Project Fletcher

Tools for Building Systems

Flask

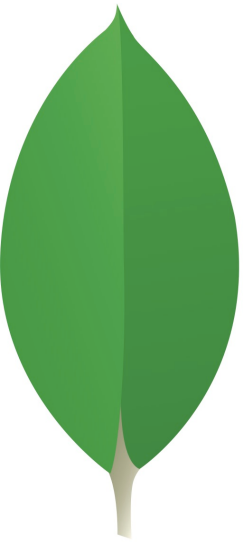
web development,
one drop at a time

"What does the model say for my user?"

"Probability of success is 0.78!"

"Thanks!"

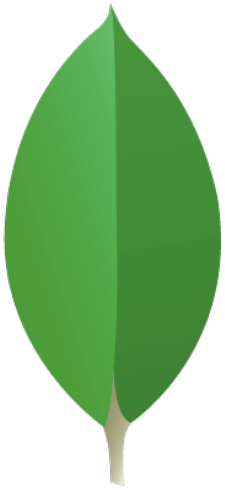
"Also here is the rest of the web site."



mongoDB

“I have data in some JSON-like structure
that could possibly change at any moment
and I just want to store it and be able to get it
back possibly with some search functionality!”

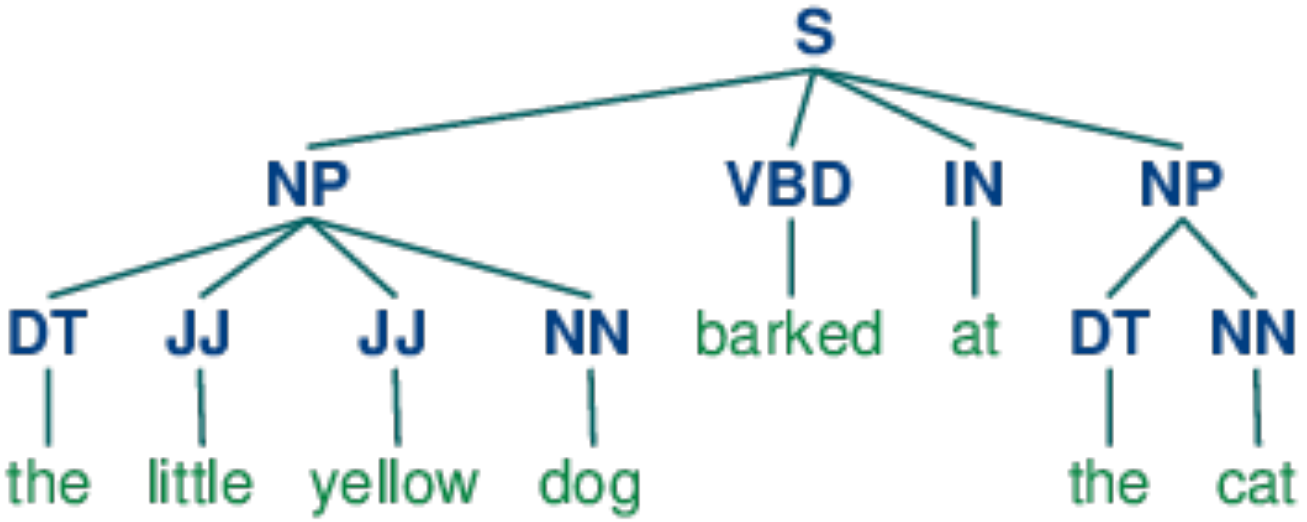
“Yes.
Yes I will do that for you.
Schemas are gross.
Never do a join.”



Natural Language Processing with Python

Tokenization, Part of Speech Tagging, Chunking, n-grams

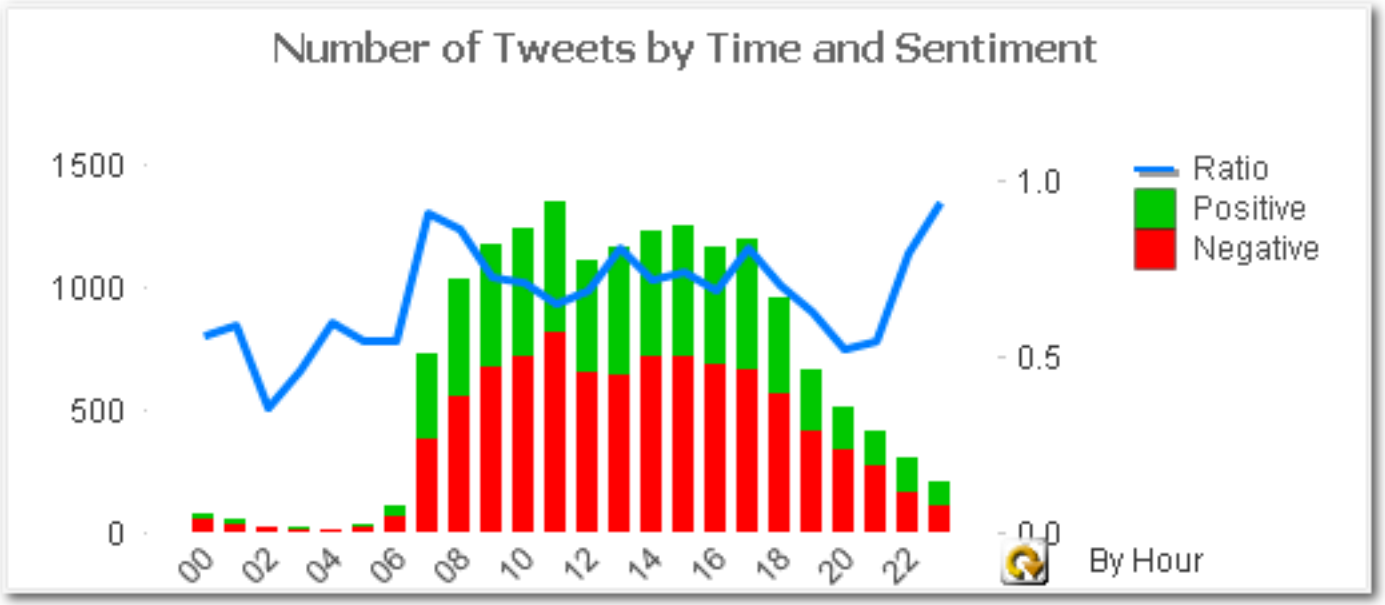
Take the text apart. Find sentences, words, adjectives, nouns, adverbs, noun phrases in it.



Sentiment Analysis

Find out how subjective or objective a text document is.

Find out if it has a positive, negative or neutral sentiment.



Counts, Frequencies, TF*IDF, Keyword Extraction

Find important words in a document or a cluster. Turn text documents into feature vectors (of numbers) so you can measure their distance or put them into machine learning algorithms.

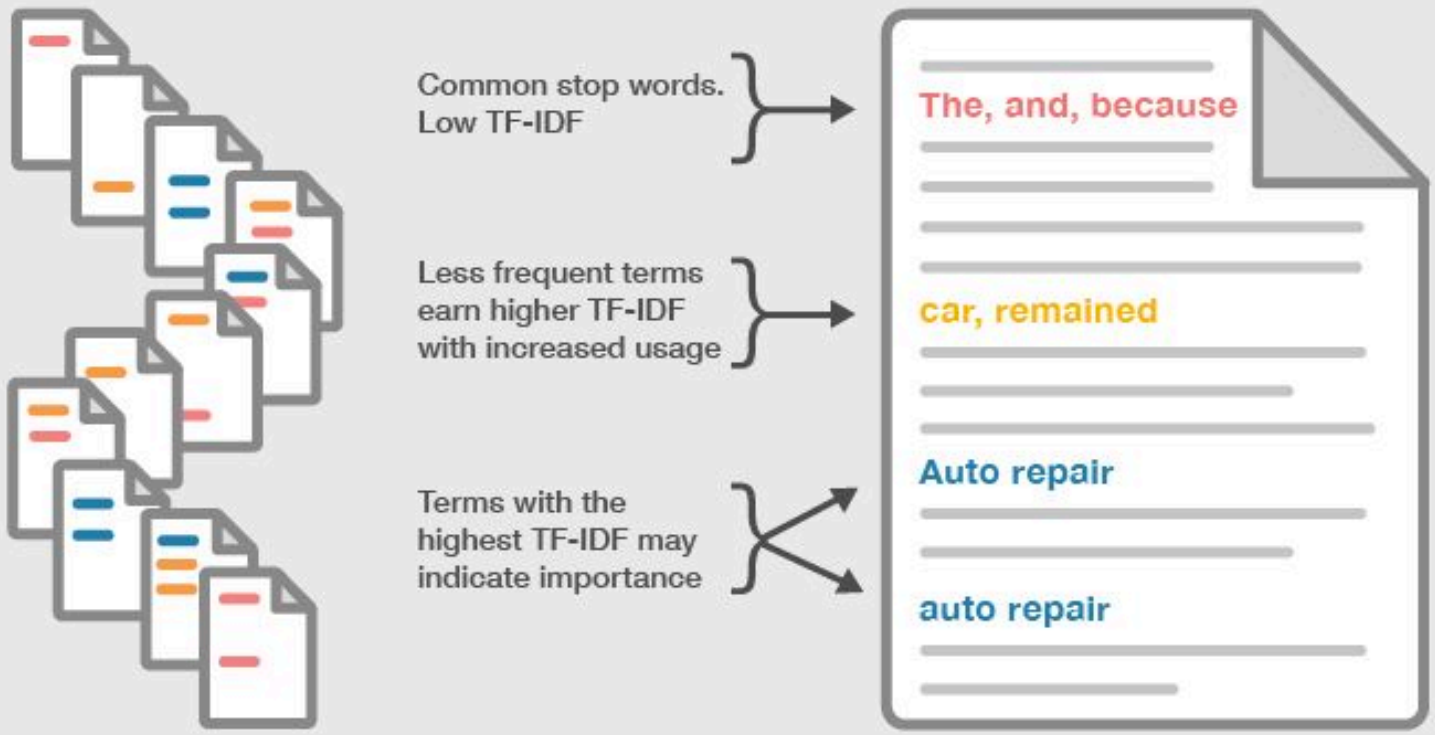
5-Star Hotels in London

5-Star Connections in London

Now you too can live in luxury, or at least for a few days, at one of three five-star hotels added to the iPass footprint

London is certainly not short of fabulous hotels, but iPass users in the UK capital can now enjoy the crème de la crème of luxurious hotels – courtesy of our latest partnership agreement with high-speed Internet access specialist RIEO Communications.

Enjoy seamless Wi-Fi connectivity with your iPass service at our five-star hotel line up including recently added Claridge's, The Berkeley and The Connaught Hotels, as well as more than 20 De Vere Hotels across the UK.



TF-IDF

Term frequency–inverse document frequency (TF-IDF) measures the importance of a keyword phrase by comparing it to the frequency of the term in a large set of documents. Many advanced textual analysis techniques use a version of TF-IDF as a base.

MOZ

Text Classification with Naïve Bayes

Naïve Bayes often shines in text classification. Nothing terribly new to this, just turn your text into vectors and let Bayes do his wonderful magic

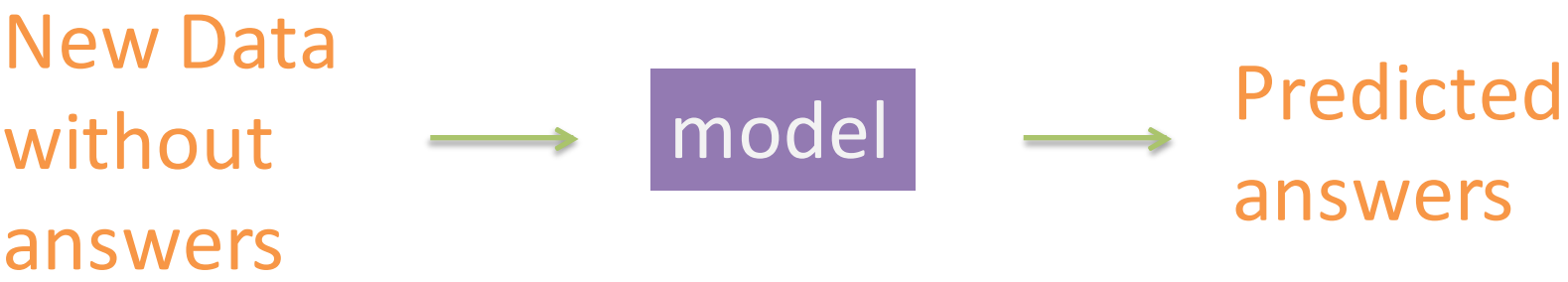
$$P(spam|penis, viagra)$$

$$= \frac{P(penis|spam) * P(viagra|spam) * P(spam)}{P(penis) * P(viagra)}$$

$$= \frac{\frac{24}{30} * \frac{20}{30} * \frac{30}{74}}{\frac{25}{74} * \frac{51}{74}} = 0.928$$

Unsupervised Learning

Supervised Learning



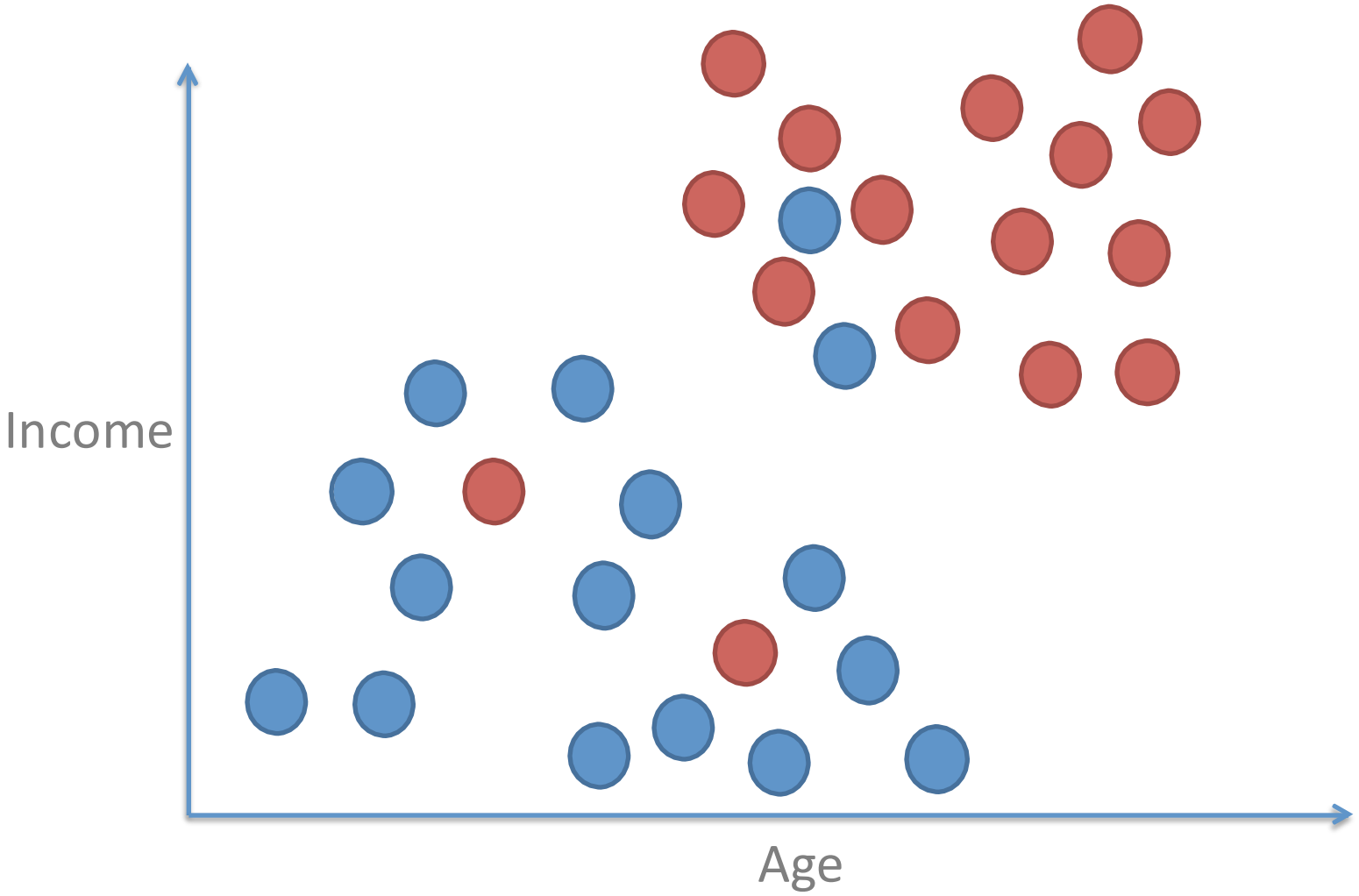
Unsupervised Learning

Unlabeled Data
(no answers)

structure

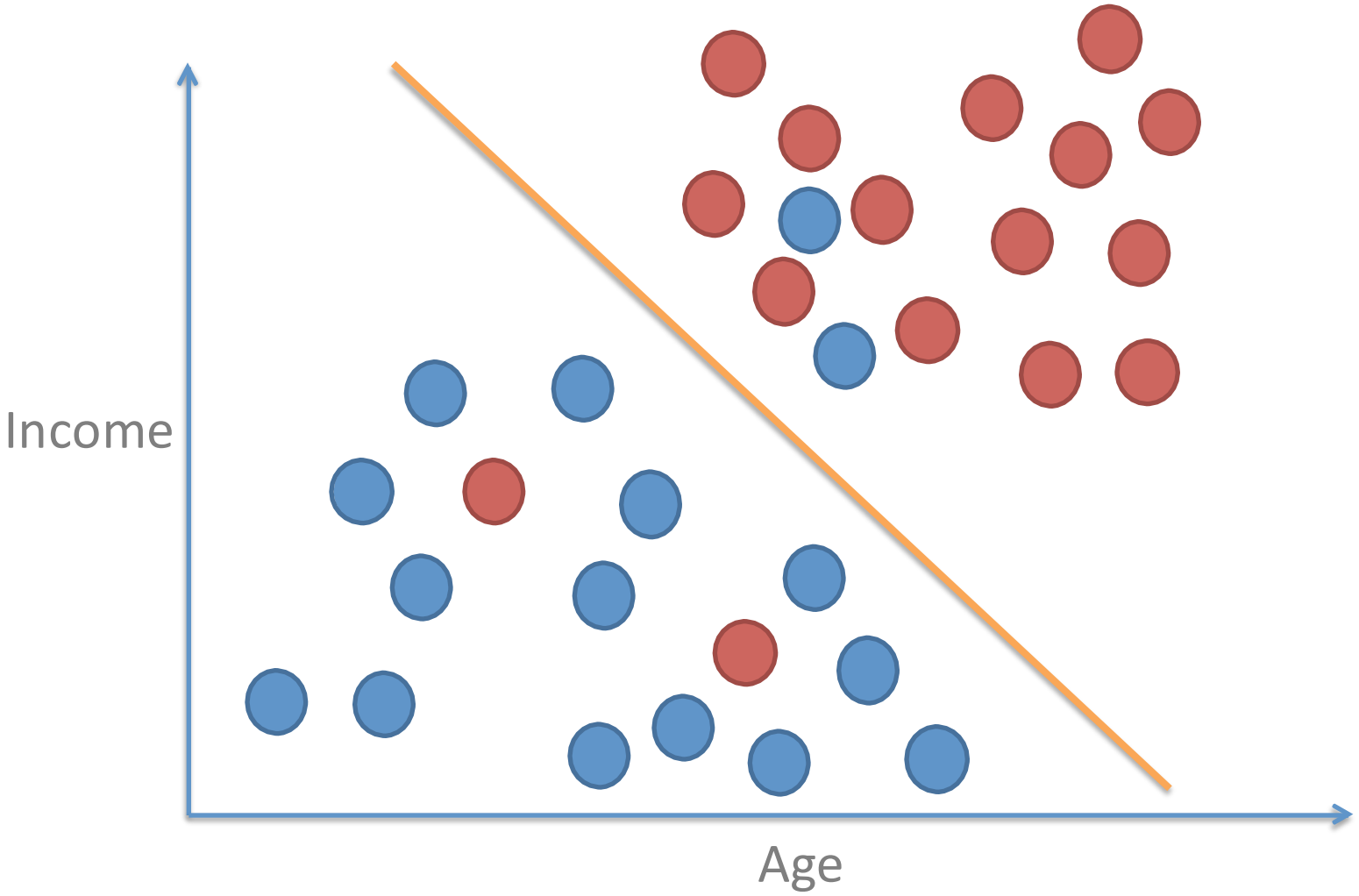
Making a map to better understand data

Supervised Learning

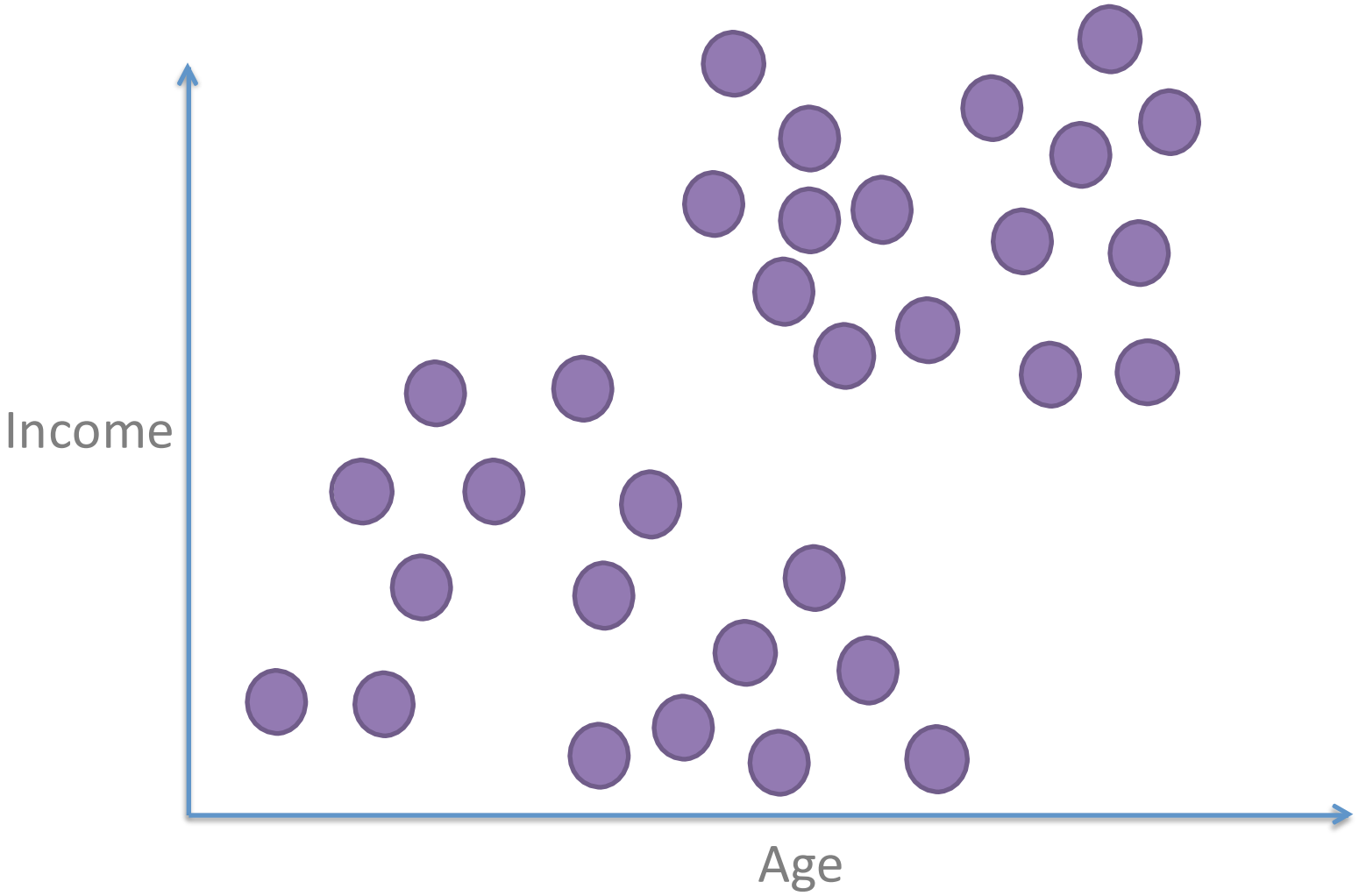


Supervised Learning

Find decision boundary using labels

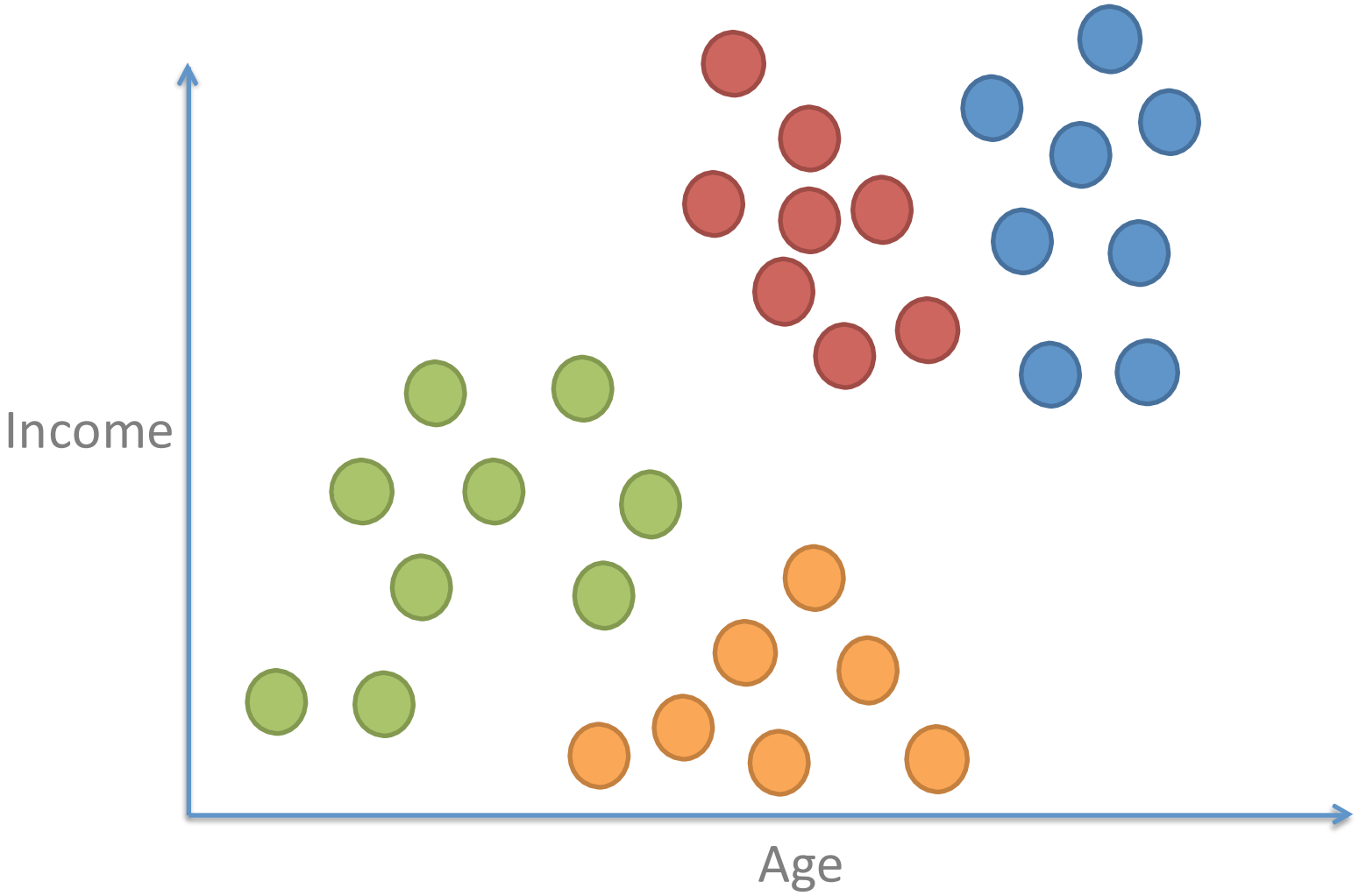


Unsupervised Learning



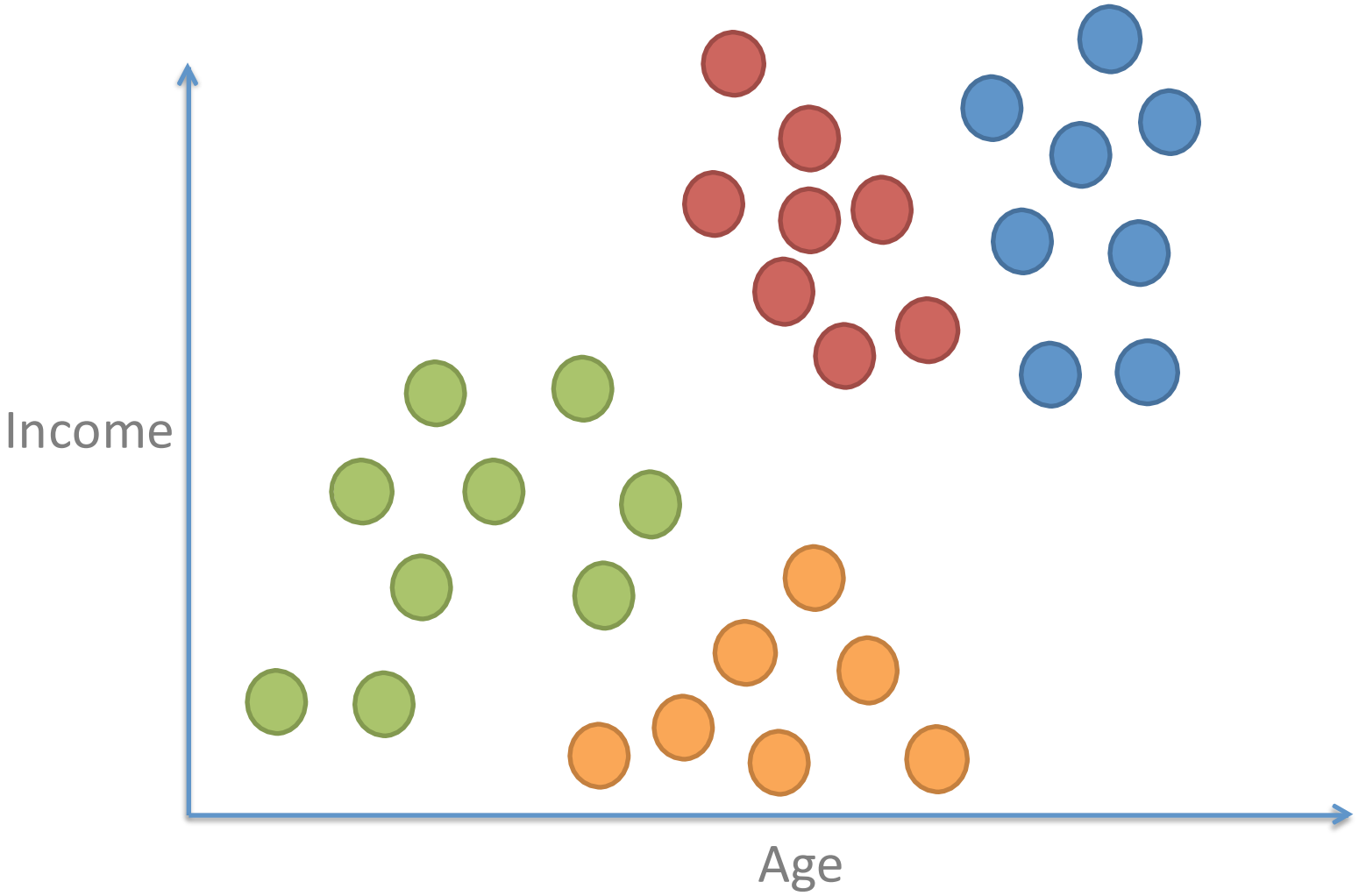
Unsupervised Learning

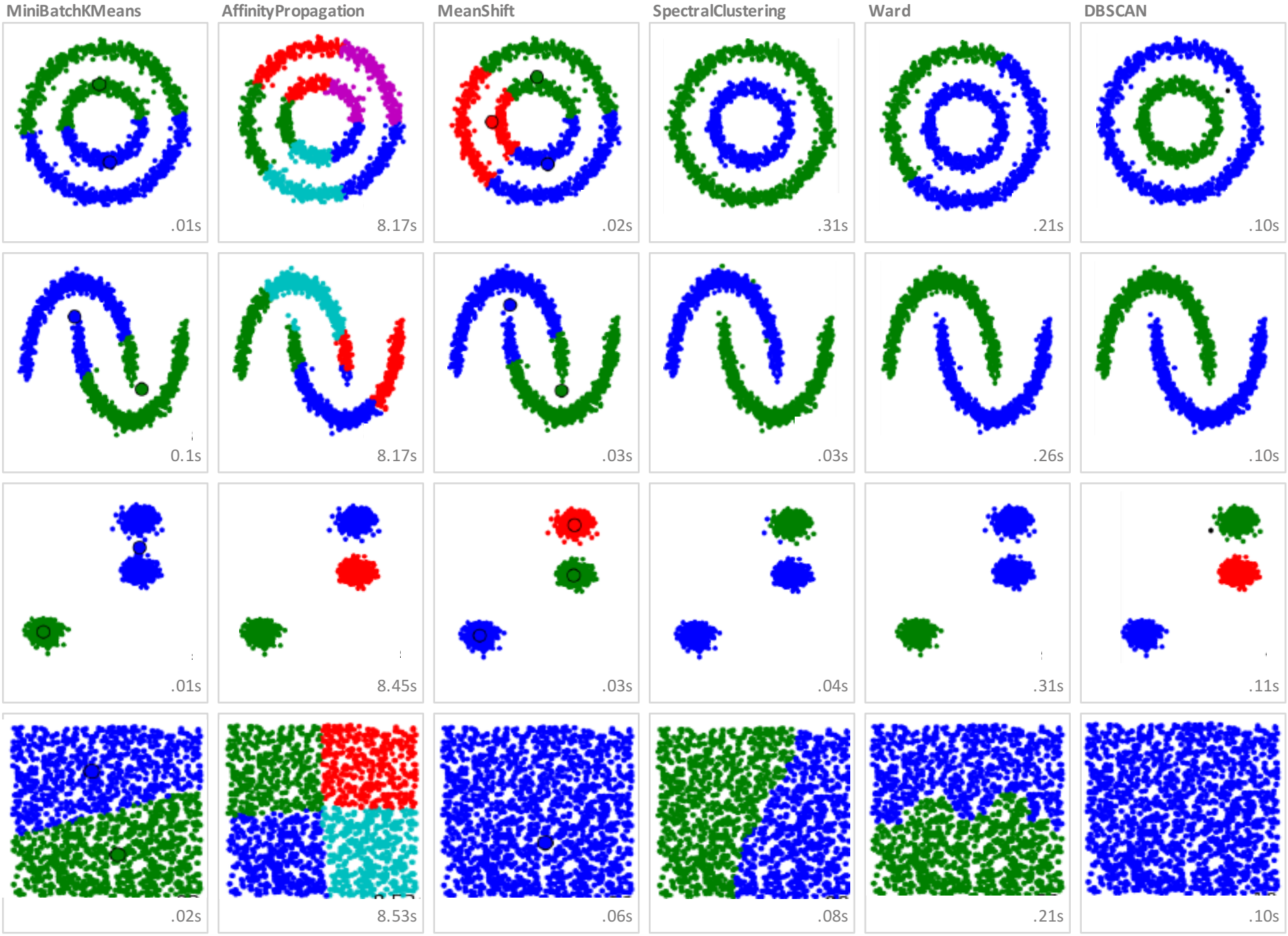
Find structure in unlabeled data



Clustering Algorithms

Finding natural groups of similar points




MiniBatchKMeans
AffinityPropagation
MeanShift
SpectralClustering
Ward
DBSCAN
.01s
8.17s
.02s
.31s
.21s
.10s
0.1s
8.17s
.03s
.03s
.26s
.10s
.01s
8.45s
.03s
.04s
.31s
.11s
.02s
8.53s
.06s
.08s
.21s
.10s

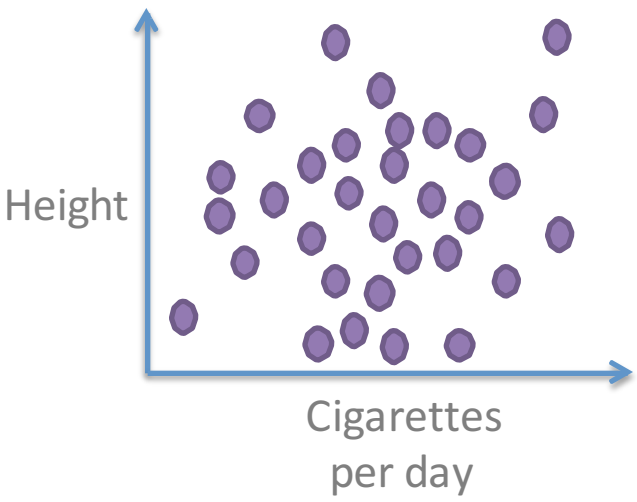
Dimensionality Reduction

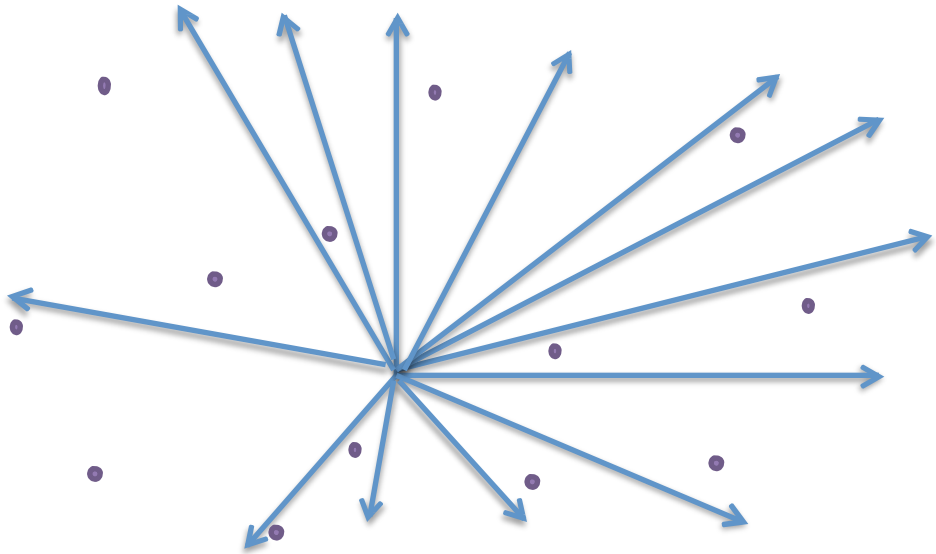
Dimensionality Reduction

What if you have a ton of features?

Not all features provide the same amount of information.

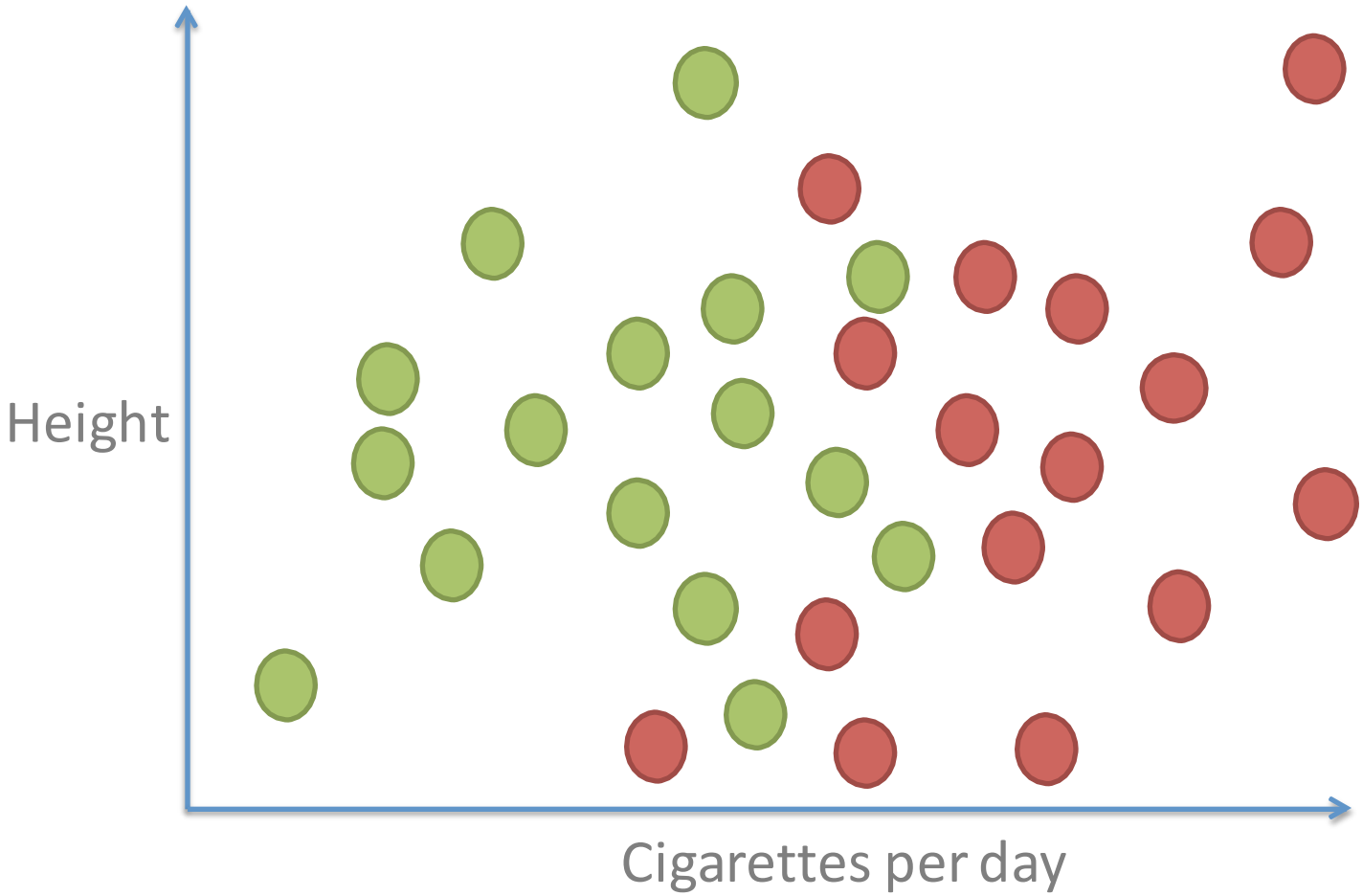
We can reduce the dimensions (compress the data) without necessarily losing too much information.





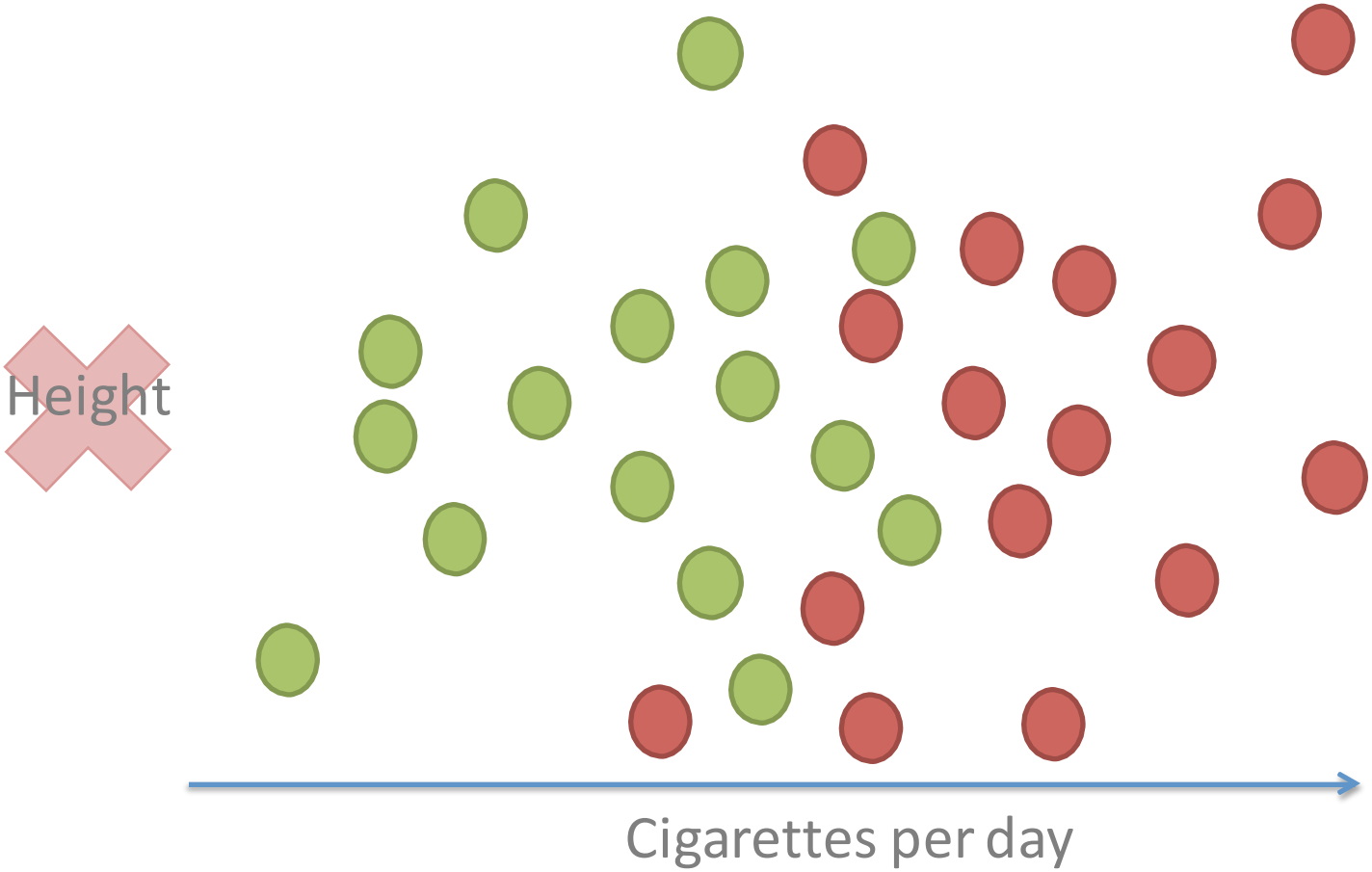
Feature Selection

Healthy / Heart Disease



Feature Selection

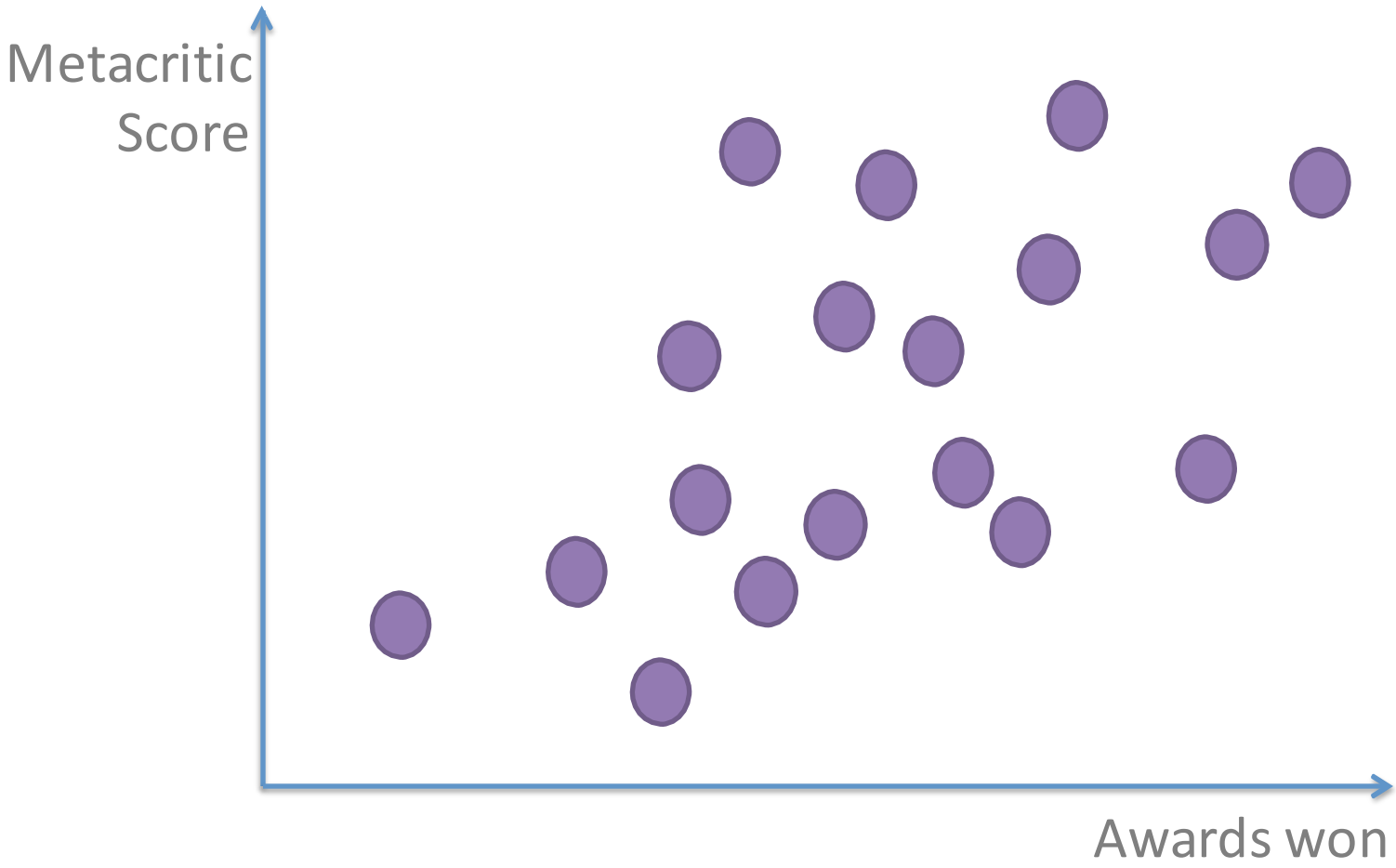
Healthy / Heart Disease



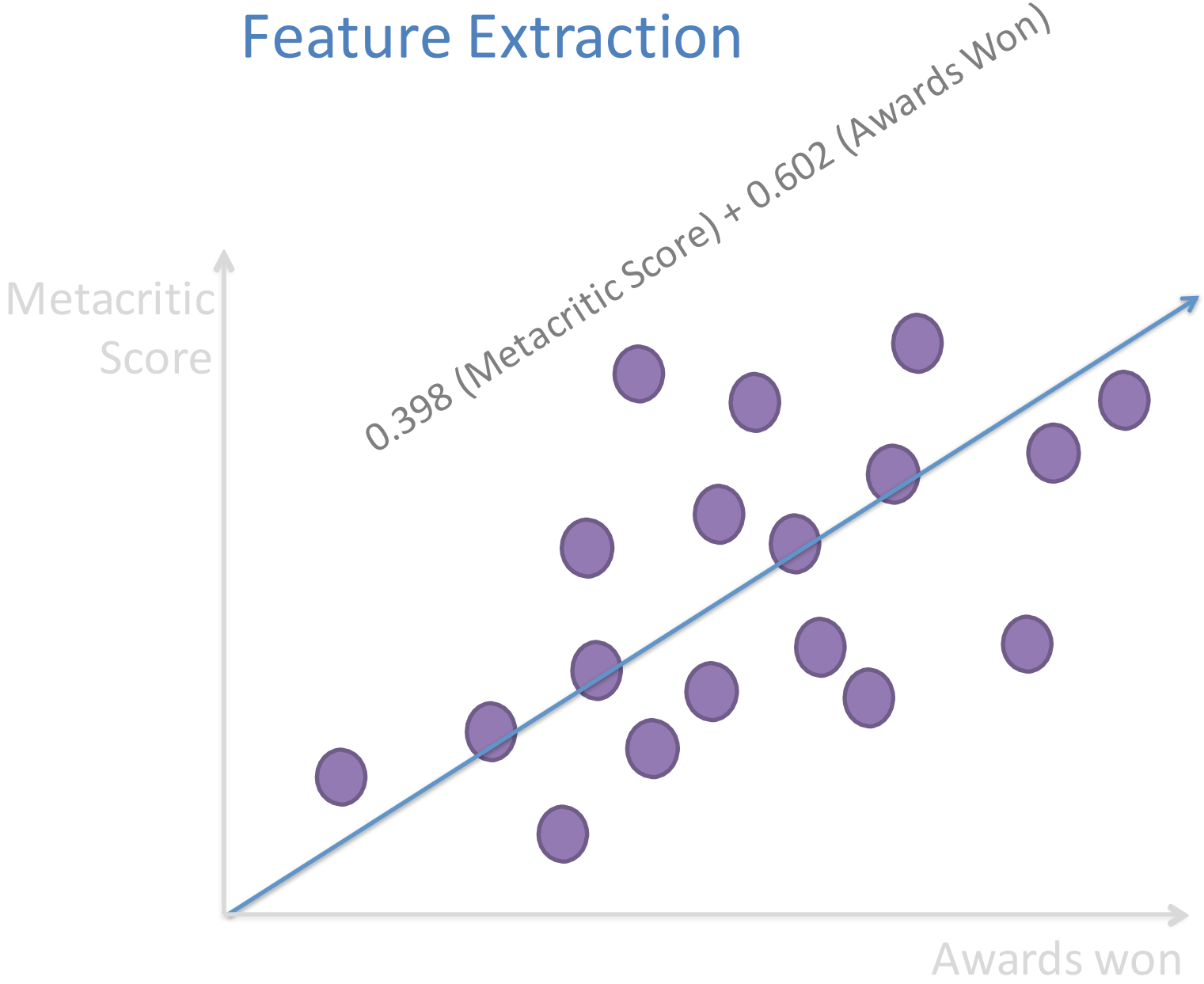
Feature Selection

Healthy / Heart Disease

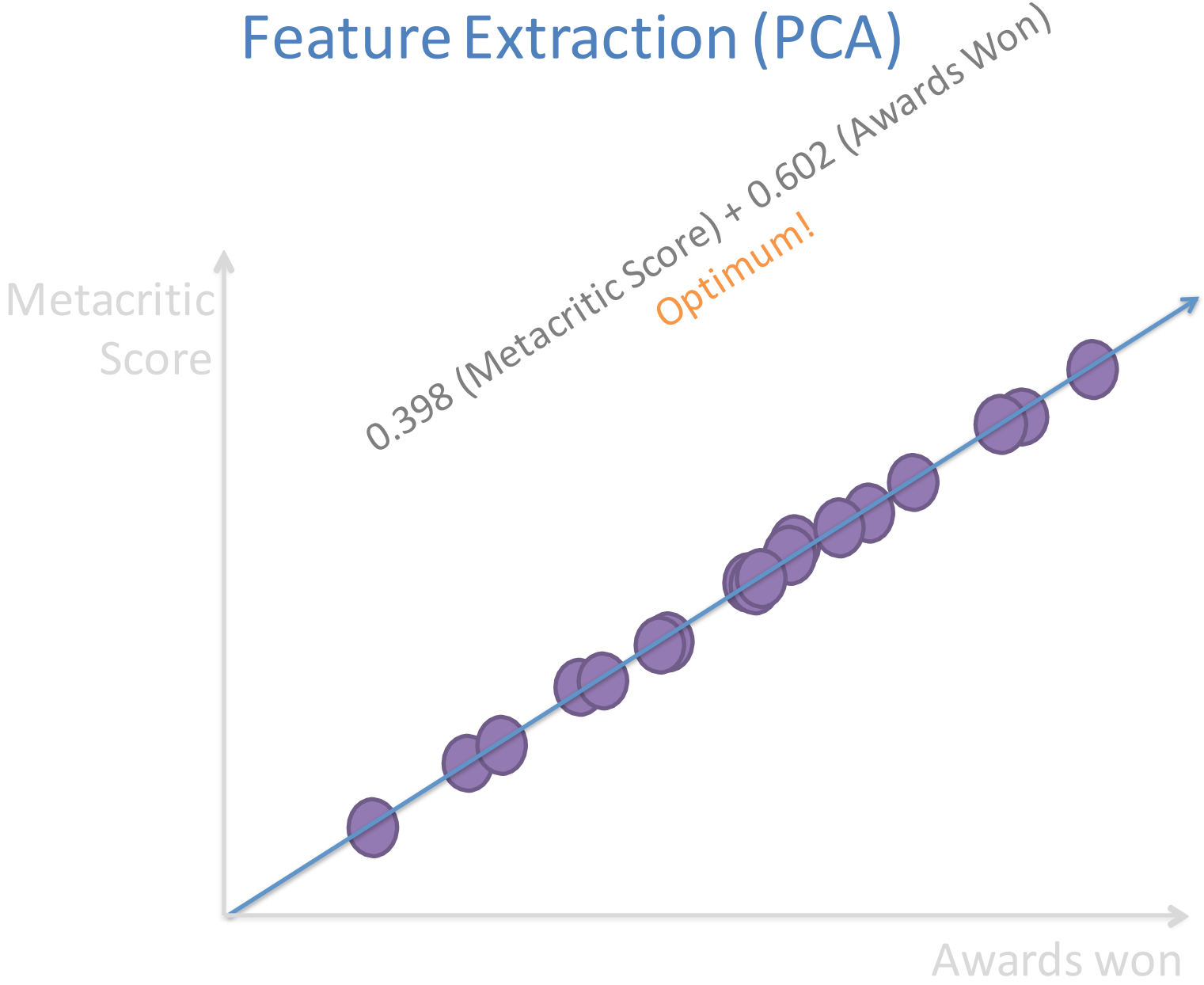
Feature Extraction



Feature Extraction



Feature Extraction (PCA)



Topic Modeling

3D → 2D Reduction with text data (bag of words model)

“I love my pet rabbit.”
“That dish yesterday was amazing.”
“She cooked the best rabbit dish ever.”
“I gave leftovers of that dish to my pet, mr. rabbit”
“Rabbits make messy pets.”
“My rabbit growls when I pet her.”
“He has five rabbits.”
“I had this weird dish with fried rabbit.”
“That’s my pet rabbit’s favorite dish.”
…

3D → 2D Reduction with text data (bag of words model)

"I love my pet rabbit."
"That dish yesterday was amazing."
"She cooked the best rabbit dish ever."
"I gave leftovers of that dish to my pet, mr. rabbit"
"Rabbits make messy pets."
"My rabbit growls when I pet her."
"He has five rabbits."
"I had this weird dish with fried rabbit."
"That's my pet rabbit's favorite dish."
...

Remove stop words, only keep nouns, end up with 3 features: "rabbit", "pet", "dish"

TOPIC 1: 1.5(Rabbit) + 1.1 (Pet) + 0.1(Dish) ← Pet rabbits, pets

TOPIC 2: 0.9(Rabbit) + 0.02(Pet) + 1.6(Dish) ← Food, rabbit dishes

"I love my **pet rabbit**."

Topic1: High "**Rabbits** make messy **pets**."

Topic2: Low "My **rabbit** growls when I **pet** her."

"He has five **rabbits**."

Topic1: Low "That **dish** yesterday was amazing."

Topic2: High "She cooked the best **rabbit dish** ever."

"I had this weird **dish** with fried **rabbit**."

Topic1: High "I gave leftovers of that **dish** to my **pet**, mr. **rabbit**"

Topic2: High "That's my **pet rabbit**'s favorite **dish**."

TOPIC 1: 1.5(Rabbit) + 1.1 (Pet) + 0.1(Dish) ← Pet rabbits, pets

TOPIC 2: 0.9(Rabbit) + 0.02(Pet) + 1.6(Dish) ← Food, rabbit dishes

What is a topic?

When writing about a specific topic (like pet rabbits), we use some words more often than others.

Words like “pet”, “rabbit”, “lettuce”, “cage”, “fluffy”, etc. are more likely to appear, words like “dish”, “transmission”, “opaque”, “affair” are less likely to appear.

A topic can be thought as a
Probability distribution over all possible words