MonkeyLearn

- Text classification

Two kinds of Models

Text Classifiers

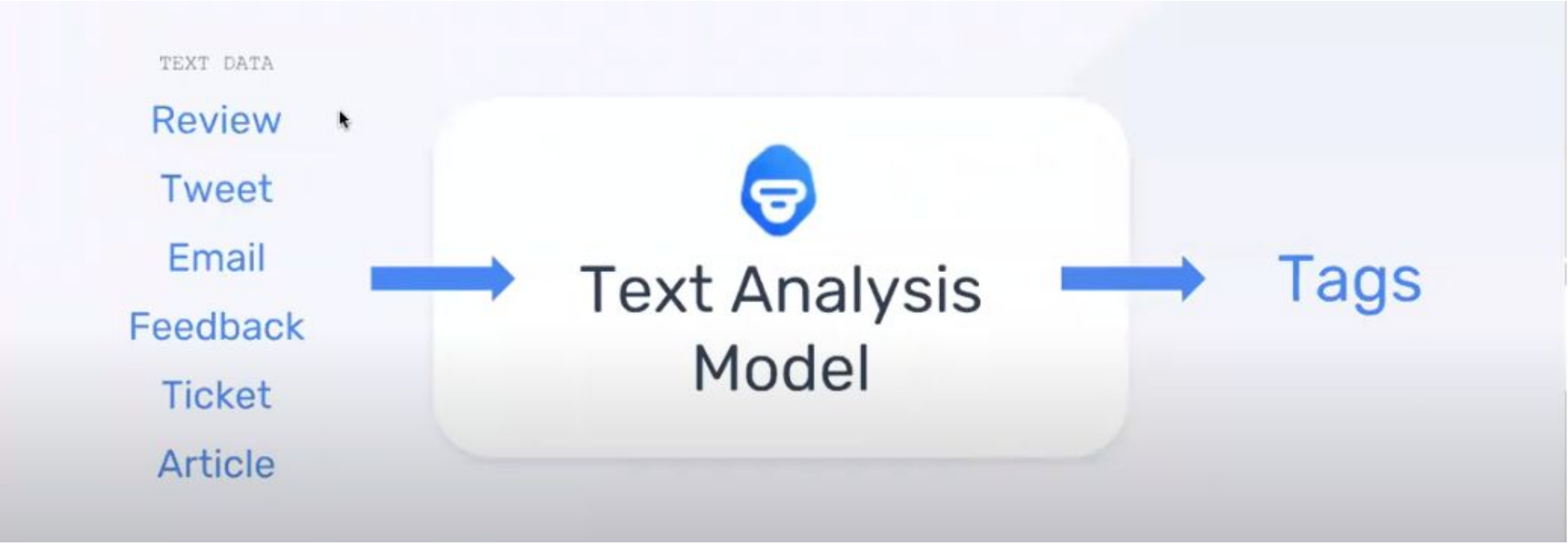
Classify text into categories or tags, e.g. sentiment, topic, urgency.

Text Extractors

Extract data from a given text, e.g. keyword, names, emails

Text Classifiers

Limitation: input data has to be CSV or Excel format



Text Classifiers

- Sentiment Analysis

Tags: positive, negative, neutral

- Topic Detector

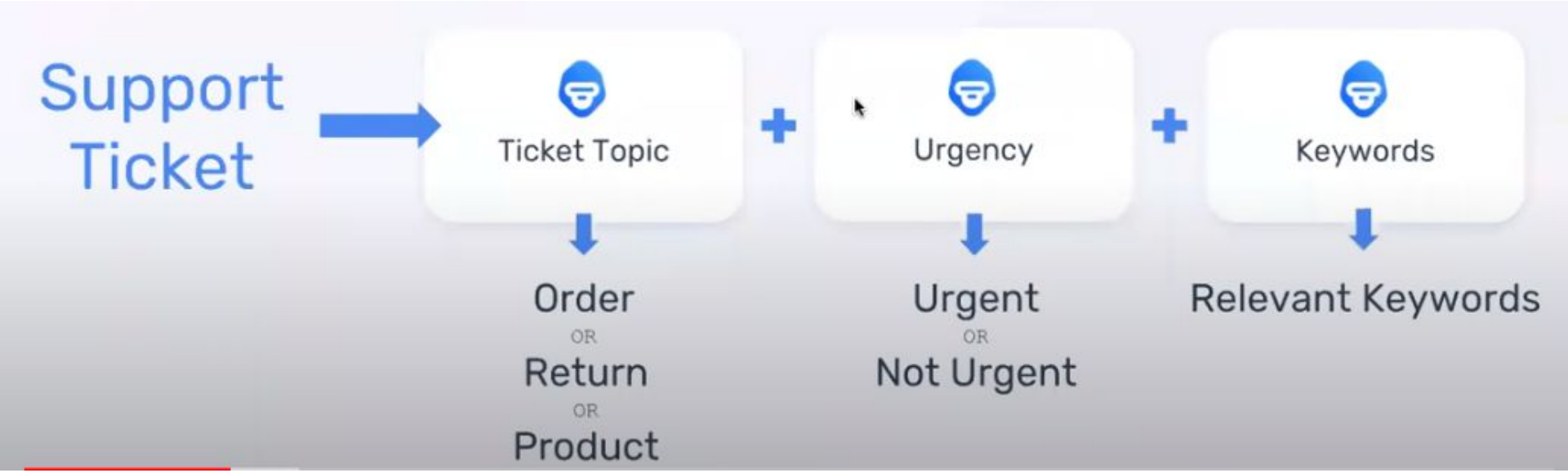
Tags: shipment, billing, return, inquiry

- Urgency Detector

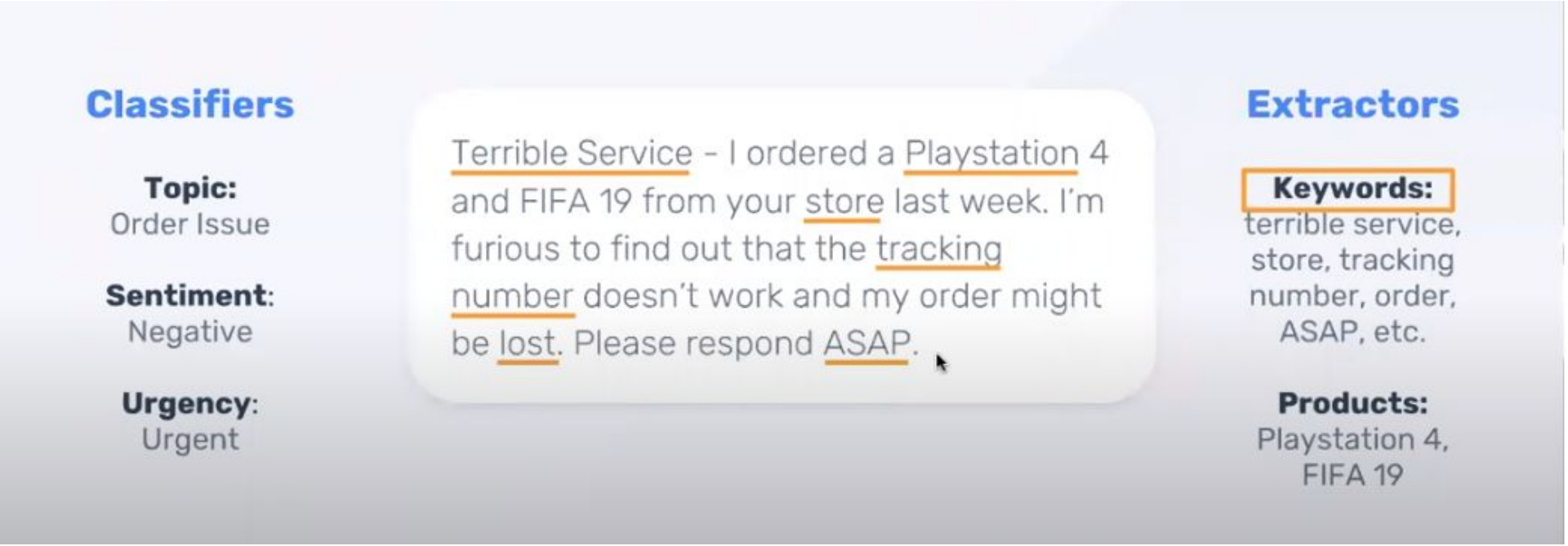
Tags: urgent, not urgent

Text Classifiers

Multiple Models Together

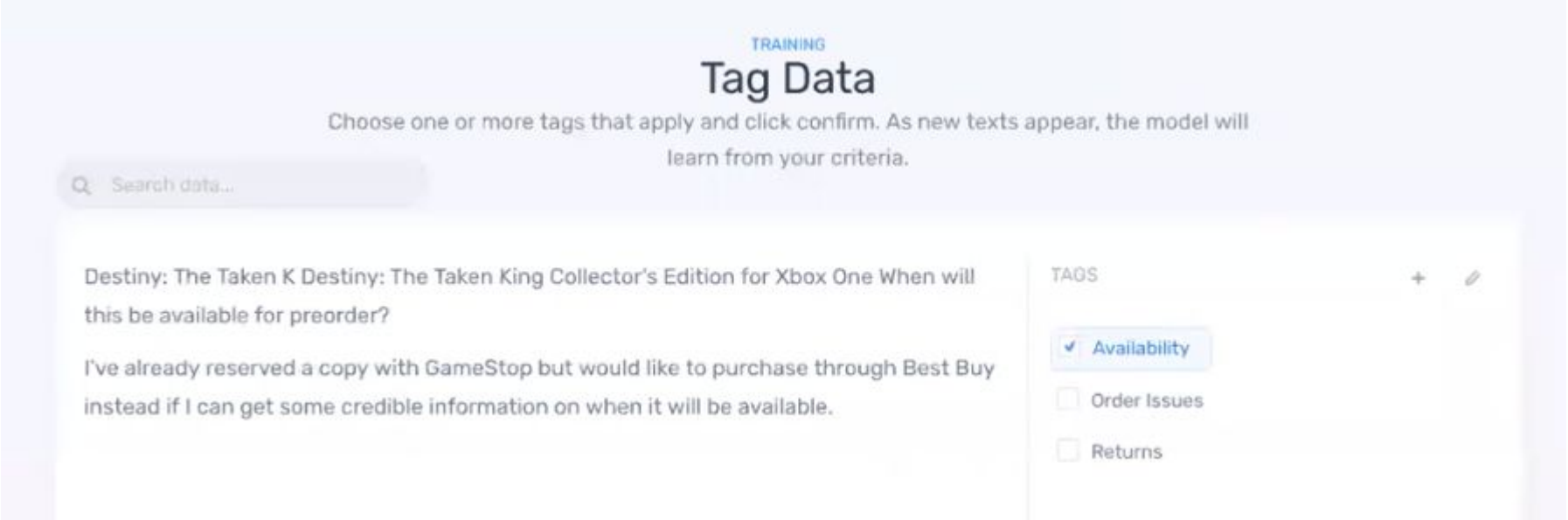


Keyword Extractor



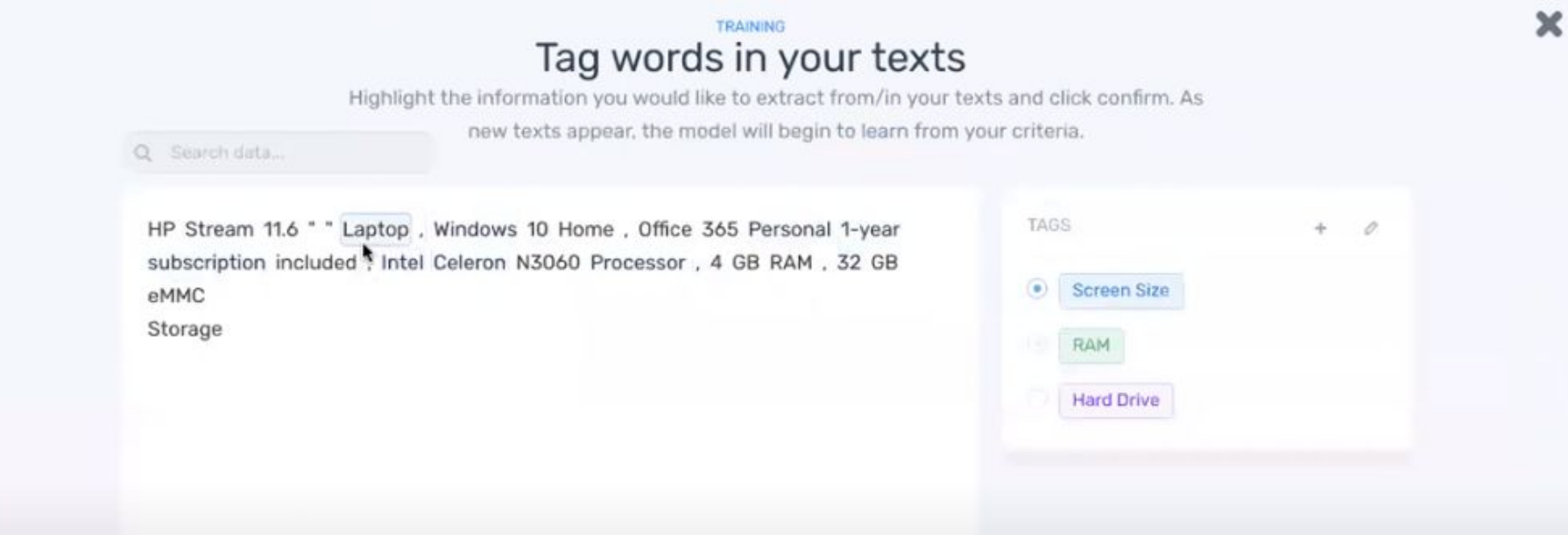
Train your own text classifier model

Upload data → define tags → tag data → test



Train your own keyword extractor model

Upload data → define tags → tag data → test



Supported algorithms

- Naive Bayes
- SVM (Support Vector Machines)

Naive Bayes

- Probabilistic algorithm
  - Calculate the probability of each tag for a given text, and then output the tag with the highest one
- Laplace smoothing

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Text	Tag
"A great game"	Sports
"The election was over"	Not sports
"Very clean match"	Sports
"A clean but forgettable game"	Sports
"It was a close election"	Not sports

$$P(sports|a\,very\,close\,game) = \frac{P(a\,very\,close\,game|sports) \times P(sports)}{P(a\,very\,close\,game)}$$

$$P(a\,very\,close\,game) = P(a) \times P(very) \times P(close) \times P(game)$$

$$P(a|Sports) \times P(very|Sports) \times 0 \times P(game|Sports)$$

Laplace smoothing:

Word	P (word | Sports)	P (word | Not Sports)
a	(2 + 1) ÷ (11 + 14)	(1 + 1) ÷ (9 + 14)
very	(1 + 1) ÷ (11 + 14)	(0 + 1) ÷ (9 + 14)
close	(0 + 1) ÷ (11 + 14)	(1 + 1) ÷ (9 + 14)
game	(2 + 1) ÷ (11 + 14)	(0 + 1) ÷ (9 + 14)

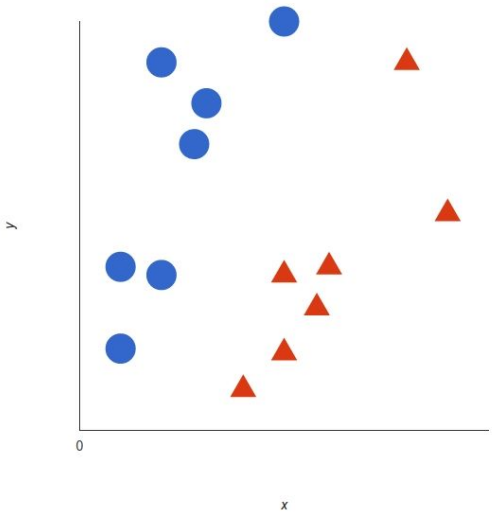
$$P(a|Sports) \times P(very|Sports) \times P(close|Sports) \times P(game|Sports) \times P(Sports)$$
$$= 2.76 \times 10^{-5}$$
$$= \underline{0.0000276}$$

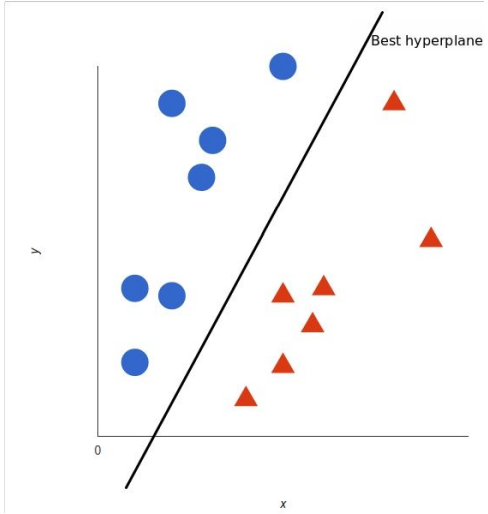
Classified as sport!

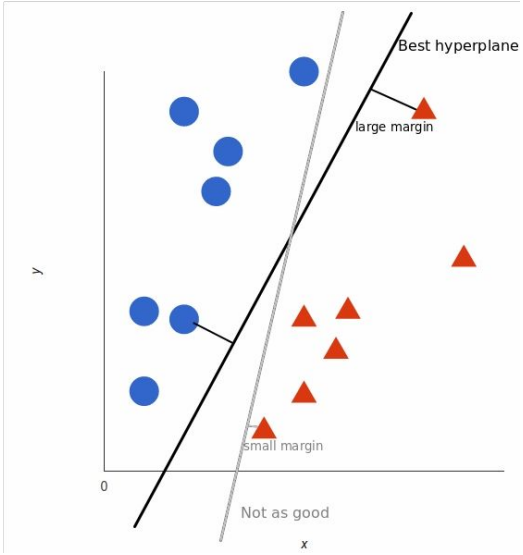
$$\mathrm{P(a}\,|Not\,Sports) \times P(very|Not\,Sports) \times P(close|Not\,Sports) \times P(game|Not\,Sports) \times P(Not\,Sports)$$
$$= 0.572 \times 10^{-5}$$
$$= \underline{0.00000572}$$

SVM (Support Vector Machines)

- Linear data

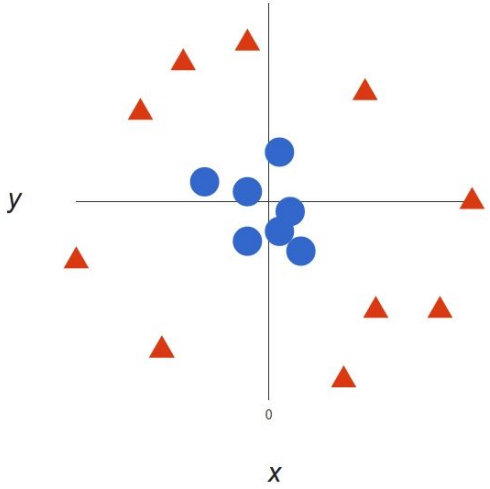


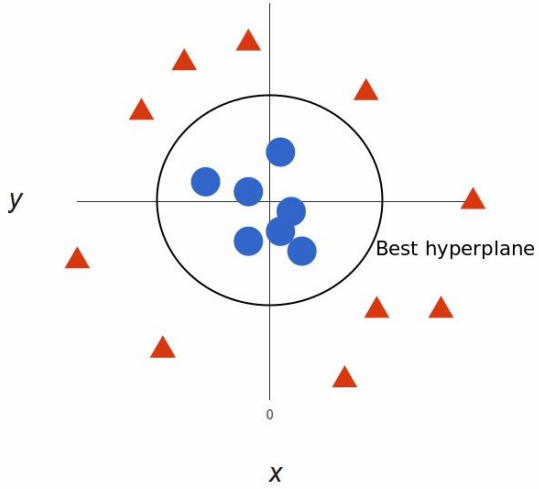




SVM (Support Vector Machines)

- Nonlinear data





References

- Text classifiers & text extractors
  - https://www.youtube.com/watch?v=vHSadS-1wOA

- Naive Bayes
  - https://monkeylearn.com/text-classification-naive-bayes/
  - https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/
- SVM
  - https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/
  - https://monkeylearn.com/text-classification-support-vector-machines-svm/