Machine Learning for Metagenomics

X Data Science Summer School 2018

git clone https://github.com/rmenegaux/2018_DS3_metagenomics.git

Introduction

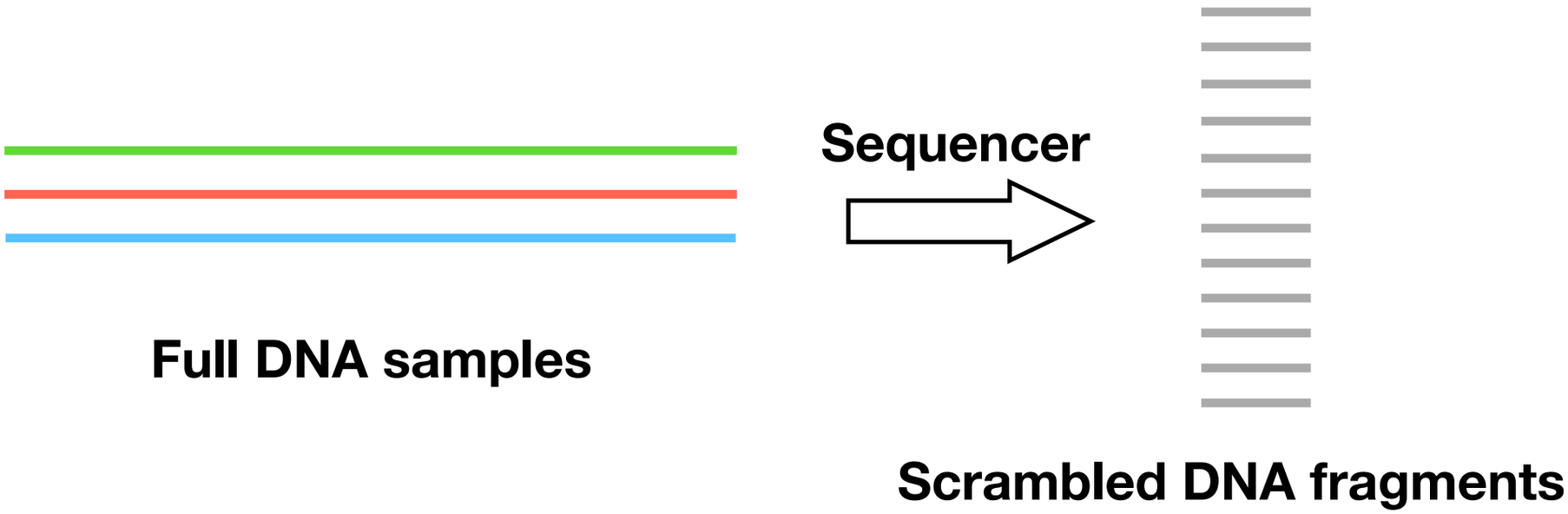
- **Metagenomics**:
  Studying an environment from its genomic material

Introduction

- **Metagenomics**:
  Studying an environment from its genomic material

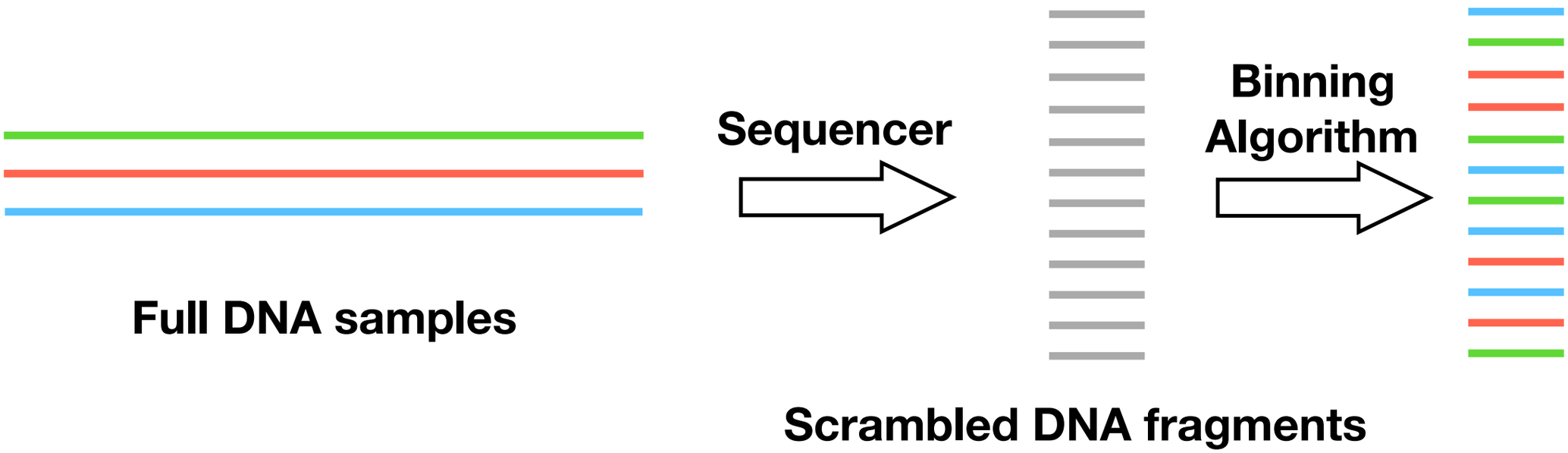
- **Uses**:
  - Bacterial fauna characterisation (e.g. stomach, mouth)
  - Medical diagnosis

Introduction



- Output of DNA sequencer:
  billions of short reads (~100-300 bp)

Introduction



- **Goal**: match reads to their parent genome

Alignment-based methods

- Based on exact matching, or matching up to a fixed number of errors
- State of the art: BWA-MEM (2009)
- Good accuracy, robust to sequencing errors

Alignment-based methods

- Based on exact matching, or matching up to a fixed number of errors
- State of the art: BWA-MEM (2009)
- Good accuracy, robust to sequencing errors
- Can we do faster?

Machine learning (compositional) approach

- Treat as a classification problem
- Class/label = species, features = ?

Linear classifiers

- **k-mer**, or bag-of-words approach
- Represent read as a binary-vector, with 4^k entries

k = 5

Linear classifiers

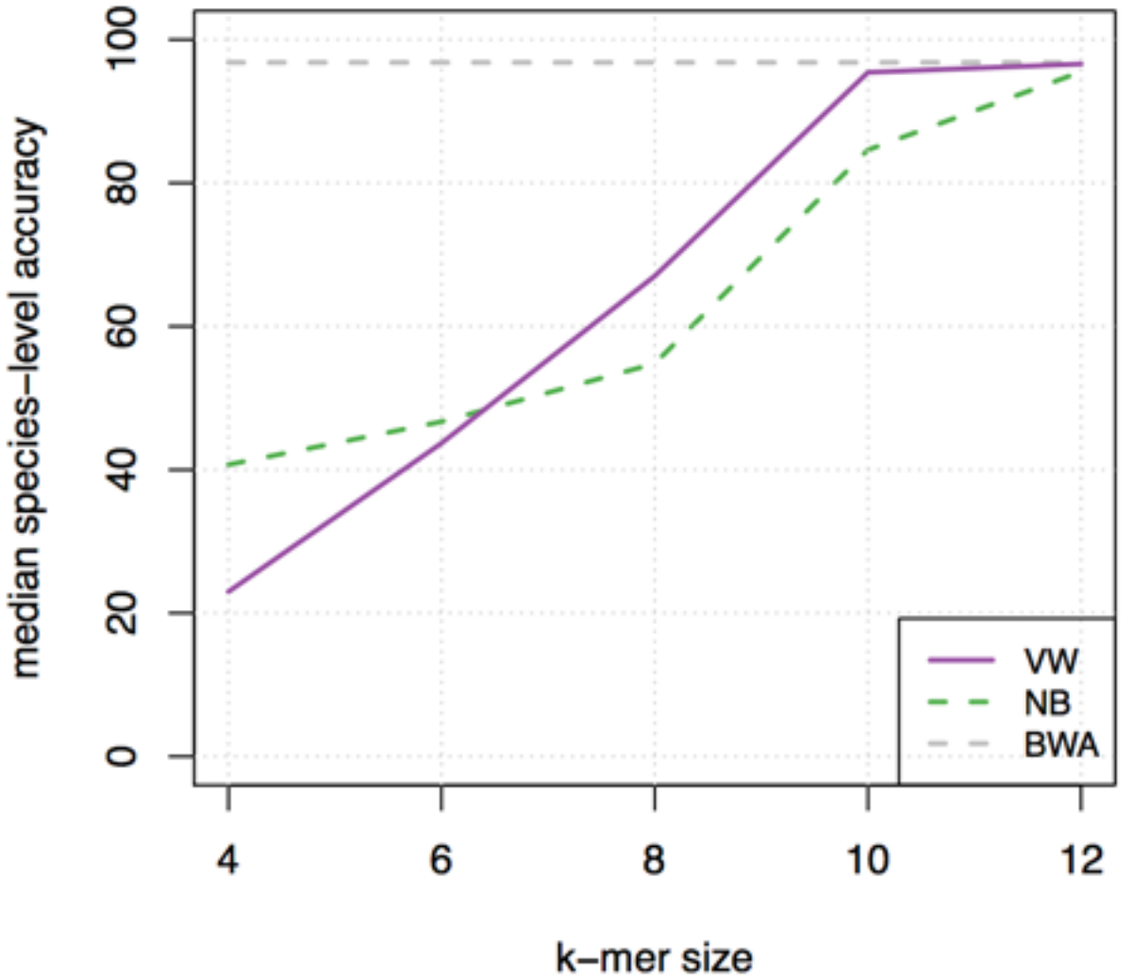
- **k-mer**, or bag-of-words approach
- Represent read as a binary-vector, with 4^k entries

Linear classifiers

- k-mer, or bag-of-words approach
- Represent read as a binary-vector, with 4^k entries
- Train a linear model

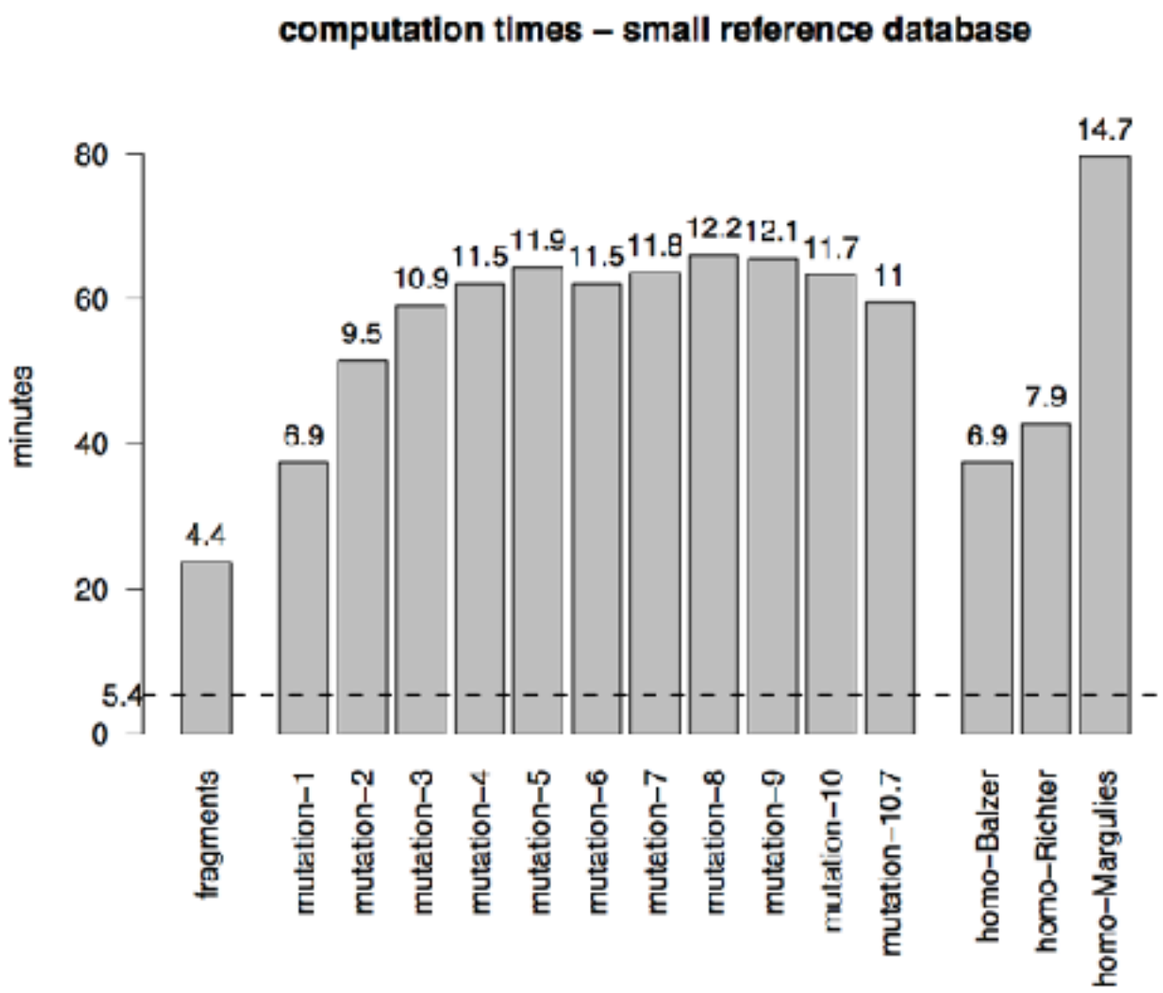
Linear classifiers

- Almost same performance as BWA



Linear classifiers

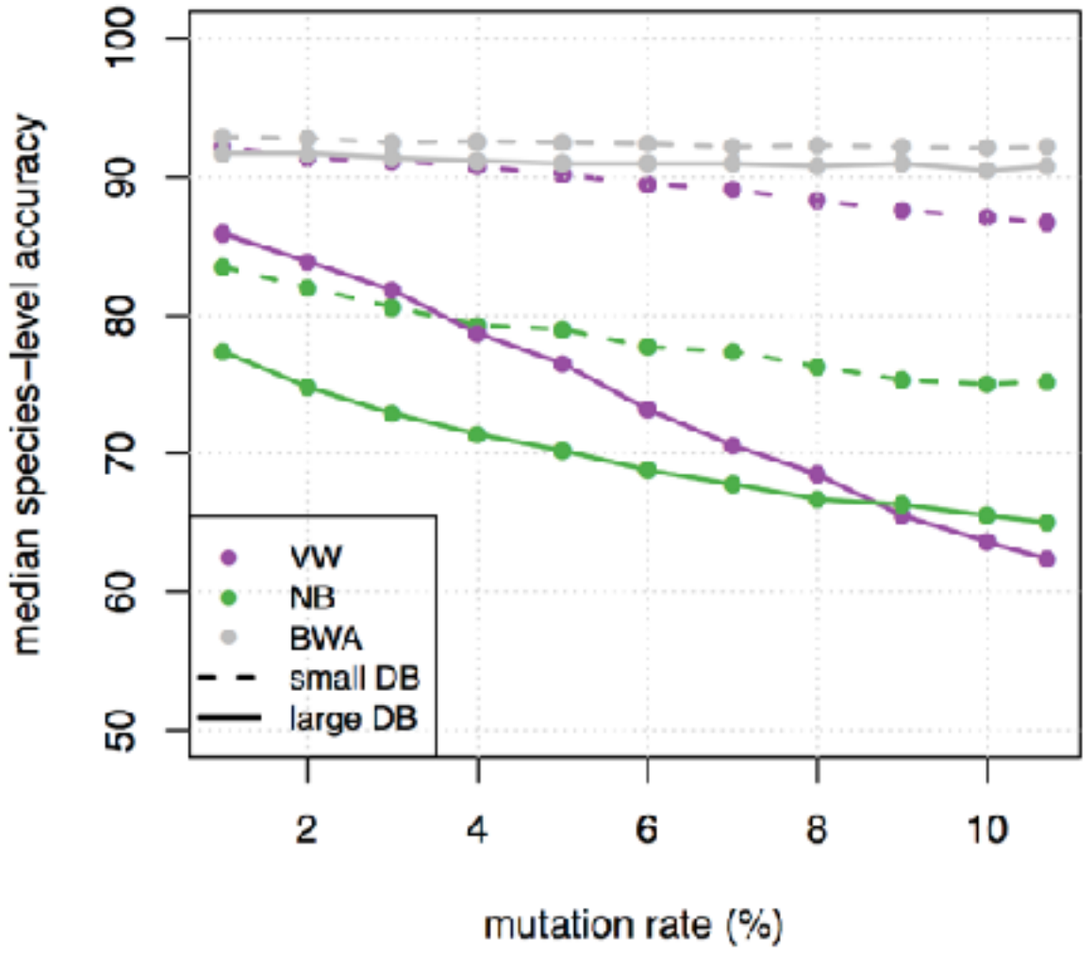
- Almost same performance as BWA
- Faster (5-15 times) prediction times



Linear classifiers

BUT:

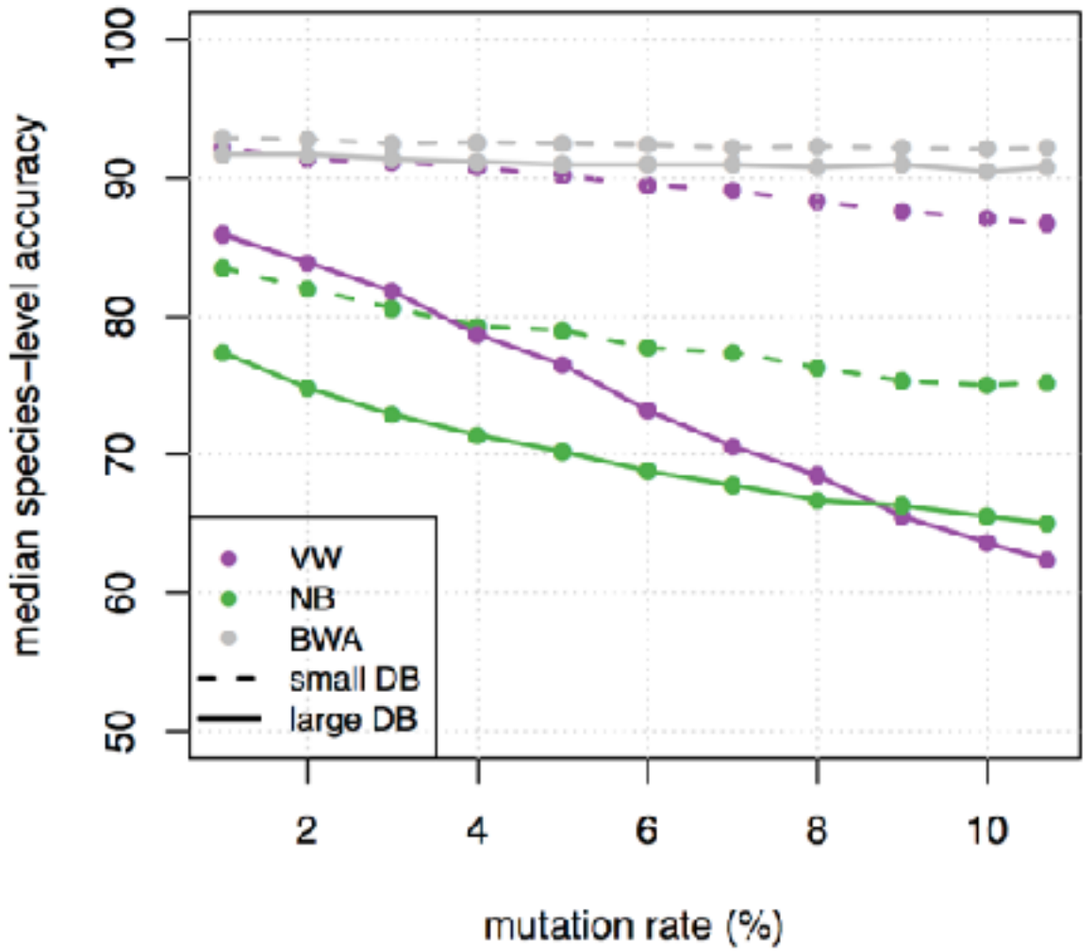
- Performance degrades with sequencing errors



Linear classifiers

BUT:

- Performance degrades with sequencing errors
- High memory cost to store the n_classes x 4^12 weights

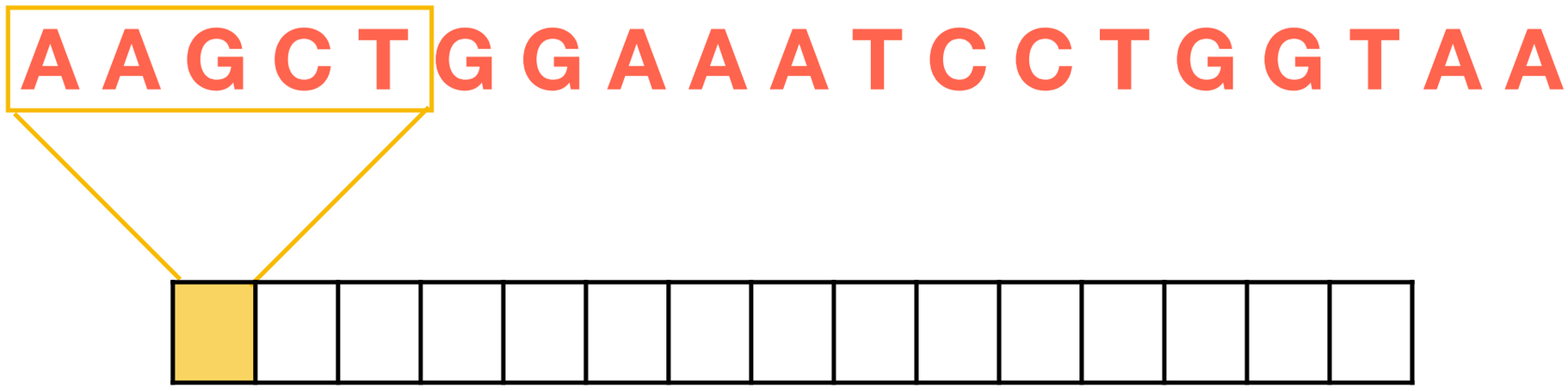


Other approach: Convolutional Nets

Filter (convolution kernel)= AAAAA

A A G C T G G A A A T C C T G G T A A

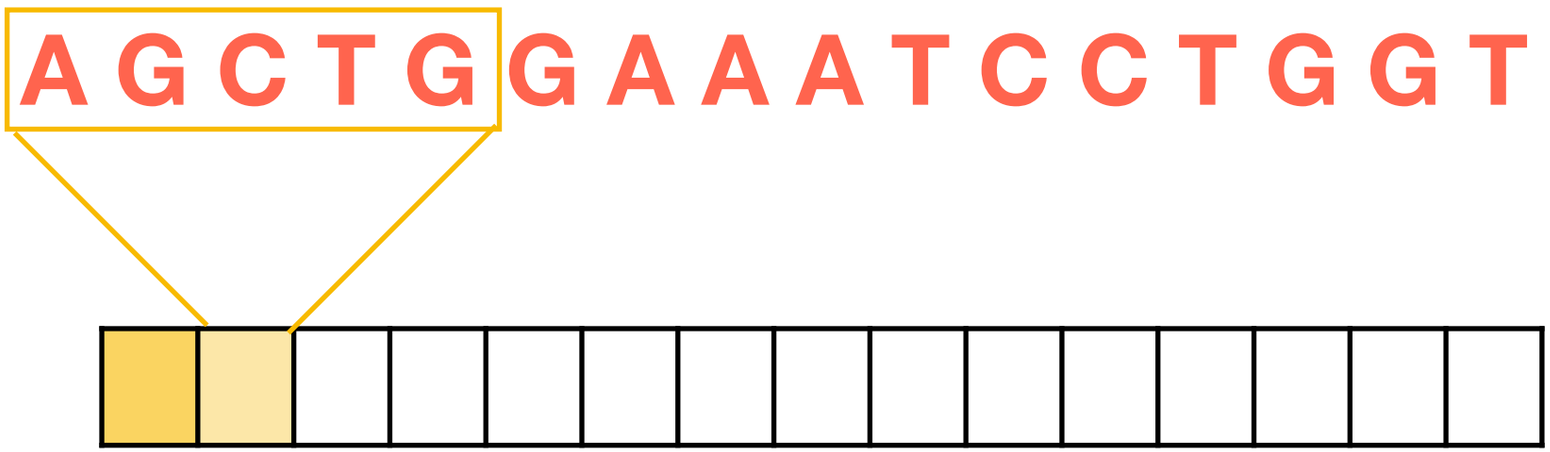
Filter (convolution kernel)= AAAAA



2/5

Filter (convolution kernel)= AAAAA

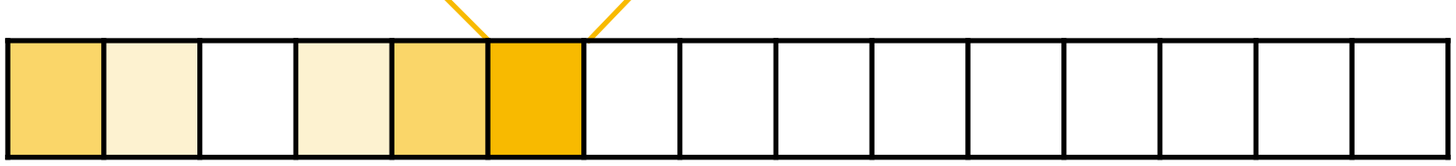
A A G C T G G A A A T C C T G G T A A



1/5

Filter (convolution kernel)= AAAAA

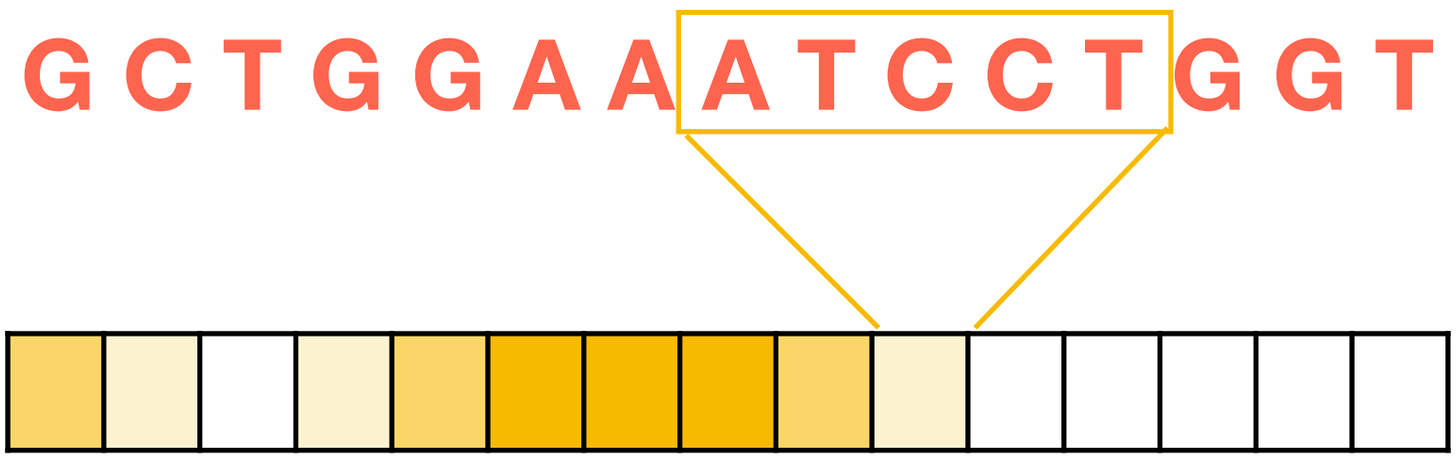
A A G C T G G A A A T C C T G G T A A



3/5

Filter (convolution kernel)= AAAAA

A A G C T G G A A A T C C T G G T A A



1/5

Filter (convolution kernel)= AAAAA

A A G C T G G A A A T C C T G G T A A

2/5

Filter (convolution kernel)= AAAAA

A A G C T G G A A A T C C T G G T A A

Advantage compared to k-mer representation:
Keep positional info

Filter

Output

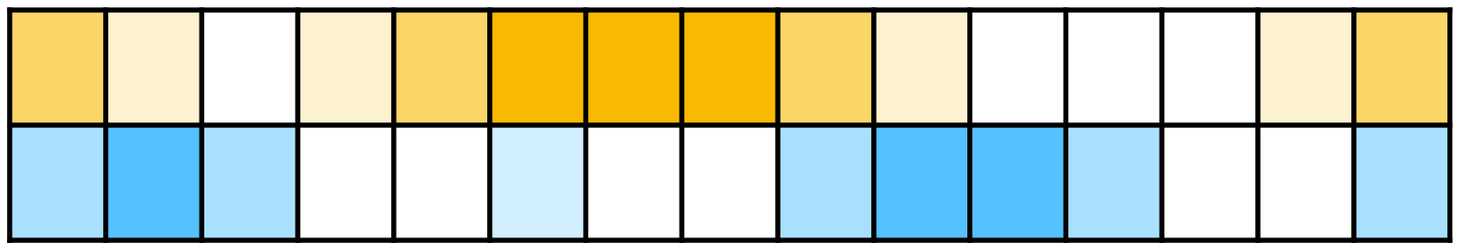
AAAAA

Filter

Output

AAAAA

GTCCA

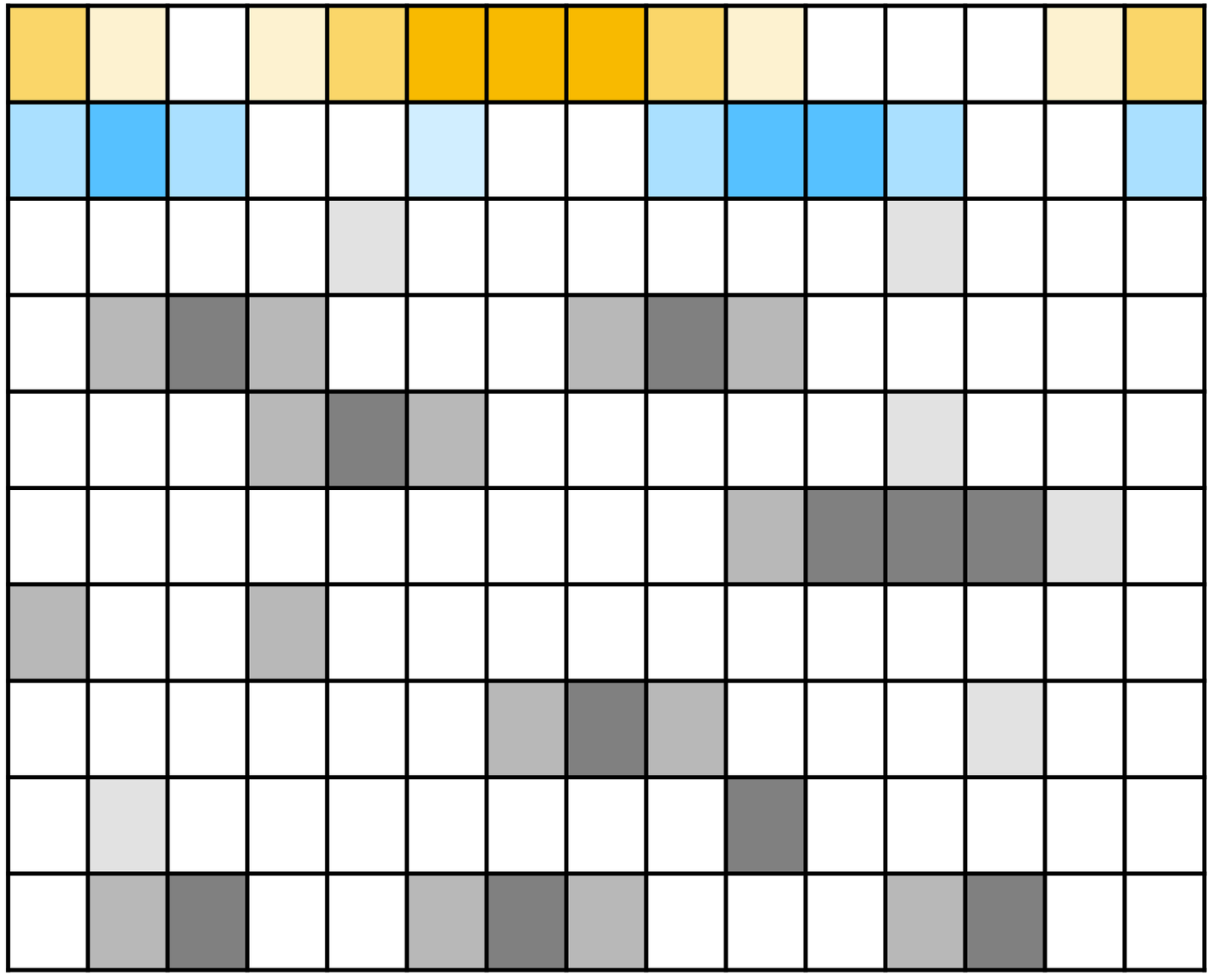


Filter

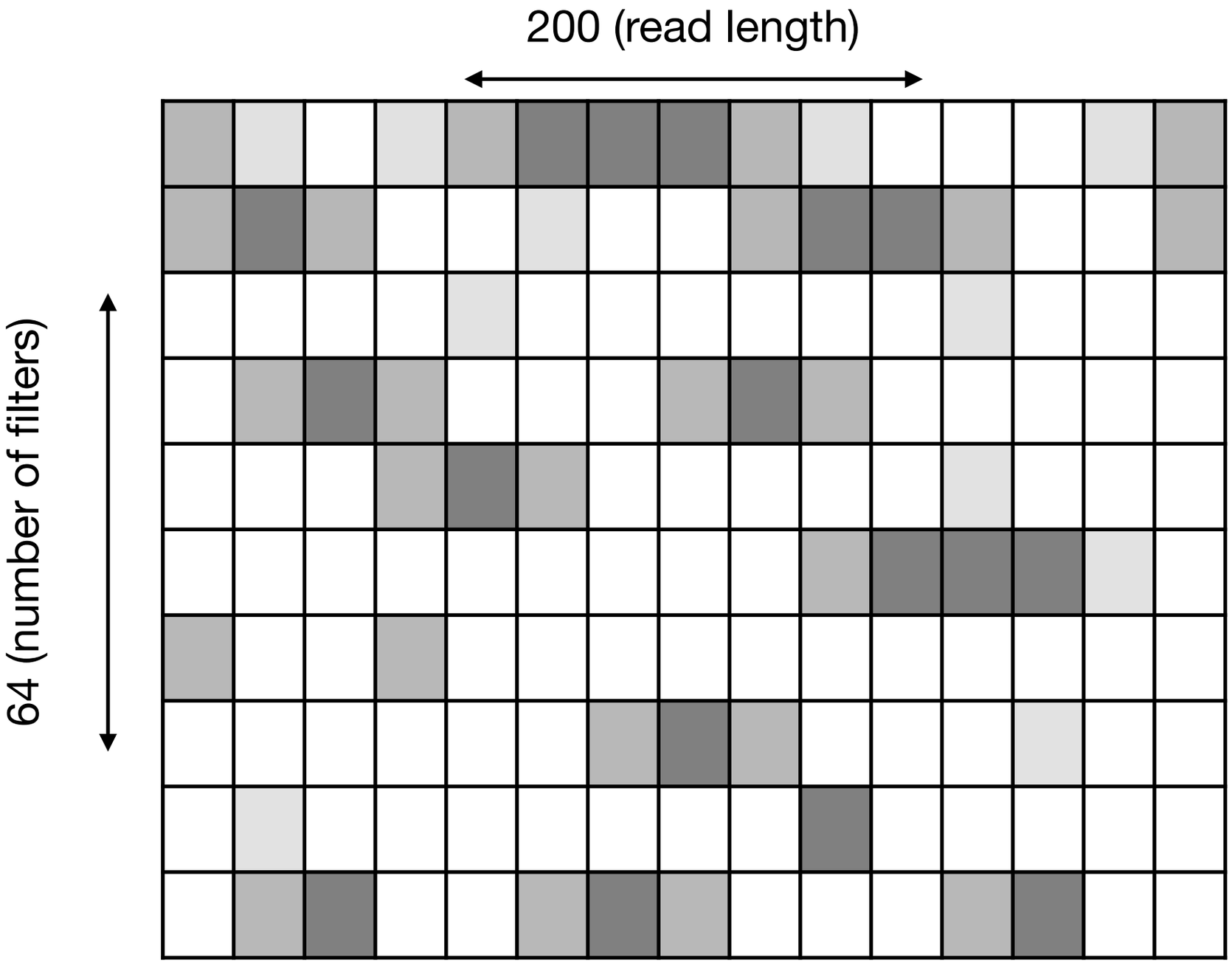
Output

AAAAA

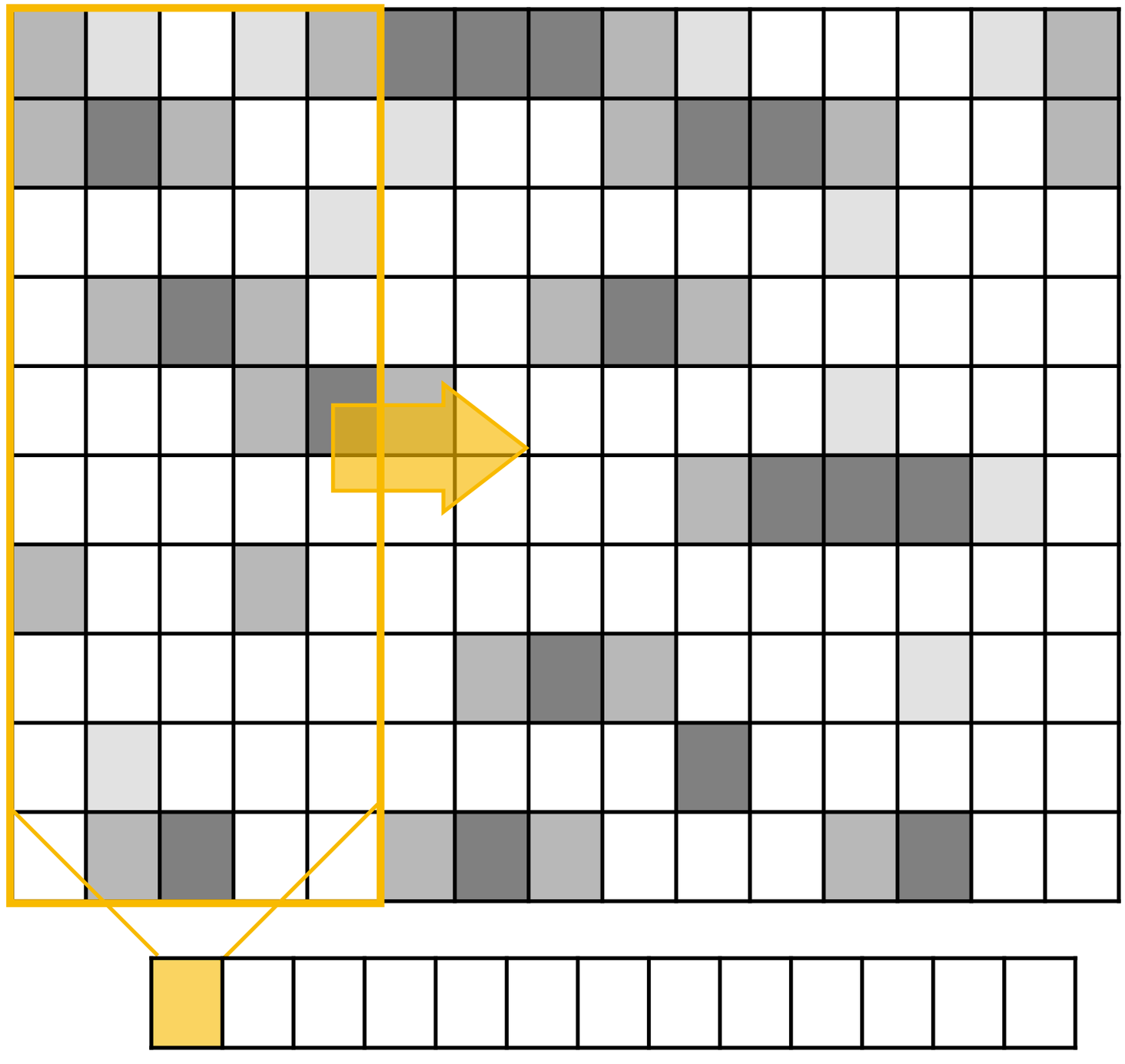
GTCCA



Output of first layer:



ITERATE!



Convolutional Nets

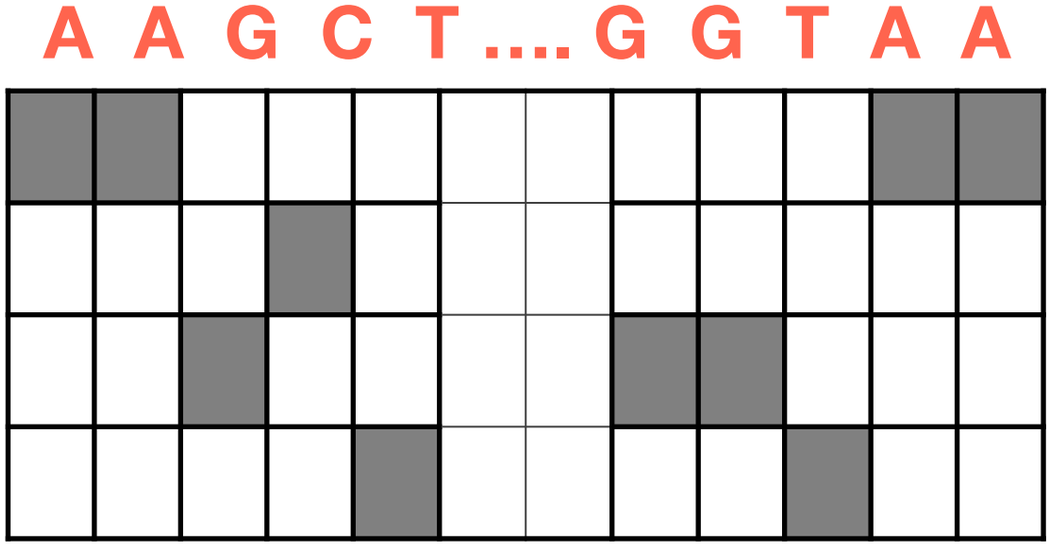
In practice:

A	A	G	C	T	....	G	G	T	A	A
1	1	0	0	0	....	0	0	0	1	1
0	0	0	1	0	....	0	0	0	0	0
0	0	1	0	0	....	1	1	0	0	0
0	0	0	0	1	....	0	0	1	0	0

One-hot encoding

Convolutional Nets

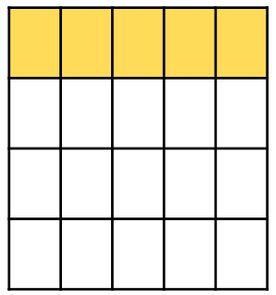
In practice:



4 x 200 binary matrix

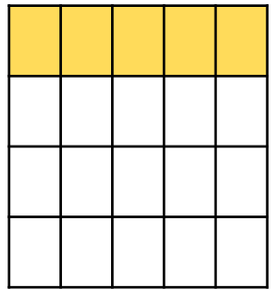
Convolutional Nets

Filter (convolution kernel)= AAAAA



Convolutional Nets

Filter (convolution kernel)= AAAAA



A	A	G	C	T	....	G	G	T	A	A
1	1	0	0	0	....	0	0	0	1	1
0	0	0	1	0	....	0	0	0	0	0
0	0	1	0	0	....	1	1	0	0	0
0	0	0	0	1	....	0	0	1	0	0

Other possible approaches:

- Recurrent Networks
- Linear models with richer features,
  e.g. Bloom filters

Practical session

- Data:
  - training: 10 full bacterial genomes
  - validation: 10000 fragments (100bp long)

- Linear model (scikit-learn)

- Convolutional network (keras)

git clone https://github.com/rmenegaux/2018_DS3_metagenomics.git