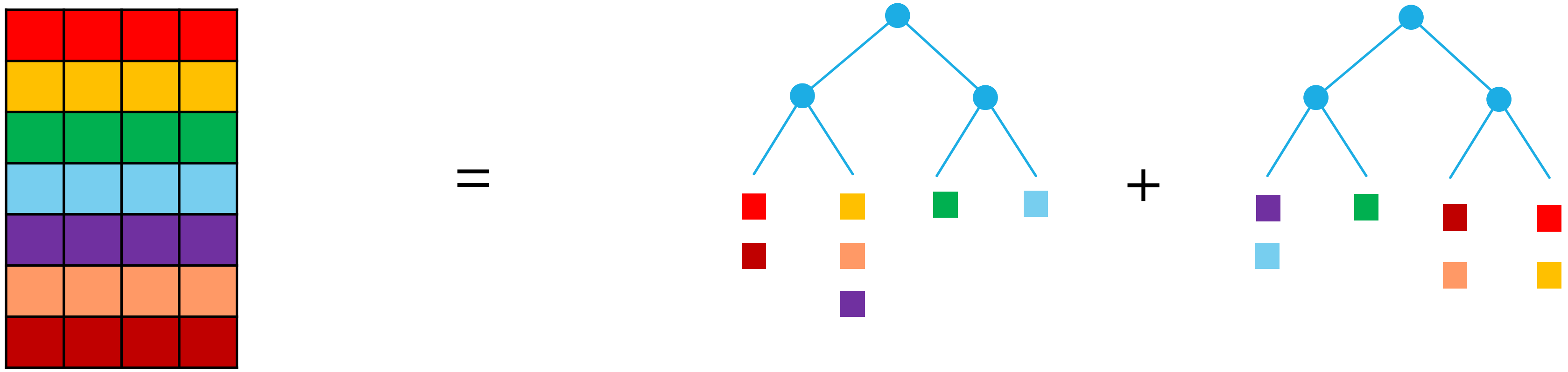
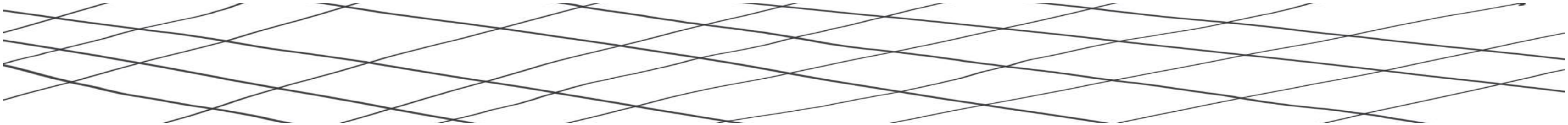




RANDOM PROJECTIONS FOR SEARCH AND MACHINE LEARNING

Stefan Savev

Berlin Buzzwords
June 2015

KEYWORD-BASED SEARCH

Document Data

- 300 unique words per document
- 300 000 words in vocabulary
- Data sparsity: 99.9%
- Words are strong query features

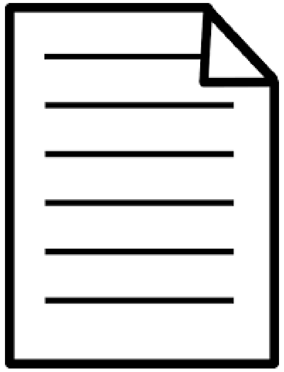
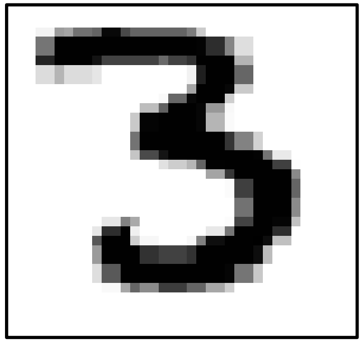


IMAGE SEARCH BY EXAMPLE

Image Data

- 800 pixels
- 200 black pixels
- Data sparsity 80%
- Pixels are weak query features



TWO KINDS OF SEARCH

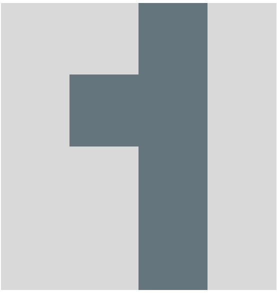
Data	Sparse	Dense
Query Size	Short	Long
Use Cases	Keyword Search	Image Search, Semantic Search, Recommendations
SEARCH METHOD	**INVERTED INDEX (LUCENE)**	**RANDOM PROJECTIONS**

OUTLINE

1. High dimensional data (images, text, clicks)
2. Random Projections: Why? What? How?
3. Image Search Benchmark

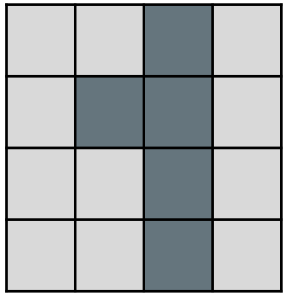
HIGH-DIMENSIONAL DATA

Images



HIGH-DIMENSIONAL DATA

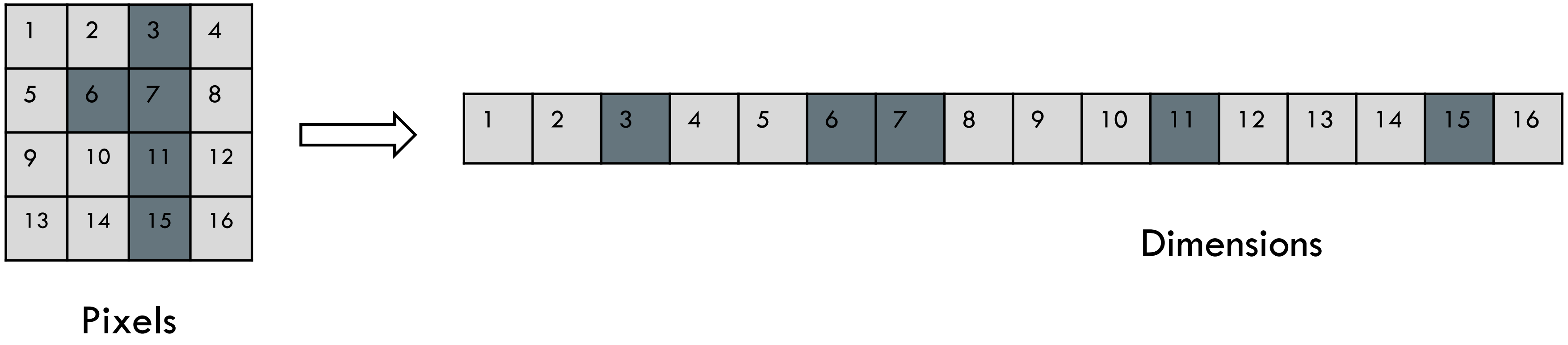
Images



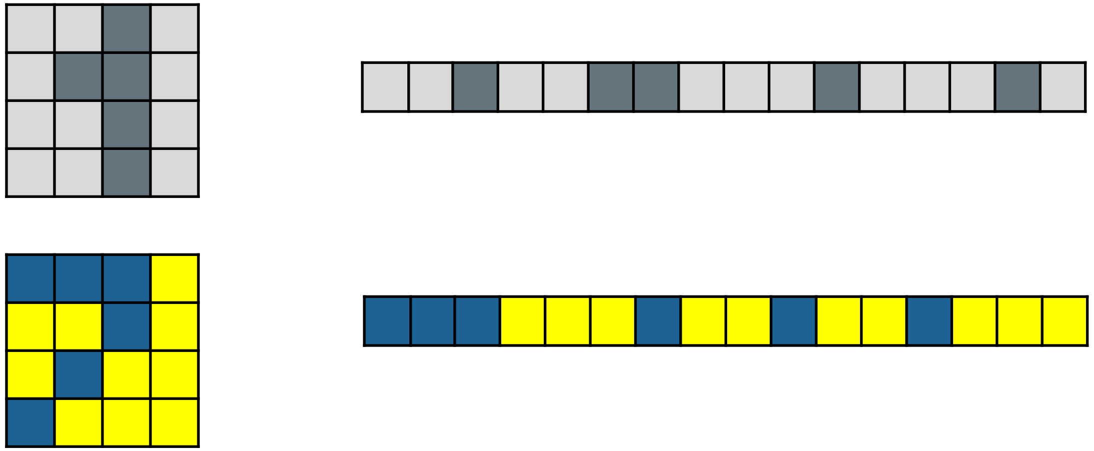
Pixels

HIGH-DIMENSIONAL DATA

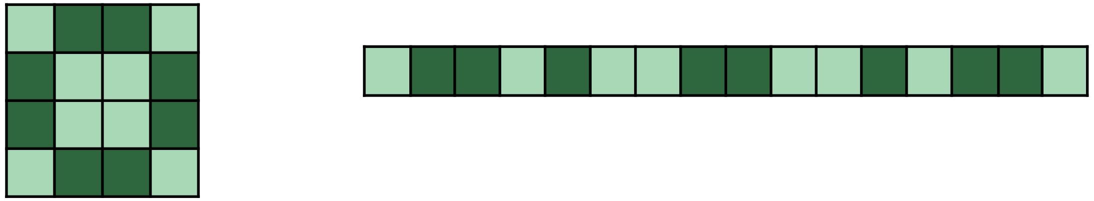
Images



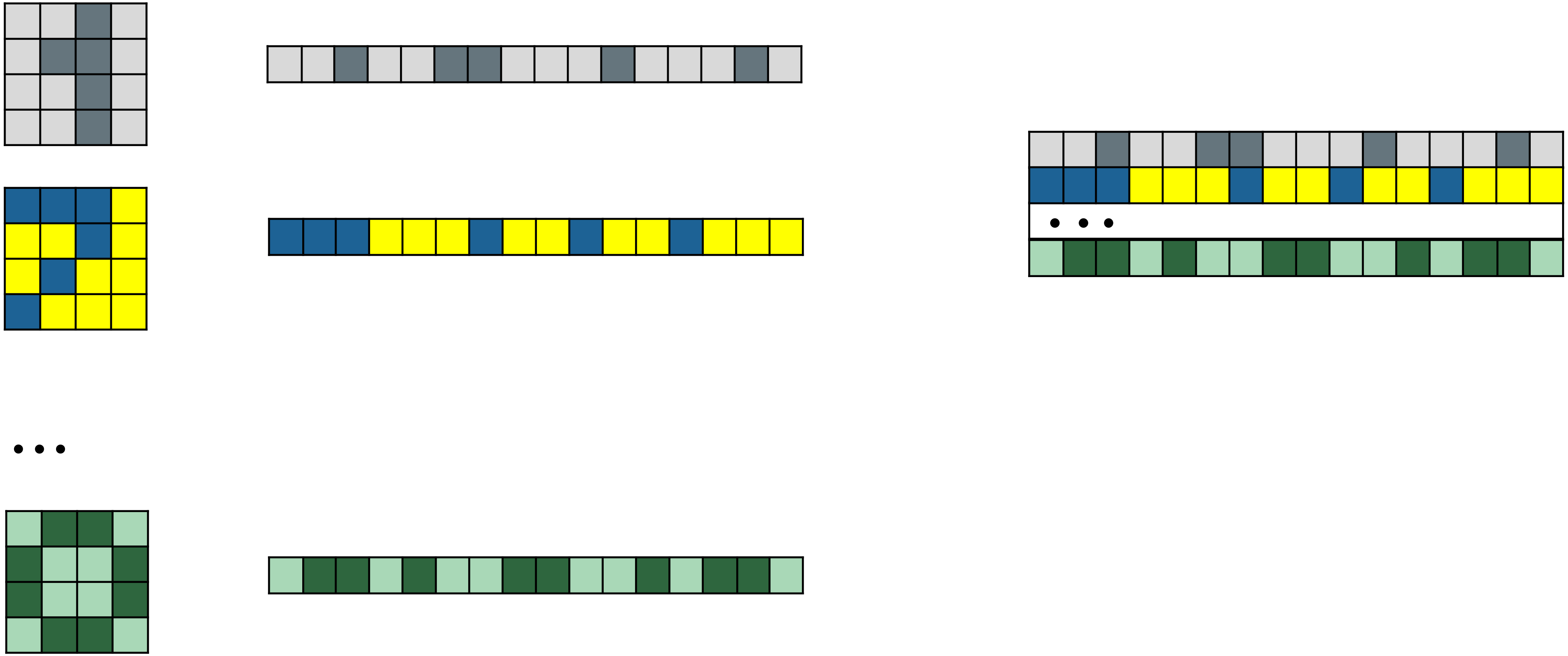
MATRIX REPRESENTATION OF DATASET



…



MATRIX REPRESENTATION OF DATASET



HIGH-DIMENSIONAL DATA

Text

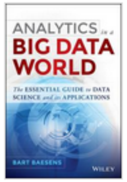
A beer can do everyone a favor.

a	…	abstract	…	be	bee	…	beer	…	can	…	do	…	everyone	…
2		0		0	0		1		1					

Dimensions

HIGH-DIMENSIONAL DATA

Clicks



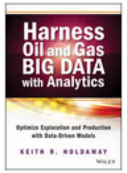
Analytics in a Big Data World

The Essential Guide to Data Science and its Applications

by *Bart Baesens*

Wiley and SAS Business Series

The guide to targeting and leveraging business opportunitiesusing big data & analytics By leveraging big data & analytics, businesses create thepotential to better understand, manage, and strategicallyexploiting the complex dynamics of customer behavior. Analyticsin... *Read more*



Harness Oil and Gas Big Data with Analytics

Optimize Exploration and Production with Data Driven Models

by *Keith Holdaway*

Wiley and SAS Business Series

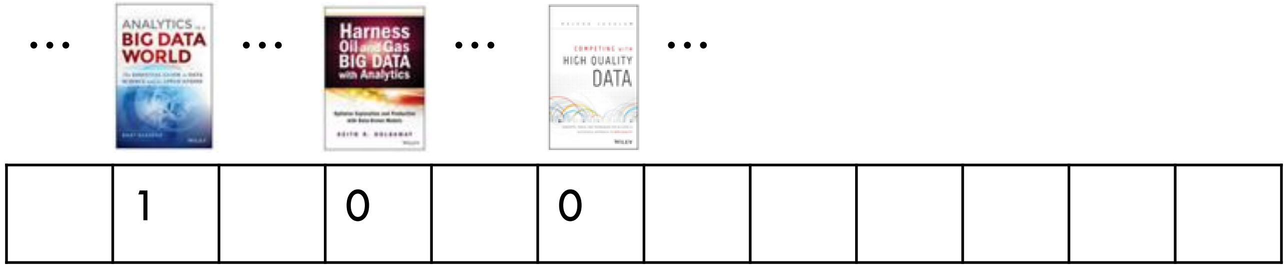
Use big data analytics to efficiently drive oil and gasexploration and production Harness Oil and Gas Big Data with Analytics provides acomplete view of big data and analytics techniques as they areapplied to the oil and gas industry. Including a compendium ofspecific case... *Read more*

Competing with High Quality Data

Concepts, Tools, and Techniques for Building a Successful Approach to Data Qu...

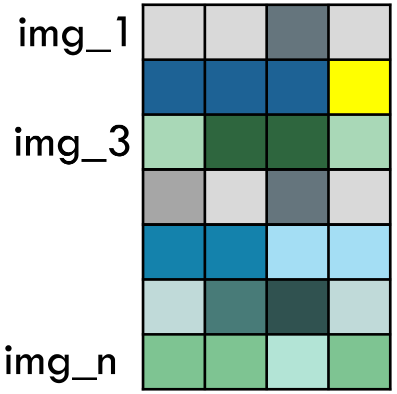
by *Rajesh Jugulum*

Create a competitive advantage with data quality Data is rapidly becoming the powerhouse of industry, butlow-quality data can actually put a company at a disadvantage. Tobe used effectively, data must accurately reflect the real-worldscenario it represents, and it ... *Read more*

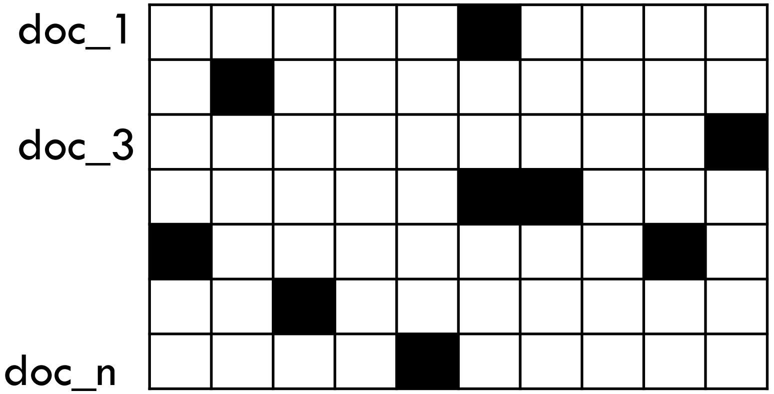


DENSE VS SPARSE MATRIX

IMAGES
1000 PIXELS



TEXT/CLICKS
300 000 WORDS

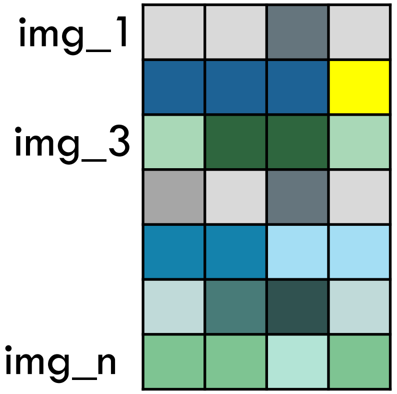


CANNOT SEARCH
WITH LUCENE

CAN SEARCH WITH LUCENE

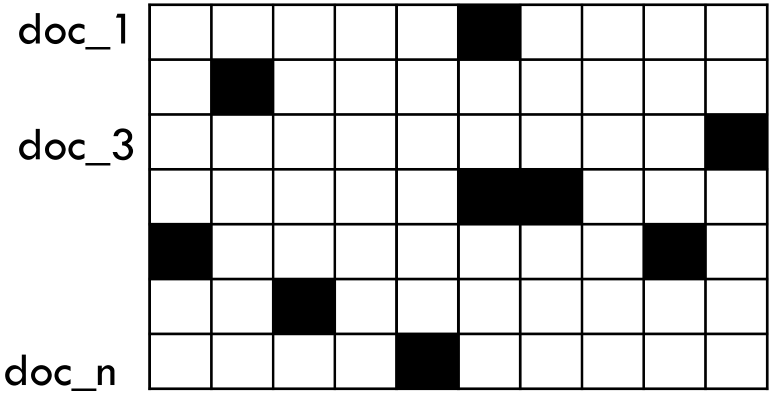
DENSE VS SPARSE MATRIX

IMAGES
1000 PIXELS



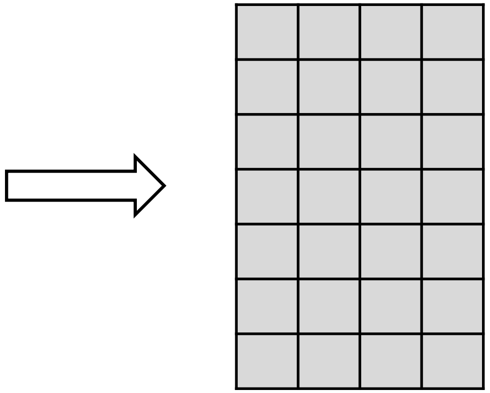
CANNOT SEARCH
WITH LUCENE

TEXT/CLICKS
300 000 WORDS



CAN SEARCH WITH LUCENE

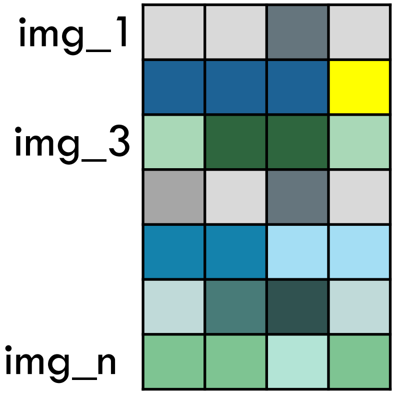
TEXT/CLICKS
500 DIMENSIONS



WITH LUCENE

DENSE VS SPARSE MATRIX

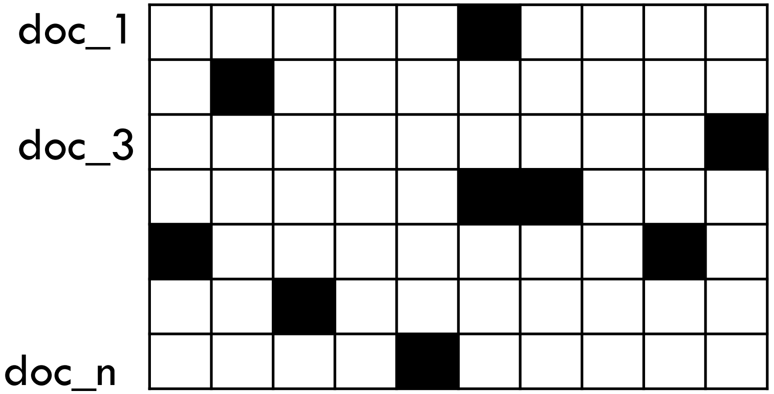
IMAGES
1000 PIXELS



CANNOT SEARCH WITH LUCENE

WITH RANDOM PROJECTIONS

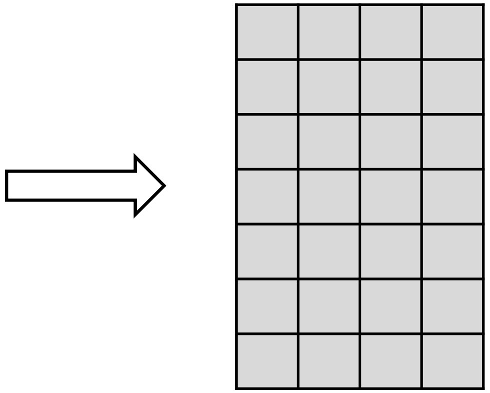
TEXT/CLICKS
300 000 WORDS



CAN SEARCH WITH LUCENE

WITH RANDOM PROJECTIONS

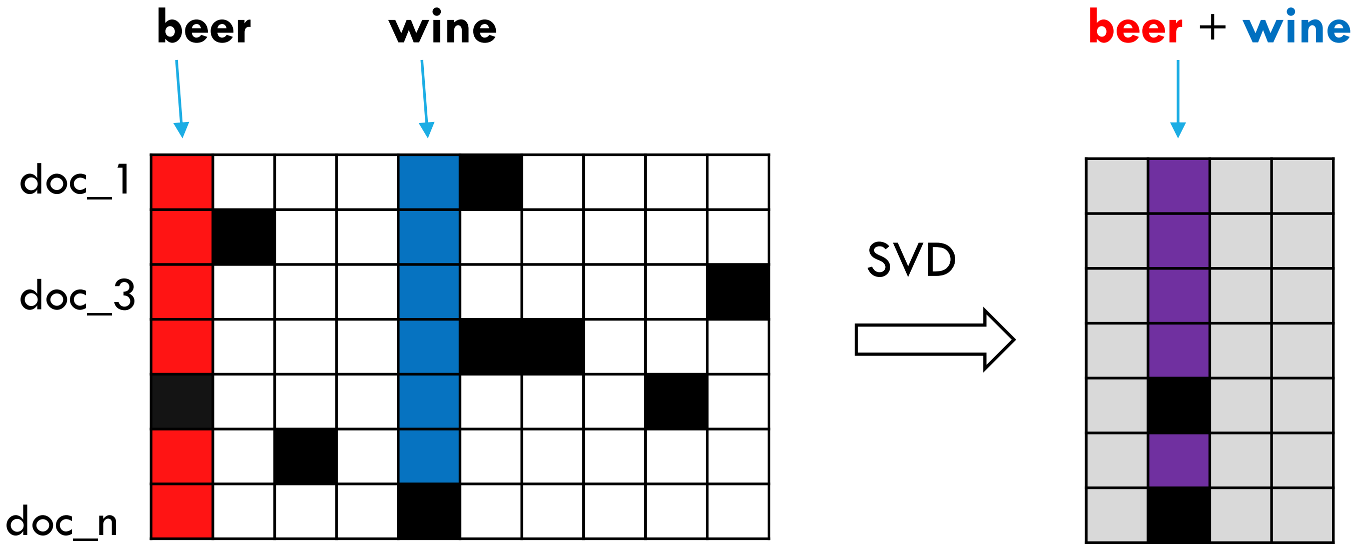
TEXT/CLICKS
500 DIMENSIONS



WITH LUCENE

WITH RANDOM PROJECTIONS

DIMENSIONALITY REDUCTION

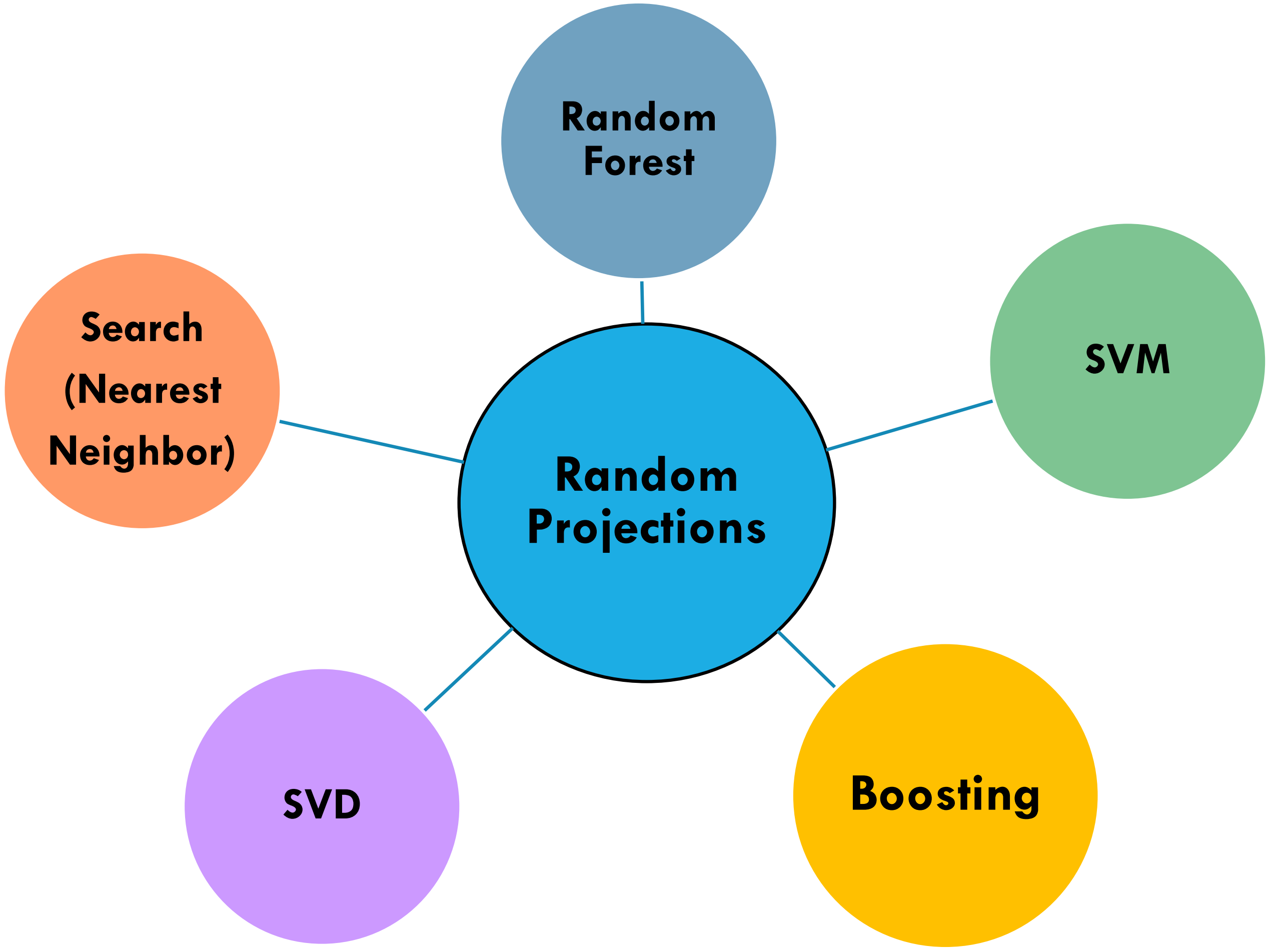


300 000 dimensions

500 dimensions

DIMENSIONALITY REDUCTION = PATTERN DISCOVERY

APPLICATIONS



RANDOM PROJECTIONS IN INDUSTRY

- **Spotify** for music recommendations
  - https://github.com/spotify/annoy

- **Etsy** for user/product recommendations
  - "Style in the Long Tail: Discovering Unique Interests with Latent Variable Models in Large Scale Social E-commerce", Diane Hu, Rob Hall, Josh Attenberg
  - Referred to as Locality Sensitive Hashing

OUTLINE

1. High dimensional data (images, text, clicks)
2. Random Projections: Why? What? How?
3. Image Search Benchmark

HOW DOES IT WORK?

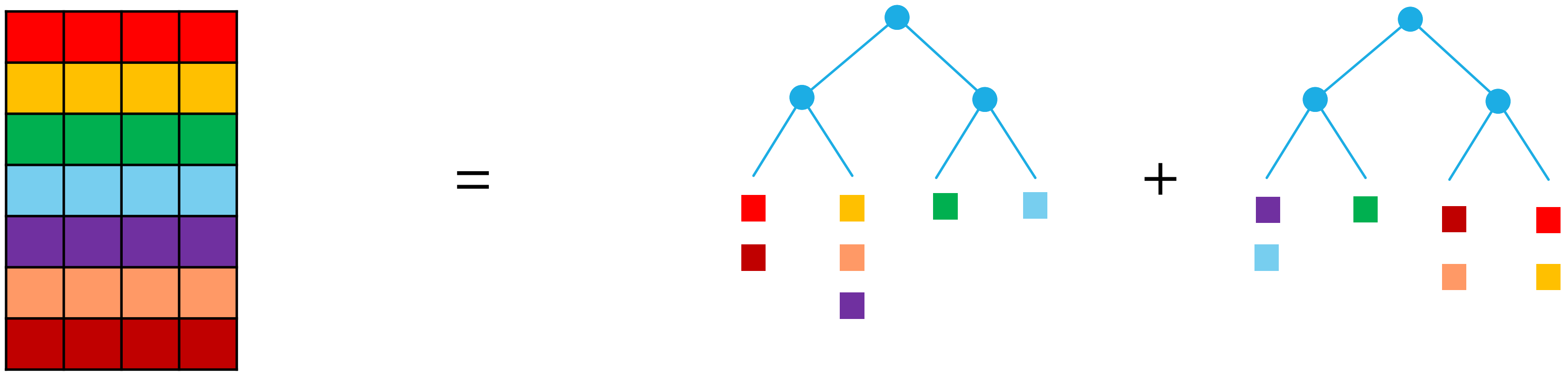
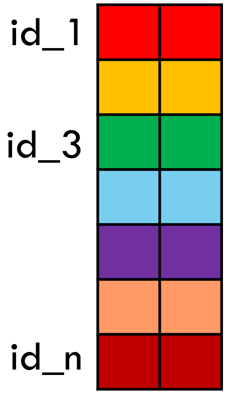


ILLUSTRATION ON ARTIFICIAL DATASET



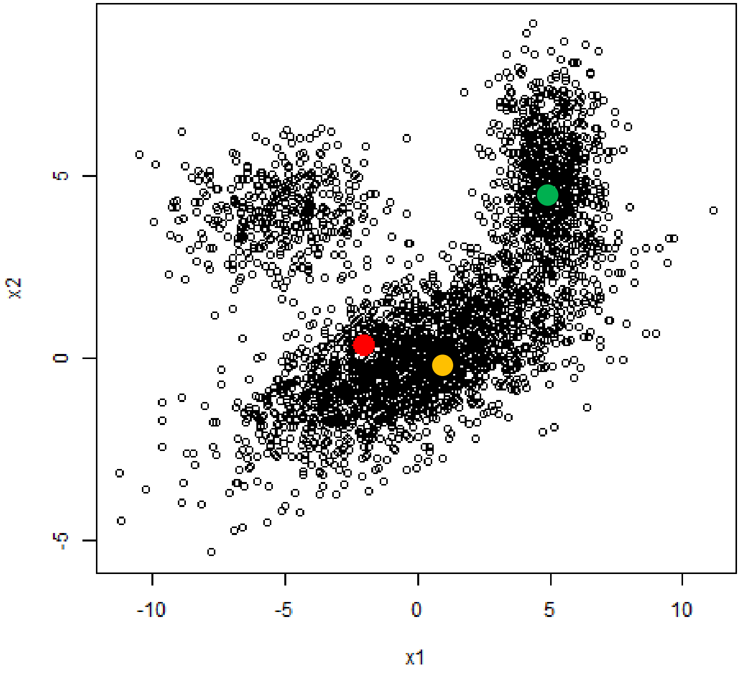


ILLUSTRATION ON ARTIFICIAL DATASET

Nearest neighbor problem: Find the closest point

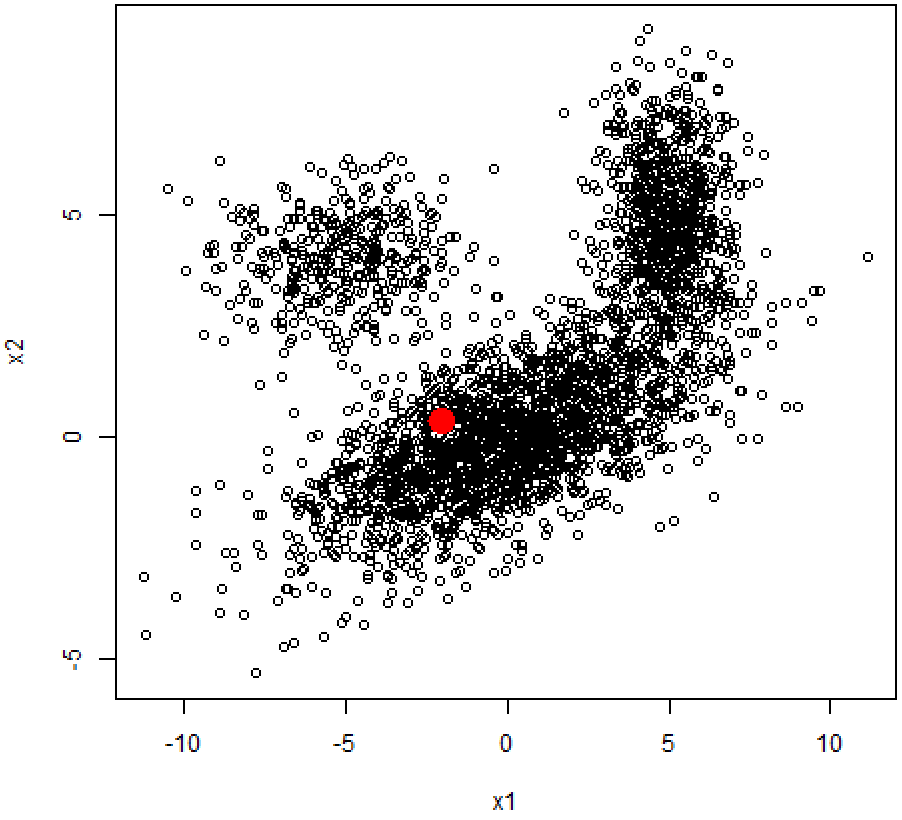
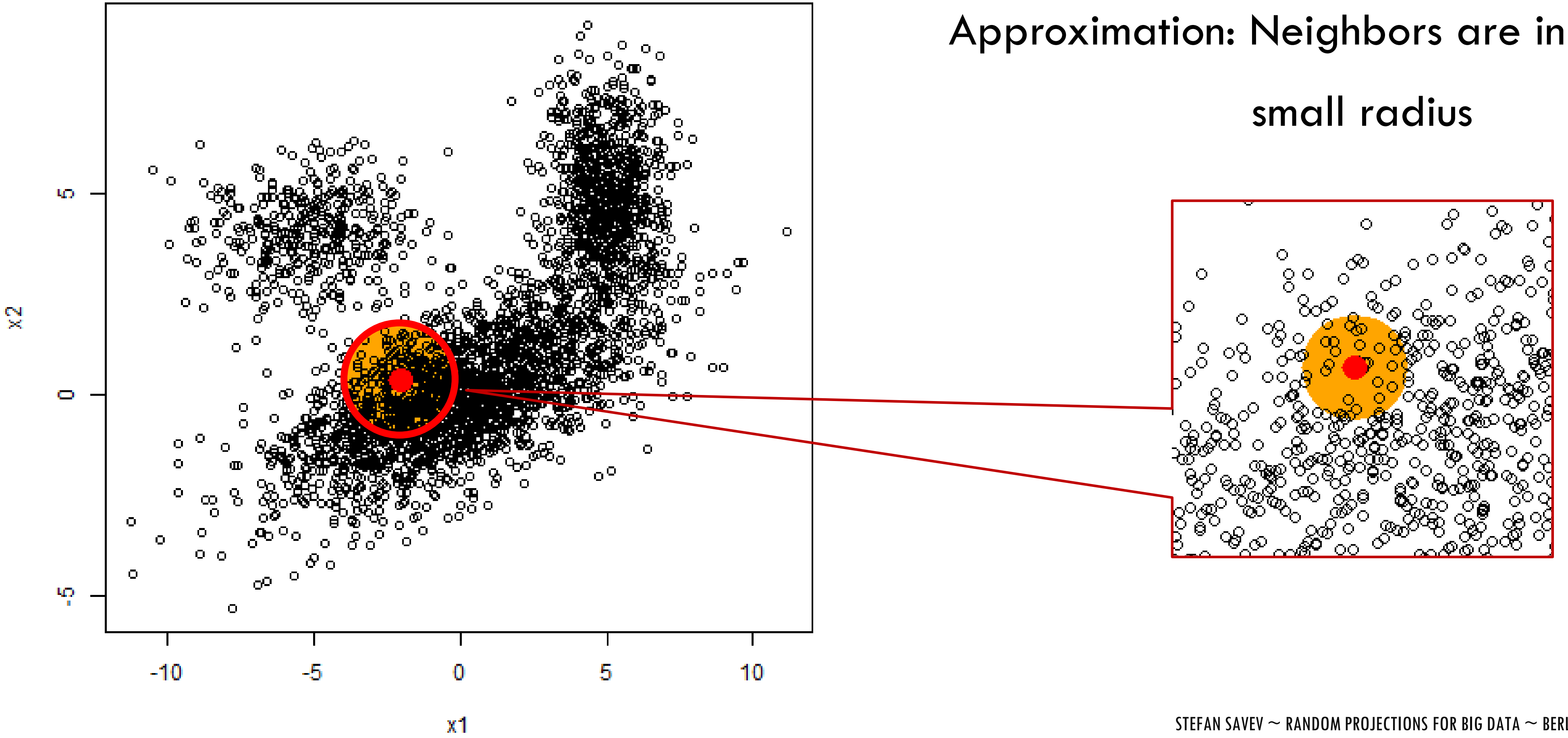
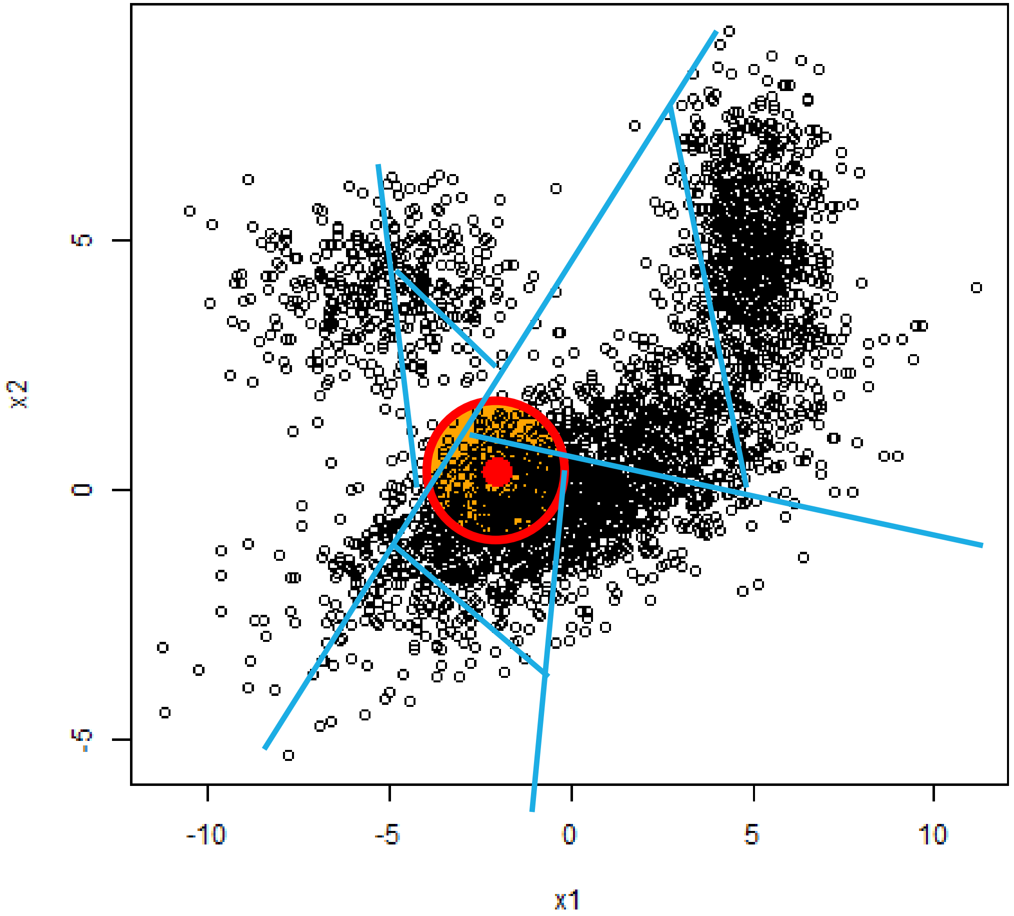


ILLUSTRATION ON ARTIFICIAL DATASET



Approximation: Neighbors are in a circle with small radius

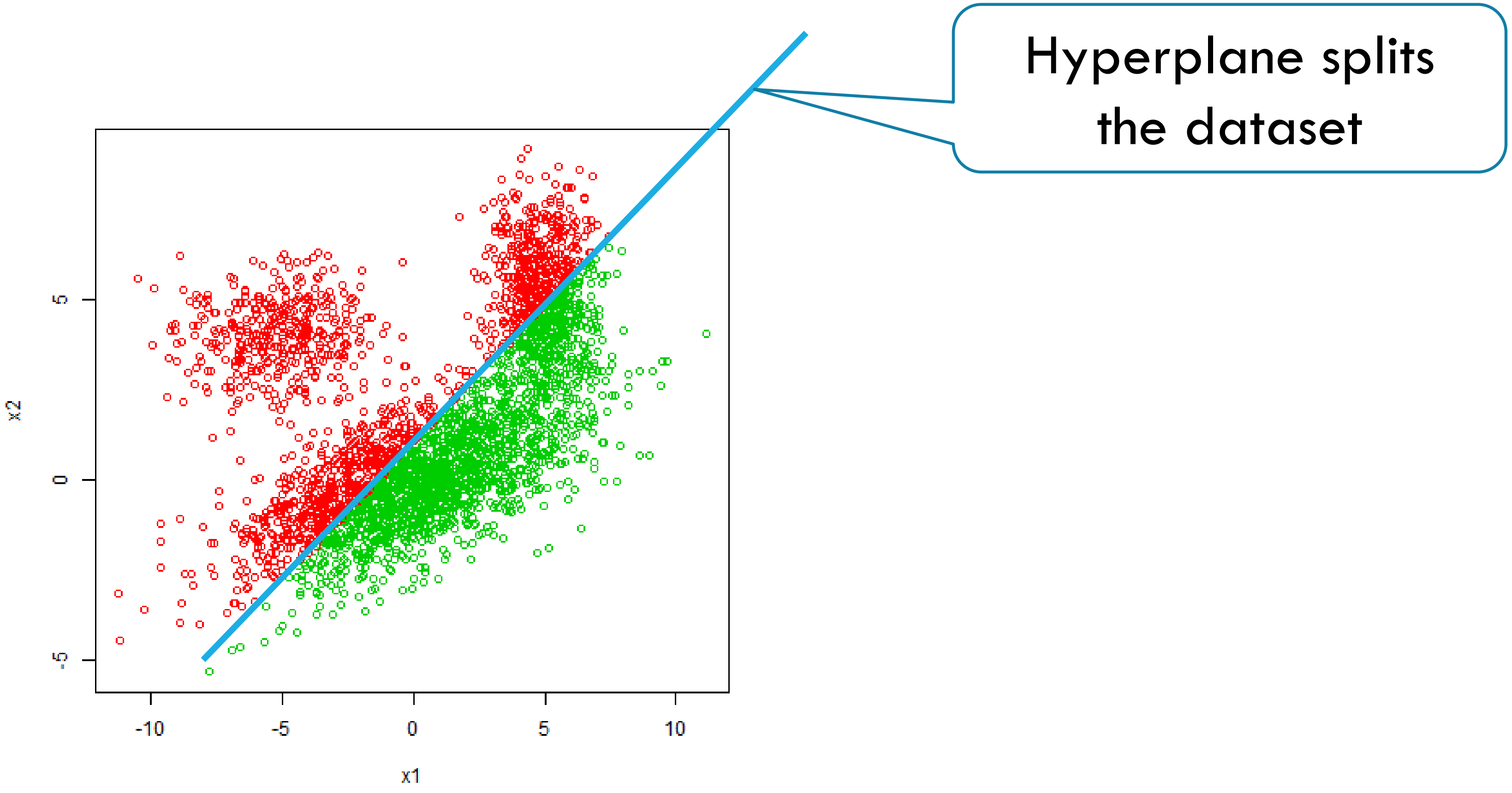
ILLUSTRATION ON ARTIFICIAL DATASET



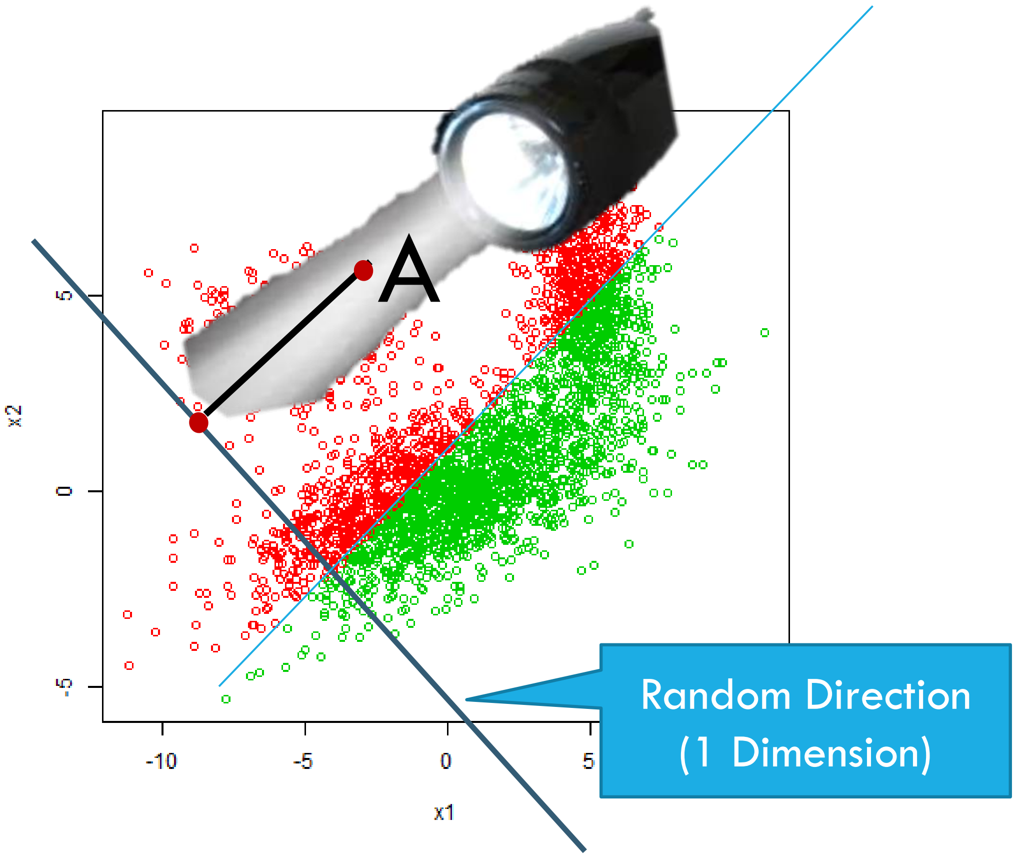
Core idea: random grids

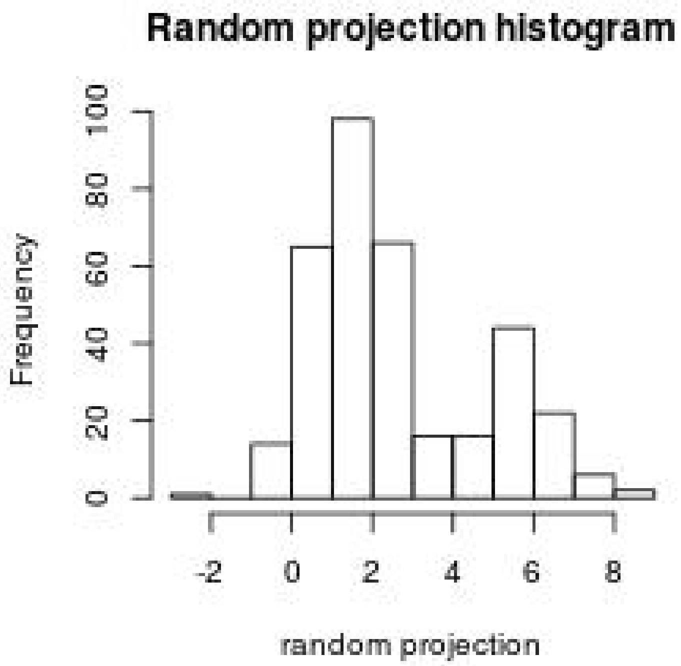
- Dynamic grids (variably sized)
- Random = cheap

PARTITION BY RANDOM SPLITS

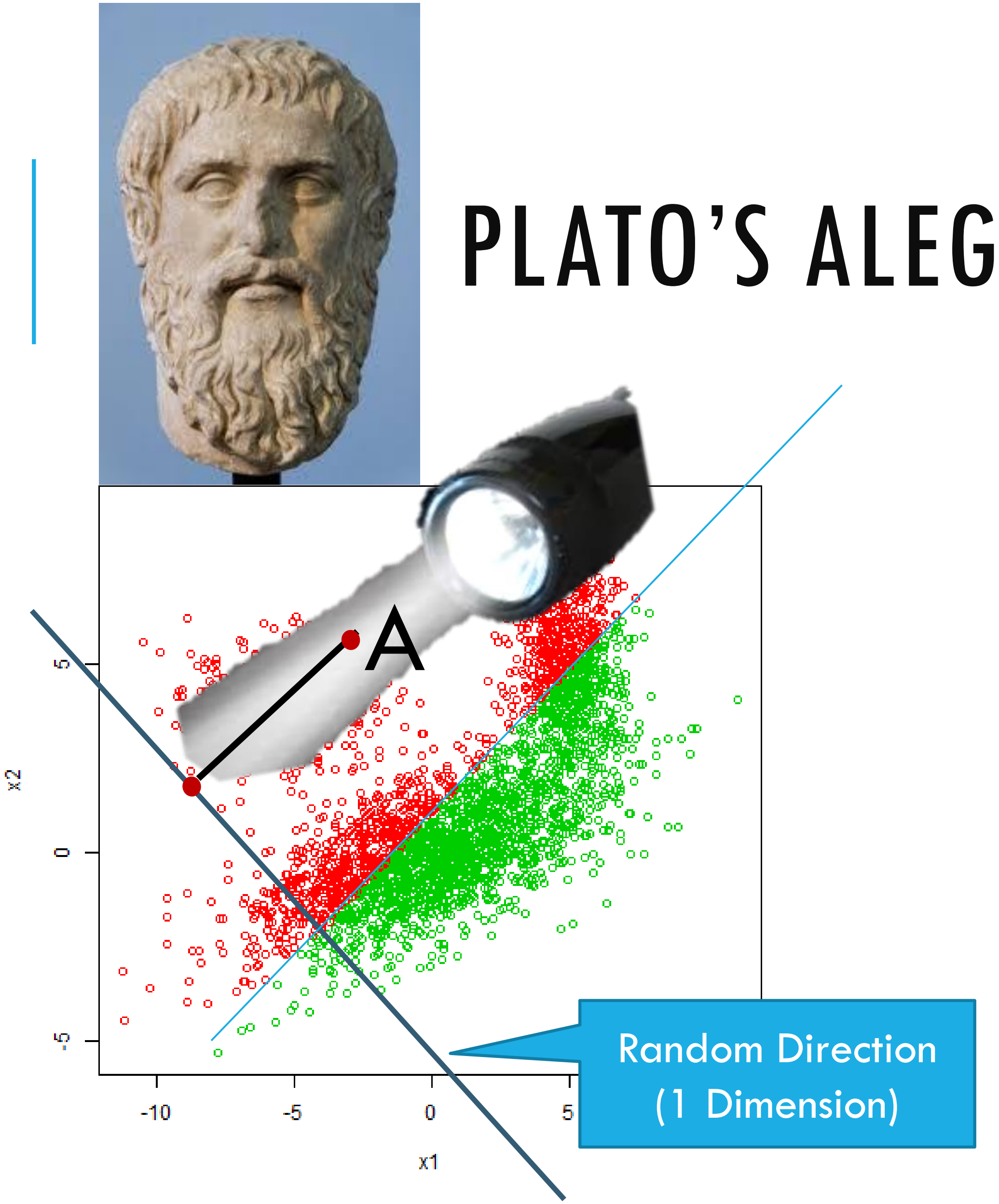


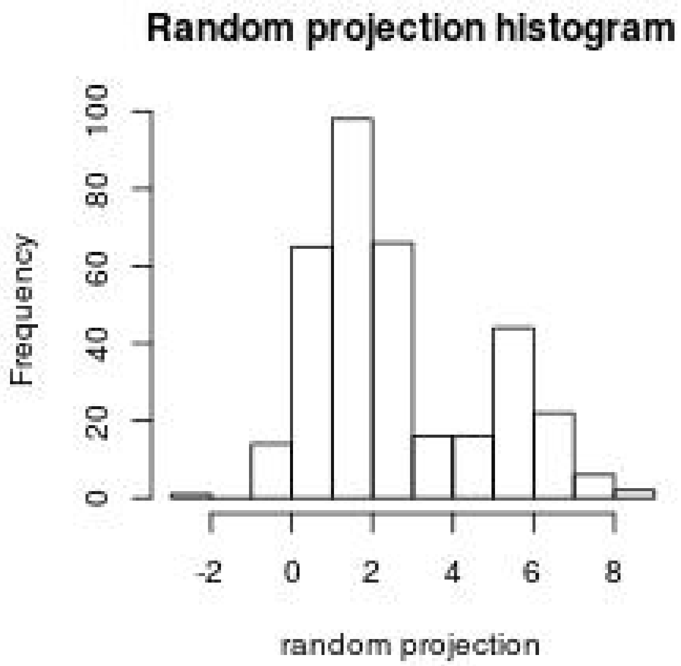
PROJECTION = ONE DIMENSIONAL VIEW OF DATASET



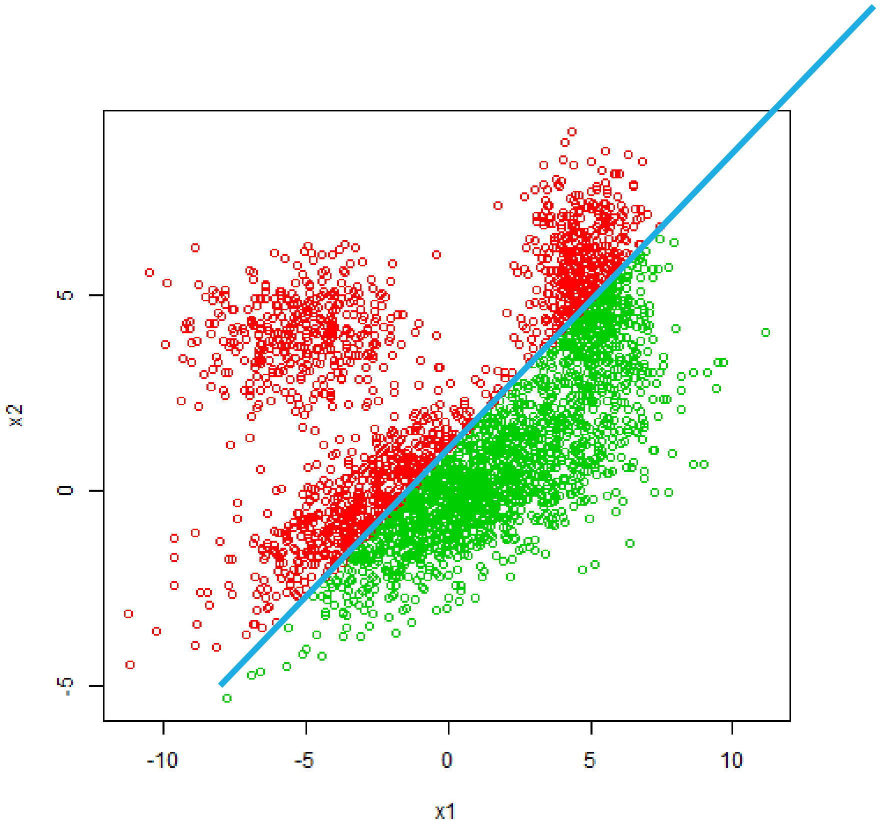


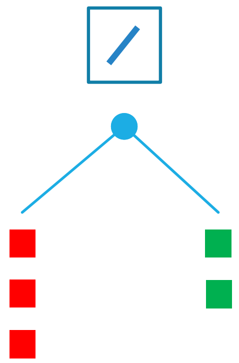
PLATO'S ALEGORY OF THE CAVE



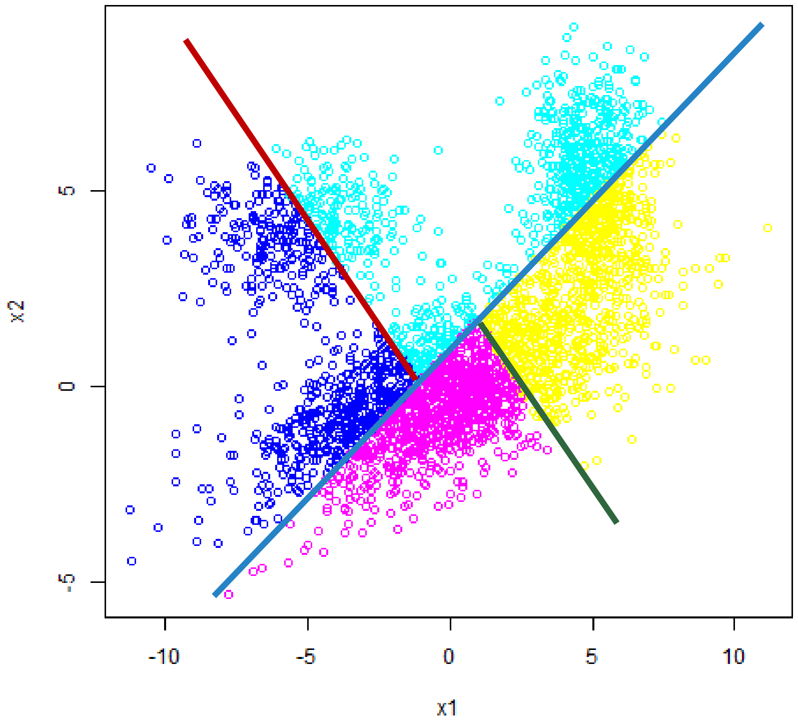


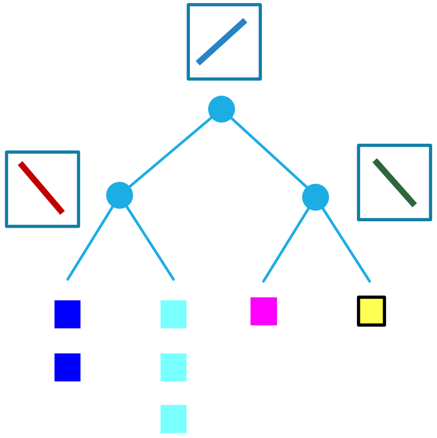
PARTITIONING BY PROJECTING ON RANDOM LINES



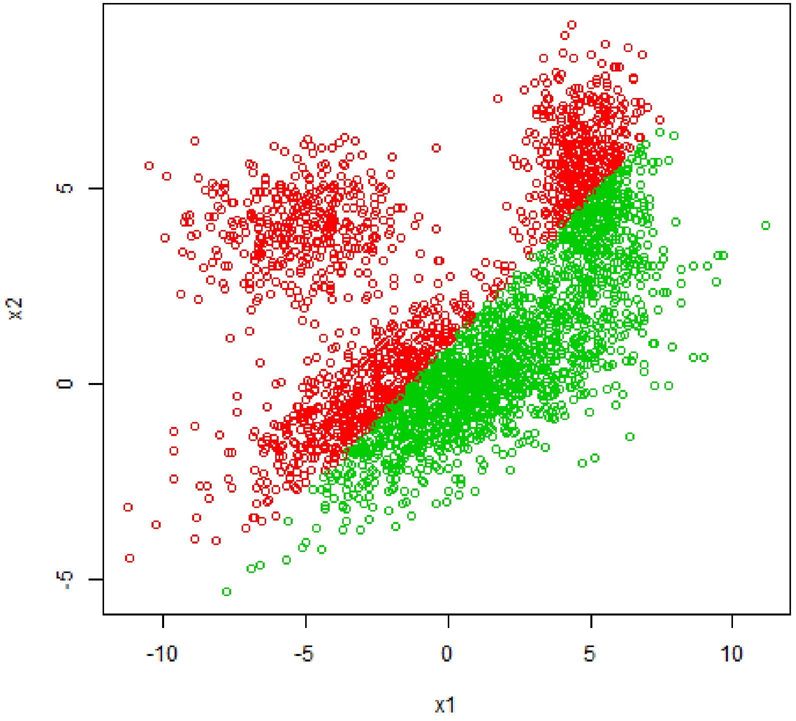


PARTITIONING BY PROJECTING ON RANDOM LINES

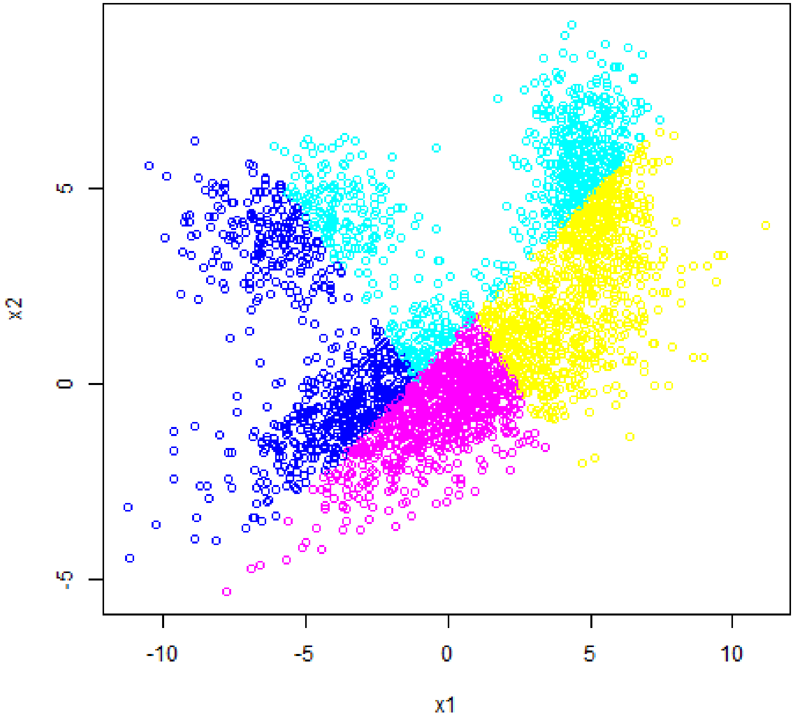




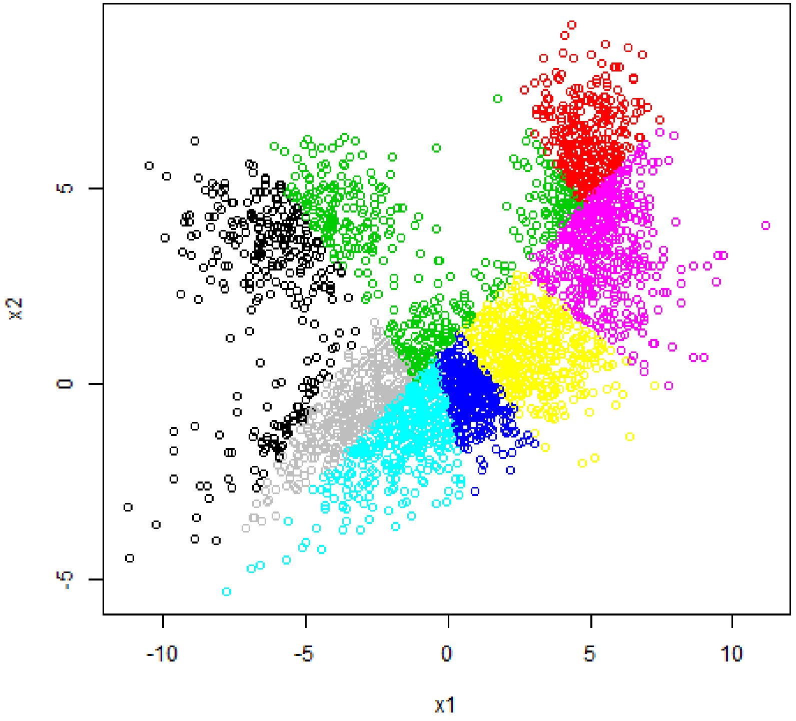
ALGORITHM VISUALIZATION



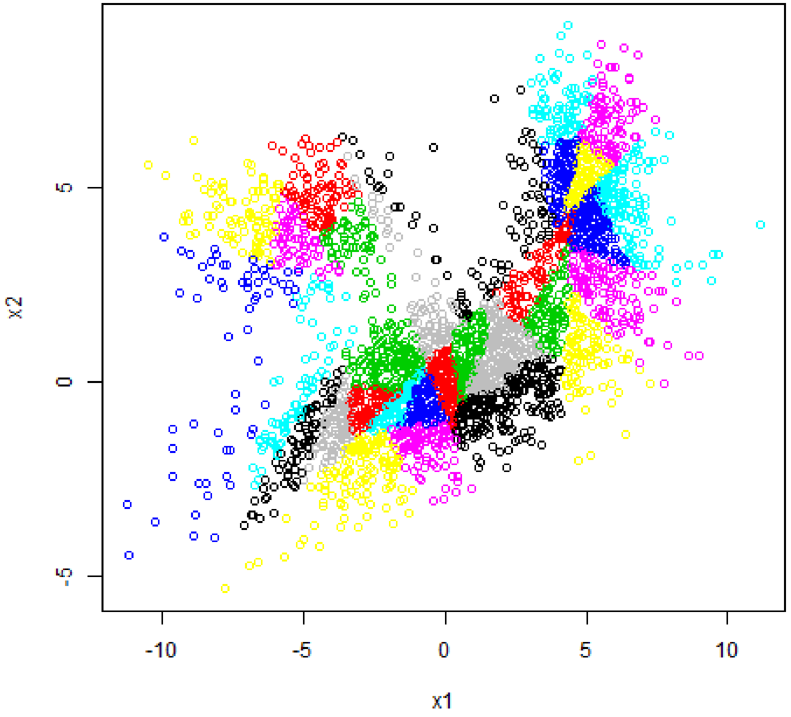
ALGORITHM VISUALIZATION



ALGORITHM VISUALIZATION



ALGORITHM VISUALIZATION



ALGORITHM VISUALIZATION

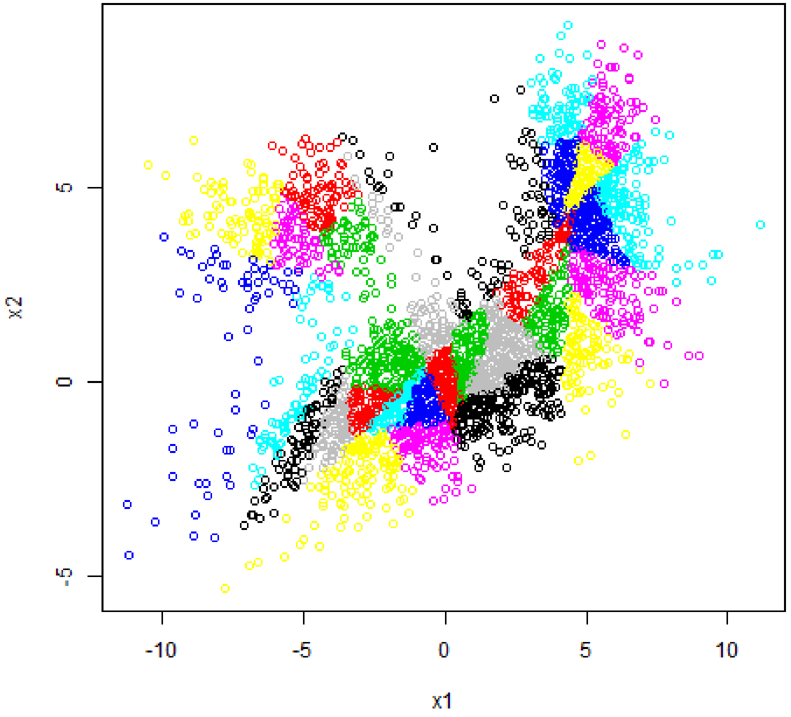
When two points are close,

they are likely to end up

in the same partition,

even under

random partitioning



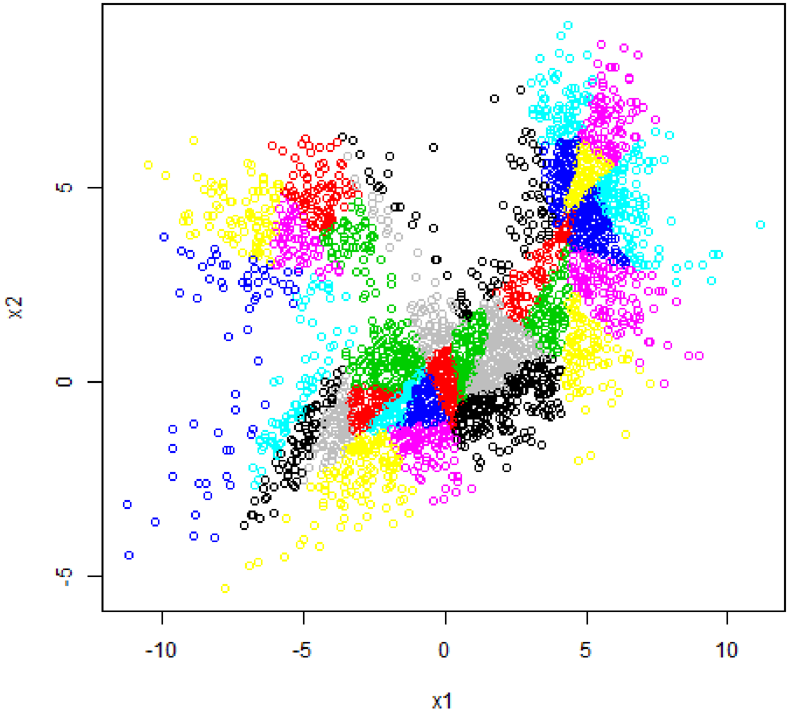
ALGORITHM VISUALIZATION

When two points are close,

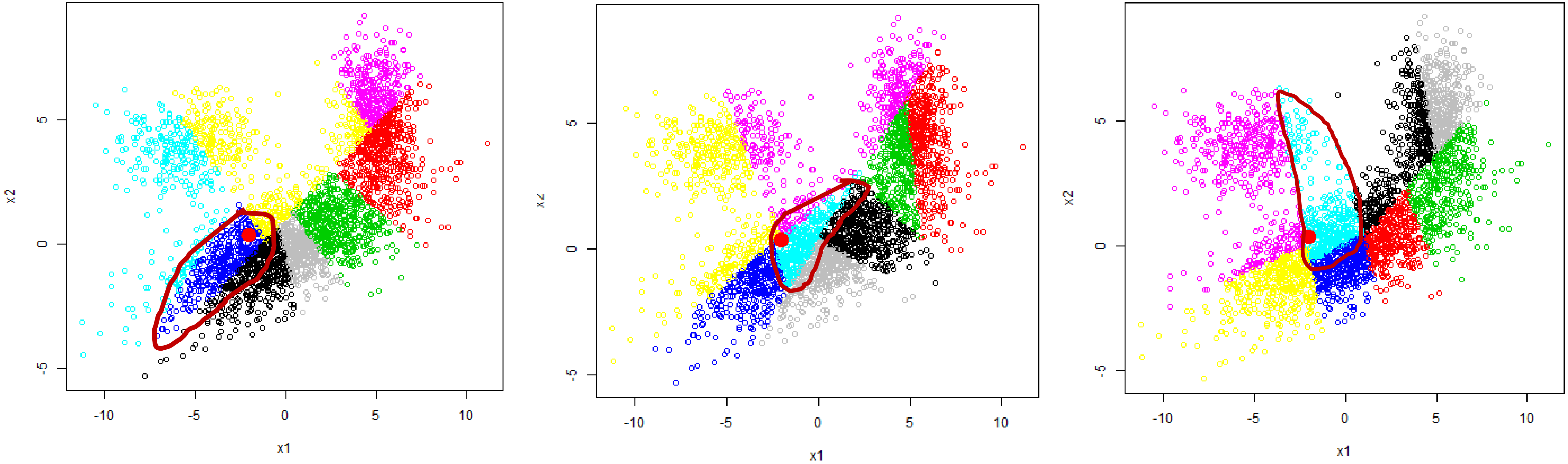
they are **LIKELY** to end up in the same partition,

even under

random partitioning

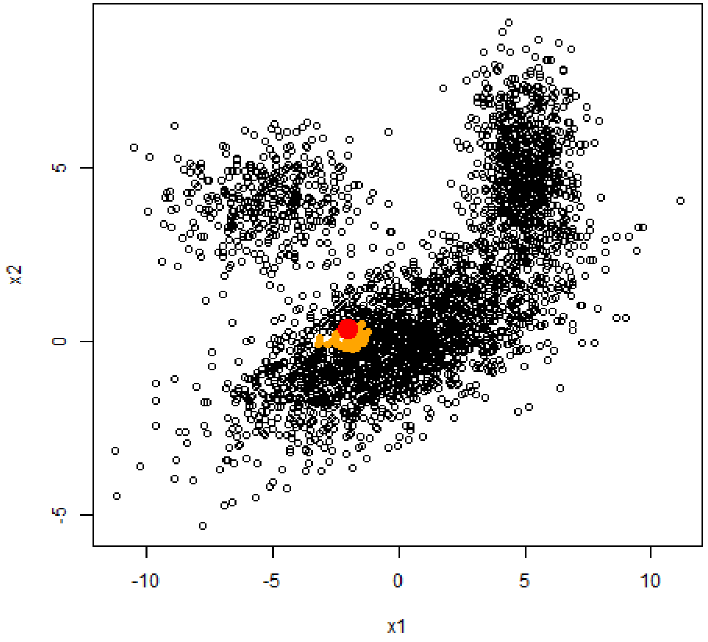


MULTIPLE TREES

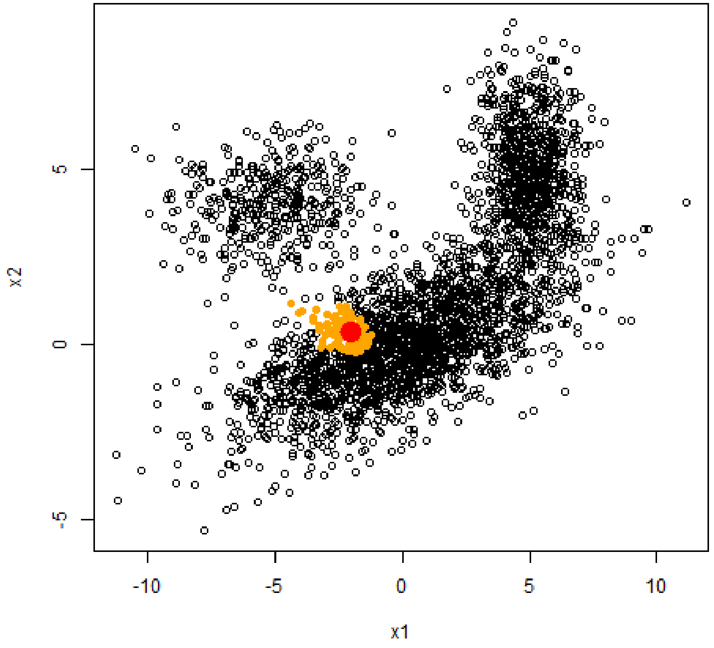


Random tree partitioning is noisy. Build more trees.

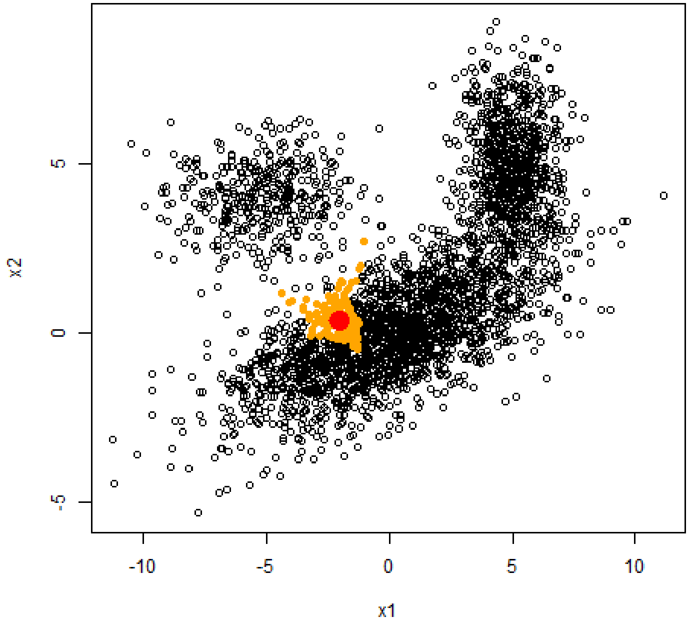
ADDING MORE TREES



Tree 1

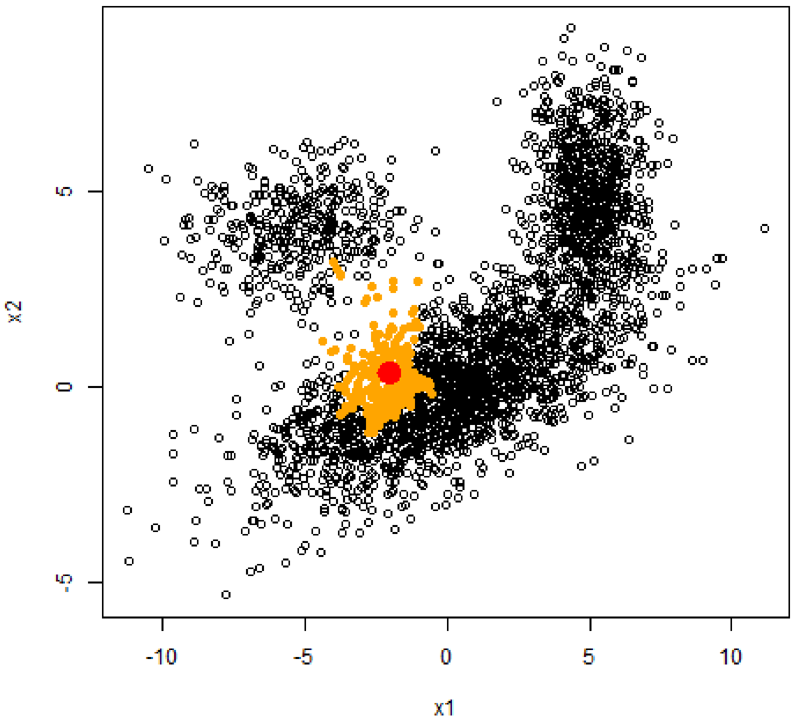


Trees 1 and 2

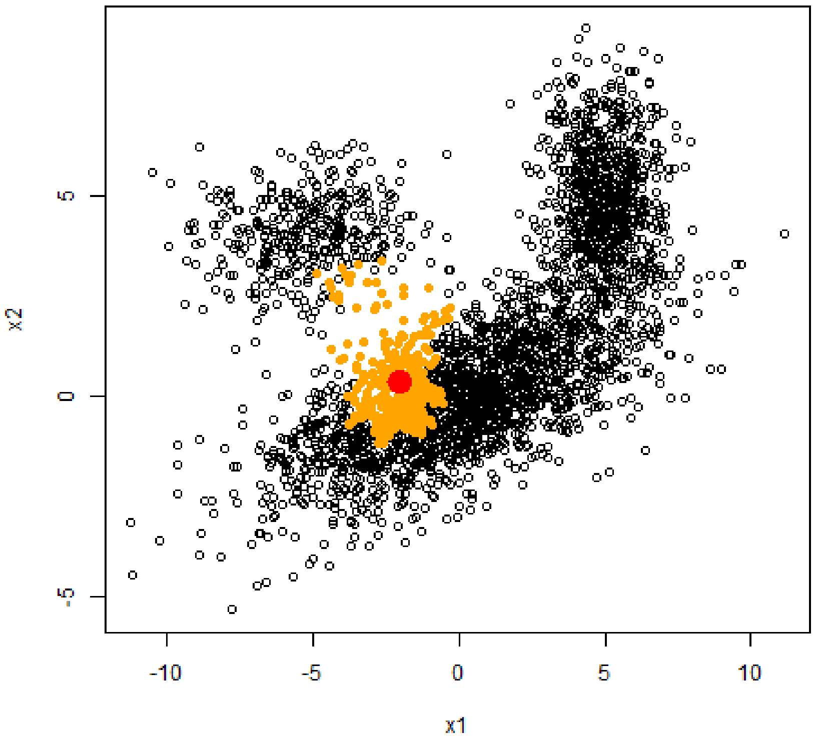


Trees 1, 2 and 3

ADDING MORE TREES



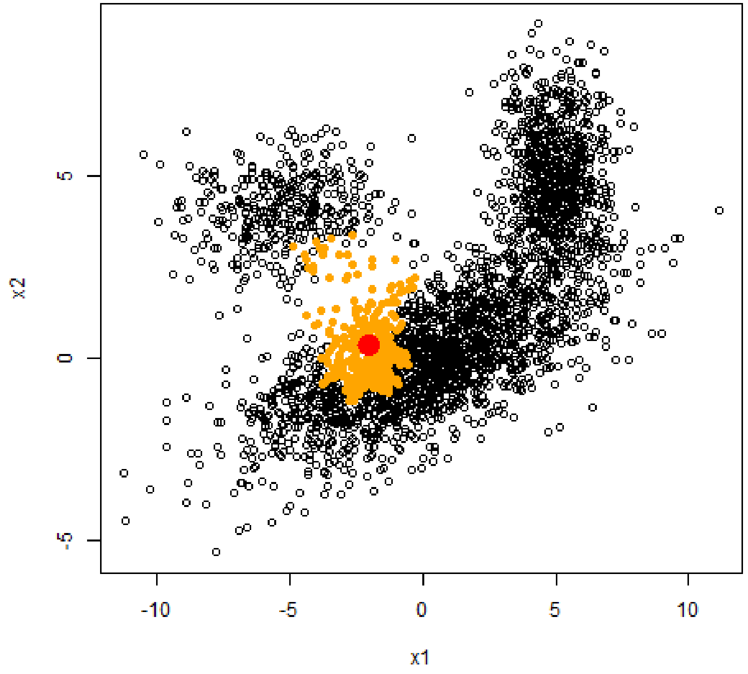
Trees 1 - 50



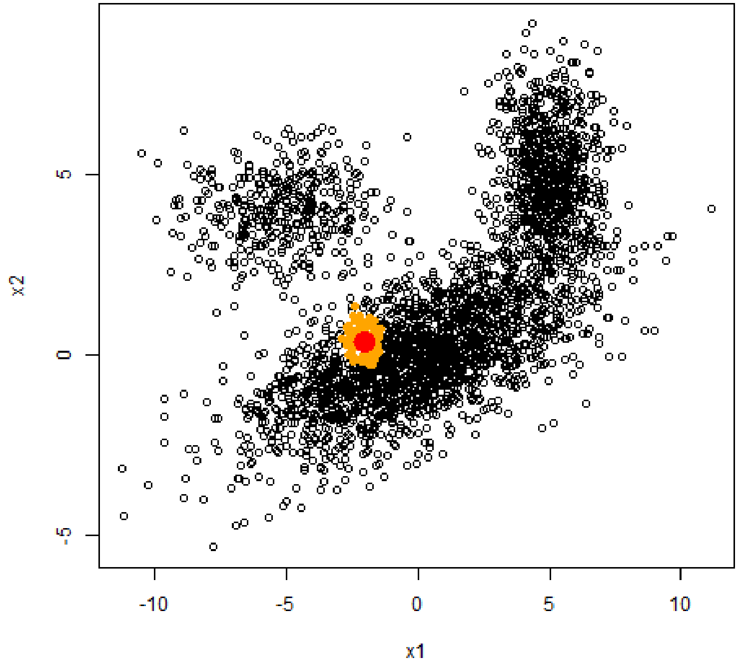
Trees 1 - 100

CONSTRAINING THE NEAREST NEIGHBOR REGION

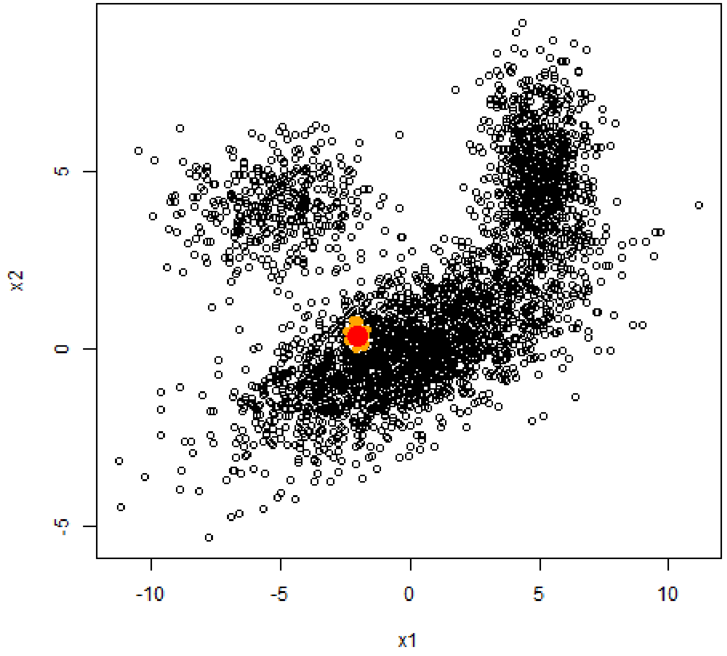
THRESHOLD = In how many trees a “NN candidate” appears



Threshold = 1



Threshold = 20



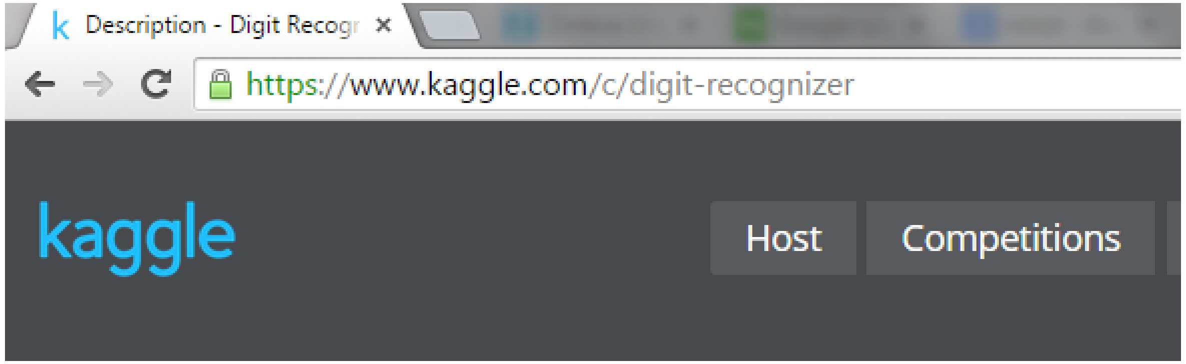
Threshold = 60

“WE’VE GOT HIM”

OUTLINE

1. High dimensional data (images, text, clicks)
2. Random Projections: Why? What? How?
3. Image Search Benchmark

IMAGE SEARCH FOR PREDICTION



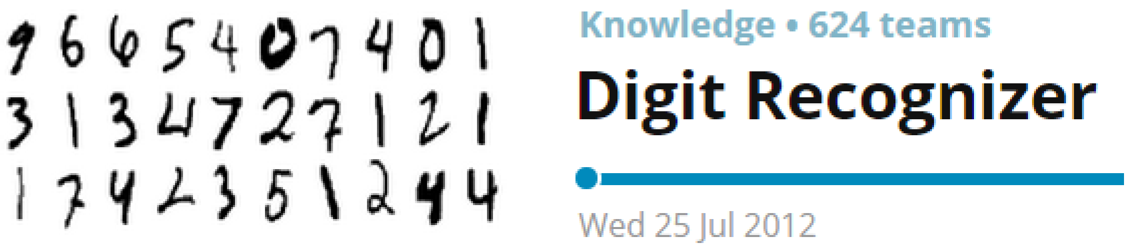
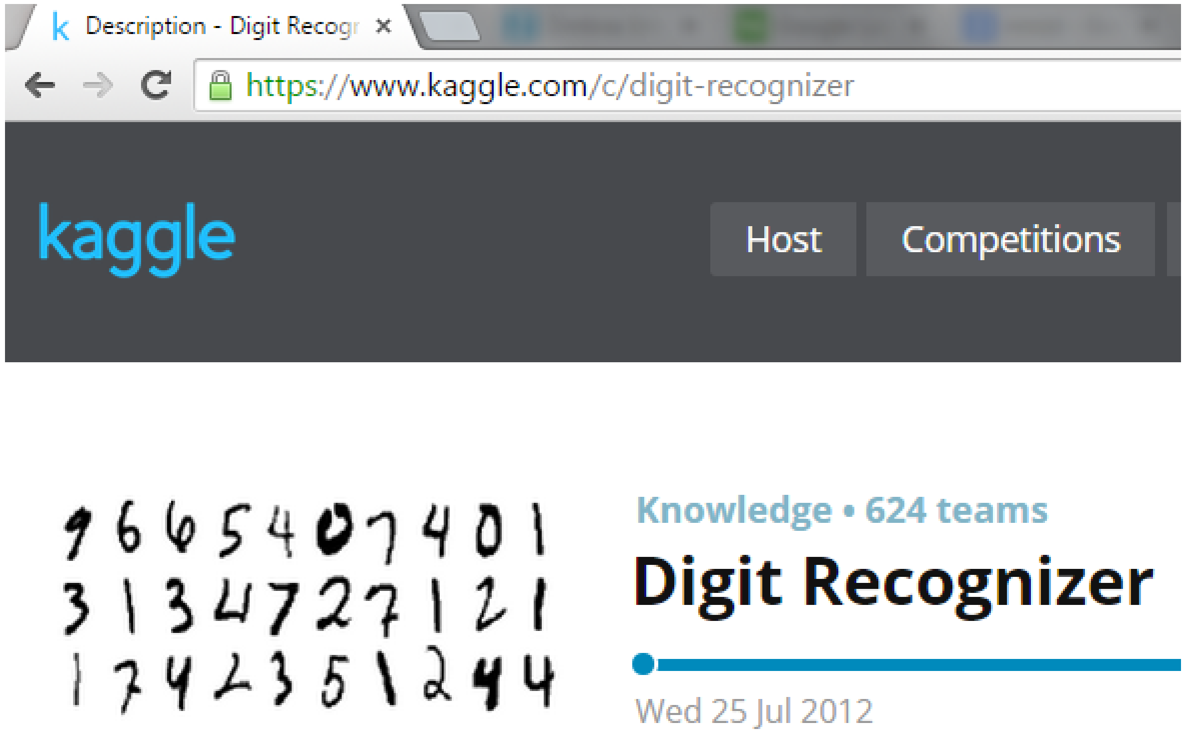
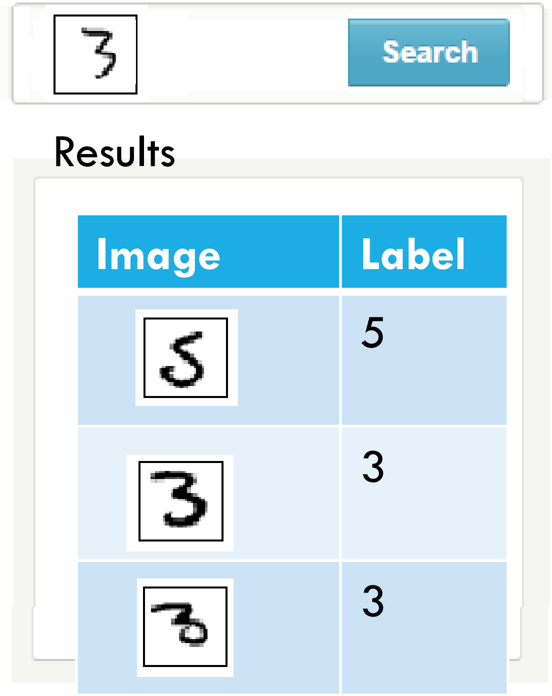


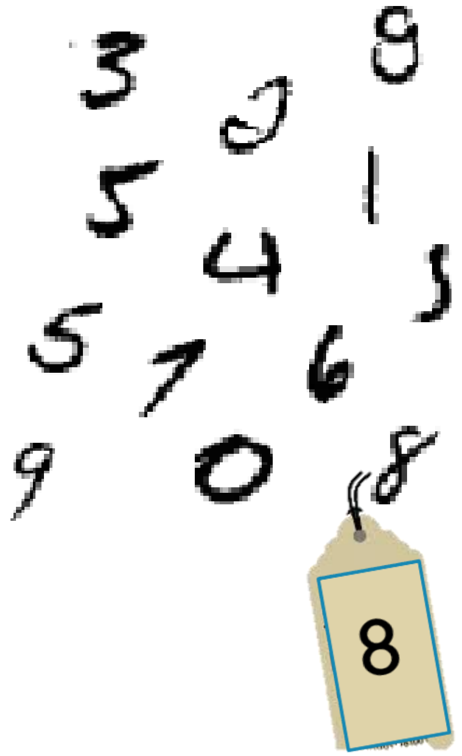
IMAGE SEARCH FOR PREDICTION





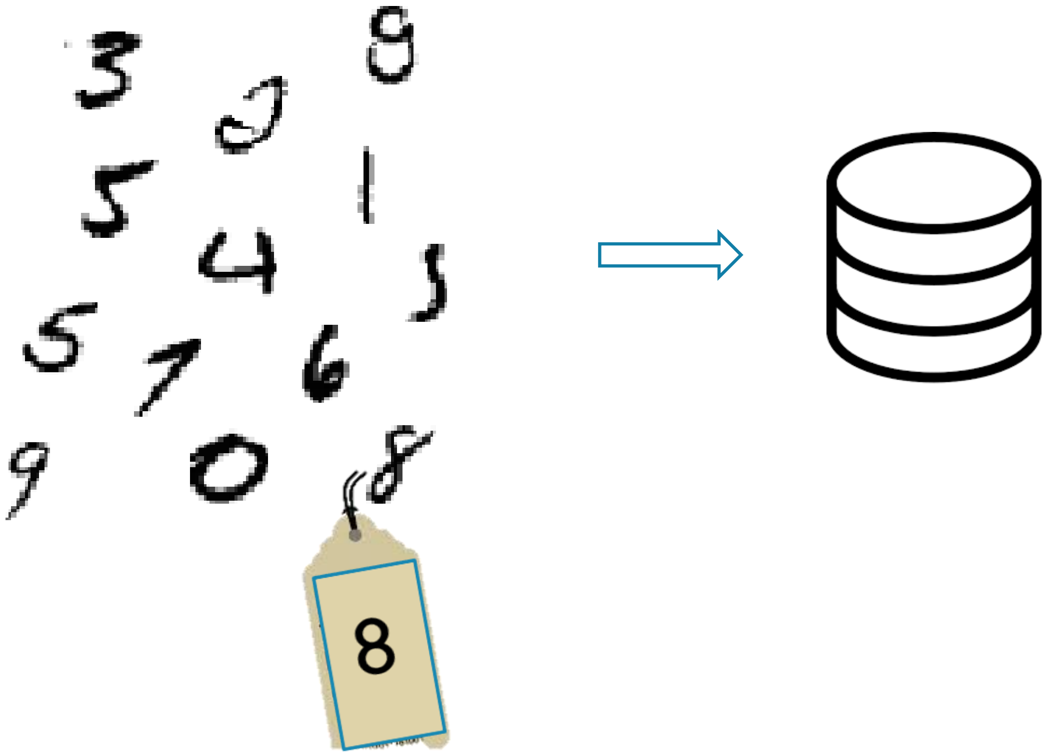
STAGE 1: INDEXING

42 000 examples with labels
784 pixels per image



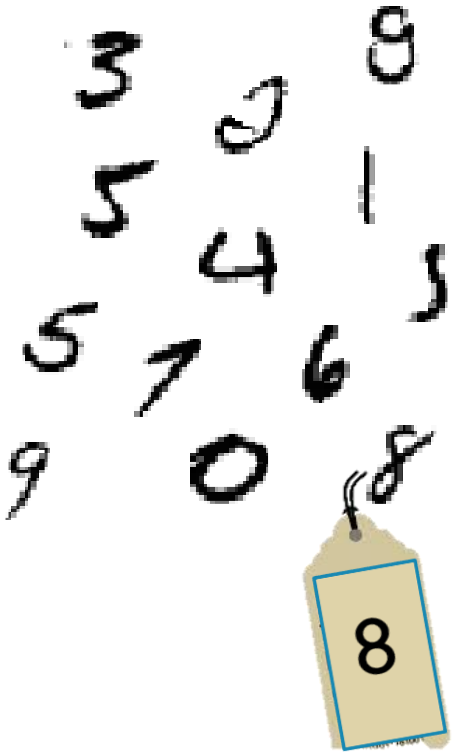
STAGE 1: INDEXING

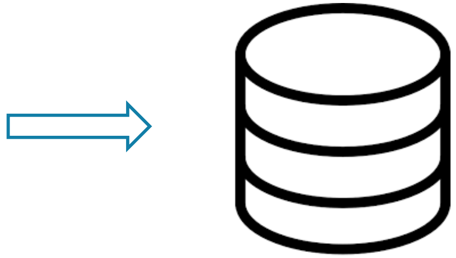
42 000 examples with labels
784 pixels per image



STAGE 1: INDEXING

42 000 examples with labels
784 pixels per image





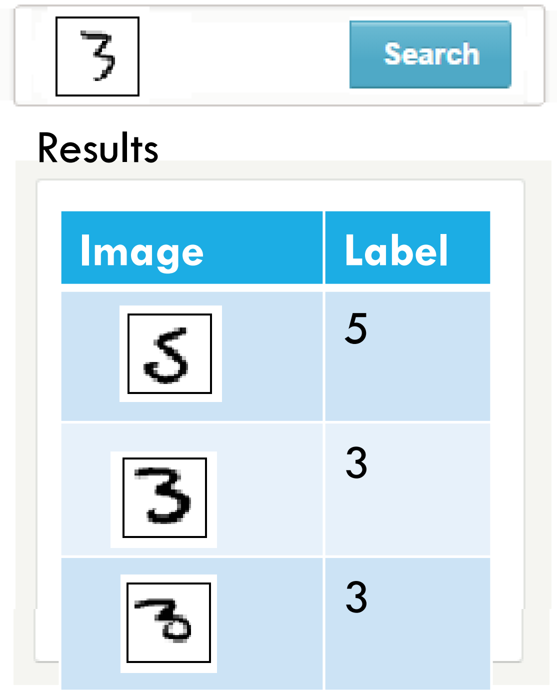
```
index = Index(...)

for image in training_examples:

    labels[image.id] = image.label
    index.add_item(image.id,
                   image.vector)

index.build(...)
```

STAGE 2: SEARCH TO PREDICT THE LABEL

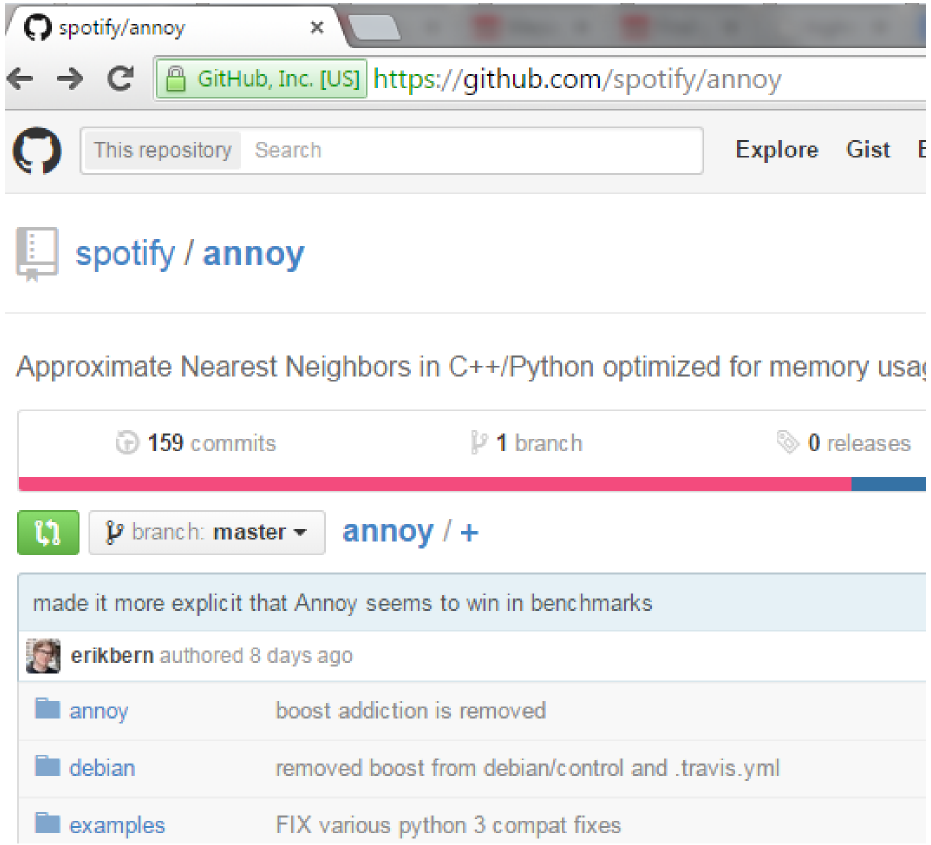



```
index.load('.../file.index')
...
image_vec = read_image(...)
nn = 100 #number of nearest neighbors
results = index.get_nns_by_vector(image_vec,nn)
top_result = results[0]
predicted_label = labels[top_result.id]
```

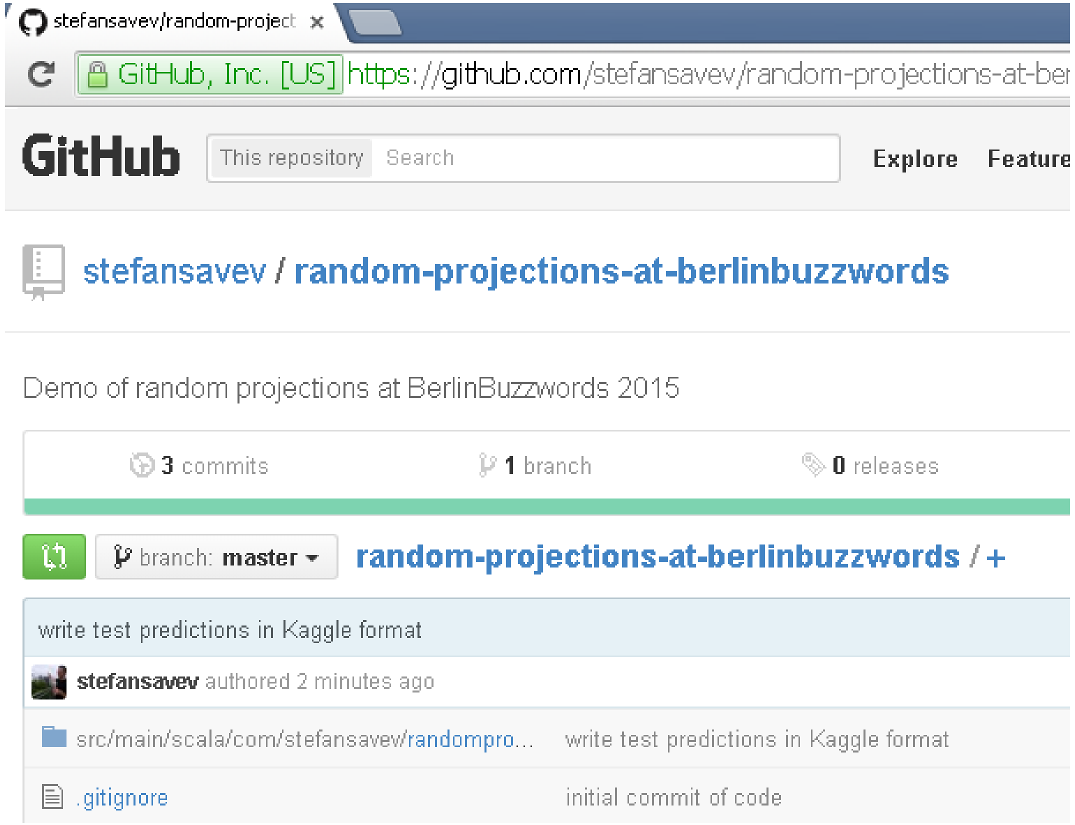
28 000 test examples

TOOLS

Spotify



stefansavev.com/randomtrees



BENCHMARK IMAGE SEARCH

DIMENSIONS	# TREES	# NN CANDIDATES	ACCURACY ON TEST	SEARCH TIME [ms/query]
784	10	1000-1010	97.0	4.85

BOTTLENECK

Reference: https://github.com/stefansavev/random-projections-talk/

BENCHMARK IMAGE SEARCH

	DIMENSIONS	# TREES	# NN CANDIDATES	ACCURACY ON TEST	SEARCH TIME [ms/query]
Spotify	784	10	1000-1010	97.0	4.85
Spotify	**100, after SVD**	10	1000-1010	**97.5**	**0.7**

Reference: https://github.com/stefansavev/random-projections-talk/

BENCHMARK IMAGE SEARCH

	DIMENSIONS	# TREES	# NN CANDIDATES	ACCURACY ON TEST	SEARCH TIME [ms/query]
Spotify	784	10	1000-1010	97.0	4.85
Spotify	**100, after SVD**	10	**1000-1010**	**97.5**	**0.7**

BOTTLENECK

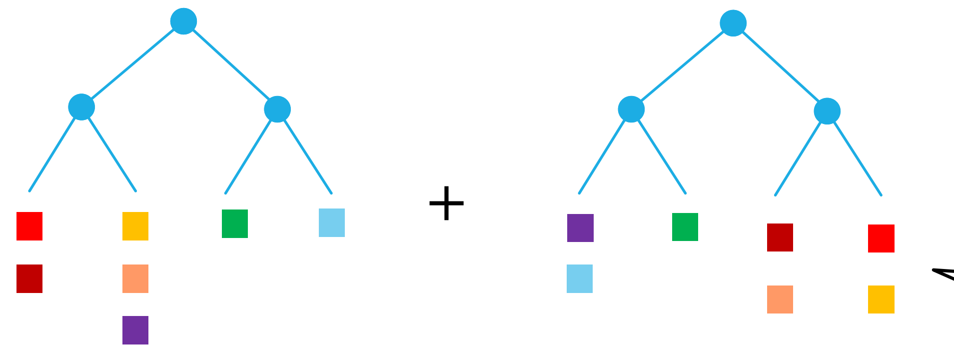
Reference: https://github.com/stefansavev/random-projections-talk/

BENCHMARK IMAGE SEARCH

	DIMENSIONS	# TREES	# NN CANDIDATES	ACCURACY ON TEST	SEARCH TIME [ms/query]
Spotify	784	10	1000-1010	97.0	4.85
Spotify	100, after SVD	10	1000-1010	97.5	0.7
stefansavev.com/randomtrees	100, after SVD	**40**	**180 (mean)**	97.5	**0.36**

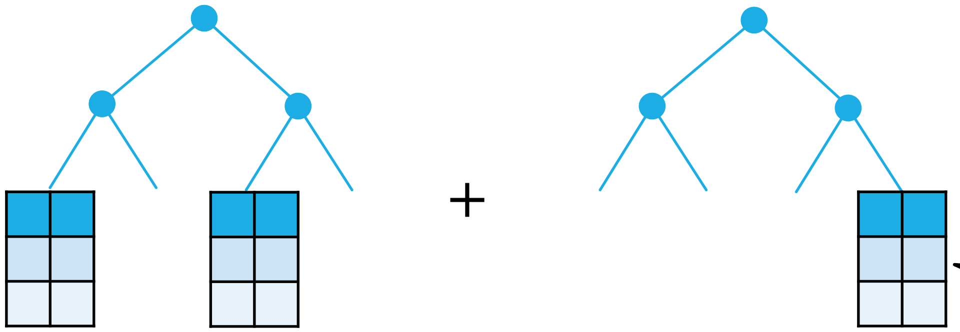
Reference: https://github.com/stefansavev/random-projections-talk/

SAME FOUNDATION FOR SEARCH AND MACHINE LEARNING



Data point id
32
56
…
89
…

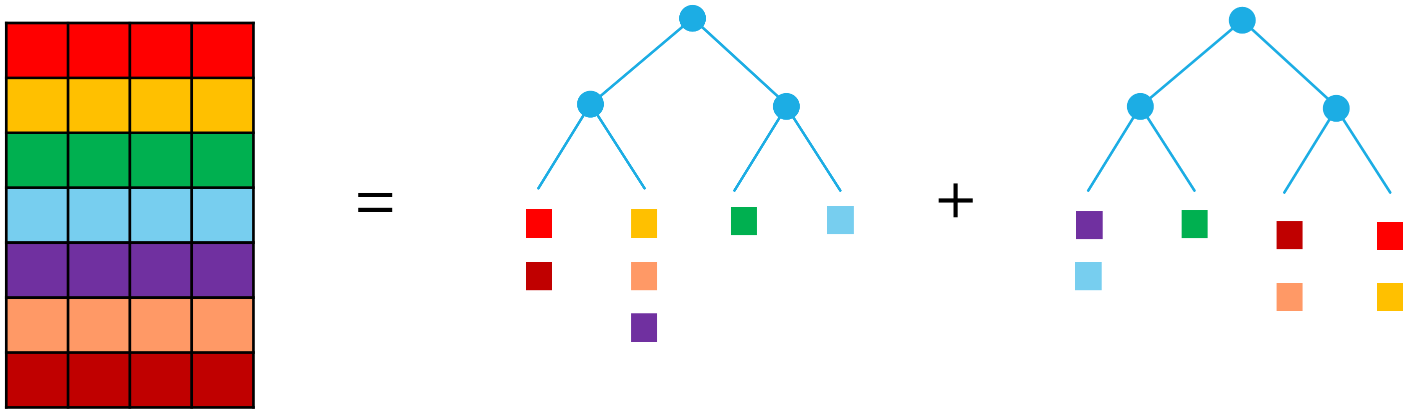
SAME FOUNDATION FOR SEARCH AND MACHINE LEARNING



Histogram

label	frequency
“0”	4
“1”	51
…	…
“7”	38
…	…

CONCLUSION



- Search in **Dense** Data
- **CHEAP** method to “explore” data; Makes algorithms FASTER (e.g. SVD, SVM)

THANK YOU!

Stefan Savev

Email: info@stefansavev.com

Blog: stefansavev.com

EXTRA SLIDES

REFERENCES

ADVANTAGES/DISADVANTAGES DIMENSIONALITY REDUCTION

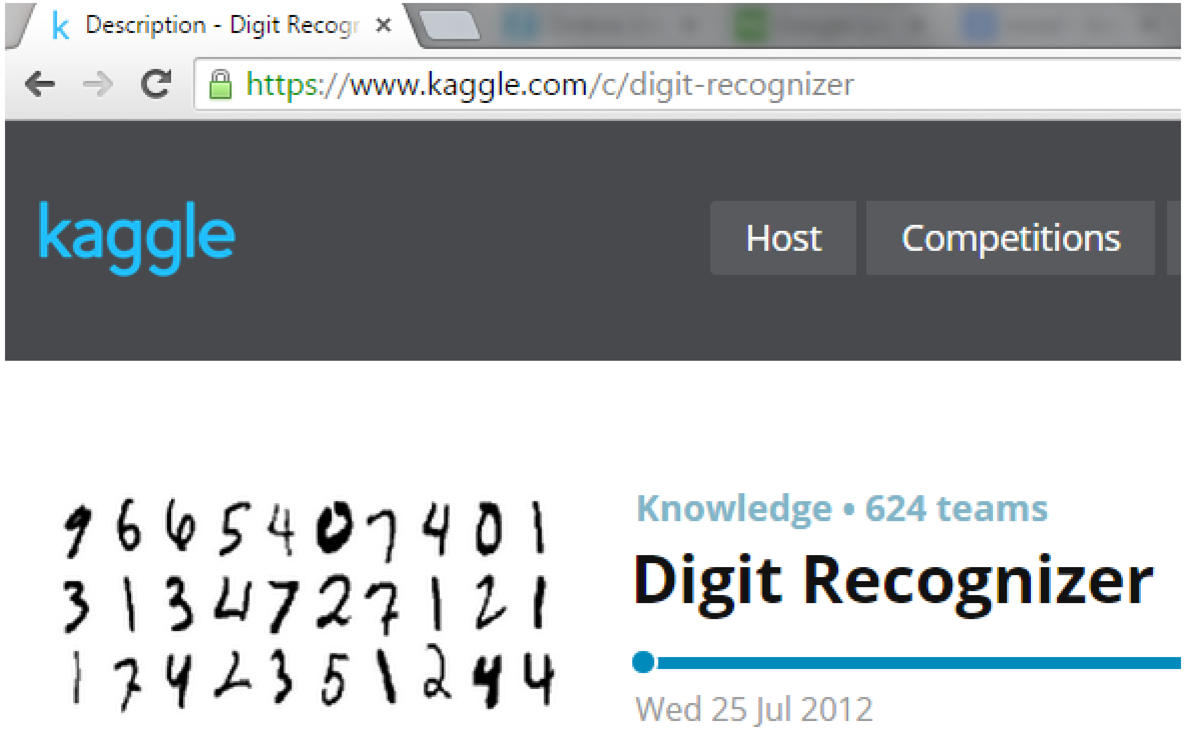
- Advantages
  - “Semantic Similarity”
  - “Concentrate the signal”
- Disadvantages
  - No exact matches possible
  - Adds noise

RANDOM PROJECTIONS AS HASH FUNCTIONS

$$
\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}
*
\begin{bmatrix} -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix}
=
\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}
$$

$1 + 1 - 1 - 1 - 1 = -1$

EXAMPLE-BASED IMAGE SEARCH



- MNIST image dataset of digits
- Images are digits from 0 to 9
- 42 000 images for training
- 28 000 images for test
- 28 x 28 pixels with values from 0 to 255
- 784 (=28*28) features
- Dataset is already preprocessed
- Goal is to predict the label of an image

SOME IMPLEMENTATION TRICKS

- sparse random projections → combine not all dimension but a small number of them
- apply dimensionality reduction first
- pick better random projections → the projected histogram is wide
- project on multiple lines simultaneously → Hadamard matrix trick
- reuse random projections via recombination
- cut out a "ball" from the densest region
- PCA may help
- use the neighbors of neighbors of a point as additional similarity candidates
- advanced search inside the trees with backtracking (KD-tree style)
- generate multiple versions of query/training data (i.e. for image data translate or rotate the image)

[Link to Blog]

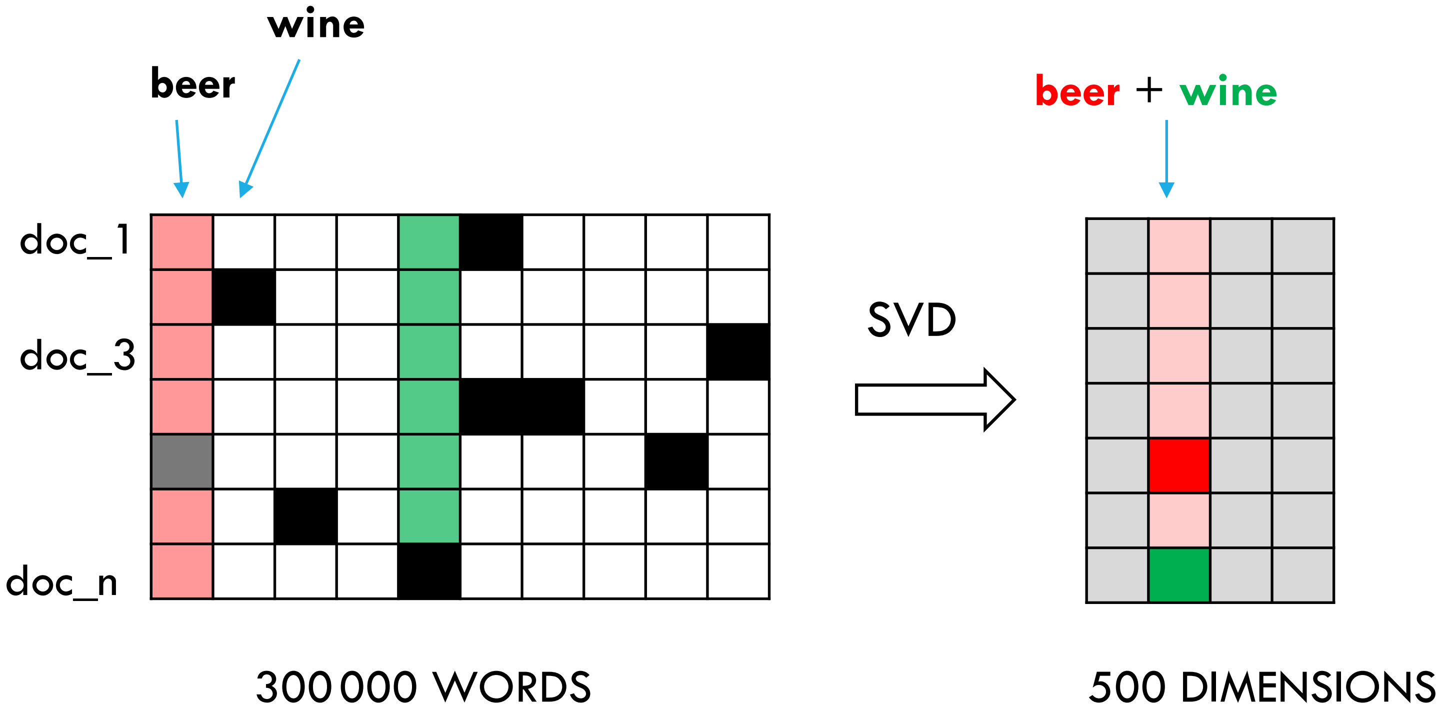
PROJECTION

- Dot product (cosine, similarity)
- View of the dataset
- Overlap with Pattern
- “Geometry” of Search
- Foundation for Search and Machine Learning

WHY RANDOM?

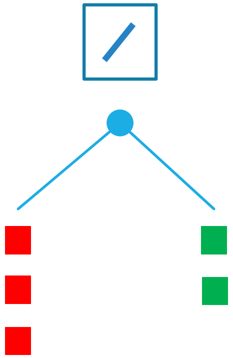
- Cheap (replace optimization with randomization); Simply build more trees
- If we build the perfect tree, it's just one tree. We need more and DIVERSE trees
- In high dimensions all algorithms will have problems, so do the cheapest
- With a few points/dimensions random is not the best choice

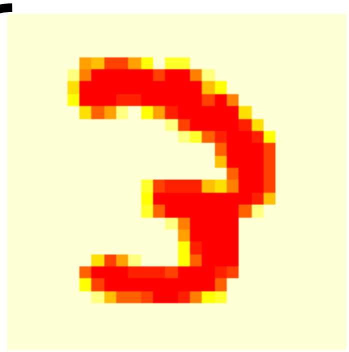
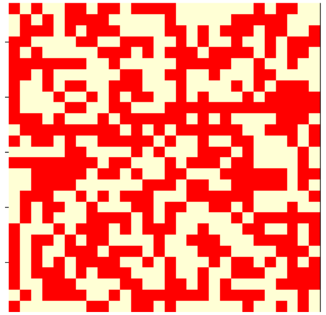
DIMENSIONALITY REDUCTION

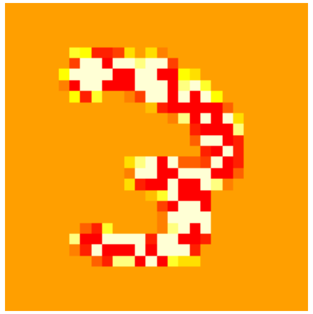


MIXING THE COLUMNS
OF THE ORIGINAL MATRIX TO MAXIMIZE SIGNAL TO NOISE

PROJECTION = OVERLAP WITH A PATTERN

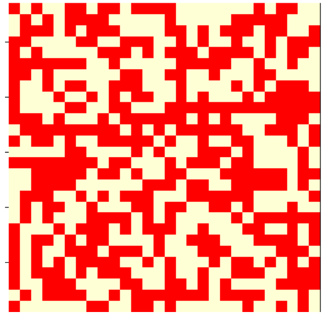
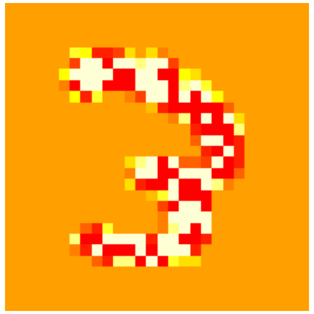


OVERLAP ( , )

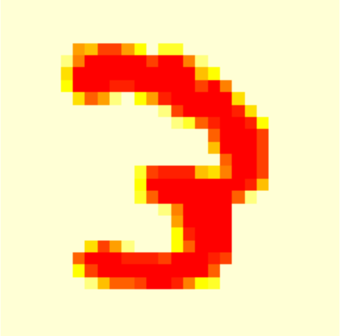
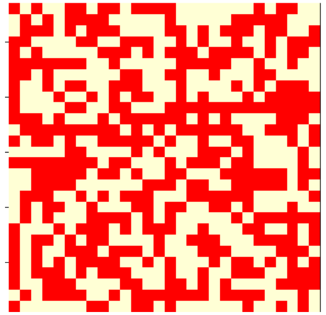
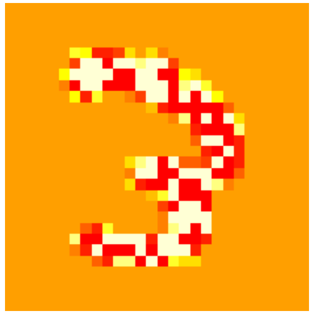
= SUM  =

- **1, more red**
- **-1, more green**

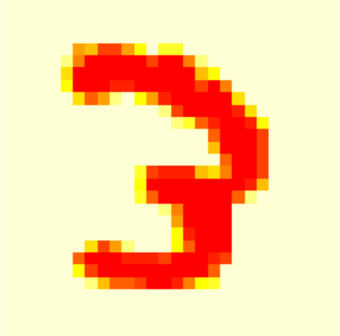
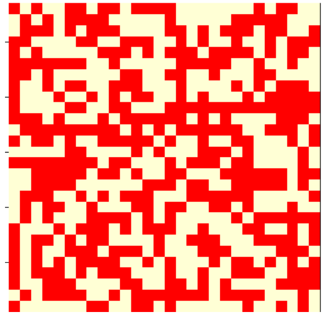
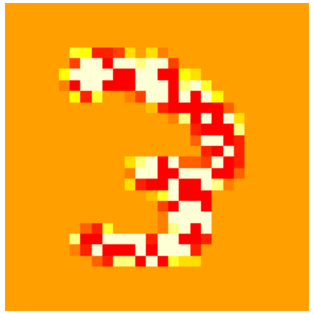
PROJECTION = OVERLAP WITH A PATTERN

OVERLAP ( , ) = SUM 

PROJECTION = OVERLAP WITH A PATTERN

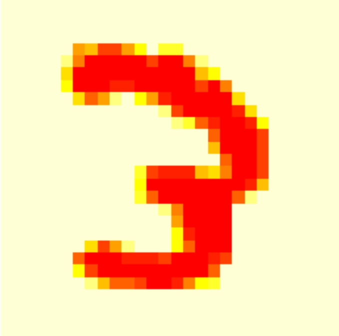
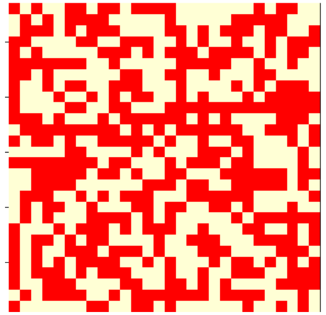
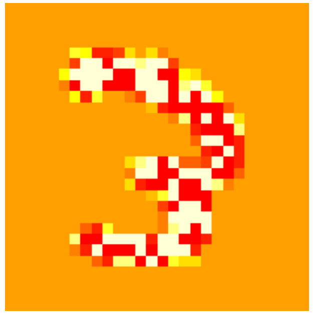
OVERLAP( , ) = SUM 

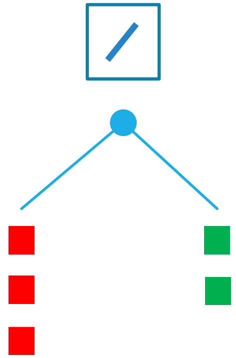
PROJECTION = OVERLAP WITH A PATTERN

OVERLAP( , ) = SUM 

= { 1, if more red pixels than white
-1, if more white pixels than red

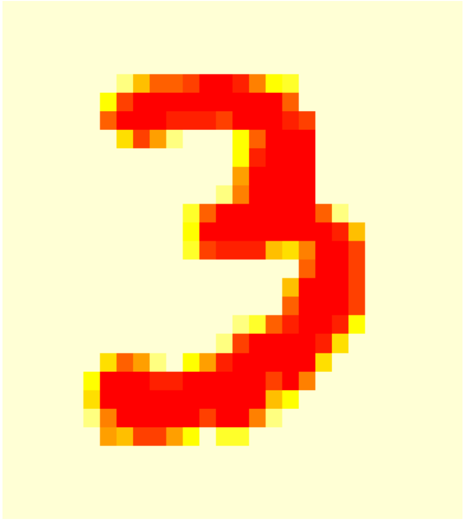
PROJECTION = OVERLAP WITH A PATTERN

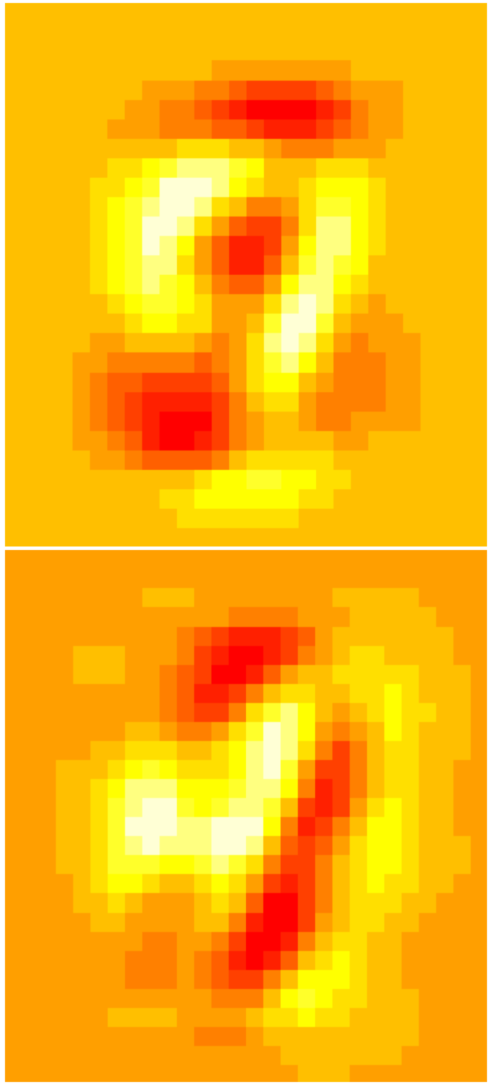
OVERLAP( , ) = SUM 

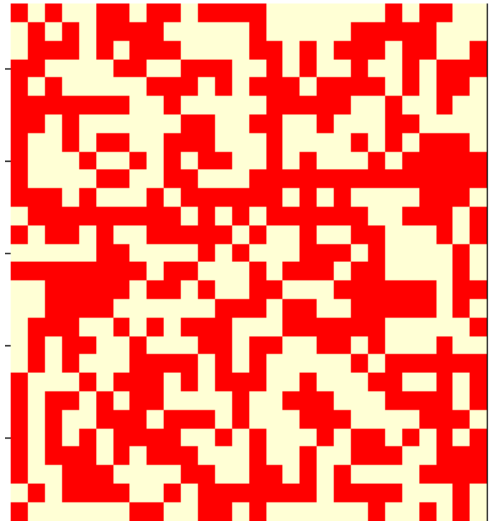


= {
1, if more red pixels than white
-1, if more white pixels than red

DATA POINT – SVD FEATURE – RANDOM FEATURE







PLATO'S ALEGORY OF THE CAVE

He then explains how the philosopher is like a prisoner who is freed from the cave and comes to understand that the shadows on the wall do not make up reality at all, as he can perceive the true form of reality rather than the mere shadows seen by the prisoners

Source: http://en.wikipedia.org/wiki/Allegory_of_the_Cave

SECRET SAUCE

- How to make it faster?
- How to make it more accurate?
- When does it work?
- Is there something better?

http://stefansavev.com/random-projections-secretsouce.html