Automated Genome Annotation

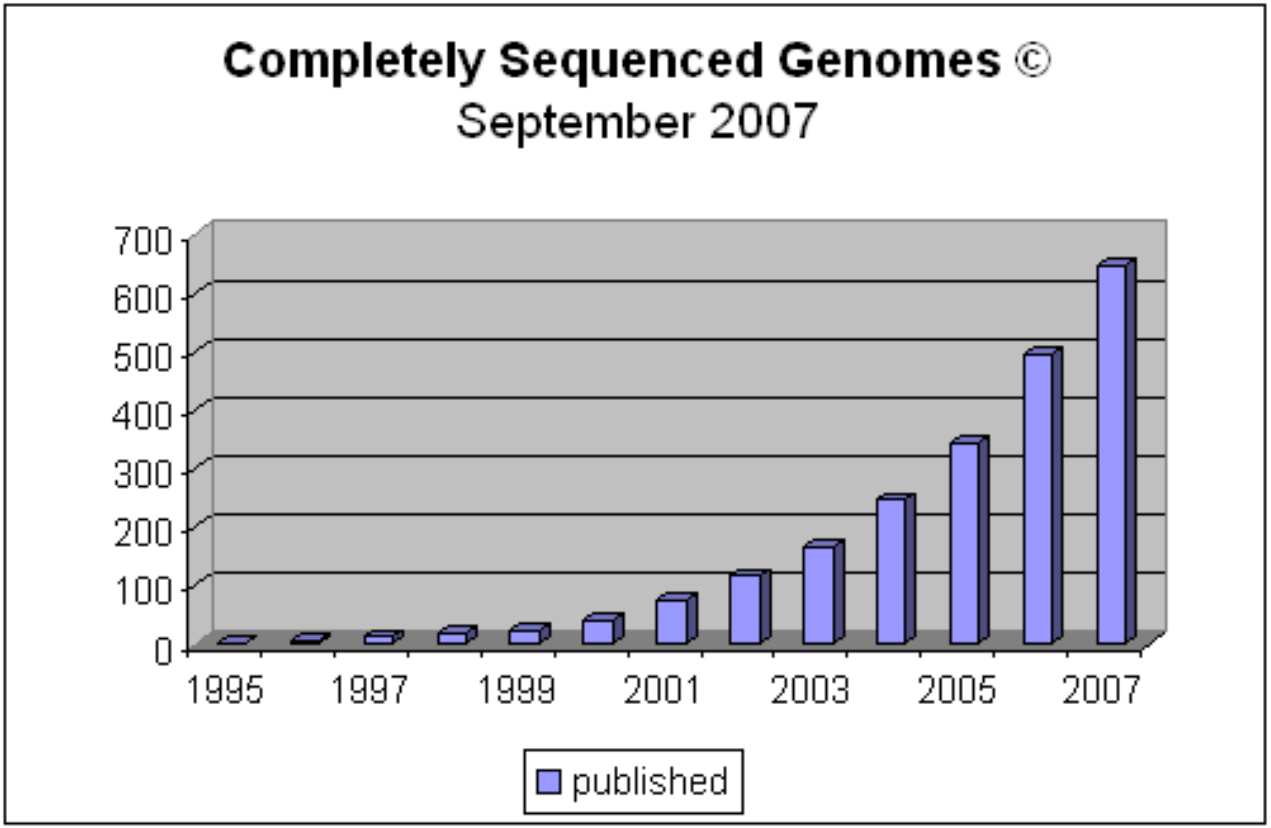
Alexander Zien[1,2] and Sebastian Schultheiss[1]
(based on slides by Cheng Soon Ong)

[1] AG Rätsch, Friedrich Miescher Laboratory, Tübingen
[2] Fraunhofer Institute FIRST, Berlin

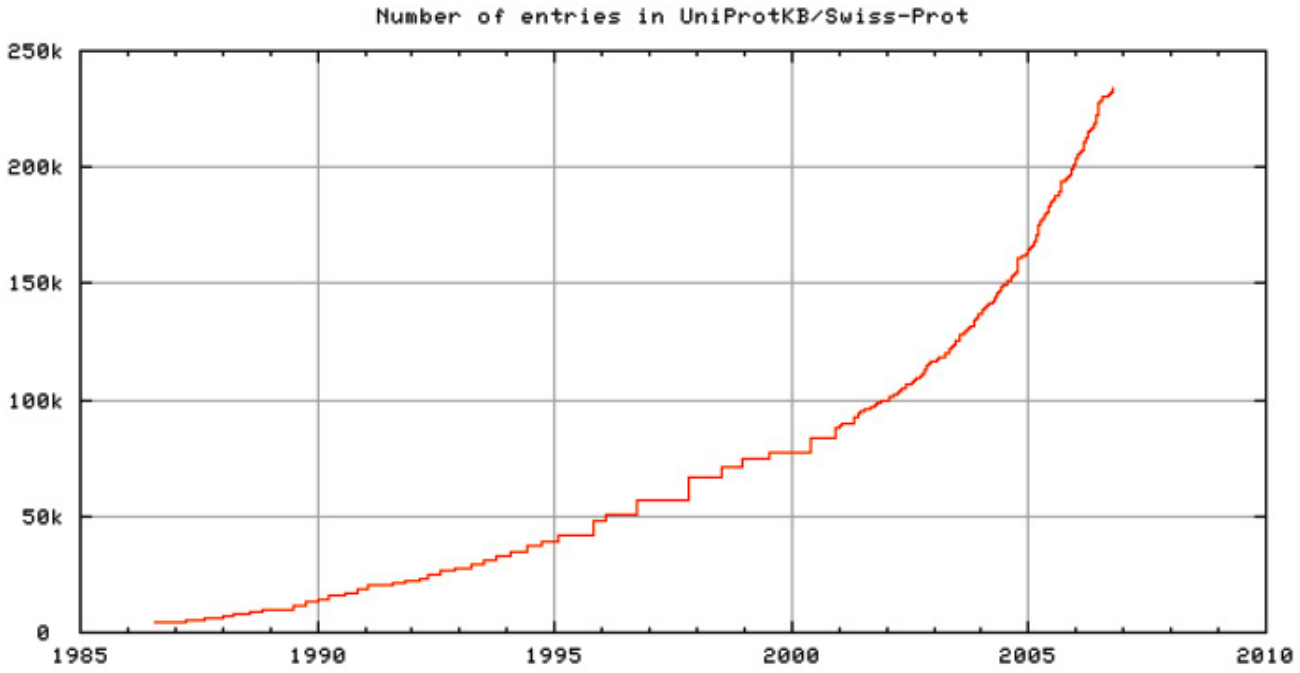
15 November 2008

http://www.fml.mpg.de/raetsch/

Current Sequencing Efforts

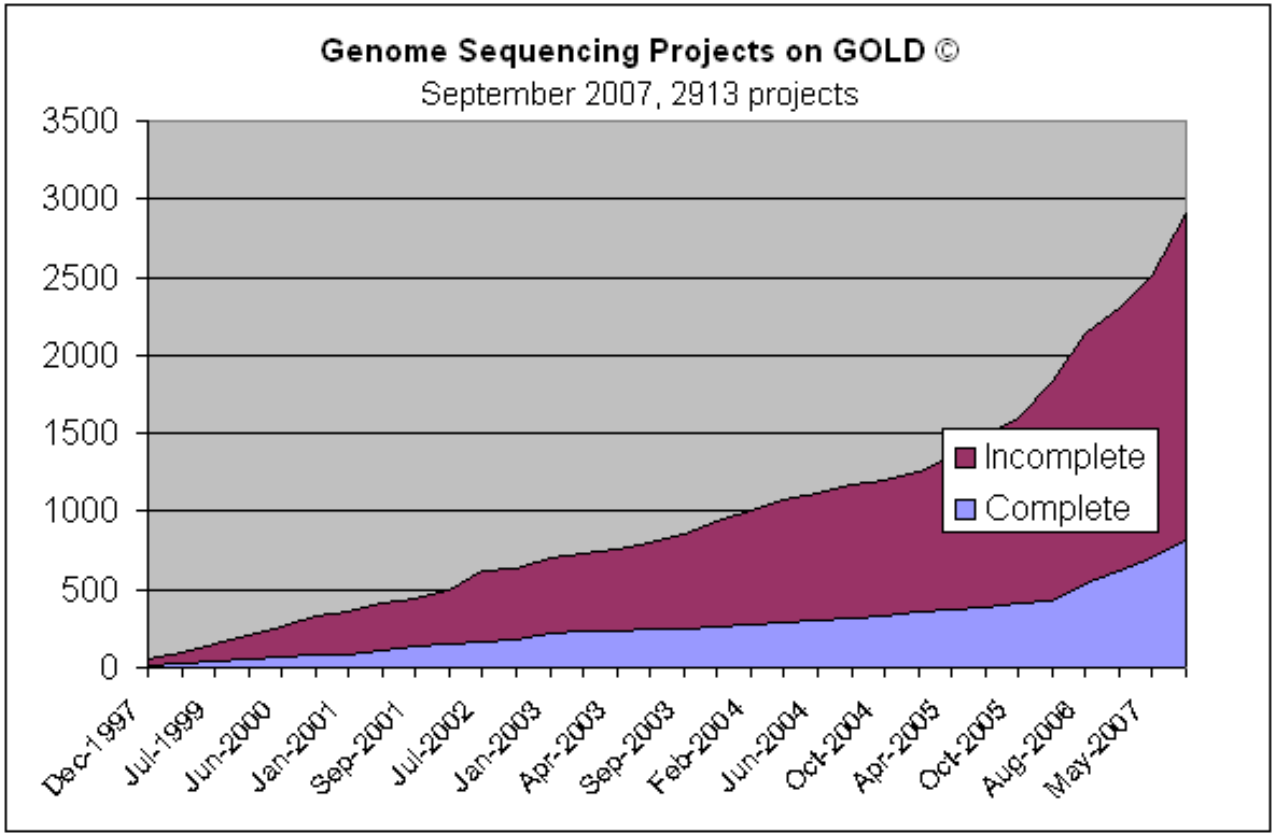


Protein Sequences

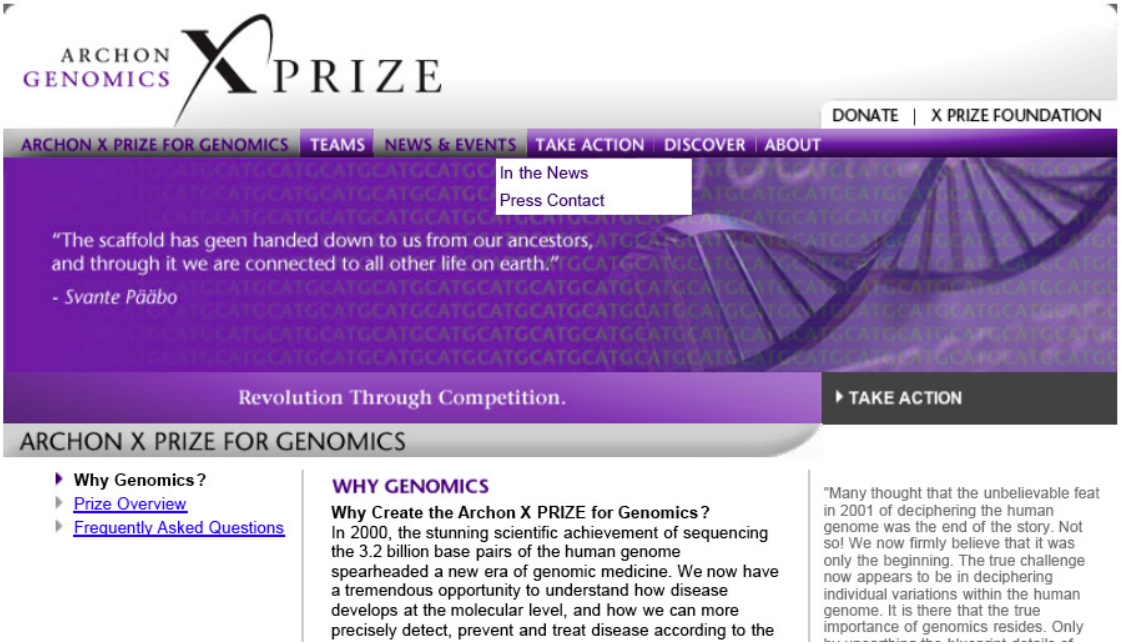


- Protein structures difficult to obtain
- Proteins interact with genome

More to Come...


Genome Sequencing Projects on GOLD ©
September 2007, 2913 projects
3500
3000
2500
2000
1500
1000
500
0
Incomplete
Complete
Dec-1997
Jul-1999
Jun-2000
Jan-2001
Sep-2001
Jul-2002
Jan-2003
Apr-2003
Sep-2003
Feb-2004
Jun-2004
Oct-2004
Apr-2005
Oct-2005
Aug-2006
May-2007

USD$10 Million



- Sequence 100 human genomes in 10 days.
- 3 billion base pairs per genome.
- 60GB of data.

Why Produce All These Data?

- Personalized medicine
- Sequence many new species
- Comparative genomics
- Study evolution
- Find useful proteins (eg, enzymes)
- . . .

Why Do Machine Learning?



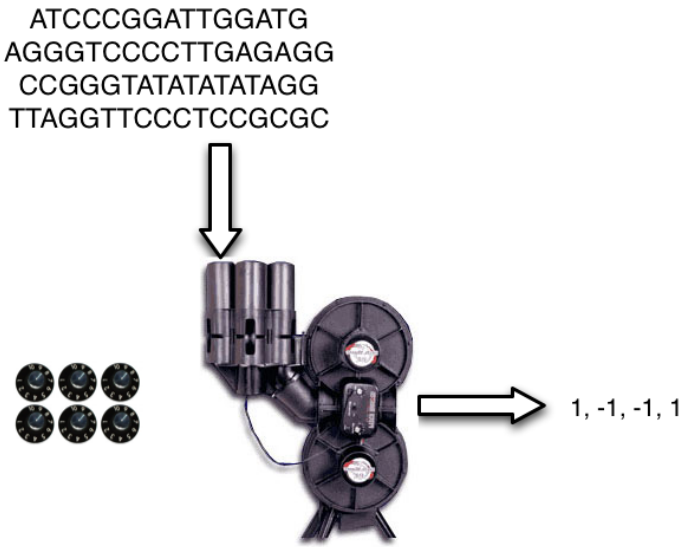
- A lot of data
- Data is noisy
- Large number of features
- No precise biological theory
- Complex relationships

Let the data do the talking!

Classification of Sequences

Example: Recognition of splice sites

- Every 'AG' is a possible acceptor splice site
- Computer has to learn what splice sites look like
  - given some known genes/splice sites . . .
- Prediction on unknown DNA



From Sequences to Features

- Many algorithms depend on numerical representations.
  - Each example is a vector of values (features).
- Use background knowledge to design good features.



	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$	$\mathbf{x}_6$	$\mathbf{x}_7$	$\mathbf{x}_8$	...
GC before	0.6	0.2	0.4	0.3	0.2	0.4	0.5	0.5	...
GC after	0.7	0.7	0.3	0.6	0.3	0.4	0.7	0.6	...
AG**AG**AAG	0	0	0	1	1	0	0	1	...
TTT**AG**	1	1	1	0	0	1	0	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱
Label	+1	+1	+1	−1	−1	+1	−1	−1	...

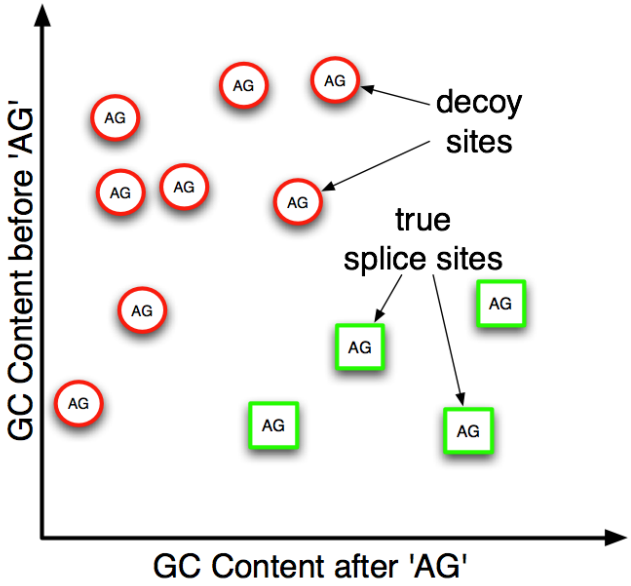
Recognition of Splice Sites

- Given: Known acceptor splice sites and pseudo sites

```
AAACAAATAAGTAACTAATCTTTTAGGAAGAACGTTTCAACCATTTTGAG
AAGATTAAAAAAAAACAAATTTTTAGCATTACAGATATAATAATCTAATT
CACTCCCCAAATCAACGATATTTTAGTTCACTAACACATCCGTCTGTGCC
TTAATTTCACTTCCACATACTTCCAGATCATCAATCTCCAAAACCAACAC
```

intron **exon**

- Goal: Rule that distinguishes true from false ones



e.g. exploit that exons have higher GC content

or

that certain motifs are located nearby

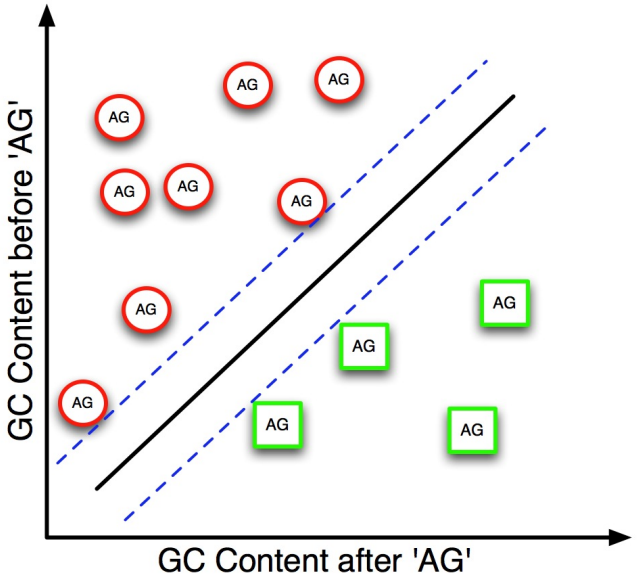
Recognition of Splice Sites

- Given: Known acceptor splice sites and pseudo sites

```
AAACAAATAAGTAACTAATCTTTTAGGAAGAACGTTTCAACCATTTTGAG
AAGATTAAAAAAAAACAAATTTTTAGCATTACAGATATAATAATCTAATT
CACTCCCCAAATCAACGATATTTTAGTTCACTAACACATCCGTCTGTGCC
TTAATTTCACTTCCACATACTTCCAGATCATCAATCTCCAAAACCAACAC
```

intron **exon**

- Goal: Rule that distinguishes true from false ones



Linear classifiers with large **margin**

Comparing Examples

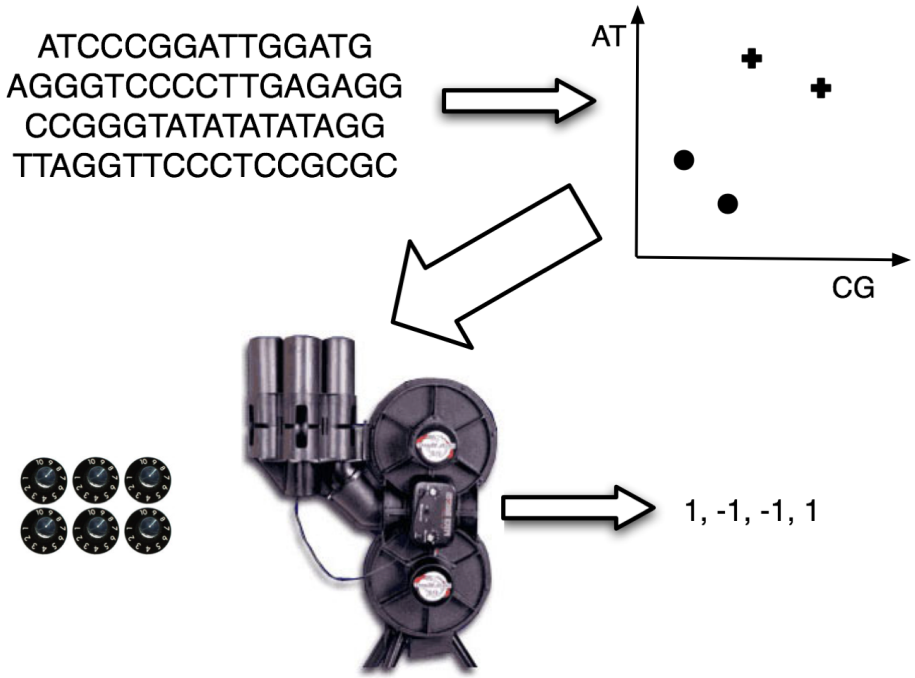
How similar are two examples?

$$\langle \mathbf{x}_1, \mathbf{x}_2 \rangle = 0.6 * 0.2 + 0.7 * 0.7 + 0 * 0 + 1 * 1 = 1.61$$
$$\langle \mathbf{x}_1, \mathbf{x}_5 \rangle = 0.6 * 0.2 + 0.7 * 0.3 + 0 * 1 + 1 * 0 = 0.33$$

Is $\mathbf{x}_1$ more similar to $\mathbf{x}_2$ or $\mathbf{x}_5$?

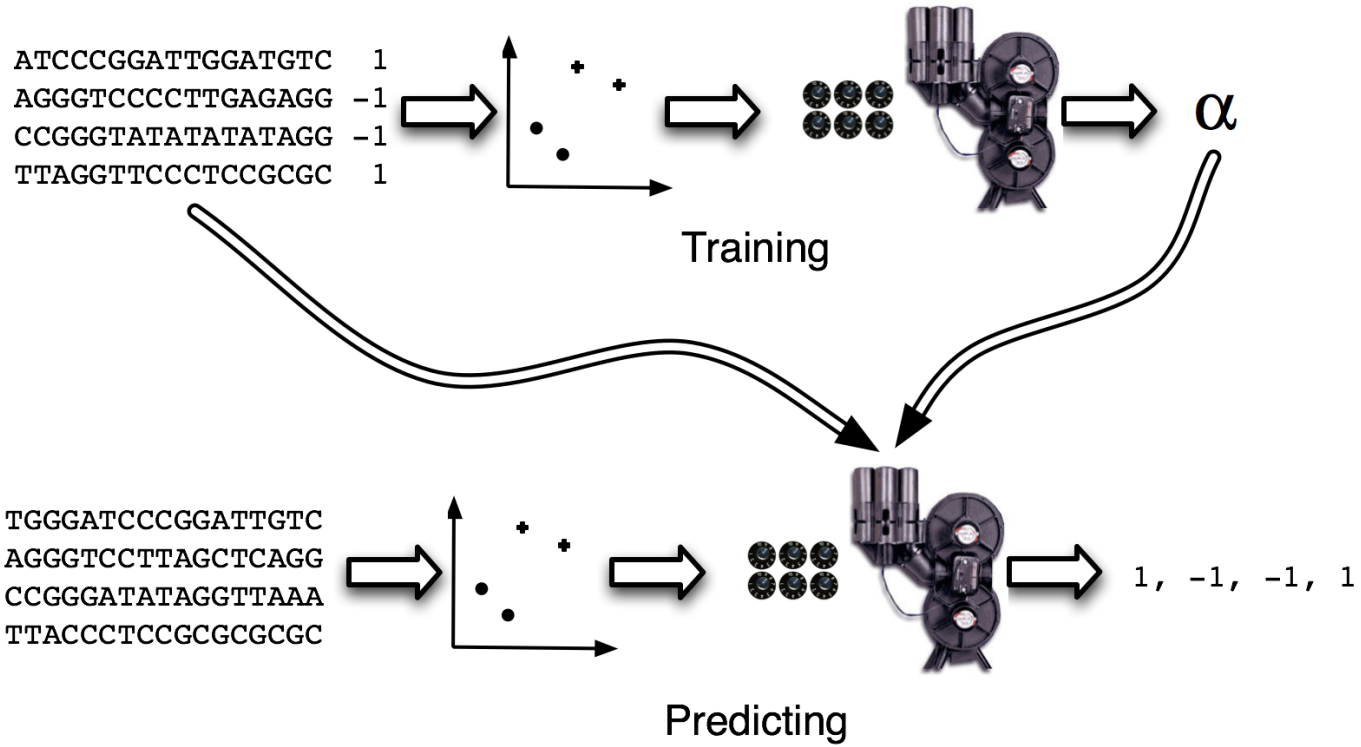
	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$	$\mathbf{x}_6$	$\mathbf{x}_7$	$\mathbf{x}_8$	...
GC before	0.6	0.2	0.4	0.3	0.2	0.4	0.5	0.5	...
GC after	0.7	0.7	0.3	0.6	0.3	0.4	0.7	0.6	...
AG**AG**AAG	0	0	0	1	1	0	0	1	...
TTT**AG**	1	1	1	0	0	1	0	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱
Label	+1	+1	+1	−1	−1	+1	−1	−1	...

Numerical Representation



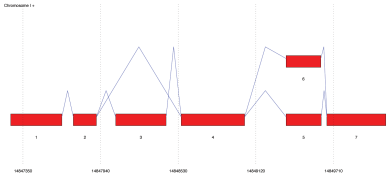
The feature space is implicitly constructed via a measure of similarity called the kernel. The algorithms which allow this are called kernel methods.

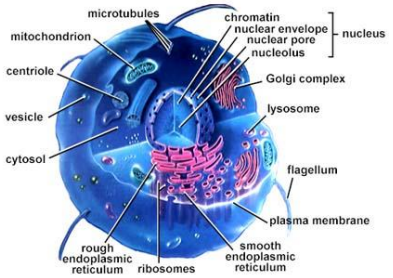
Empirical Inference (SVM)

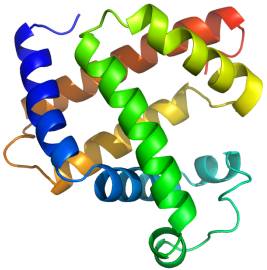


Current Projects

- *Ab initio* gene finding
  - Prediction of (alternative) splicing
  - Transcription starts and stops
  - Improved alignment algorithms
- Prediction of polymorphisms
- Function Prediction
  - Gene regulation
  - RNA secondary structure
  - Protein subcellular localization



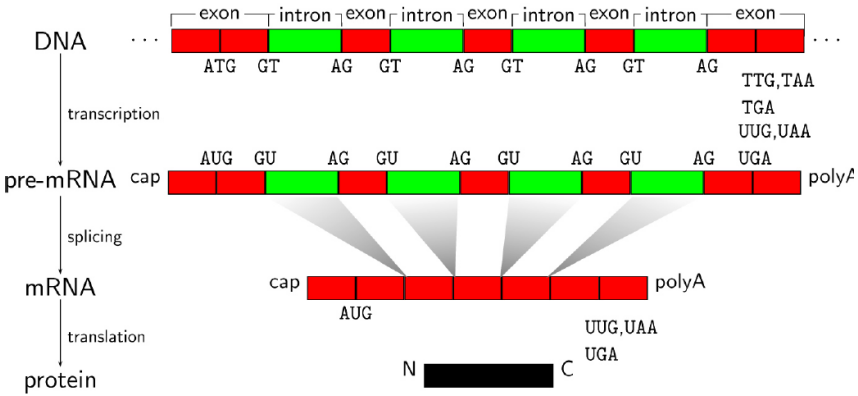




Gene Finding I

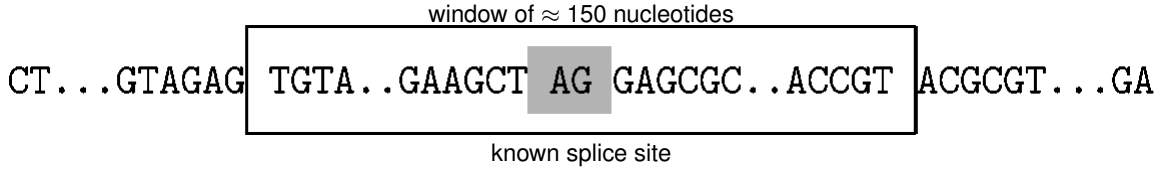
Given the DNA, predict resulting mRNA and protein

- Requires very accurate identification of
  - splice sites, translation & transcription starts & stops
  - sites of regulation (transcription, splicing, etc.)



Gene Finding II: Splice Sites

Finding Intron-Exon junctions



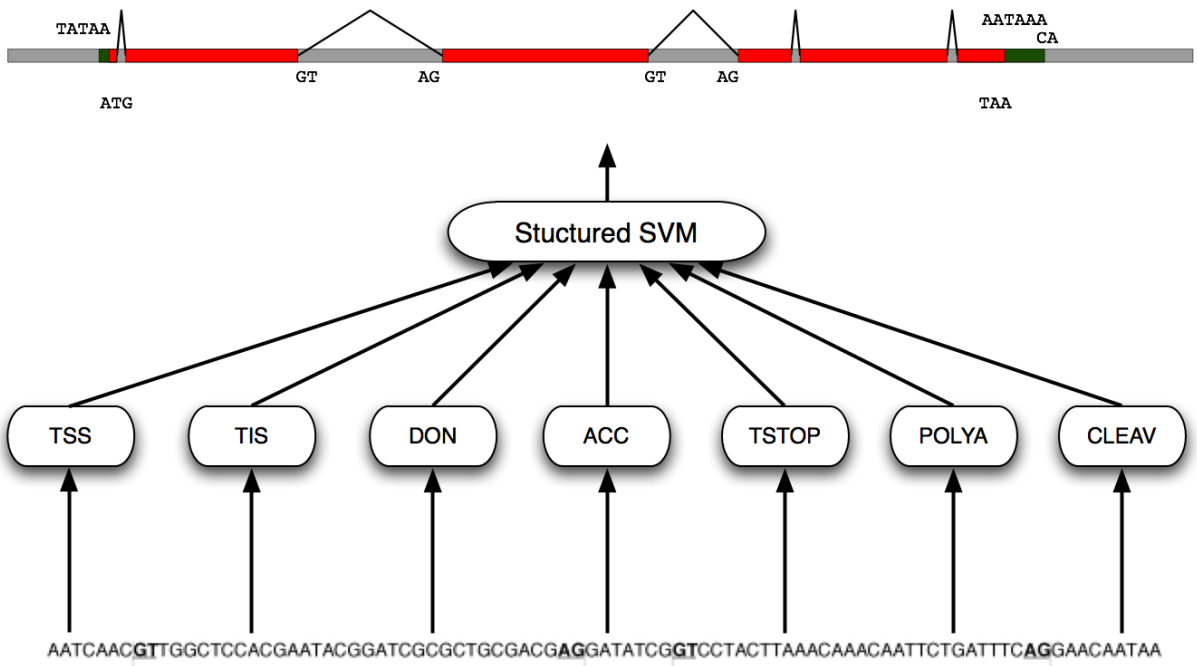
- true sites: fixed-length window around splice site
- decoys sites: generated by shifting the window

```
AAACAAATAAGTAACTAATCTTTTAGGAAGAACGTTTCAACCATTTTGAG
AAGATTAAAAAAAAAACAAATTTTTAGCATTACAGATATAATAATCTAATT
CACTCCCCAAATCAACGATATTTTAGTTCACTAACACATCCGTCTGTGCC
TTAATTTCACTTCCACATACTTCCAGATCATCAATCTCCAAAACCAACAC
```

$\Rightarrow$ Very unbalanced problem (1:200)
$\Rightarrow$ Millions of points from EST databases
$\Rightarrow$ Large scale methods necessary

Gene Finding III: Whole Gene

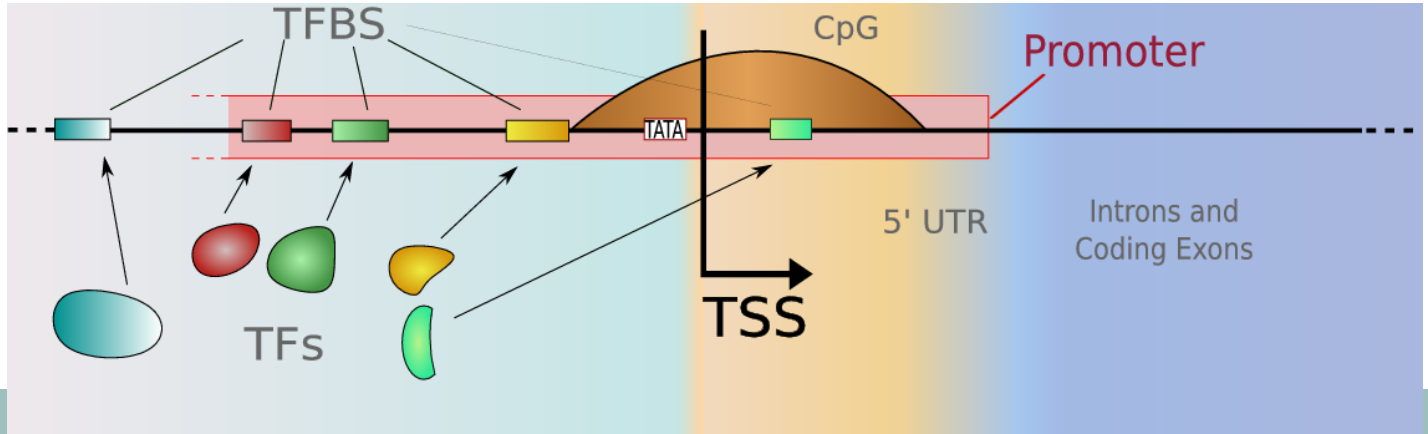
mGene: An ab initio gene finding system



won nGASP competition on $C.\ elegans$

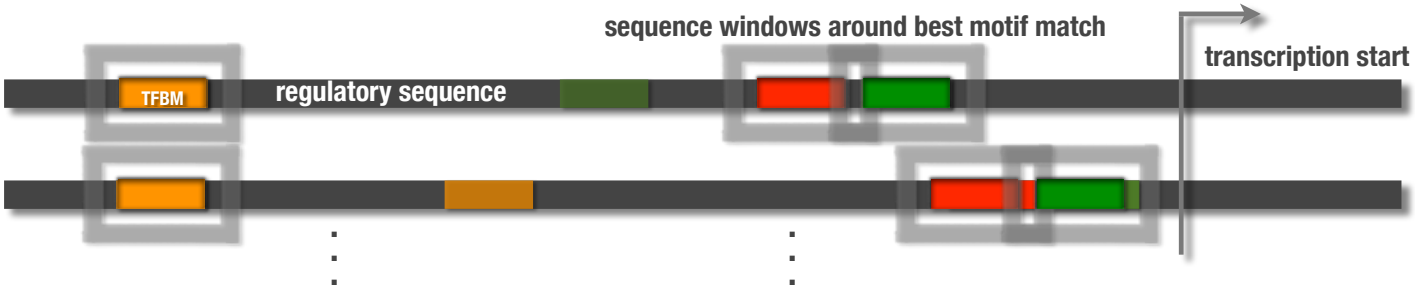
Regulation: Use Case

- **Microarray data** with different experimental conditions
- Apply statistical methods to identify co-expressed genes
- **Control:** invariably highly expressed genes
- **Goal:** find all genes regulated by the same mechanism
- Select appropriate **regulatory regions** (promoter, introns, downstream)
- Learn **co-regulation** from different sequence features



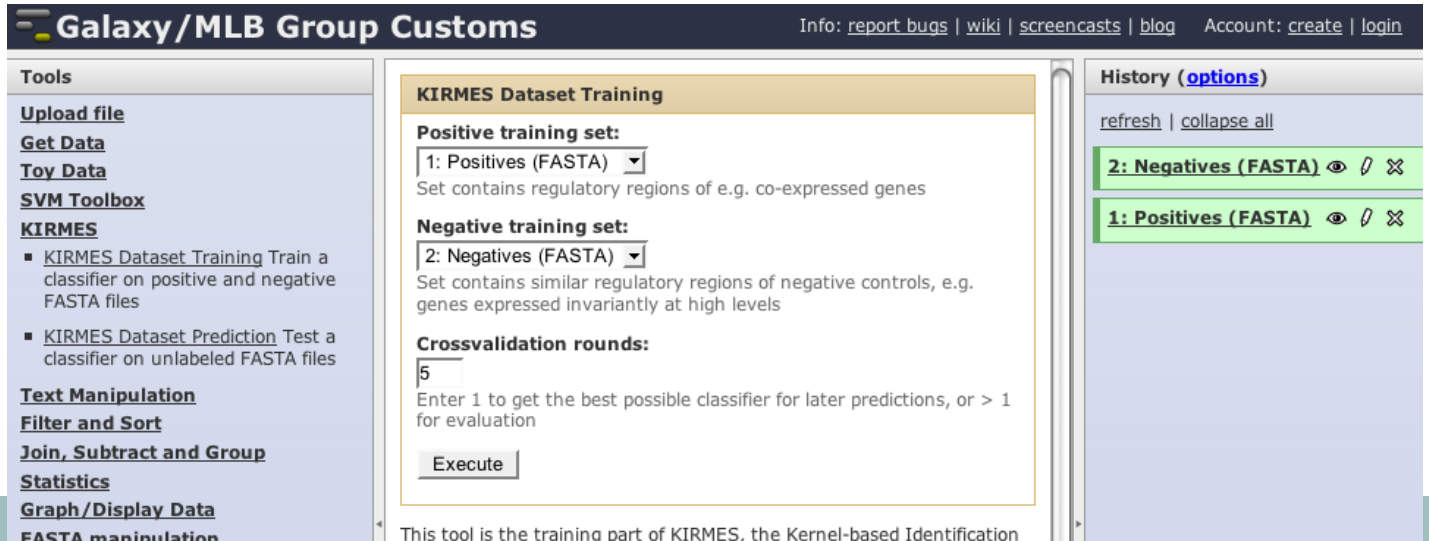
Regulation: Background

- Naïve approach: compare region similarity
- Novel machine learning method
  - Windows around motif positions
  - Positional information, relative distances
  - Conservation information from related organisms



Regulation: KIRMES

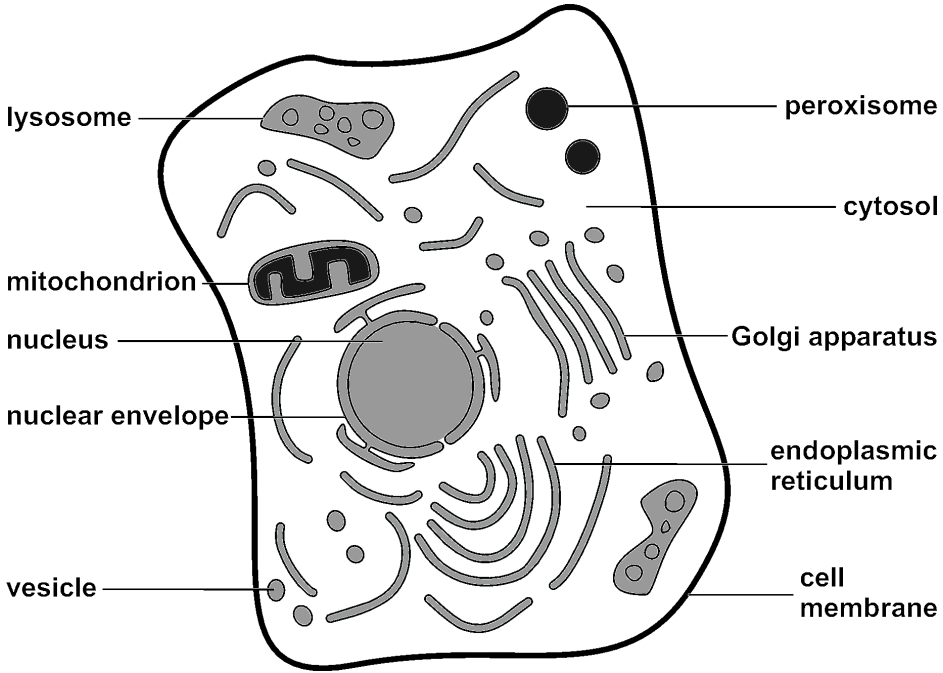
- **Input:** FASTA files containing regulatory regions of co-expressed genes (*e.g.* from microarray experiments)
- **Output:** Classify genes as co-regulated or not
- **Novelty:** Kernel that accepts input vectors with small sequence windows, conservation
- **Galaxy** server at http://galaxy.tuebingen.mpg.de



Subcellular Localization

Input: protein sequence
Output: target location of the protein

- different kernels for different sequence patterns
- automatically combine kernels



Which Problems?

MAX-PLANCK-GESELLSCHAFT

SVMs are for you if:

- There is relatively large amounts of data (examples).
- The outcomes (labels) are well defined, and there are labeled examples.
- Something is known about the contributing factors (features for the kernel).
- No good model of the phenomenon exists.

Questions machine learners will (probably) ask you:

- What is an example? Sequence, structure, vector, etc.
- What type of label do you have? $\{-1, +1\}$, $\{1, 2, 3, ...\}$, real value, structure, etc.
- What features/motifs are useful?

Summary

Machine Learning

- Given some labelled training data, construct a predictor for unlabelled data.
- Relevant features are captured using a measure of similarity called the kernel.

Gene Finding

- Highly accurate recognition of genomic signal sites:
  - splice sites (acceptor, donor)
  - transcription start and end
  - translation start and end
- Splice form prediction
  - including alternative splicing

Any questions?

