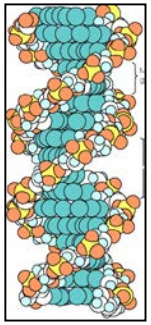
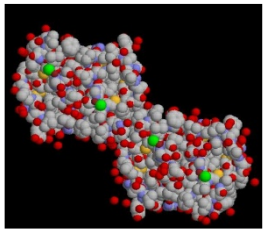
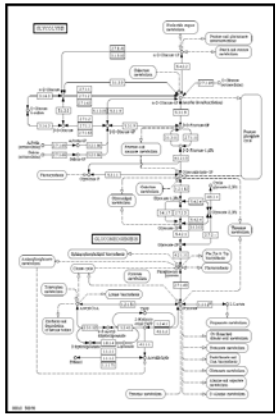
Introduction to Bioinformatics Finish

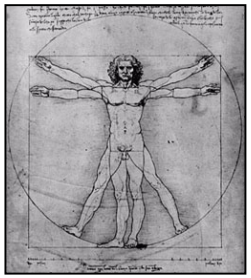
Johannes Starlinger

This Lecture









Genomics

Sequencing

Gene prediction

Evolutionary relationships

Motifs - TFBS

Transcriptomics

Alignment

...

Proteomics

Structure prediction

... comparison

Motives, active sites

Docking

Protein-Protein Interaction

Proteomics

...

Systems Biology

Pathway analysis

Gene regulation

Signaling

Metabolism

Quantitative models

Network reconstruction

...

Medicine

Phenotype – genotype

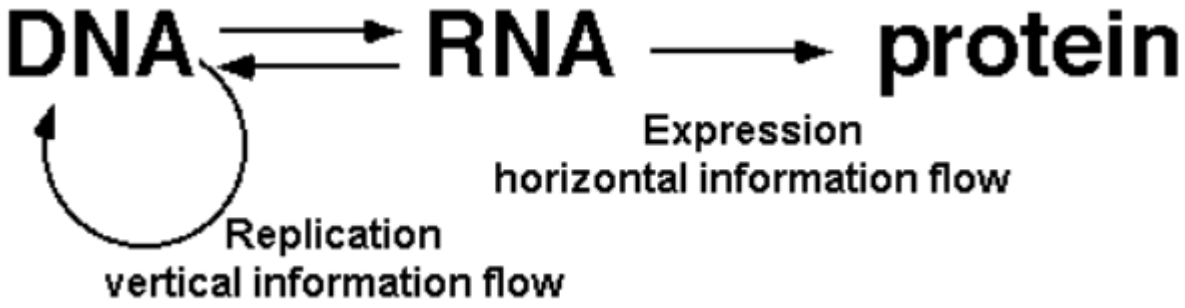
Mutations and risk

Population genetics

Adverse effects

...

Central Dogma of Molecular Biology



	U		C		A		G		
U	UUU UUC	Phenyl-alanine	UCU UCC UCA UCG	Serine	UAU UAC	Tyrosine	UGU UGC	Cysteine	U C A G
	UUA UUG	Leucine			UAA UAG	Stop codon Stop codon	UGA	Stop codon	
							UGG	Tryptophan	
C	CUU CUC CUA CUG	Leucine	CCU CCC CCA CCG	Proline	CAU CAC	Histidine	CGU CGC CGA CGG	Arginine	U C A G
					CAA CAG	Glutamine			
A	AUU AUC AUA	Isoleucine	ACU ACC ACA ACG	Threonine	AAU AAC	Asparagine	AGU AGC	Serine	U C A G
	AUG	Methionine; initiation codon			AAA AAG	Lysine	AGA AGG	Arginine	
G	GUU GUC GUA GUG	Valine	GCU GCC GCA GCG	Alanine	GAU GAC	Aspartic acid	GGU GGC GGA GGG	Glycine	U C A G
					GAA GAG	Glutamic acid			

Bioinformatics / Computational Biology

- Computer Science methods for
  - Solving biologically relevant problems
  - Analyzing and managing experimental data sets
- Empirical: Data from high throughput experiments
- Mostly focused on developing algorithms
- Problem are typically complex, data full of errors – importance of heuristics and approximate methods
- Reductionist – Strings, graphs, sequences, signals
- Interdisciplinary: Biology, Computer Science, Physics, Mathematics, Genetics, ...

Typical Approach

1. Observe and learn about biological background
2. Abstract
3. Formalize (Definitions + Formulas)
4. Create algorithms
   - Based on observations
   - Using formalization
5. Learn from outcome
6. Reiterate

Searching Sequences (Strings)

- A chromosome is a string
- A sequencing machine generates strings
- Substrings may represent biologically important areas
  - Genes on a chromosome
  - Transcription factor binding sites
  - Same gene in a different species
  - Similar gene in a different species
  - ...
- Exact or approximate string search
  - Naive and Boyer-Moore algorithm
  - PSWM: Approximate gap-free matching
  - Local and global alignment

Example

$$d(i,j) = \min \begin{cases} d(i,j-1)+1 \\ d(i-1,j)+1 \\ d(i-1,j-1)+t(i,j) \end{cases}$$

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1							
T	2							
G	3							
G	4							

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1	0						
T	2							
G	3							
G	4							

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1	0	1	2	3	4	5	6
T	2							
G	3							
G	4							

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1	0	1	2	3	4	5	6
T	2	1	0	1	2	3	4	5
G	3							
G	4							

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1	0	1	2	3	4	5	6
T	2	1	0	1	2	3	4	5
G	3	2	1	0	1	2	3	4
G	4							

		A	T	G	C	G	G	T
	0	1	2	3	4	5	6	7
A	1	0	1	2	3	4	5	6
T	2	1	0	1	2	3	4	5
G	3	2	1	0	1	2	3	4
G	4	3	2	1	1	1	2	3

PAM as Distance Measure

- Definition
  Let *S_1, S_2 be two protein sequences with $|S_1|=|S_2|$. We say S_1 and S_2 are x PAM distant, iff S_1 most probably was produced from S_2 with x mutations per 100 AAs*
- Remarks
  - PAM is motivated by evolution
  - Assumptions: Mutations happen with the same rate at every position of a sequence
  - If mutation rate is high, mutations will occur again and again at the same position
  - PAM ≠ %-sequence-identity

Example

$S_{1,1}$: **ACGTGAC**
$S_{2,1}$: **AGGTGCC**

$S_{1,2}$: **GTTAGTA**
$S_{2,2}$: **TTTAGTA**

$S_{1,3}$: **GGTCA**
$S_{2,3}$: **AGTCA**

Relative frequencies

A: 10/38	C: 6/38	G: 11/38	T: 11/38

$$M_x(i, j) = \log\left(\frac{f(i, j)}{f(i) * f(j)}\right)$$

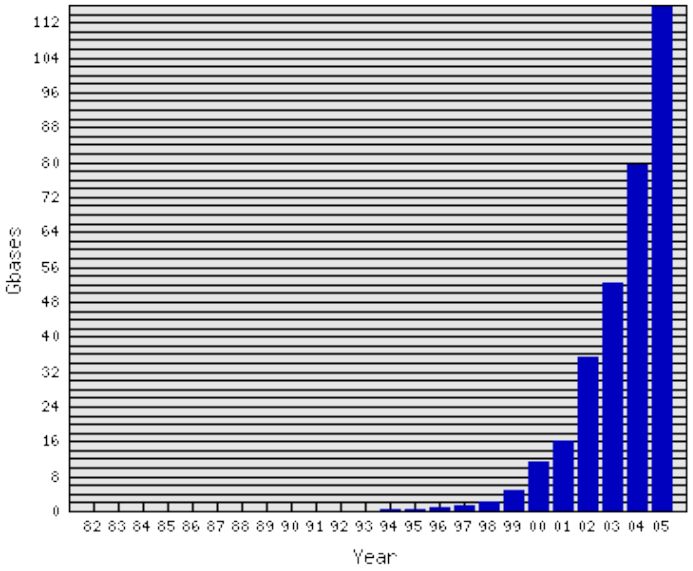
Mutation rates

	A	C	G	T
A	4/19	1/19	1/19	0/19
C		2/19	1/19	0/19
G			4/19	1/19
T				5/19

Matrix

	A	C	G	T
A	0,48	0,10	-0,16	-
C		0,63	0,06	-
G			0,40	-0,20
T				0,50

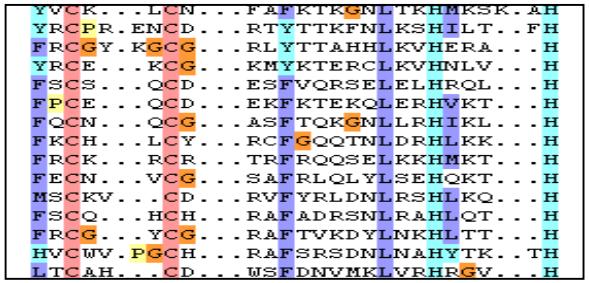
Searching a Database of Strings

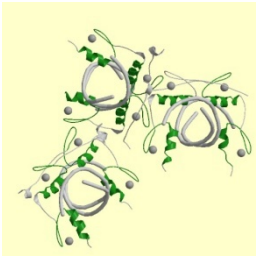


- Comparing two sequences is costly
- Given s, assume we want to find the most similar s' in a database of all known sequences
  - Naïve: Compare s with all strings in DB
  - Will take years and years
- BLAST: Basic local alignment search tool
  - Ranks all strings in DB according to similarity to s
  - Similarity: High is s, s' contain substrings that are highly similar
  - Heuristic: Might miss certain similar sequences
  - Extremely popular: You can "blast a sequence"

Multiple Sequence Alignment

- Given a set S of sequences: Find an arrangement of all strings in S in columns such that there are (a) few columns and (b) columns are maximally homogeneous
  - Additional spaces allowed

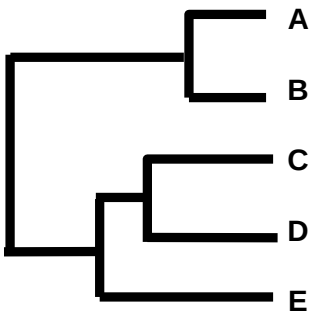




Source: Pfam, Zinc finger domain

- Goal: Find commonality between a set of functionally related sequences
  - Proteins are composed of different functional domains
  - Which domain performs a certain function?

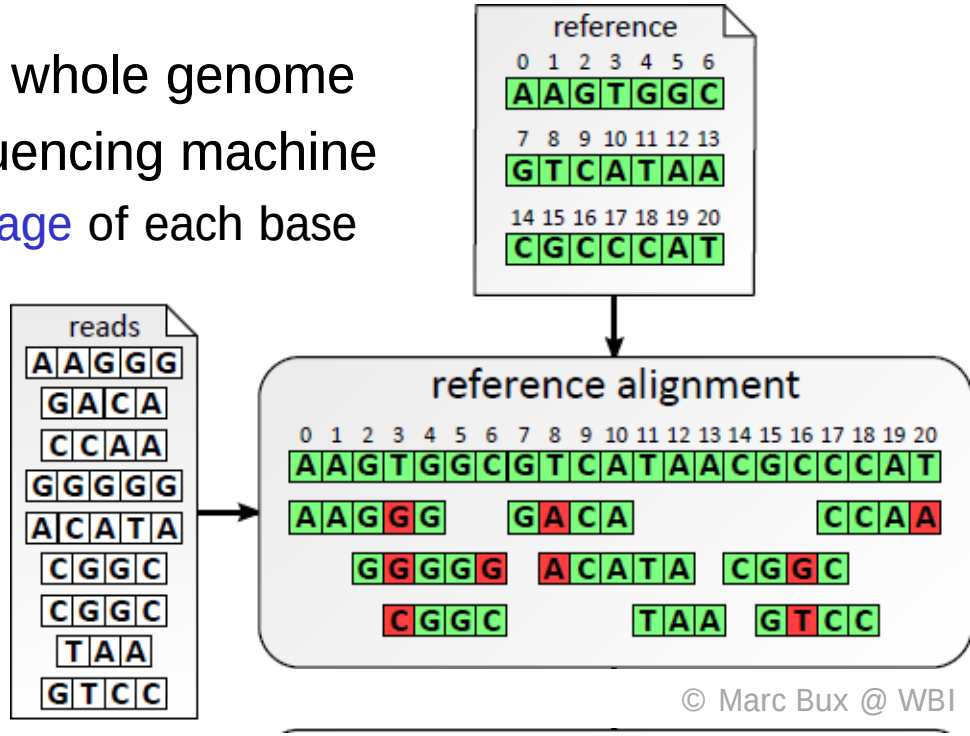
Example




```
C   PADKTNVKAAWGKVGAHAGEYGA
D   AADKTNVKAAWSKVGGHAGEYGA

A   PEEKSAVTALWGKVNVDEYGG
B   GEEKAAVLALWDKVNEEEYGG

C   PADKTNVKAAWG_KVGAHAGEYGA
D   AADKTNVKAAWS_KVGGHAGEYGA
E   AA__TNVKTAWSSKVGGHAPA__A

 A   PEEKSAV_TALWG_KVN__VDEYGG
 B   GEEKAAV_LALWD_KVN__EEEYGG
 C   PADKTNVKAA_WG_KVGAHAGEYGA
 D   AADKTNVKAA_WS_KVGGHAGEYGA
 E   AA__TNVKTA_WSSKVGGHAPA__A
```

Read Mapping

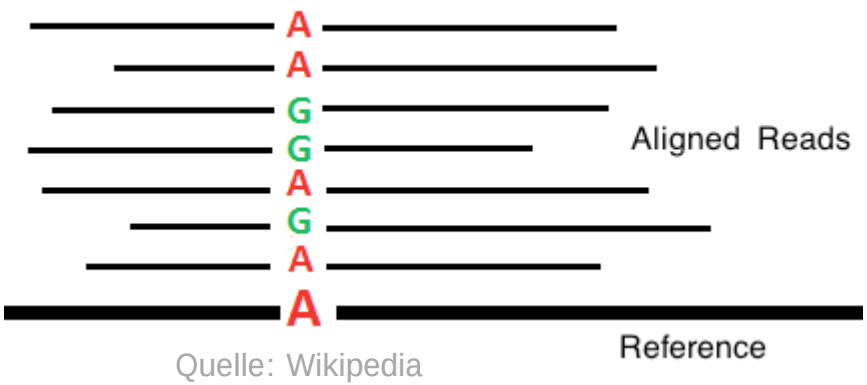
- A sequencing machine outputs short sequence reads
  - Not whole genome or chromosome as one long sequence
- Need to reconstruct to whole sequence from the reads
- General Approach:
  - Given a reference build of the whole genome
  - Given the reads from the sequencing machine
    - With a certain depth of coverage of each base
  - Find the best alignment for each read within the reference sequence
  - Together with information about matches, mismatches indels
    - Similar to an edit script



Variant Calling: Problem definition

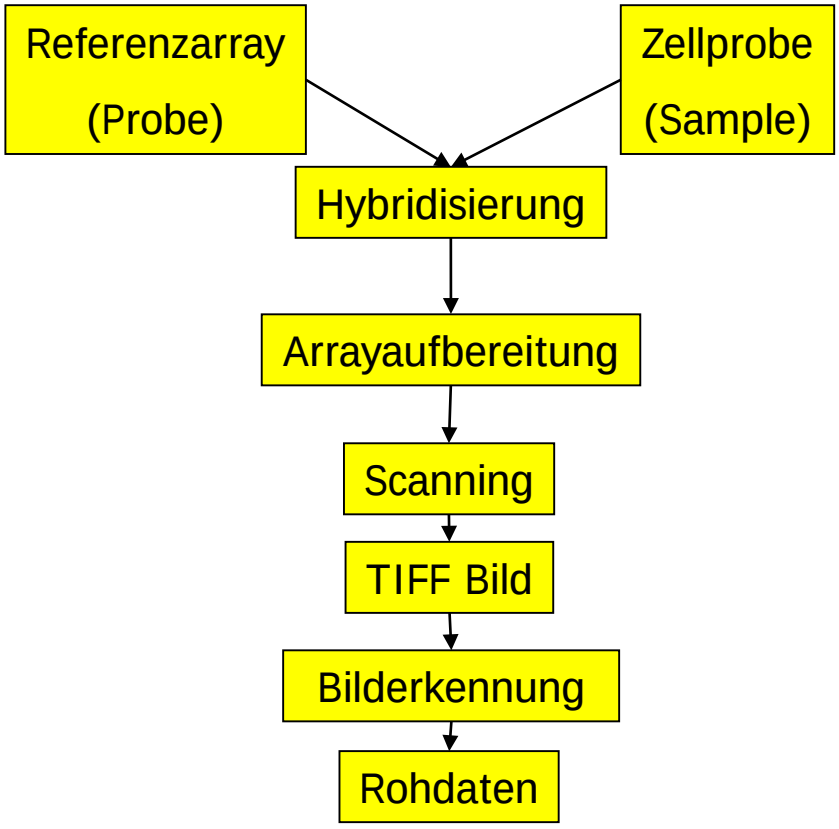
- Input for each position
  - A column of bases (cmp *coverage*)
  - Mapping quality score for each read
  - Base call quality for each position in the read (from sequencing)

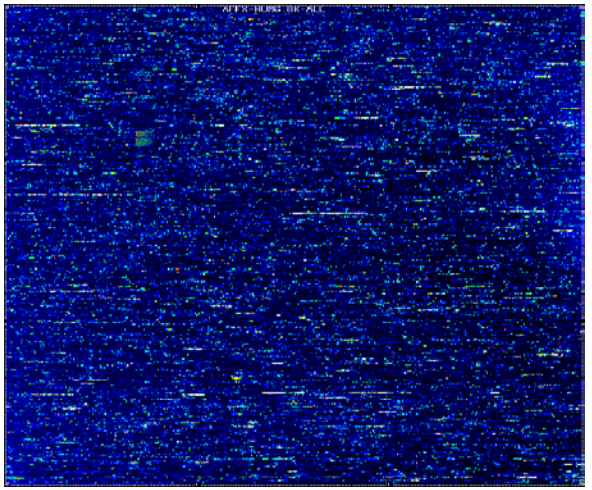
- Output for each position:
  - Whether the genomic position is
    - Homozygous wildtype (as per reference)
    - Heterozygous
    - Homozygous variant



Quelle: Wikipedia

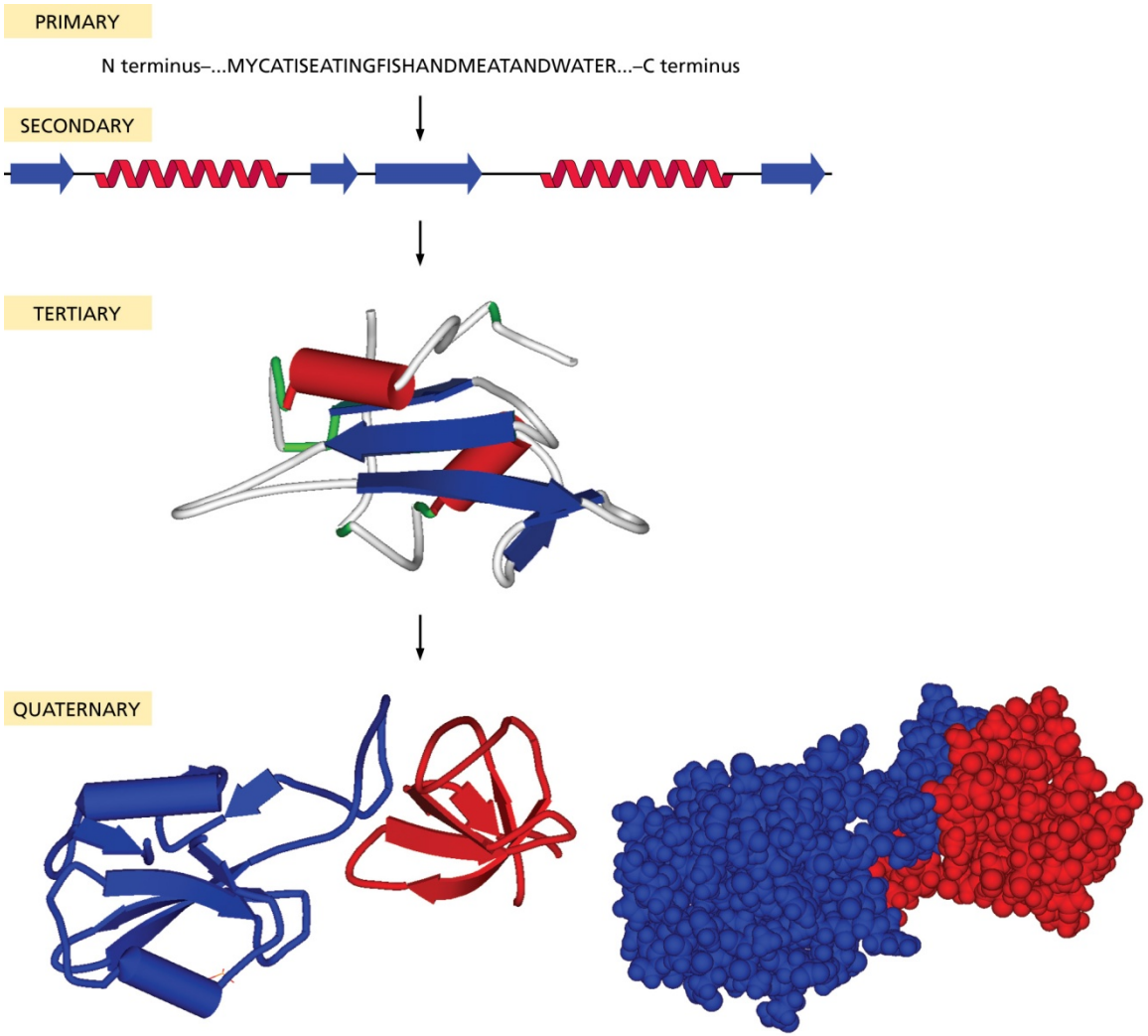
Microarrays / Transcriptomics





Protein Structure

- Primary
  - 1D-Seq. of AA
- Secondary
  - 1D-Seq. of "subfolds"
- Tertiary
  - 3D-Structure
- Quaternary
  - Assembled complexes



Predicting Secondary Structure

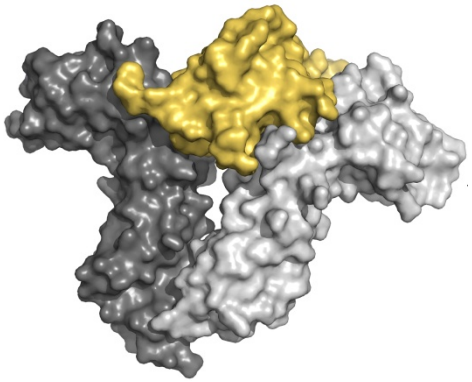
- SSP: Given a protein sequence, assign each AA in the sequence to one of the three classes Helix (H), Strand (E), or Coil (_)

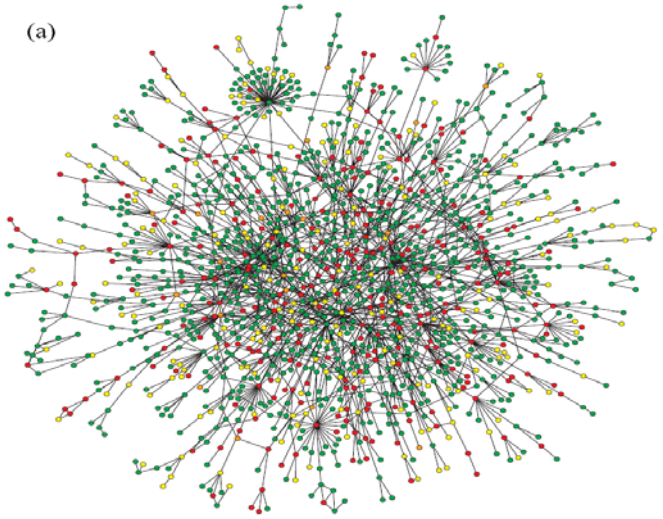
```
KVYGRCELAAAMKRLGLDNYRGYSLGNWVCAAKFESNFNTHATNRNTD
GSTDYGILQINSRWWCNDGRTPGSKNLCNIPCSALLSSDITASVNCAK
        KIASGGNGMNAWVAWRNRCKGTDVHAWIRGCRL
```

```
KVYGRCELAAAMKRLGLDNYRGYSLGNWVCAAKFESNFNTHATNRNTD
-----HHHHHHHHH------------EEEEE---------------
GSTDYGILQINSRWWCNDGRTPGSKNLCNIPCSALLSSDITASVNCAK
----EEEEEE--------------------------------HHHHHH
        KIASGGNGMNAWVAWRNRCKGTDVHAWIRGCRL
        HHH-------EEE--------------------
```

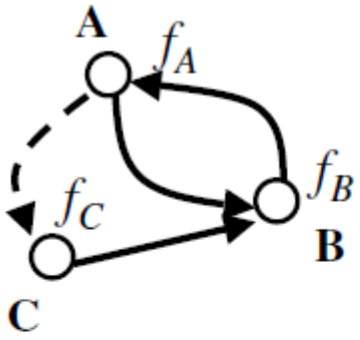
Protein-Protein-Interactions

- Proteins do not work in isolation but interact with each other
  - Metabolism, complex formation, signal transduction, transport, ...
- PPI networks
  - Neighbors tend to have similar functions
  - Interactions tend to be evolutionary conserved
  - Dense subgraphs (cliques) tend to perform distinct functions
  - Are not random at all





Regulartory Network Reconstruction

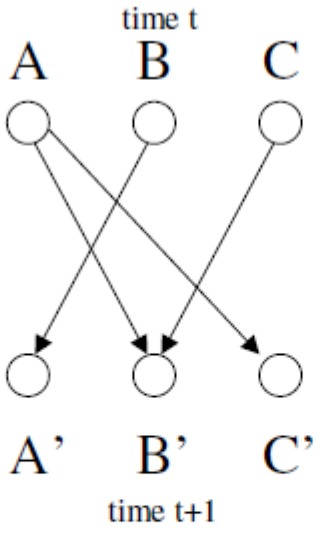


$f_A(B) = B$

$f_B(A, C) = A \text{ and } C$

$f_C(A) = \text{not } A$

Boolean Network



Wiring Diagram

State	INPUT			OUTPUT		
	A	B	C	A'	B'	C'
1	0	0	0	0	0	1
2	0	0	1	0	0	1
3	0	1	0	1	0	1
4	0	1	1	1	0	1
5	1	0	0	0	0	0
6	1	0	1	0	1	0
7	1	1	0	1	0	0
8	1	1	1	1	1	0

Transition table

Source: Filkov, „Modeling Gene Regulation“, 2003

Topics Not Covered

- Phylogenetic algorithms
- Gene prediction
- Protein 3D-structure prediction
- Docking
- RNA Seq
- Genotype / Phenotype association studies (GWAS)
- Biological Databases
- Machine Learning in Life Science Data
- ...

Evaluation

Klausur

- Zulassung
- Bücher versus Folien
- Lerngruppen
- Ablauf Klausur
- Ergebnisse
- Klausureinsicht
- Wiederholungen

Klausurtermin

- Montag, 14.8.2017, 11-14 (11.30 – 13.30) Uhr
- Raum: 3.001
- Keine Hilfsmittel erlaubt

- Anmelden
- Übungsschein

Wiederholungstermine

- Mündliche Wiederholungsprüfungen
- Termine: 18. / 19. / 20.9.2017
- Anmeldung ab 21.8.

Wissensmanagement in der Bioinformatik

- Research: Scientific database systems, Biomedical Text Mining, Statistical analysis, Scientific Workflows
- Our topics in teaching
  - Bachelor
    - Grundlagen der Bioinformatik
    - Information Retrieval
    - Seminare, Semesterprojekte
  - Master
    - Algorithmische Bioinformatik
    - Data Warehousing und Data Mining
    - Informationsintegration
    - Maschinelle Sprachverarbeitung
    - Implementierung von Datenbanksystemen
    - Seminare

- Wenn Sie beim Lernen Fragen haben – Mail
- Wenn Sie beim Lernen Fehler in den Folien finden – Mail

- Viel Erfolg bei der Klausur