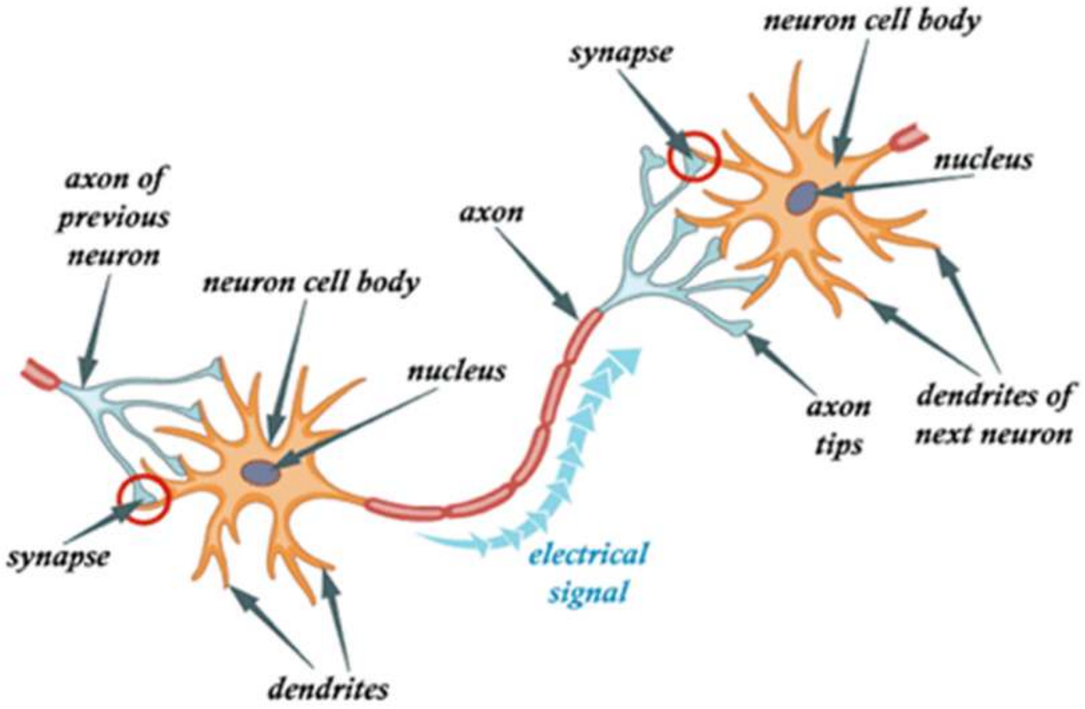
# Decision Trees

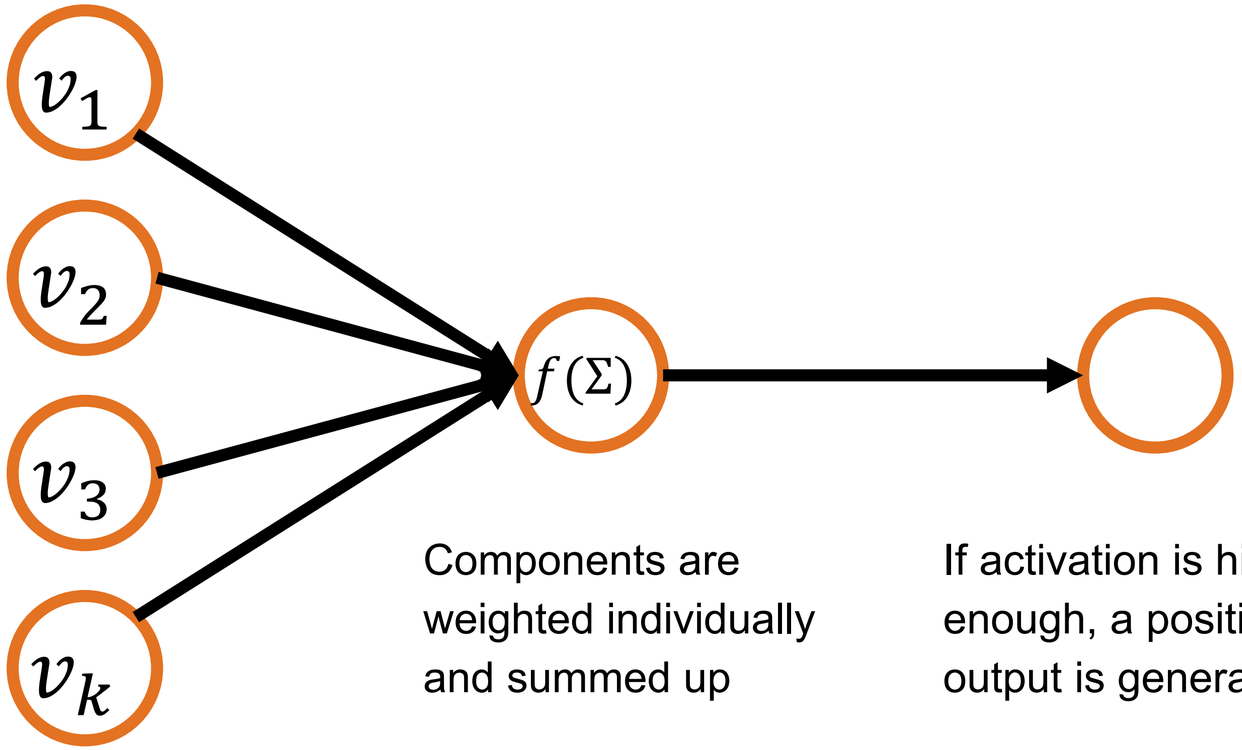





https://commons.wikimedia.org/wiki/File:WD-40_Smart_Straw.JPG
https://commons.wikimedia.org/wiki/File:Duct-tape.jpg
https://commons.wikimedia.org/wiki/File:Polyken_Duct_Tape.jpg

# Perceptrons



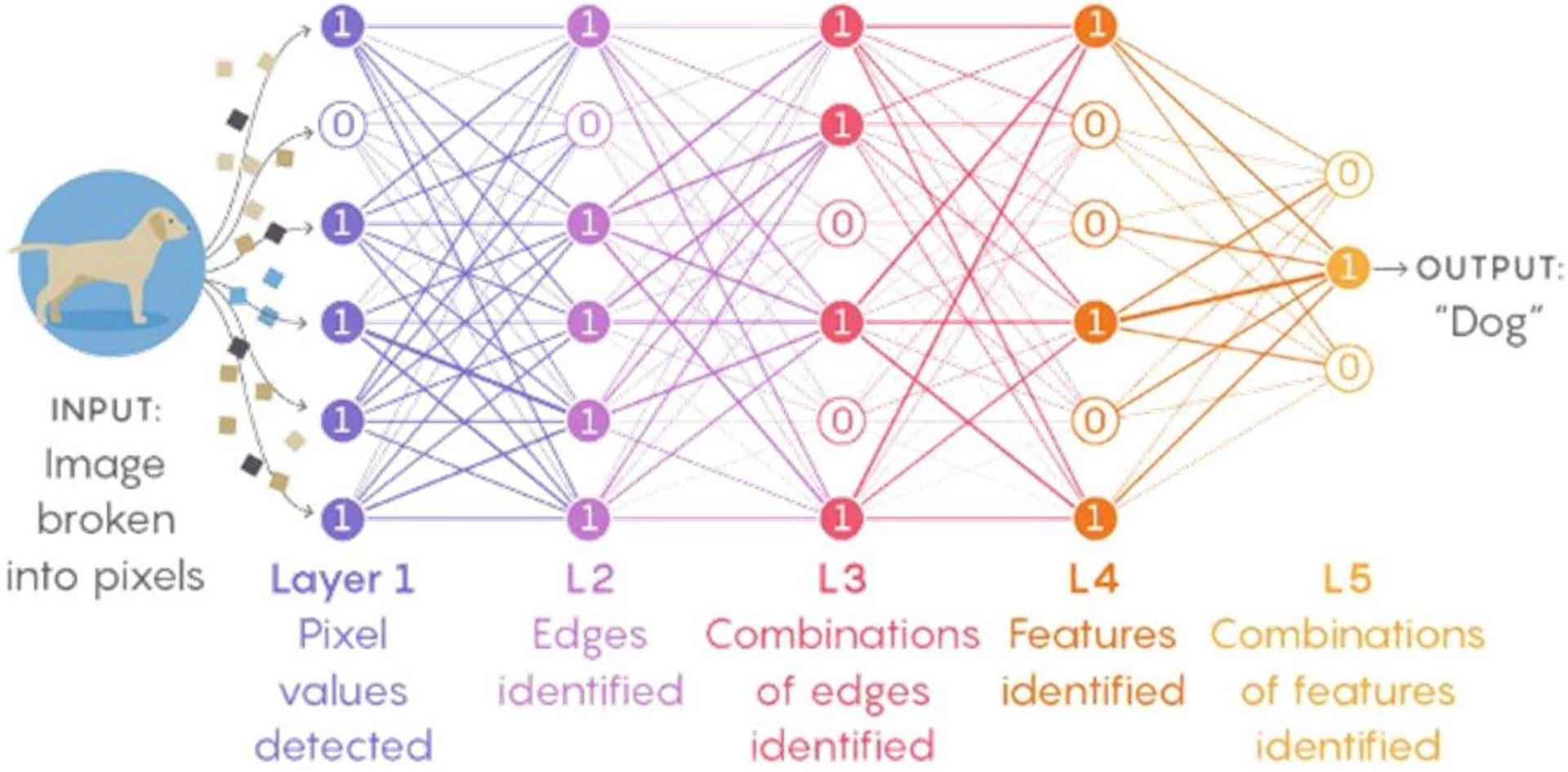

Components are weighted individually and summed up

If activation is high enough, a positive output is generated

Components of instances

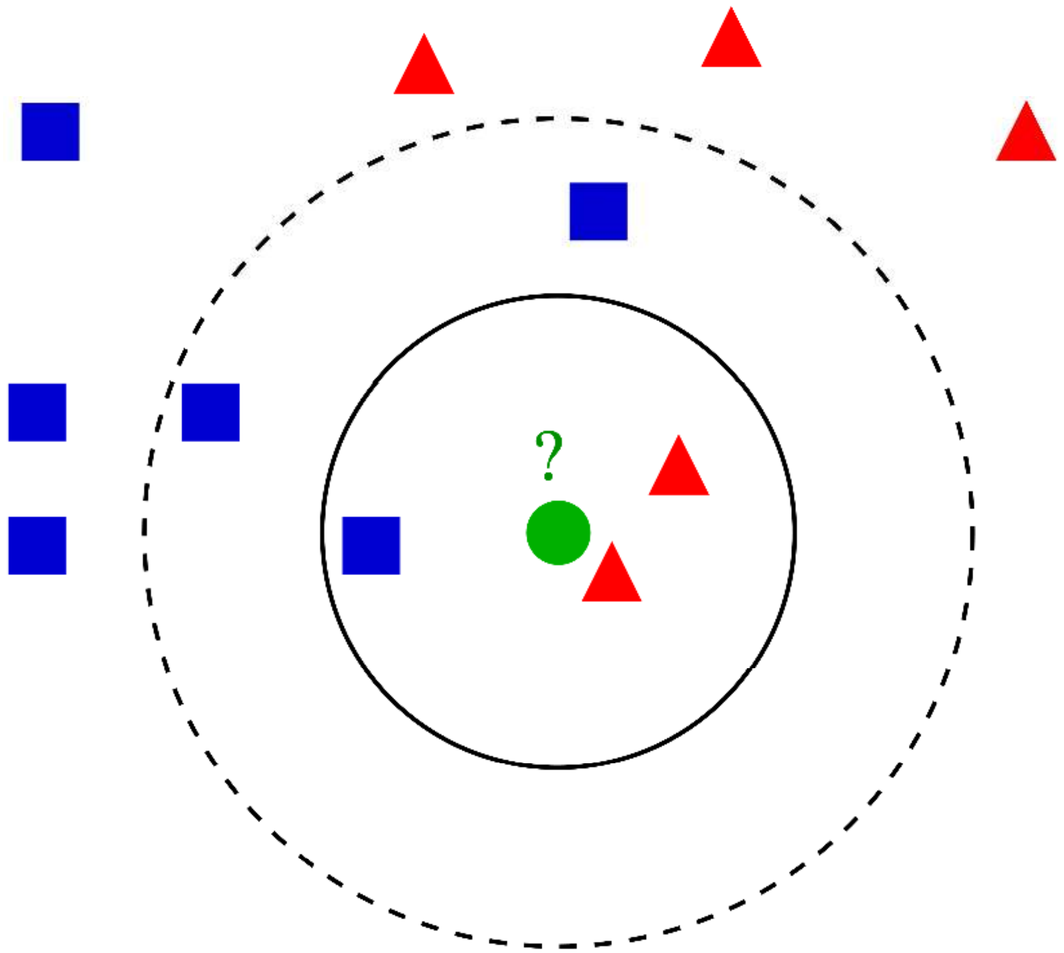
https://www.deeplearningtrack.com/single-post/2017/07/09/Introduction-to-NEURAL-NETWORKS-Advantages-and-Applications

# Neural Networks



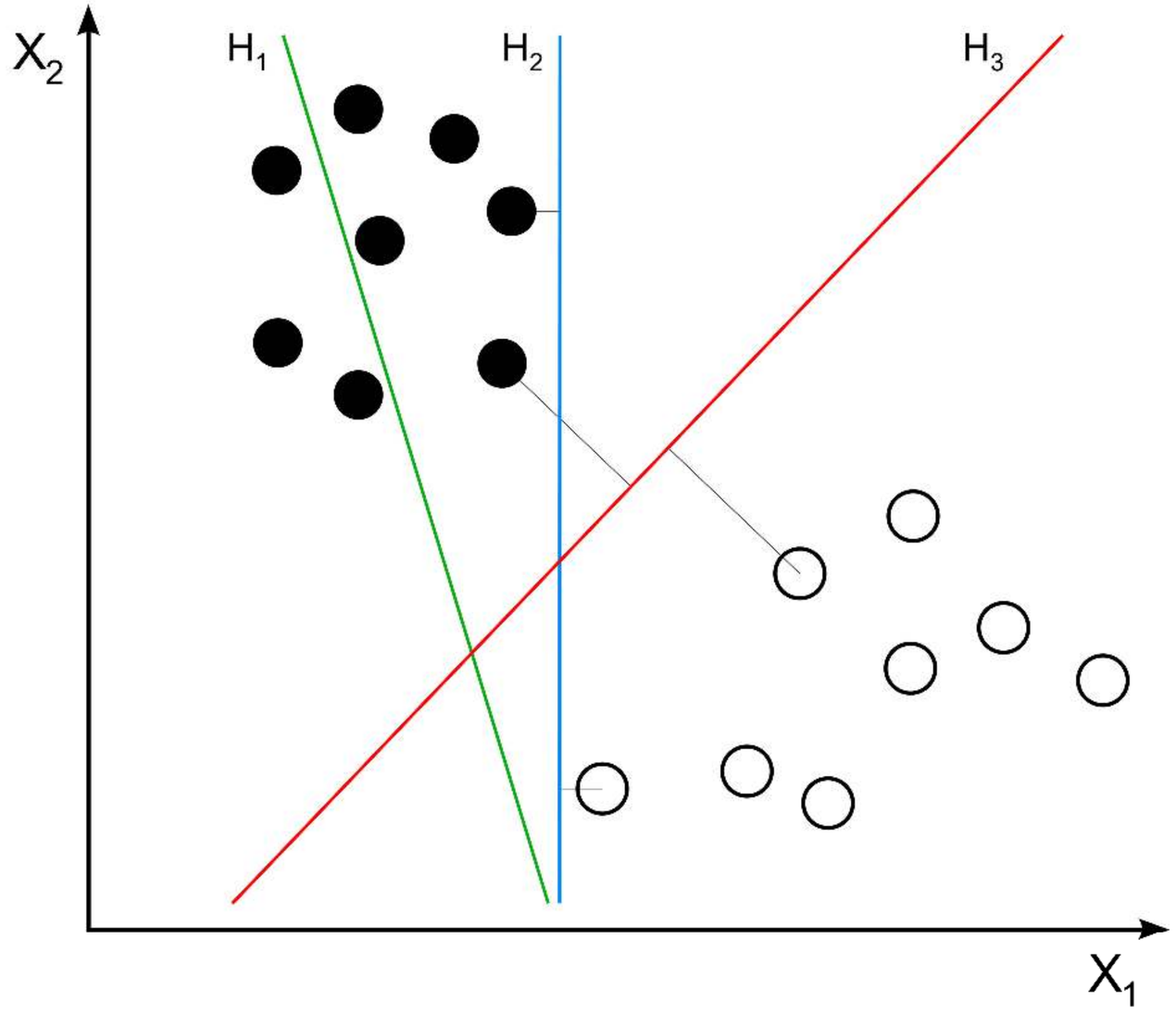

https://towardsdatascience.com/designing-your-neural-networks-a5e4617027ed

# *k*-Nearest Neighbors



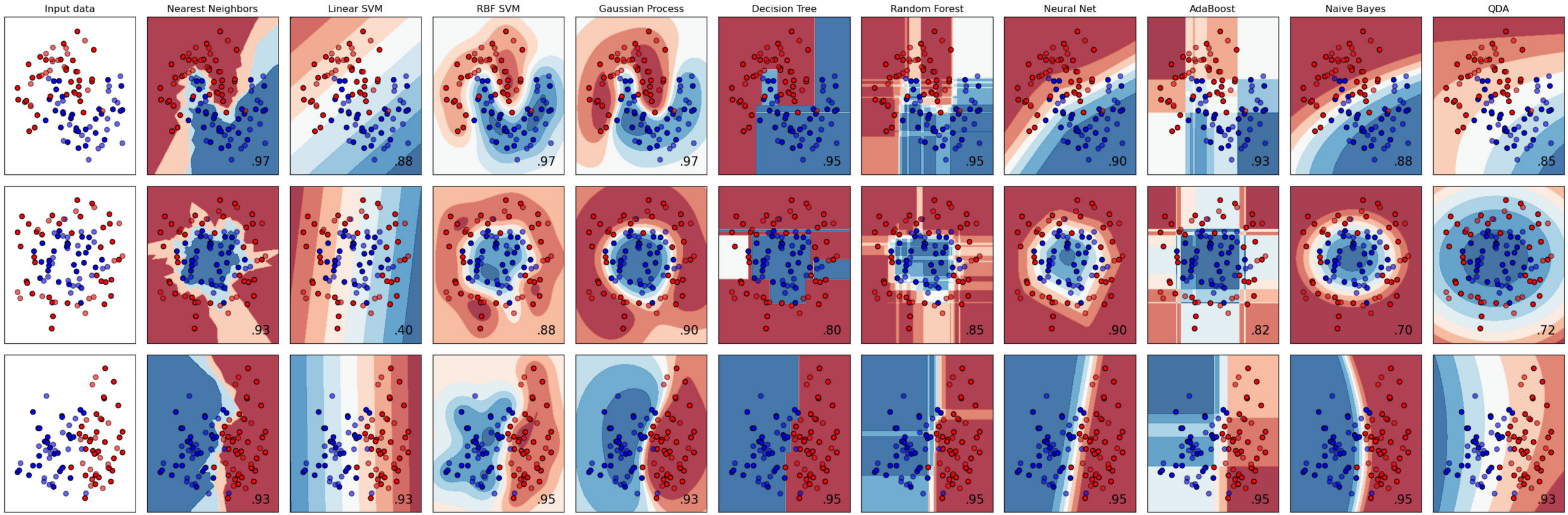

https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

# Support Vector Machines




https://en.wikipedia.org/wiki/Support-vector_machine

# Some Classification Algorithms Compared




https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

# Ensemble Models


Inspectable models ease debugging problems in
data collection, feature engineering, etc

Ensemble models provide ways to restrict features to separate models that otherwise would interact in unintended ways

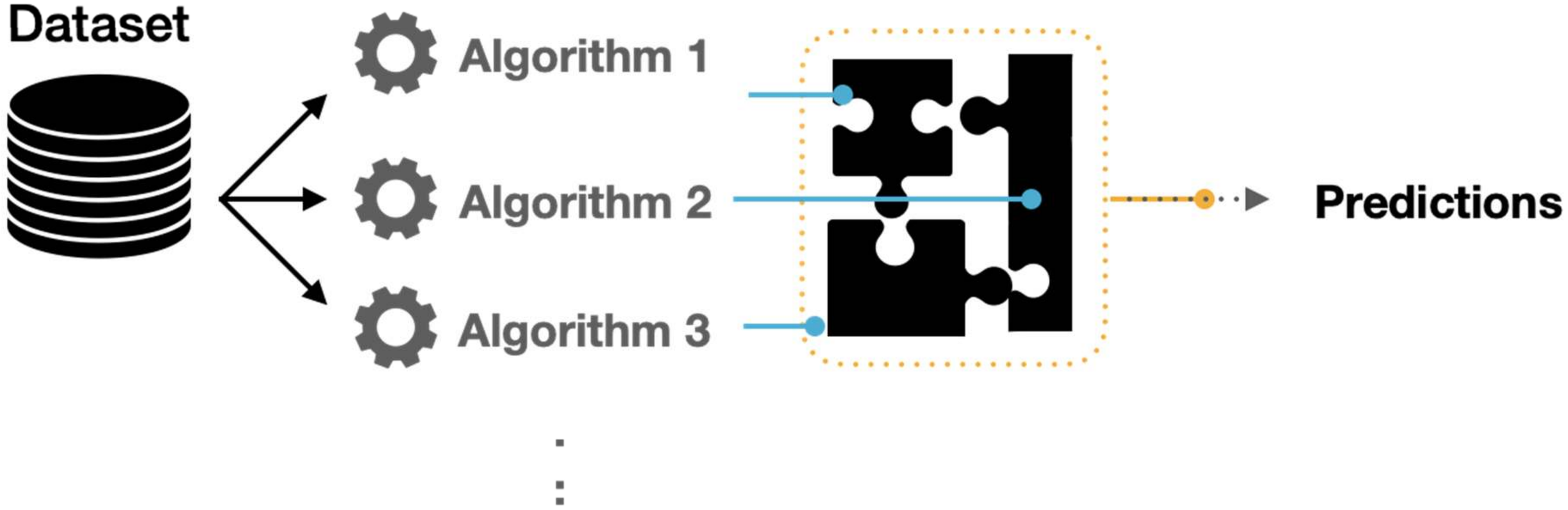
Helps overcoming

- high variance: single estimators are very sensitive to inputs to the learned features
- low accuracy: single models/algorithms might not be good enough
- features noise and bias: single estimators might rely heavily on one or few features

Various ways to combine models

# Ensemble Models

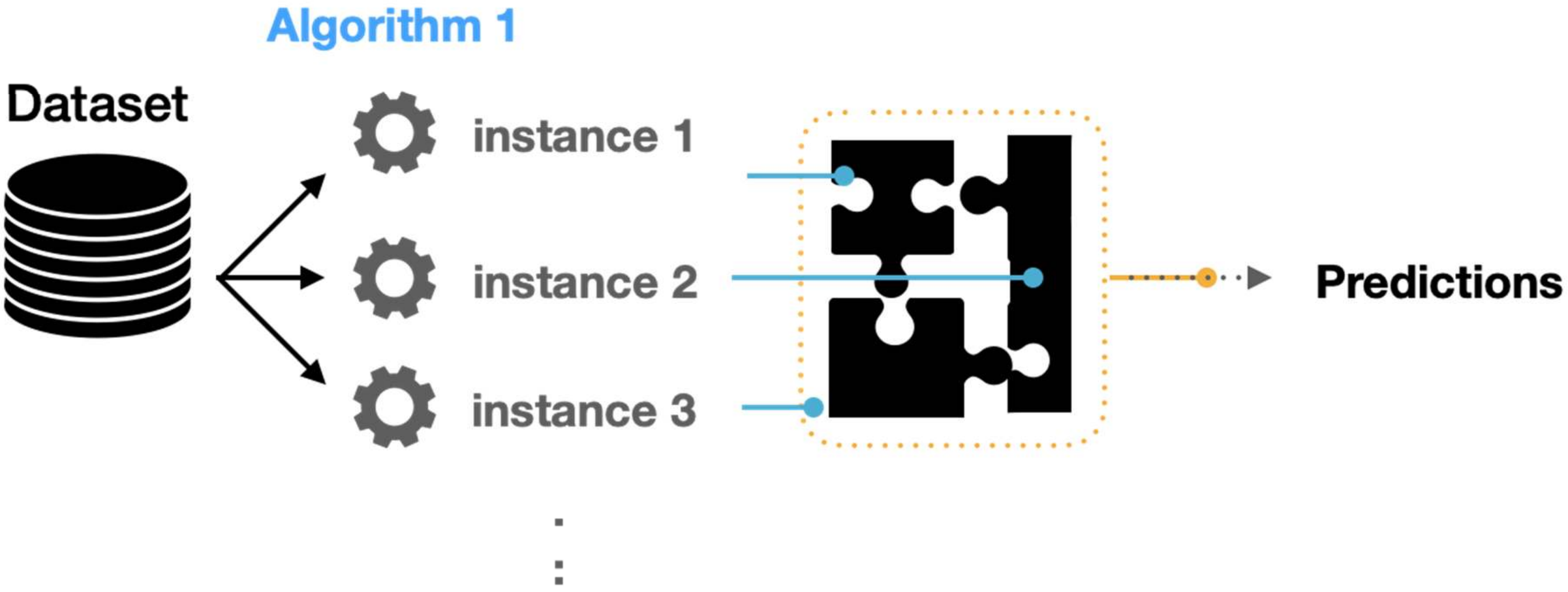
Using multiple algorithms to diversify model predictions




https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

# Ensemble Models

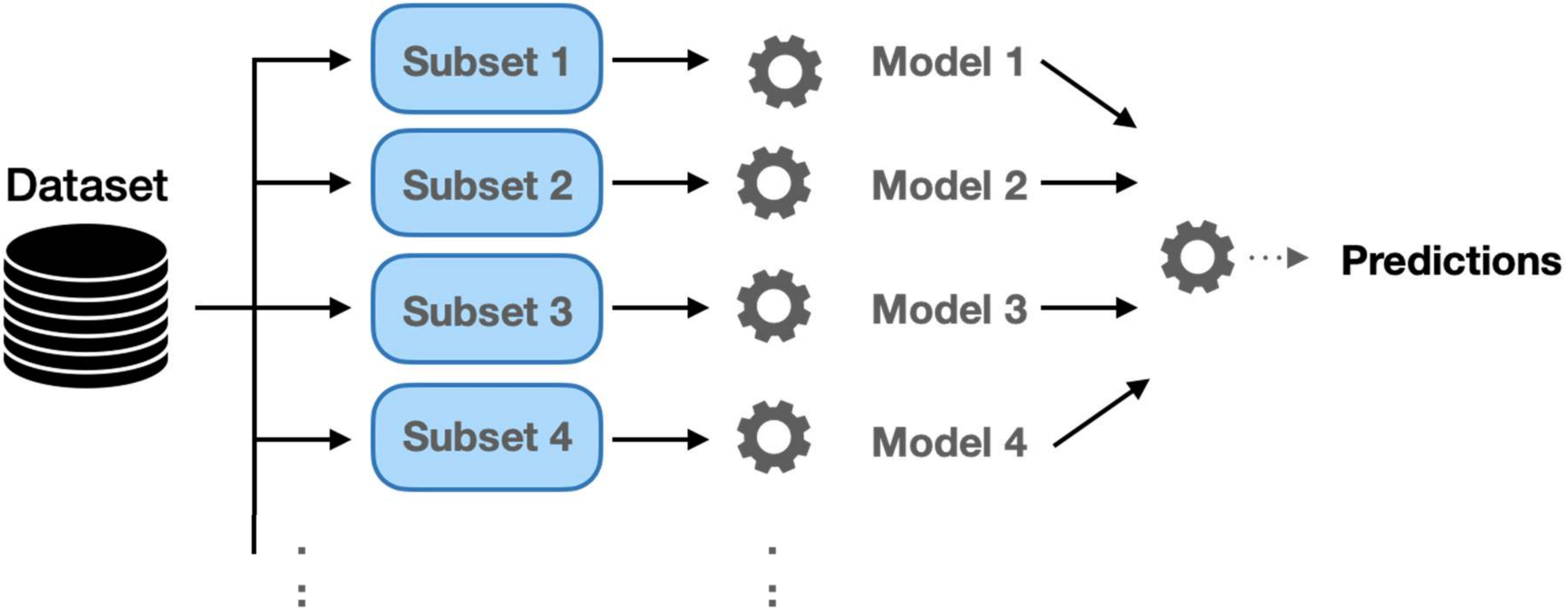
Using multiple weak instances of the same algorithm




https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

# Ensemble Models: Bagging & Bootstrapping

Combine predictions from multiple models

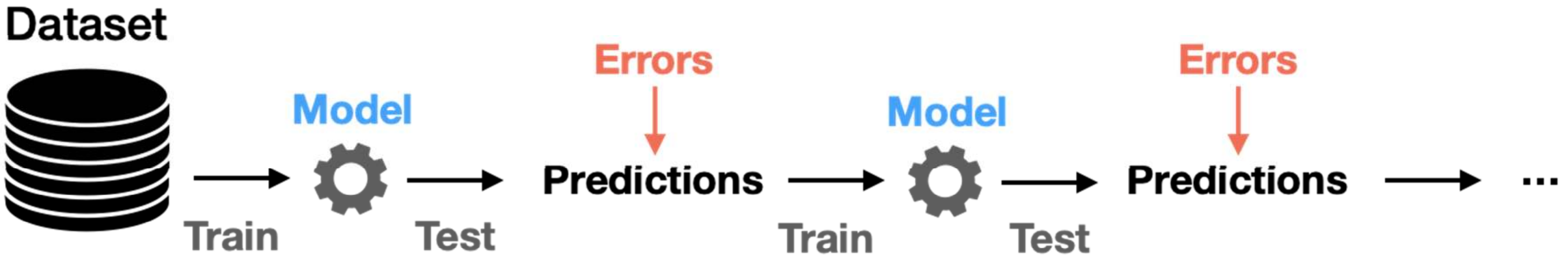



https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

# Ensemble Models: Boosting

Ensemble of algorithms that builds models on top of several weak learners

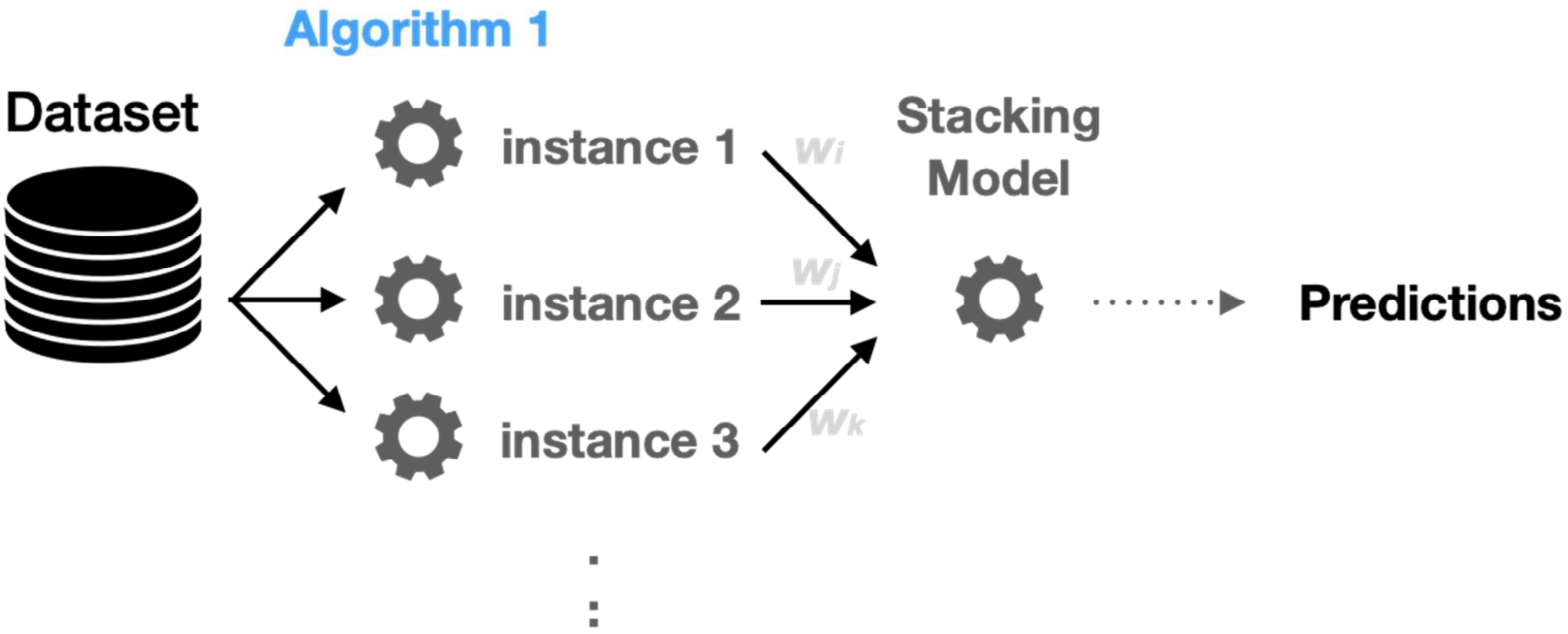
Here: Sequental Adaptive Boosting (AdaBoost)




https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

# Ensemble Models: Stacking

Stacking intermediate predictions to make a final prediction



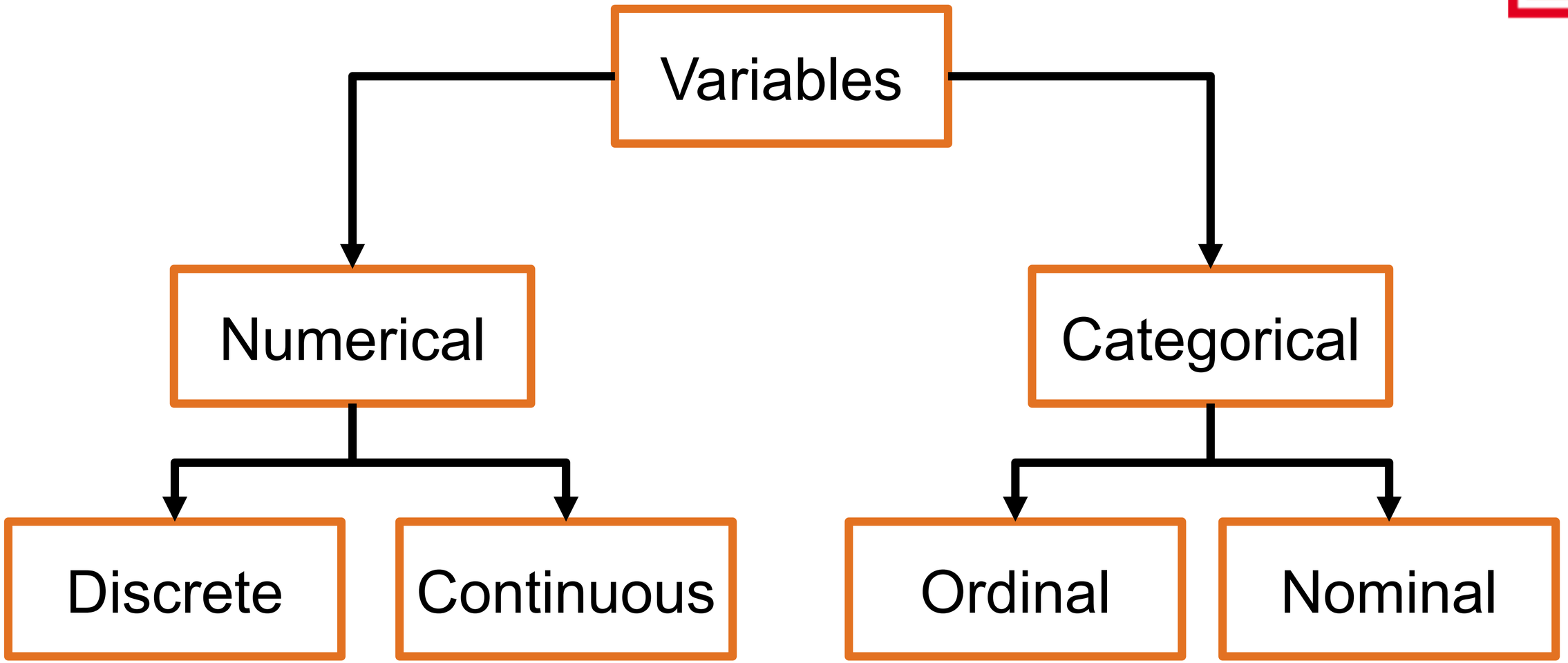

https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

# How to select a model?

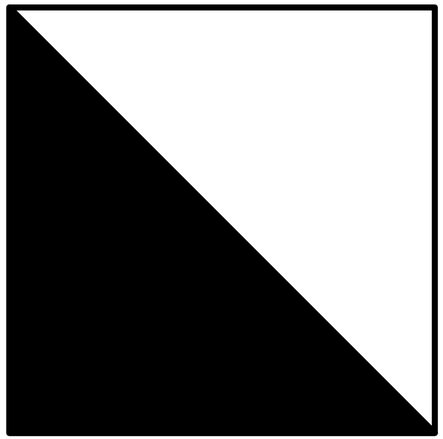

- Quality of predictions
  i.e. performance in terms of a quality metric
- Speed
  i.e. training time, prediction time
- Robustness
  i.e. handling noise or missing values and still classify correctly
- Scalability
  i.e. computational efficiency
- Interpretability
  subjective means
- Other

# Different Types of Data

# Hello Data


Variables
Numerical
Categorical
Discrete
Continuous
Ordinal
Nominal

# Discrete vs. Continuous



## Dichotomous data:

Data points can only take up 2 values


## Discrete data:

Data points can only take up values from a set of possible values.

Either finite or infinite but countable


## Continuous data:

Data points can be measured to an arbitrary level of exactness

# Scales

## Nominal

Values divide space of possible values into different segments

No order between segments

Example: gender, nationality

## Ordinal

Allows for rank order – data can be sorted

No relative degree or relative difference between groups

Example: school grades

## Interval

Order and difference between groups is defined

No natural „zero“ & ratio between points is undefined

Example: temperature in Celsius

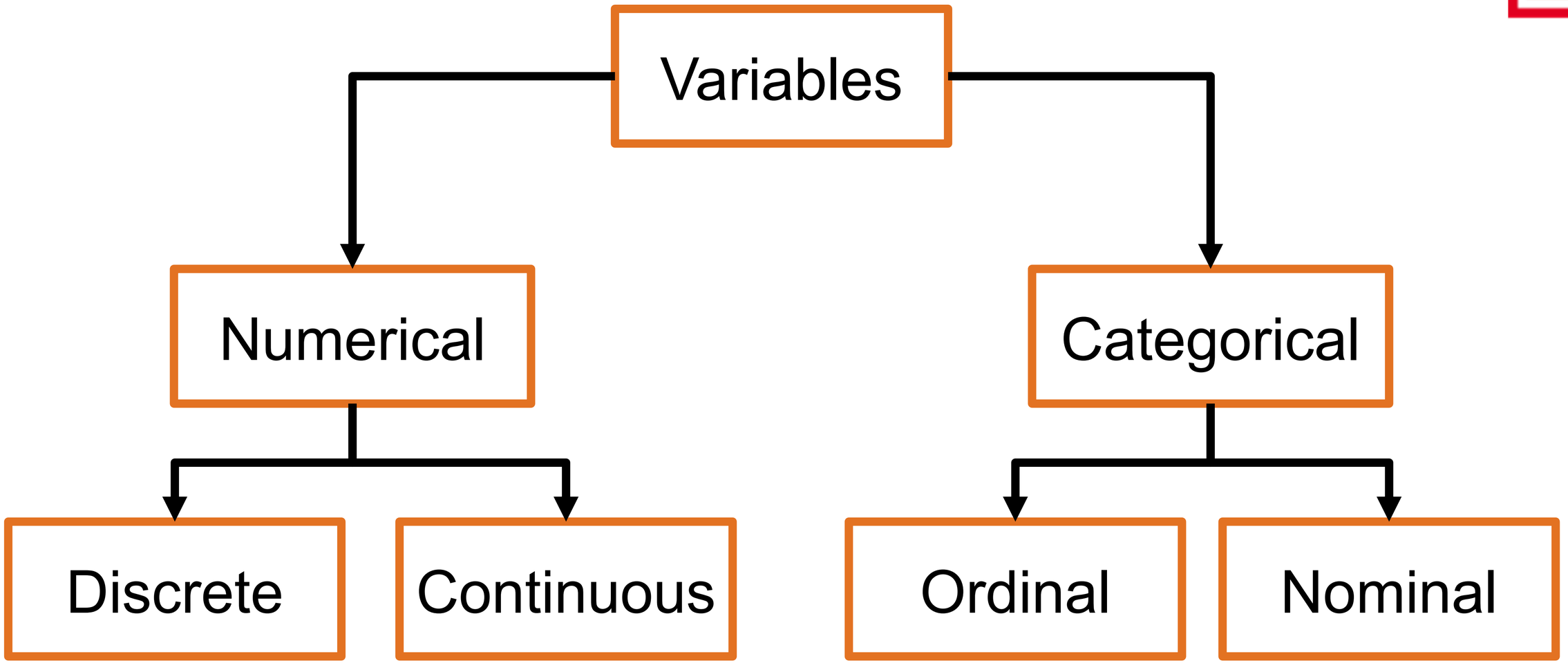
## Ratio

Meaningful zero, i.e. unique and non-arbitrary

Example: temperature in Kelvin, age, weight, height

S. S. Stevens: On the Theory of Scales of Measurement. In: Science. 1946, 103, S. 677–680.

# Hello Data


Variables
Numerical
Categorical
Discrete
Continuous
Ordinal
Nominal


https://xkcd.com/1838/

Thanks.

mirco.schoenfeld@uni-bayreuth.de