Advanced Methods for Sequence Analysis

Gunnar Rätsch

Friedrich Miescher Laboratory, Tübingen

Vorlesung WS 2006/2007
Eberhard-Karls-Universität Tübingen

14 February, 2007

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

Today

- Two applications
  - Learning parameters for spliced alignments
  - Analysis of Resequencing Data
- Summary of previous lectures

Max-Margin Structured Output Learning

- Learn function $f(\mathbf{y}|\mathbf{x})$ scoring segmentations $\mathbf{y}$ for $\mathbf{x}$
- Maximize $f(\mathbf{y}|\mathbf{x})$ w.r.t. $\mathbf{y}$ for prediction:

$$\operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}^*} f(\mathbf{y}|\mathbf{x})$$

- Given N sequence pairs $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)$ for training
- Determine f such that there is a large margin between true and wrong segmentations

$$\begin{aligned} \min_f \quad & C\sum_{n=1}^{N}\xi_n + \mathbf{P}[f] \\ \text{w.r.t.} \quad & f(\mathbf{y}_n|\mathbf{x}_n) - f(\mathbf{y}|\mathbf{x}_n) \geq 1 - \xi_n \\ & \text{for all } \mathbf{y}_n \neq \mathbf{y} \in \mathcal{Y}^*, n = 1, \ldots, N \end{aligned}$$

- Exponentially many constraints!

Joint Feature Map

Recall the kernel trick

For each kernel, there exists a corresponding feature mapping $\Phi(\mathbf{x})$ on the inputs such that $k(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle$.

Joint kernel on $\mathcal{X}$ and $\mathcal{Y}$

We define a joint feature map on $\mathcal{X} \times \mathcal{Y}$, denoted by $\Phi(\mathbf{x}, y)$. Then the corresponding kernel function is

$$k((\mathbf{x}, y), (\mathbf{x}', y')) := \langle \Phi(\mathbf{x}, y), \Phi(\mathbf{x}', y') \rangle.$$

For multiclass

For normal multiclass classification, the joint feature map decomposes and the kernels on $\mathcal{Y}$ is the identity, that is

$$k((\mathbf{x}, y), (\mathbf{x}', y')) := [[y = y']] k(\mathbf{x}, \mathbf{x}').$$

Algorithm

1. $\mathcal{Y}_n^1 = \emptyset$, for $n = 1, \ldots, N$
2. Solve

$$(\mathbf{w}^t, \boldsymbol{\xi}^t) = \underset{\mathbf{w}\in\mathcal{F},\boldsymbol{\xi}}{\operatorname{argmin}} \quad C\sum_{n=1}^{N}\xi_n + \|w\|^2$$

$$\text{w.r.t.} \quad \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}_n) - \Phi(\mathbf{x}, \mathbf{y})\rangle \geq 1 - \xi_n$$

$$\text{for all } \mathbf{y}_n \neq \mathbf{y} \in \mathcal{Y}_n^t, n = 1, \ldots, N$$

3. Find violated constraints ($n = 1, \ldots, N$)

$$\mathbf{y}_n^t = \underset{\mathbf{y}_n\neq\mathbf{y}\in\mathcal{Y}^*}{\operatorname{argmax}} \langle \mathbf{w}^t, \Phi(\mathbf{x}, \mathbf{y})\rangle$$

   If $\langle \mathbf{w}^t, \Phi(\mathbf{x}, \mathbf{y}_n) - \Phi(\mathbf{x}, \mathbf{y}_n^t)\rangle < 1 - \xi_n^t$, set $\mathcal{Y}_n^{t+1} = \mathcal{Y}_n^t \cup \{\mathbf{y}_n^t\}$
4. If violated constraint exists then go to 2
5. Otherwise terminate $\Rightarrow$ Optimal solution

Current Projects

- *Ab initio* gene finding
  - Prediction of alternative splicing
  - Great improvements on *C. elegans*
- Prediction of polymorphisms from genome tiling arrays
  - Prediction of SNPs & highly polymorphic regions
  - Predicting changes in gene products
- Accurate alignments, RNA secondary structure, etc.
- Improved learning algorithms
  - Large scale ($>$10 millions of examples
  - Structured outputs (segmentations, graphs)
- Methods for understanding learning results
- Theoretical analysis of iterative algorithms

PALMA: Perfect Alignments using Large Margins

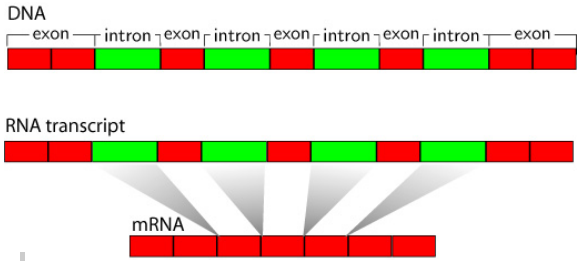
Gunnar Rätsch,[1] Bettina Hepp,[2] Uta Schulze,[1,3] and Cheng Soon Ong[1,4]

[1] Friedrich Miescher Laboratory, Tübingen
[2] Fraunhofer FIRST, Berlin,
[3] University of Leipzig
[4] Max Planck Institute for Biological Cybernetics, Tübingen

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

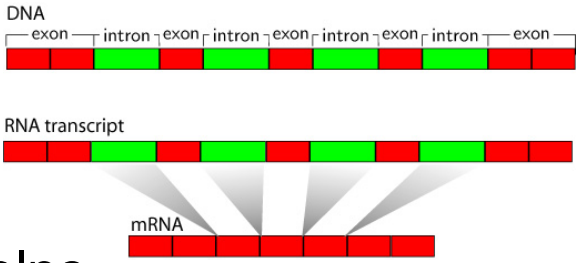
Motivation & Background

- Abundant experimental data:
  - Expressed Sequence Tags (EST)
  - Full length mRNAs
- Alignment to genomic sequences helps
  - Discovery of new genes,
  - Delineation exon/intron boundaries,
  - Identification alternative splice forms,
  - Finding SNPs, . . .
- Problems
  - Repetitive elements, paralogs, pseudo-genes
  - Sequencing errors, polymorphisms
  - Non-canonical splice sites
  - Microexons

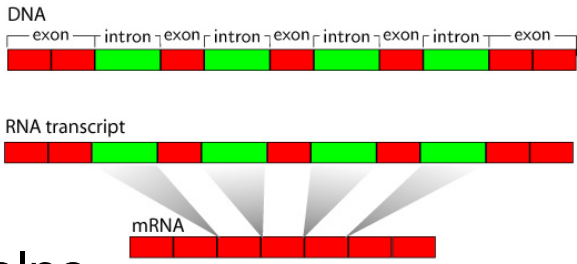


Motivation & Background

- Abundant experimental data:
  - Expressed Sequence Tags (EST)
  - Full length mRNAs
- Alignment to genomic sequences helps
  - Discovery of new genes,
  - Delineation exon/intron boundaries,
  - Identification alternative splice forms,
  - Finding SNPs, . . .
- Problems
  - Repetitive elements, paralogs, pseudo-genes
  - Sequencing errors, polymorphisms
  - Non-canonical splice sites
  - Microexons



Motivation & Background



- Abundant experimental data:
  - Expressed Sequence Tags (EST)
  - Full length mRNAs
- Alignment to genomic sequences helps
  - Discovery of new genes,
  - Delineation exon/intron boundaries,
  - Identification alternative splice forms,
  - Finding SNPs, . . .
- Problems
  - Repetitive elements, paralogs, pseudo-genes
  - Sequencing errors, polymorphisms
  - Non-canonical splice sites
  - Microexons

Previous Work

More than 10 years of research on spliced alignments

- Greedy algorithms (extend seed words or `Blast` based)
  - `Sim4` [Florea et al., 1998], `Spidey` [Wheelan et al., 2001]
  - `Blat` (prefers `AG/GT`) [Kent, 2002]
- `EST_Genome` (DP based, prefers `AG/GS`) [Mott, 1997]
- `Exalin` (DP based, `AG/GS` only) [Zhang and Gish, 2006]
  - Fixed substitution and gap costs
  - Splice site model (PWMs)

  } Maximum likelihood combination

Why another tool?

- More accurate splice site models (SVM based)
- Intron length model
- More principled combination (based on large margins)

Previous Work

More than 10 years of research on spliced alignments

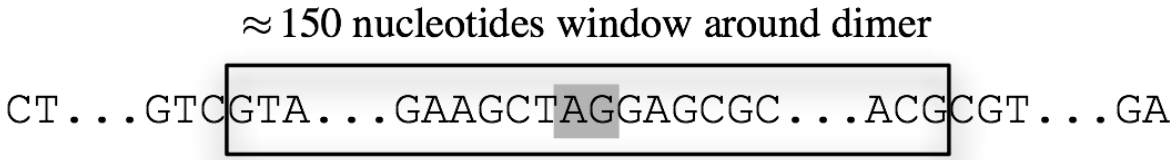
- Greedy algorithms (extend seed words or `Blast` based)
  - `Sim4` [Florea et al., 1998], `Spidey` [Wheelan et al., 2001]
  - `Blat` (prefers `AG/GT`) [Kent, 2002]
- `EST_Genome` (DP based, prefers `AG/GS`) [Mott, 1997]
- `Exalin` (DP based, `AG/GS` only) [Zhang and Gish, 2006]
  - Fixed substitution and gap costs
  - Splice site model (PWMs)

  } Maximum likelihood combination

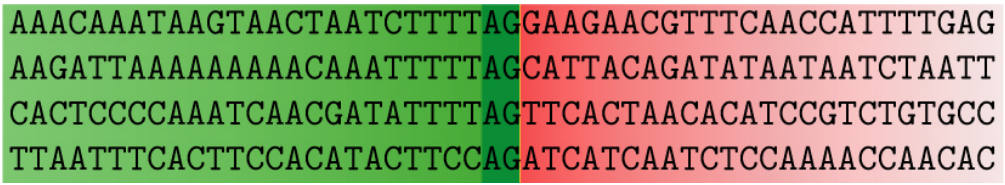
Why another tool?

- More accurate splice site models (SVM based)
- Intron length model
- More principled combination (based on large margins)

2-Class Splice Site Detection



- True sites: fixed window around a true splice site
- Decoys sites: generated by shifting the window



⇒ Very unbalanced problem (1:200)
⇒ Millions of points from EST databases
⇒ Large scale methods necessary
(here: Support Vector Machines)

2-Class Splice Site Detection

Results on *C. elegans* [Rätsch and Sonnenburg, 2004]	1-ROC Score	
	Donor	Acceptor
Higher order PWM	1.77%	1.12%
SVM w/ TOP Kernel	1.68%	1.30%
SVM w/ Polynomial	1.59%	1.05%
SVM w/ Locality Improved	*1.52%*	*0.92%*
SVM w/ Weighted Degree '04	*1.53%*	*0.95%*
SVM w/ Weighted Degree '05	*0.38%*	*0.26%*

Weighted Degree kernel is simple and accurate:

$k(s_1,s_2) = w_7 + w_1 + w_2 + w_2 + w_3$

Computes similarity between sequences



Allows training on 10 million examples [Sonnenburg et al., 2006]

Alignment Algorithms

Input

- Two sequences over the alphabet $\{A, C, G, T, N\}$
  - EST sequence S_E of length m
  - DNA sequence S_D of length n
- Substitution matrix $M : \Sigma \times \Sigma \to \mathbb{R}$,
  where $\Sigma := \{A, C, G, T, N, -\}$

Output

- Sequence alignment $\mathcal{A}$
  - Sequence of pairs, i.e. $\mathcal{A} = (a_r, b_r)_{r=1,\ldots,R}$, $a_r, b_r \in \Sigma$
  - $R \leq m + n$ depends on the alignment
- Alignment that maximizes the alignment score

$$s(\mathcal{A}) = \sum_{r=1}^{R} M(a_r, b_r)$$

Maximizing the Alignment Score

Needleman-Wunsch algorithm

- Maximizes alignment score by dynamic programming
- Fills $m \cdot n$ alignment matrix V:
  - $V(i,0) := 0$ and $V(0,j) := 0$ for all i, j
  - Recursion

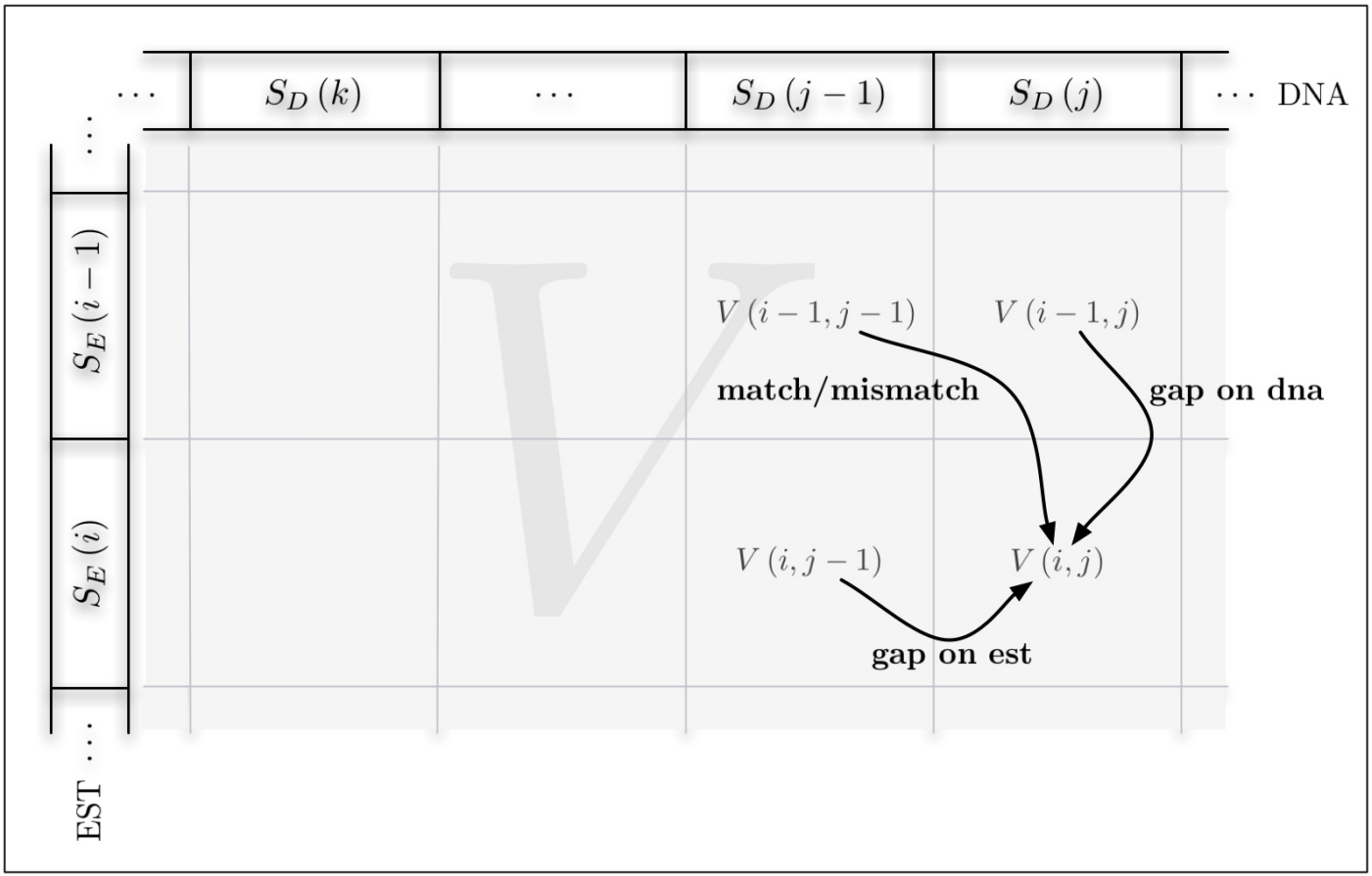
$$V(i,j) = \max \begin{cases} V(i-1,j-1) + M(S_E(i), S_D(j)) \\ V(i-1,j) + M(S_E(i), '-') \\ V(i,j-1) + M('-', S_D(j)) \end{cases}$$

- Runtime time and space complexity: $\mathcal{O}(m \cdot n)$

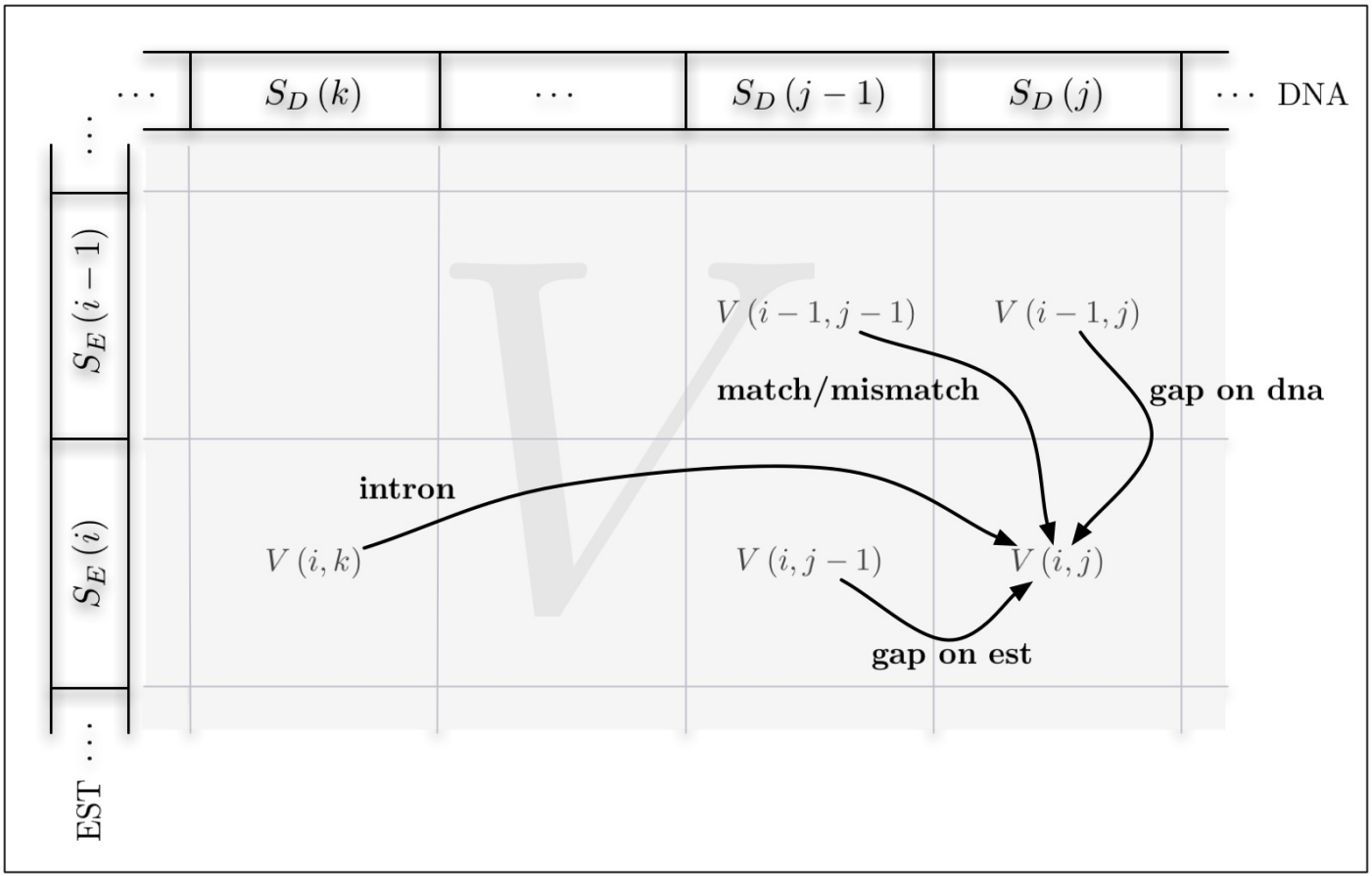
Problems:

- Does not distinguish between gaps and intron
- How to choose M? No splice site model!
- Too expensive for alignments against whole genomes

Needleman-Wunsch algorithm



Needleman-Wunsch with Introns



Recursion with Intron Model

- Extended recursion formula ($\forall i = 1 \ldots m, j = 1 \ldots n$)

$$V(i,j) = \max \begin{cases} V(i-1,j-1) + M(S_E(i), S_D(j)) \\ V(i-1,j) + M(S_E(i), '-') \\ V(i,j-1) + M('-', S_D(j)) \\ \max_{1 \le k \le j-1} (V(i,k) + f_I(k,j)) \end{cases}$$

- For intron score $f_I(k,j)$ consider
  - Splice sites scores s_k^{Don} and s_k^{Acc} (SVM predictions)
    $\Rightarrow$ Contribute $f^{\text{Don}}(s_k^{\text{Don}}) + f^{\text{Acc}}(s_k^{\text{Acc}})$
  - Length of intron
    $\Rightarrow$ Contributes $f^{\text{Len}}(j-k)$
- Unspecified functions $f^{\text{Don}}, f^{\text{Acc}}, f^{\text{Len}}$ as well as M!

Idea: Learn functions on training set with known alignments

Parameterization

- Substitution matrix $M : \Sigma \times \Sigma \to \mathbb{R}$
- Functions f^{Len}, f^{Acc} and f^{Don}
  - Piecewise linear functions (support points $x_1, \ldots, x_s$):

$$f(x) = \begin{cases} \theta_1 & x \leq x_1 \\ \frac{\theta_i(x_{i+1}-x)+\theta_{i+1}(x-x_i)}{x_{i+1}-x_i} & x_i \leq x \leq x_{i+1} \\ \theta_s & x \geq x_s \end{cases}.$$

  - $\boldsymbol{\theta} := (\theta_1, \ldots, \theta_s)$ parametrizes function
- Let $\boldsymbol{\theta} := (\boldsymbol{\theta}^{\text{Acc}}, \boldsymbol{\theta}^{\text{Don}}, \boldsymbol{\theta}^{\text{Len}}, \boldsymbol{\theta}^{M})$
- Given $\boldsymbol{\theta}$, alignment score $s_{\boldsymbol{\theta}}(\mathcal{A})$ is fully defined

Parameters: Optimization

Idea

Find $\boldsymbol{\theta}$ such that for a known alignment $\mathcal{A}^+$

$$s_{\boldsymbol{\theta}}(\mathcal{A}^+) \gg s_{\boldsymbol{\theta}}(\mathcal{A}^-)$$

where $\mathcal{A}^- \neq \mathcal{A}^+$ is any wrong alignment

Given N known alignments $\mathcal{A}_i^+, i = 1, \ldots, N$

Solve quadratic optimization problem (QP)

$$\min_{\boldsymbol{\xi} \geq \mathbf{0}, \boldsymbol{\theta}} \quad \frac{1}{N} \sum_{i=1}^{N} \xi_i + \mathbf{P}(\boldsymbol{\theta})$$

$$\text{s.t.} \quad s_{\boldsymbol{\theta}}(\mathcal{A}_i^+) - s_{\boldsymbol{\theta}}(\mathcal{A}_i^-) \geq 1 - \xi_i \quad \forall \mathcal{A}_i^- \neq \mathcal{A}_i^+, i = 1, \ldots, N$$

- ξ_i: Slack-variables to implement a soft-margin
- $\mathbf{P}(\boldsymbol{\theta})$: Regularizer leading to smooth functions

Problem: Exponentially many constraints

Iterative Algorithm

- Set $\boldsymbol{\theta} := (\boldsymbol{\theta}^{\mathrm{Acc}}, \boldsymbol{\theta}^{\mathrm{Don}}, \boldsymbol{\theta}^{\mathrm{Len}}, \boldsymbol{\theta}^{M})$ randomly, $A_i^- = \emptyset$
- For $t = 1, \ldots, T$
  - For $i = 1, \ldots, N$
    - Compute (wrong) alignments $\mathcal{A}^-$ based on $\boldsymbol{\theta}$
    - If $\mathcal{A}^- \neq \mathcal{A}_i^+$, then $A_i^- := A_i^- \cup \{\mathcal{A}^-\}$
  - Obtain new parameters $\boldsymbol{\theta}$ by solving the restricted QP

$$\begin{aligned} \min_{\boldsymbol{\xi} \geq \mathbf{0}, \boldsymbol{\theta}} \quad & \frac{1}{N} \sum_{i=1}^{N} \xi_i + \mathbf{P}(\boldsymbol{\theta}) \\ \text{s.t.} \quad & s_{\boldsymbol{\theta}}(\mathcal{A}_i^+) - s_{\boldsymbol{\theta}}(\mathcal{A}^-) \geq 1 - \xi_i \quad \forall \mathcal{A}^- \in A_i^-, i = 1, \ldots, N \end{aligned}$$

- Only need to solve small optimization problems!
- Guaranteed convergence!

Microexon Simulation Study

Artificial Data

- Consider EST confirmed exon triples (*C. elegans*)
- Shorten middle exon in central region (EST & DNA)



(found by `blat` and strict filtering)

Microexon generated
$\Rightarrow$ Splice sites still intact

- Generate insertions/deletions/mutations in artificial EST
  ($\sigma = 0\%, 1\%, 2\%, 10\%, 20\%, 50\%$)
- Train `PALMA` on 4608 exon triples ($\approx 1h$)
- Test `blat`, `sim4`, `exalin` and `PALMA` on 4358 tripples
- Correct only if all boundaries are correctly predicted

Microexon Simulation Study

Artificial Data

- Consider EST confirmed exon triples (*C. elegans*)
- Shorten middle exon in central region (EST & DNA)



(found by `blat` and strict filtering)

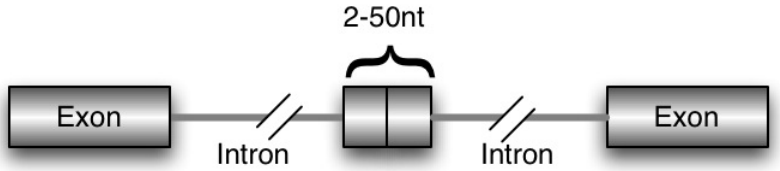
Microexon generated
$\Rightarrow$ Splice sites still intact

- Generate insertions/deletions/mutations in artificial EST
  ($\sigma = 0\%, 1\%, 2\%, 10\%, 20\%, 50\%$)
- Train `PALMA` on 4608 exon triples ($\approx 1h$)
- Test `blat`, `sim4`, `exalin` and `PALMA` on 4358 tripples
- Correct only if all boundaries are correctly predicted

Microexon Simulation Study

Artificial Data

- Consider EST confirmed exon triples (*C. elegans*)
- Shorten middle exon in central region (EST & DNA)



(found by `blat` with strict post-processing)

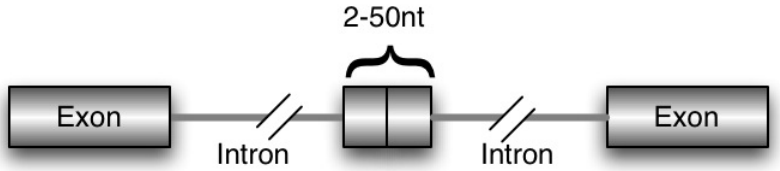
$\Rightarrow$ Microexon generated
$\Rightarrow$ Splice sites still intact

- Generate insertions/deletions/mutations in artificial EST ($\sigma = 0\%, 1\%, 2\%, 10\%, 20\%, 50\%$)
- Train `PALMA` on 4608 exon triples ($\approx 1h$)
- Test `blat`, `sim4`, `exalin` and `PALMA` on 4358 tripples
- Correct only if all boundaries are correctly predicted

Microexon Simulation Study

Artificial Data

- Consider EST confirmed exon triples (*C. elegans*)
- Shorten middle exon in central region (EST & DNA)

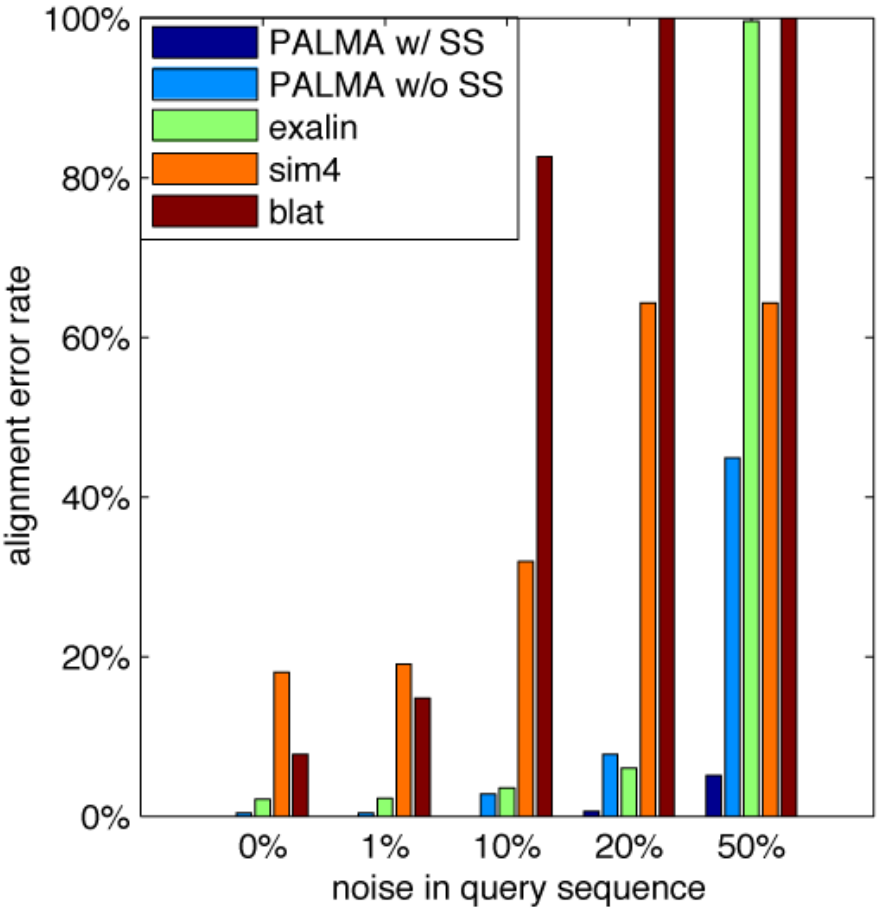


(found by `blat` with strict post-processing)

$\Rightarrow$ Microexon generated
$\Rightarrow$ Splice sites still intact

- Generate insertions/deletions/mutations in artificial EST ($\sigma = 0\%, 1\%, 2\%, 10\%, 20\%, 50\%$)
- Train `PALMA` on 4608 exon triples ($\approx 1h$)
- Test `blat`, `sim4`, `exalin` and `PALMA` on 4358 tripples
- Correct only if all boundaries are correctly predicted

Results (*C. elegans*)



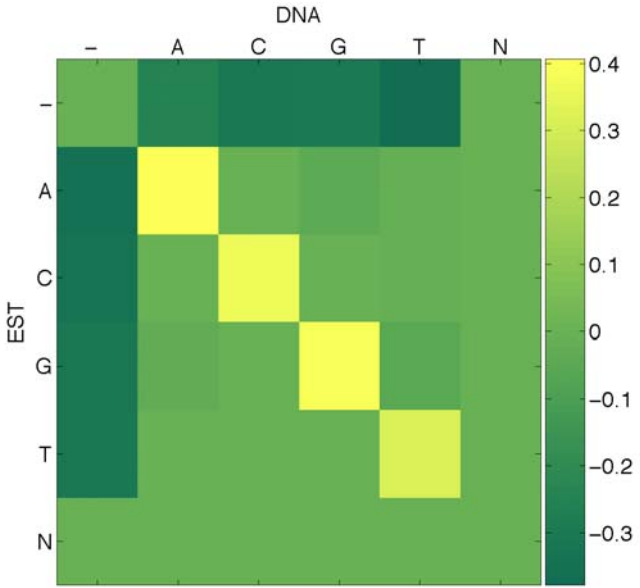
- Error rate increases drastically for `blat` and `sim4`
- `Exalin` performs quite well, but always $> 3\%$ wrong
- `PALMA` is always correct for $\sigma \leq 10\%$
- `PALMA w/o SS` only slightly worse

Conclusion

- New alignment algorithm for *perfect* alignments of mRNA and DNA
- Exploits very accurate SVM-based splice site predictions
- New idea of combining different sources of information:
  - Similarity, splice site scores and intron lengths
  - Large margin based iterative algorithm
  - Guaranteed convergence
- Significantly reduced error rates on test data (short exons/much noise)
- Better detection of microexons & altern. spliced exons
- Current work: Reduce computational complexity
- Source code (Python/C++, GPL) and data available at
  `http://www.fml.mpg.de/raetsch/projects/palma`

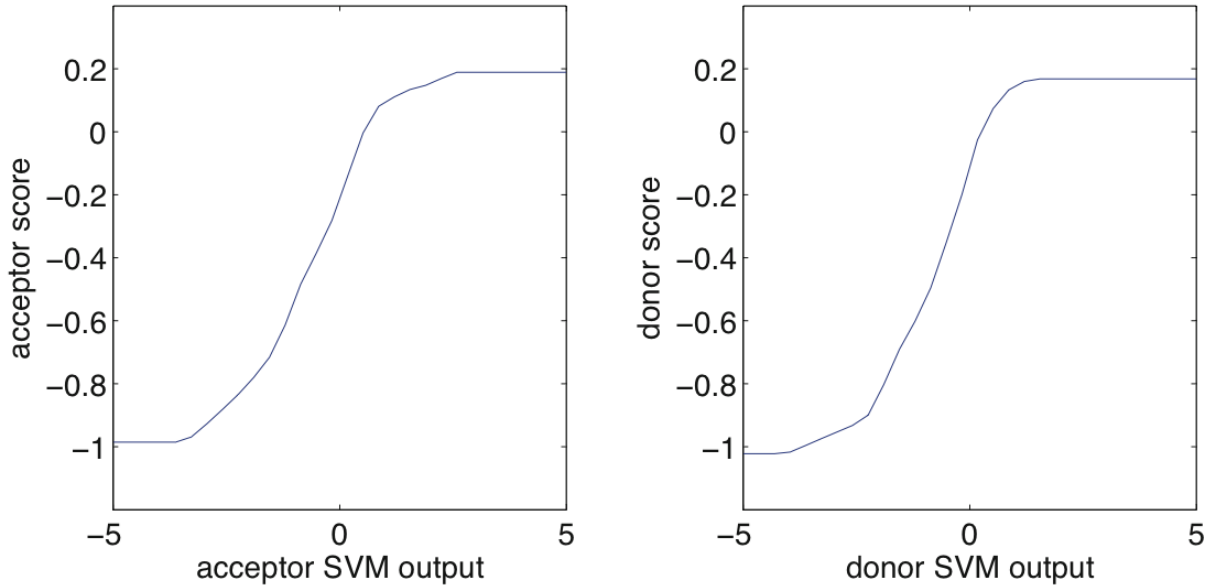
Substitution Matrix

- $M : \Sigma \times \Sigma \to \mathbb{R}$
- matches score high, gaps score low
- mismatch scores all close to zero



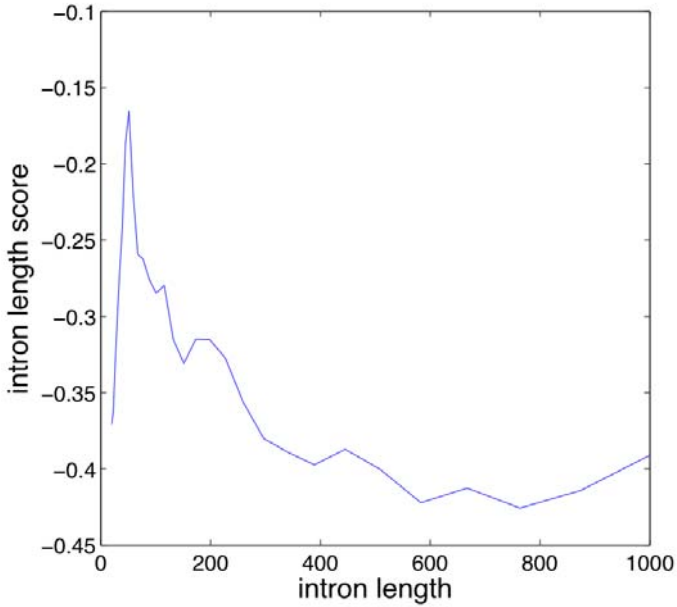
Donor/Acceptor Score

- f^{Acc} and f^{Don}
- Score the acceptor and donor SVM outputs



Intron Length Score

- f^{Len}
- Maximum near 50nt (most frequent intron length)

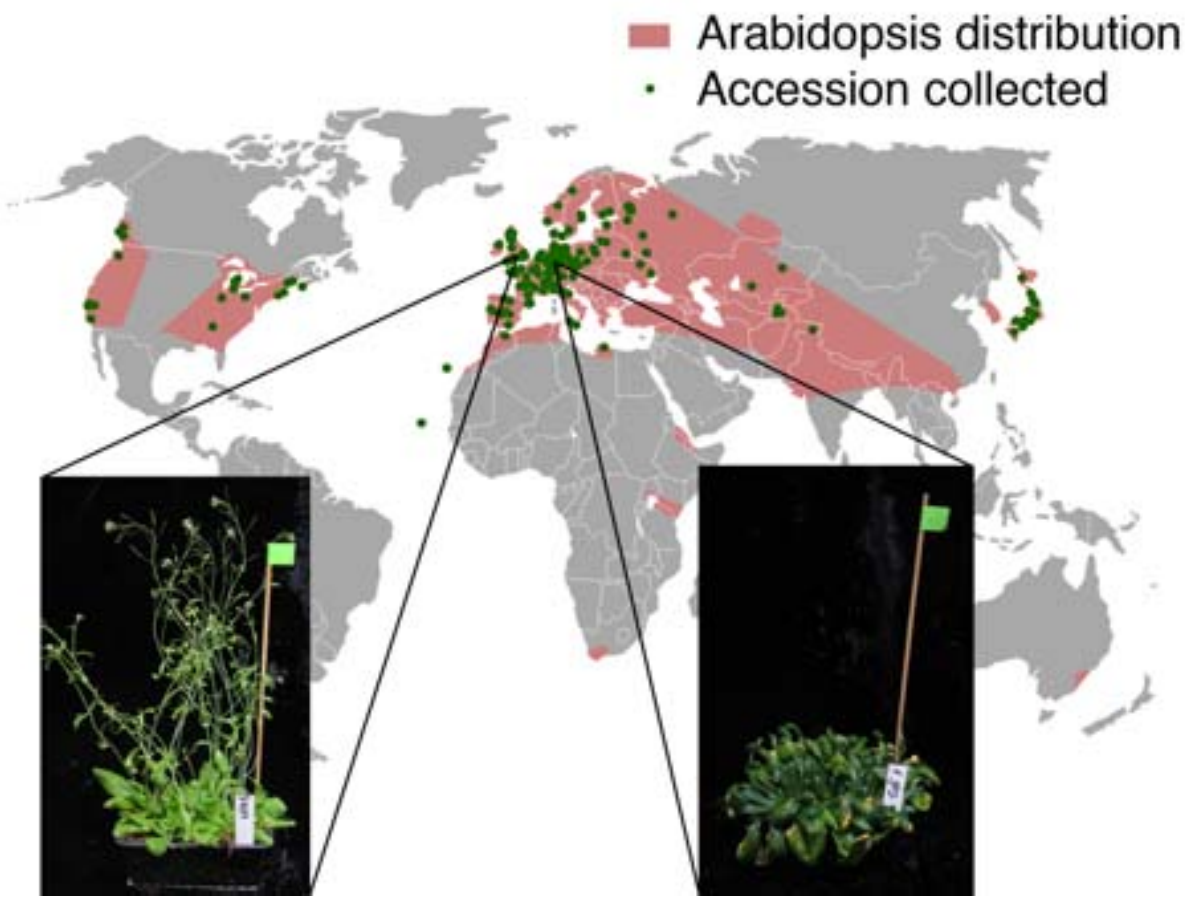


Computational Approaches to the Analysis of Whole Genome Resequencing Data

Gunnar Rätsch

Friedrich Miescher Laboratory, Tübingen

Introduction



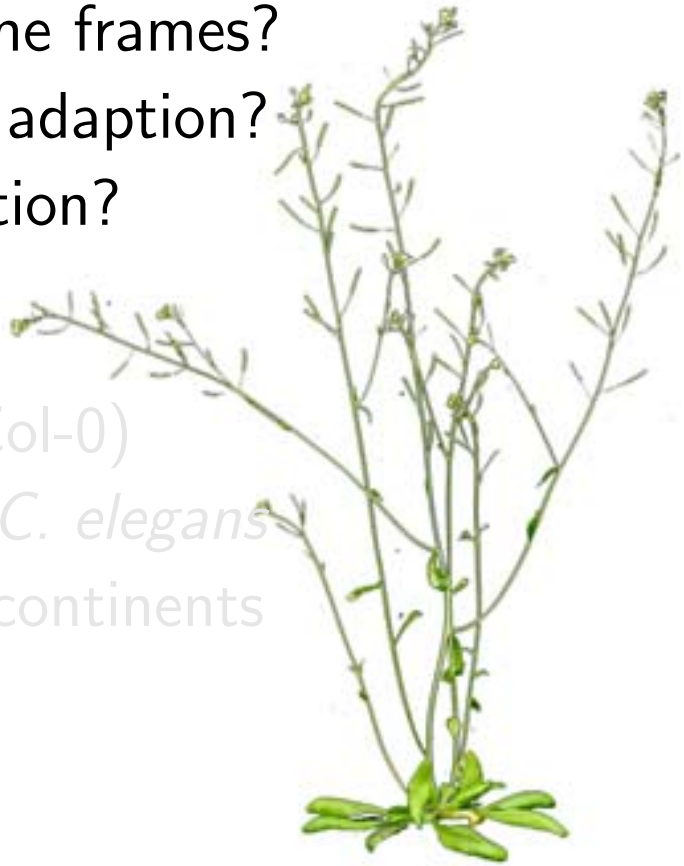
What is the genetic basis of variation?

Modified from Koornneef et al. 2004. *Ann. Rev. Plant Biology*, 55, 141-172.

Introduction

Questions:

- What sequence changes occur in short time frames?
- Which polymorphisms and genes underlie adaption?
- What are the consequences for gene function?



Arabidopsis thaliana:

- 119 Mb finished euchromatic sequence (Col-0)
- Resources comparable to *Drosophila* and *C. elegans*
- Collections of >1000 wild strains from 3 continents
- Strains are largely homozygous

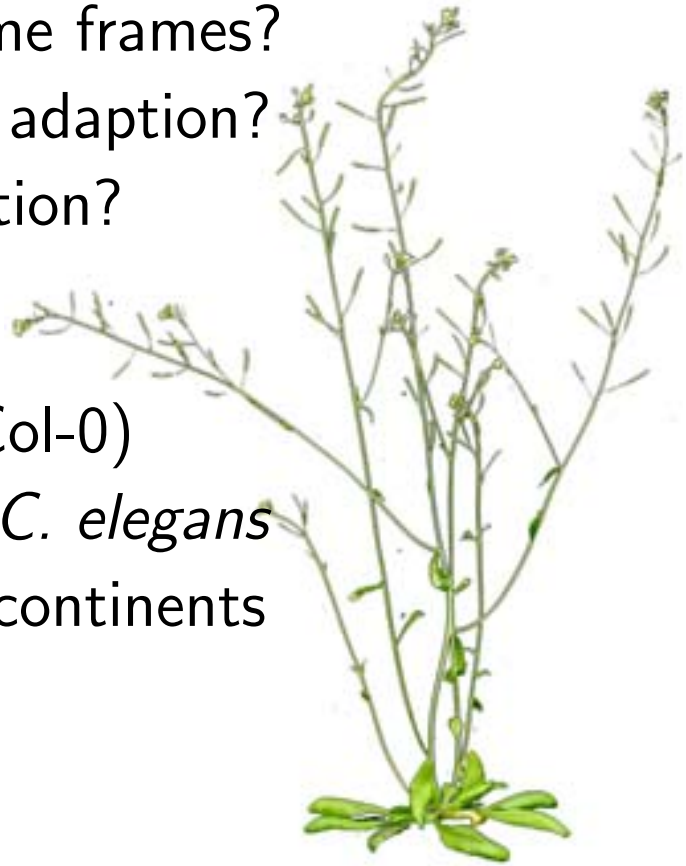
Resequencing of 20 wild strains

- genome-wide identification of sequence polymorphisms
- High-density oligo-nucleotide arrays for high-throughput resequencing

Introduction

Questions:

- What sequence changes occur in short time frames?
- Which polymorphisms and genes underlie adaption?
- What are the consequences for gene function?



Arabidopsis thaliana:

- 119 Mb finished euchromatic sequence (Col-0)
- Resources comparable to *Drosophila* and *C. elegans*
- Collections of >1000 wild strains from 3 continents
- Strains are largely homozygous

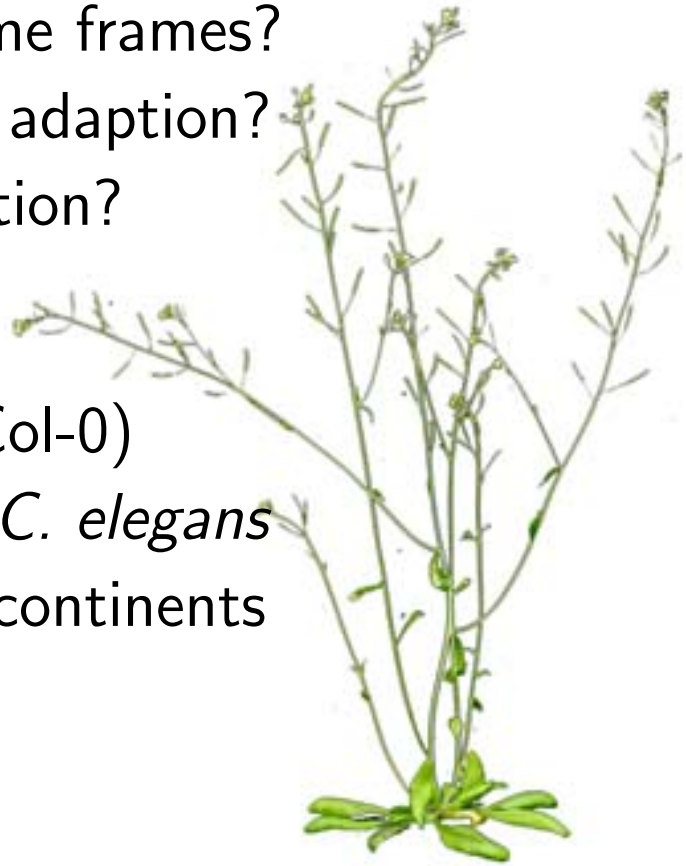
Resequencing of 20 wild strains

- genome-wide identification of sequence polymorphisms
- High-density oligo-nucleotide arrays for high-throughput resequencing

Introduction

Questions:

- What sequence changes occur in short time frames?
- Which polymorphisms and genes underlie adaption?
- What are the consequences for gene function?



Arabidopsis thaliana:

- 119 Mb finished euchromatic sequence (Col-0)
- Resources comparable to *Drosophila* and *C. elegans*
- Collections of >1000 wild strains from 3 continents
- Strains are largely homozygous

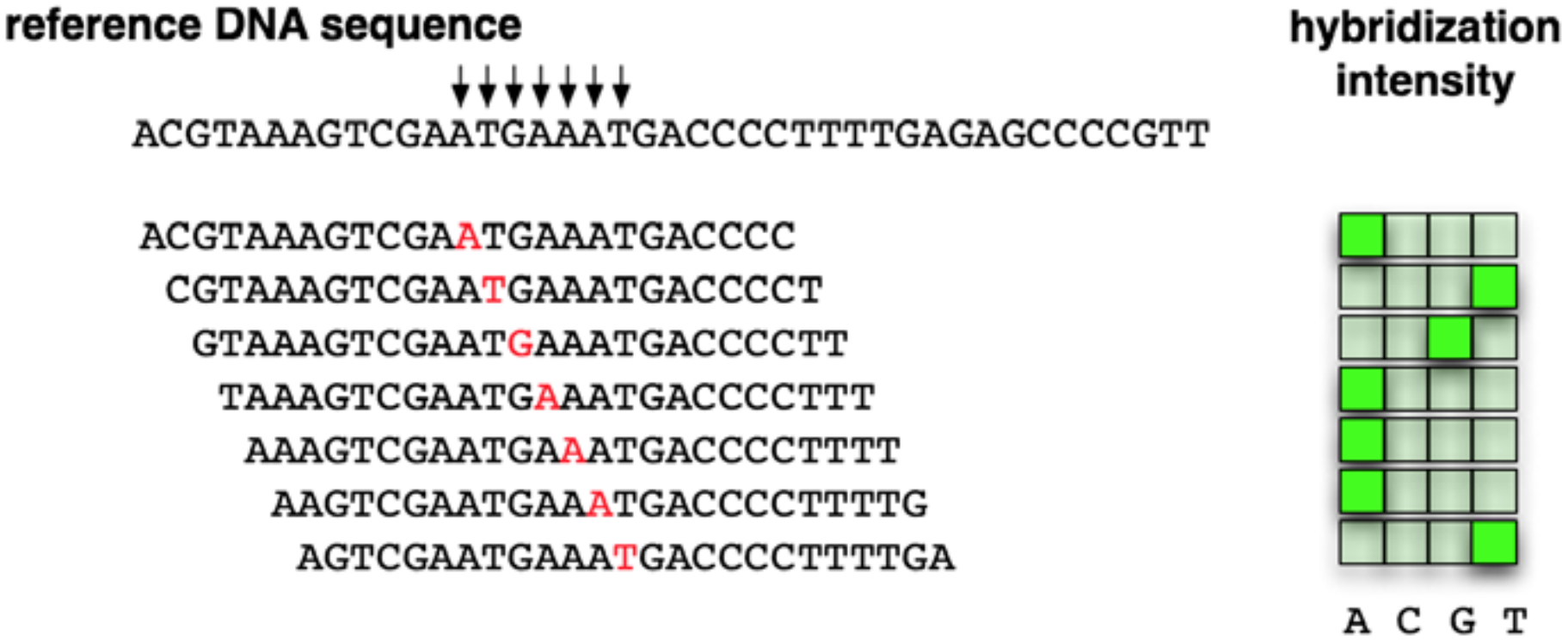
Resequencing of 20 wild strains

- genome-wide identification of sequence polymorphisms
- High-density oligo-nucleotide arrays for high-throughput resequencing

Resequencing Array Basics I

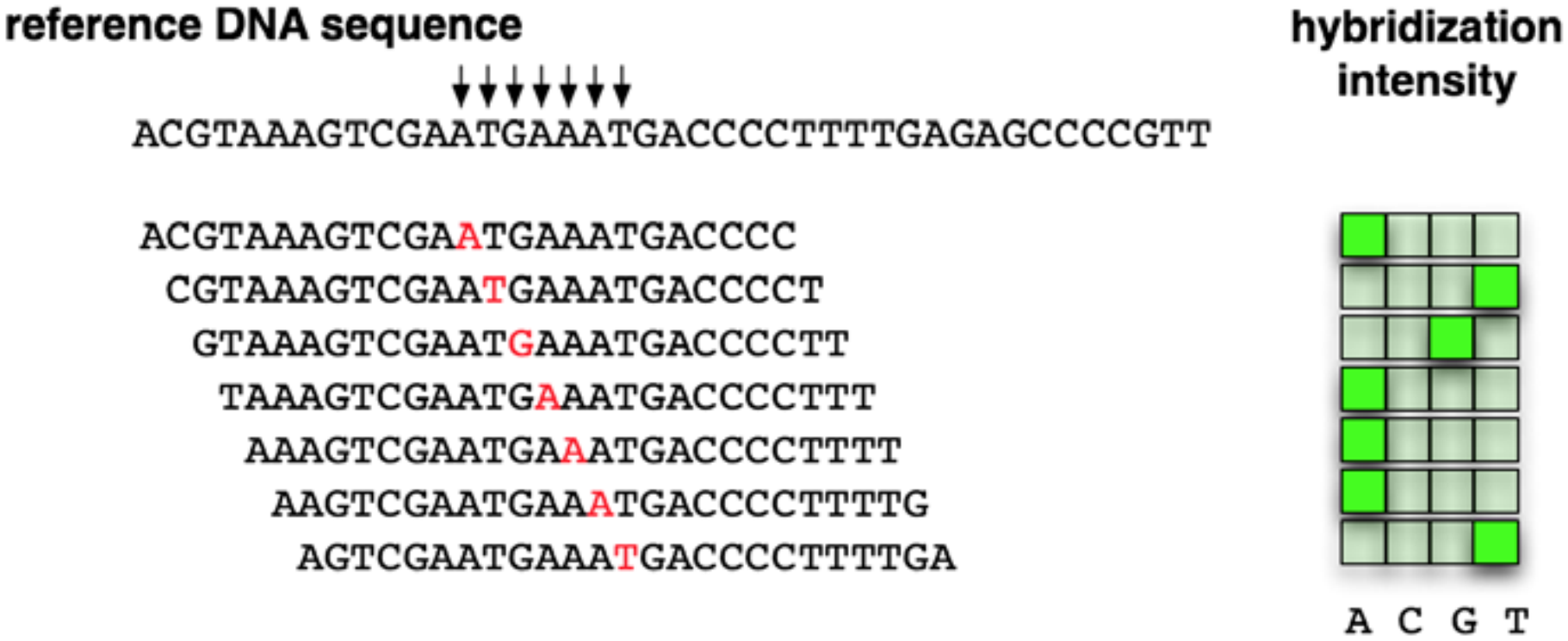


Resequencing Array Basics II



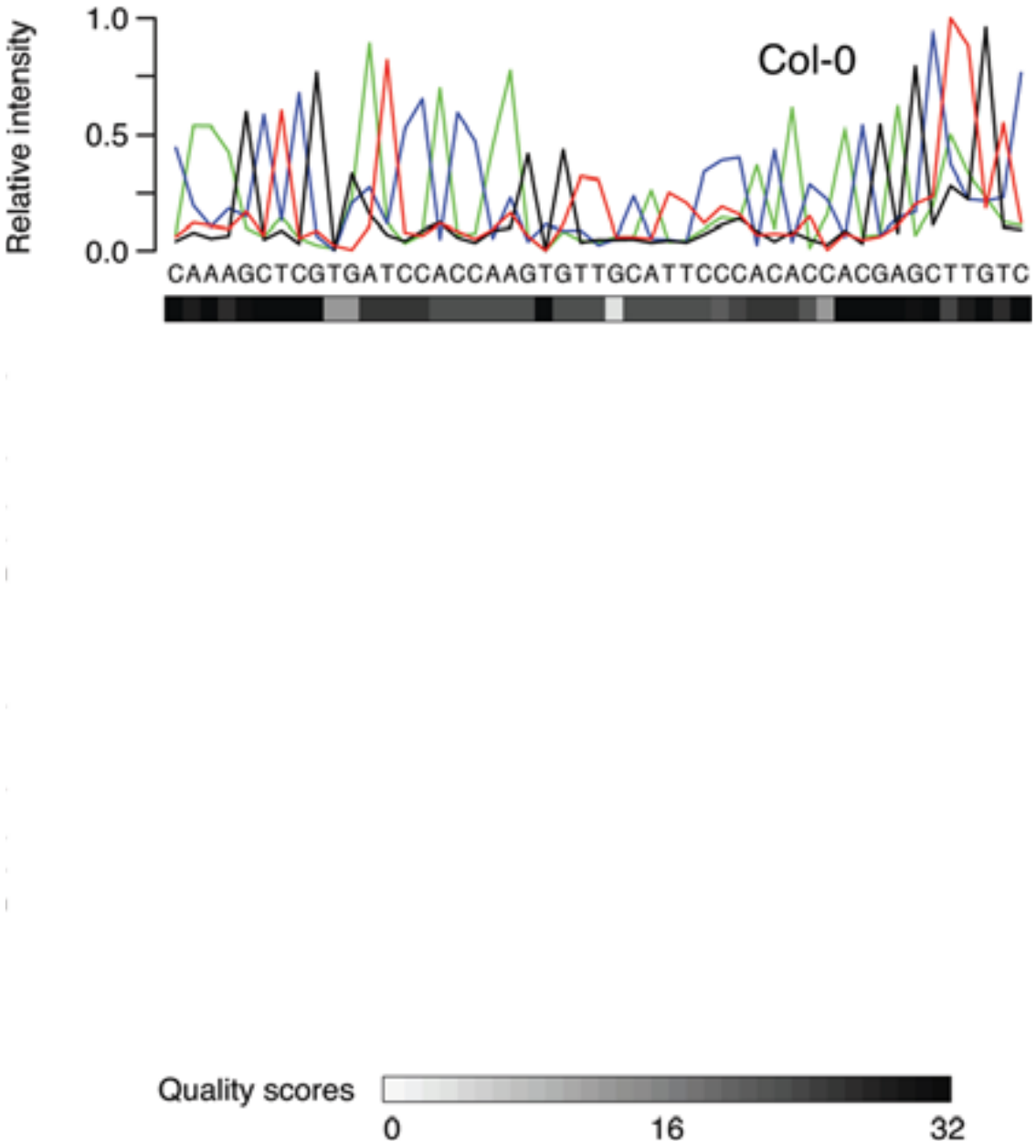
- >99.99% of bases represented
- Each base queried with forward and reverse strand probe quartets
- Nearly 1 billion oligos per ecotype
- 19+1 ecotypes surveyed representing worldwide distribution

Resequencing Array Basics II



- >99.99% of bases represented
- Each base queried with forward and reverse strand probe quartets
- Nearly 1 billion oligos per ecotype
- 19+1 ecotypes surveyed representing worldwide distribution

Resequencing Data



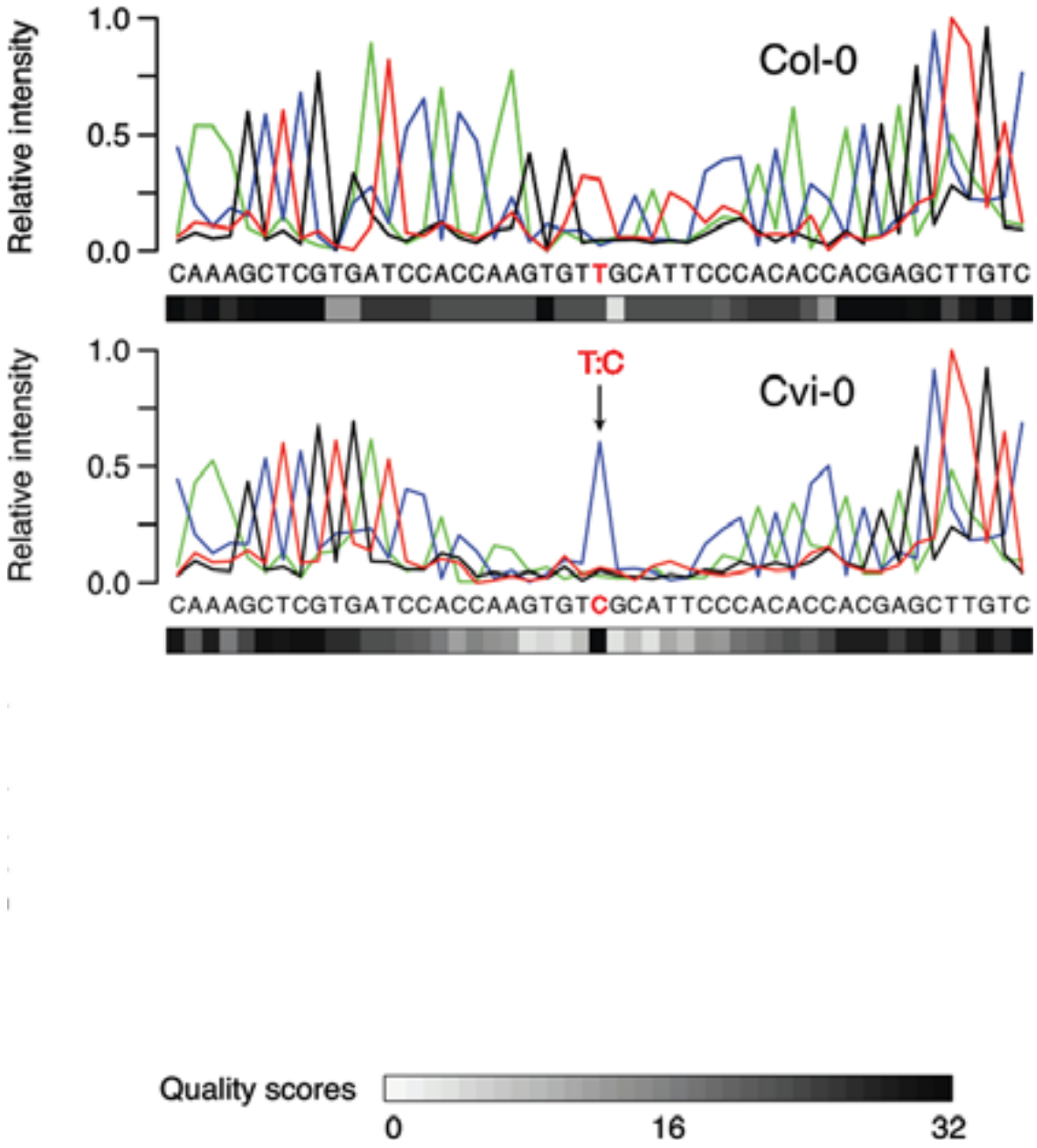
Data analysis challenge

- Hybridization intensities depend on
  - Oligomer
  - Ecotype
  - Repeats
- Measurement noise
- Identify SNPs

Problematic cases

- Highly polymorphic regions
- Deletions/insertions

Resequencing Data



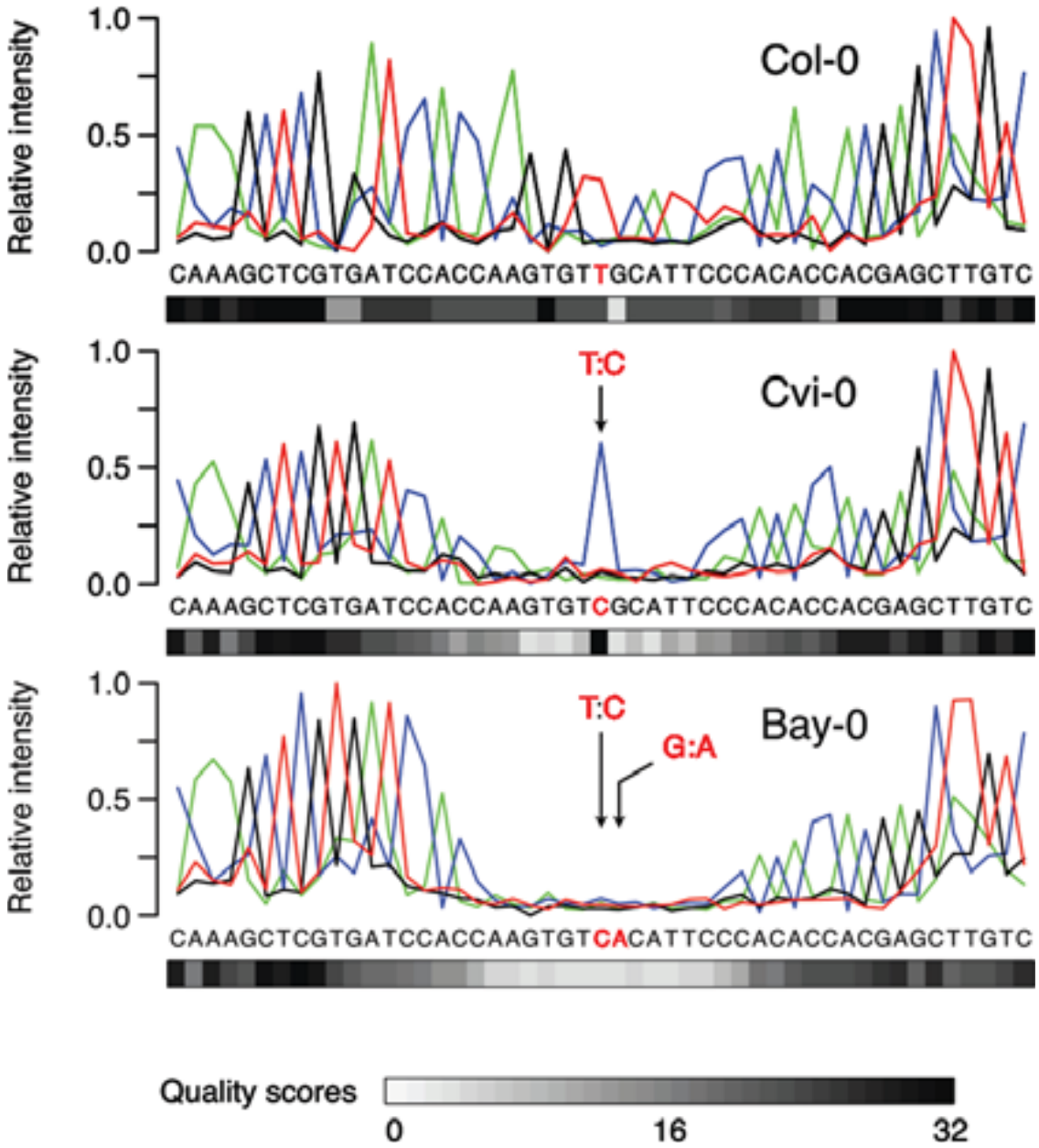
Data analysis challenge

- Hybridization intensities depend on
  - Oligomer
  - Ecotype
  - Repeats
- Measurement noise
- Identify SNPs

Problematic cases

- Highly polymorphic regions
- Deletions/insertions

Resequencing Data



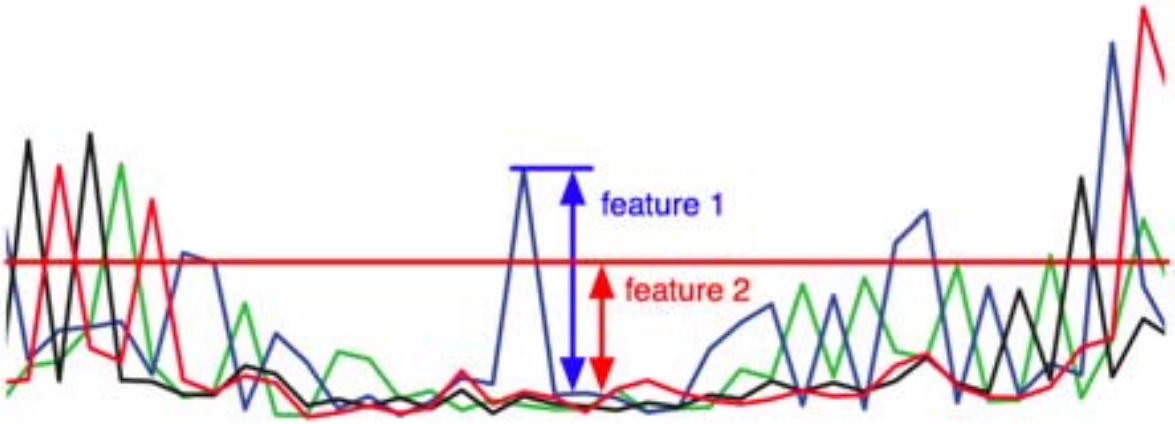
Data analysis challenge

- Hybridization intensities depend on
  - Oligomer
  - Ecotype
  - Repeats
- Measurement noise
- Identify SNPs

Problematic cases

- Highly polymorphic regions
- Deletions/insertions

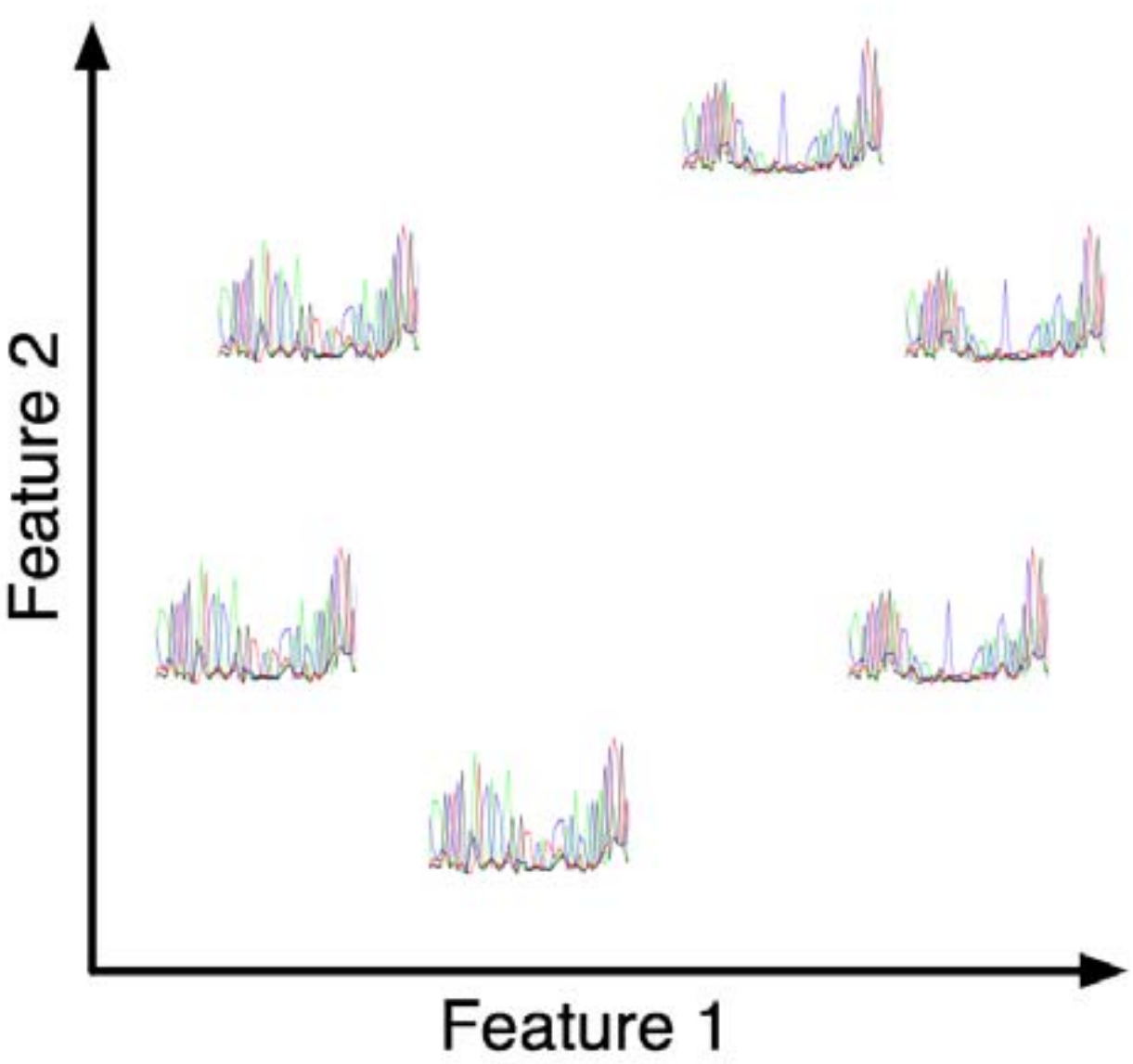
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
- Elegant theory
- "Support Vector Machines"

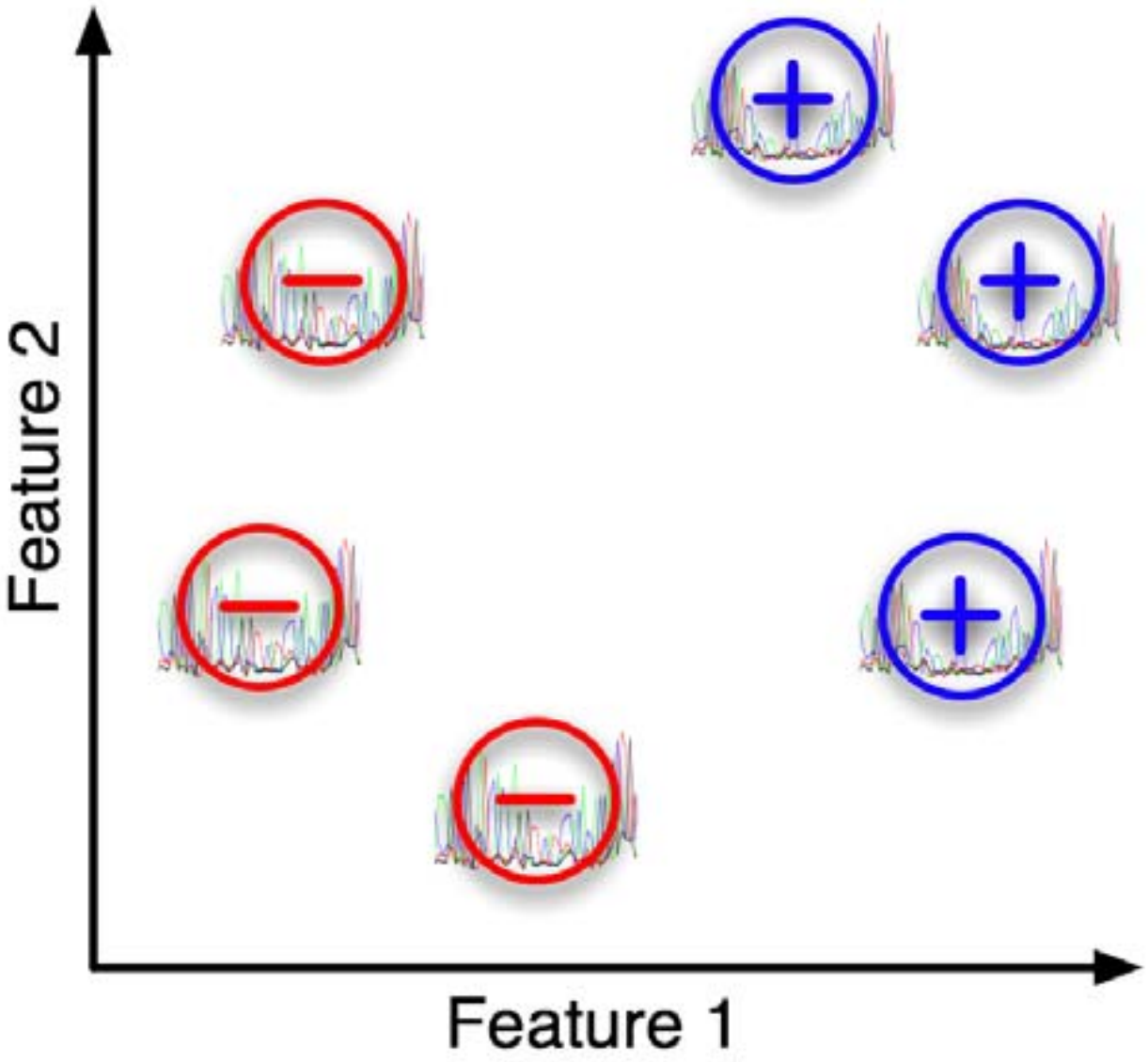
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
- Elegant theory
- "Support Vector Machines"

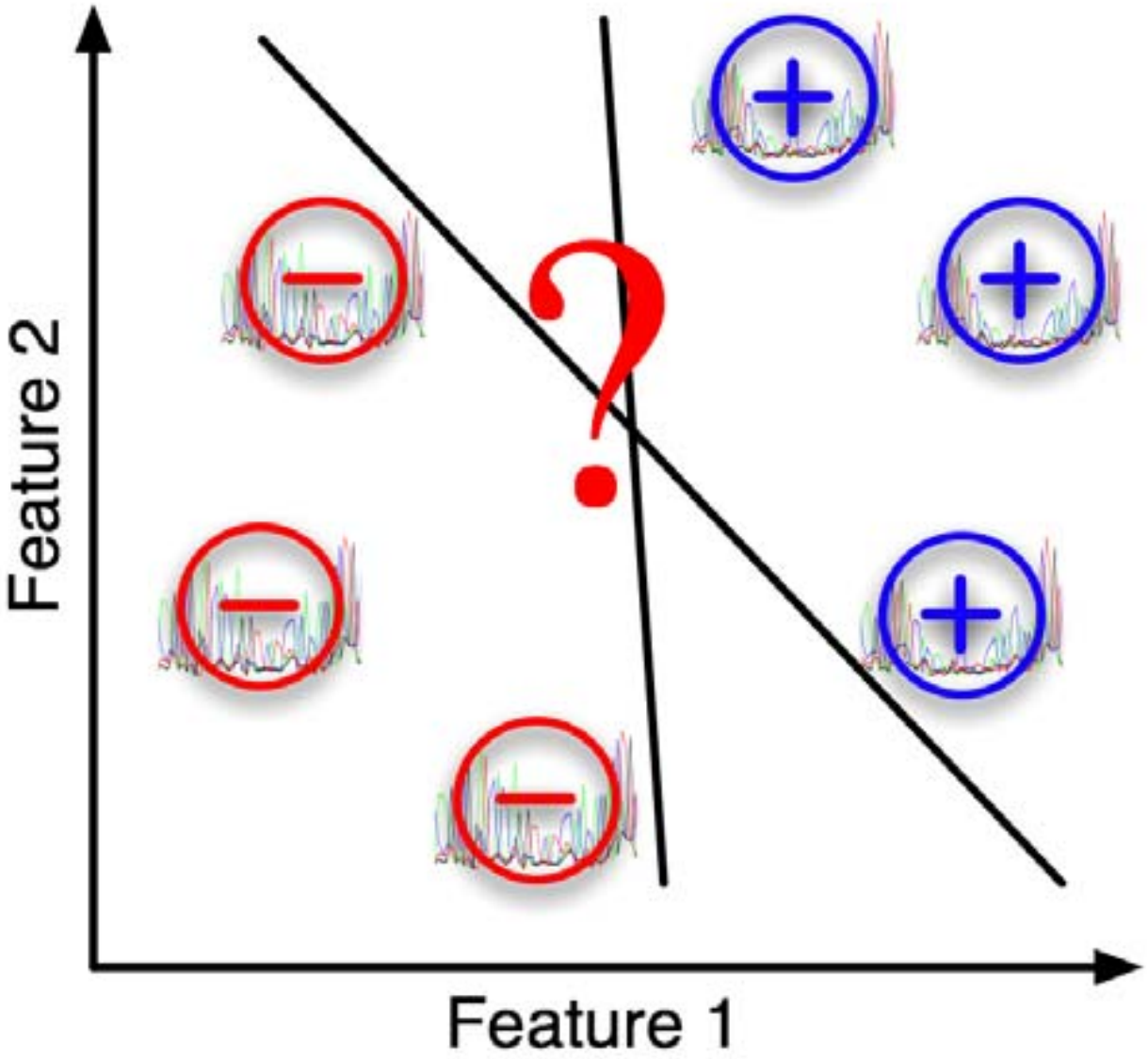
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
- Elegant theory
- "Support Vector Machines"

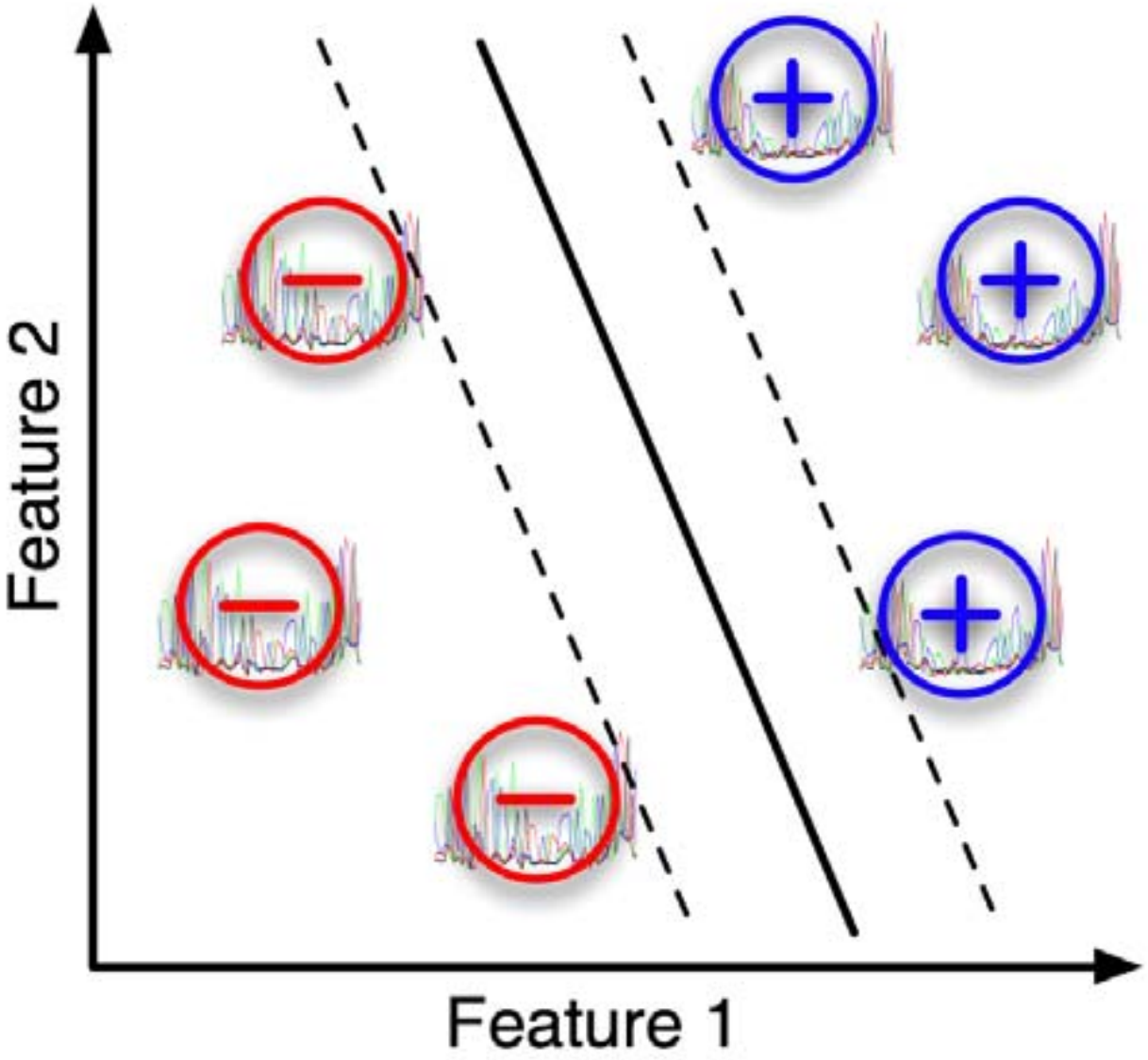
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
- Elegant theory
- "Support Vector Machines"

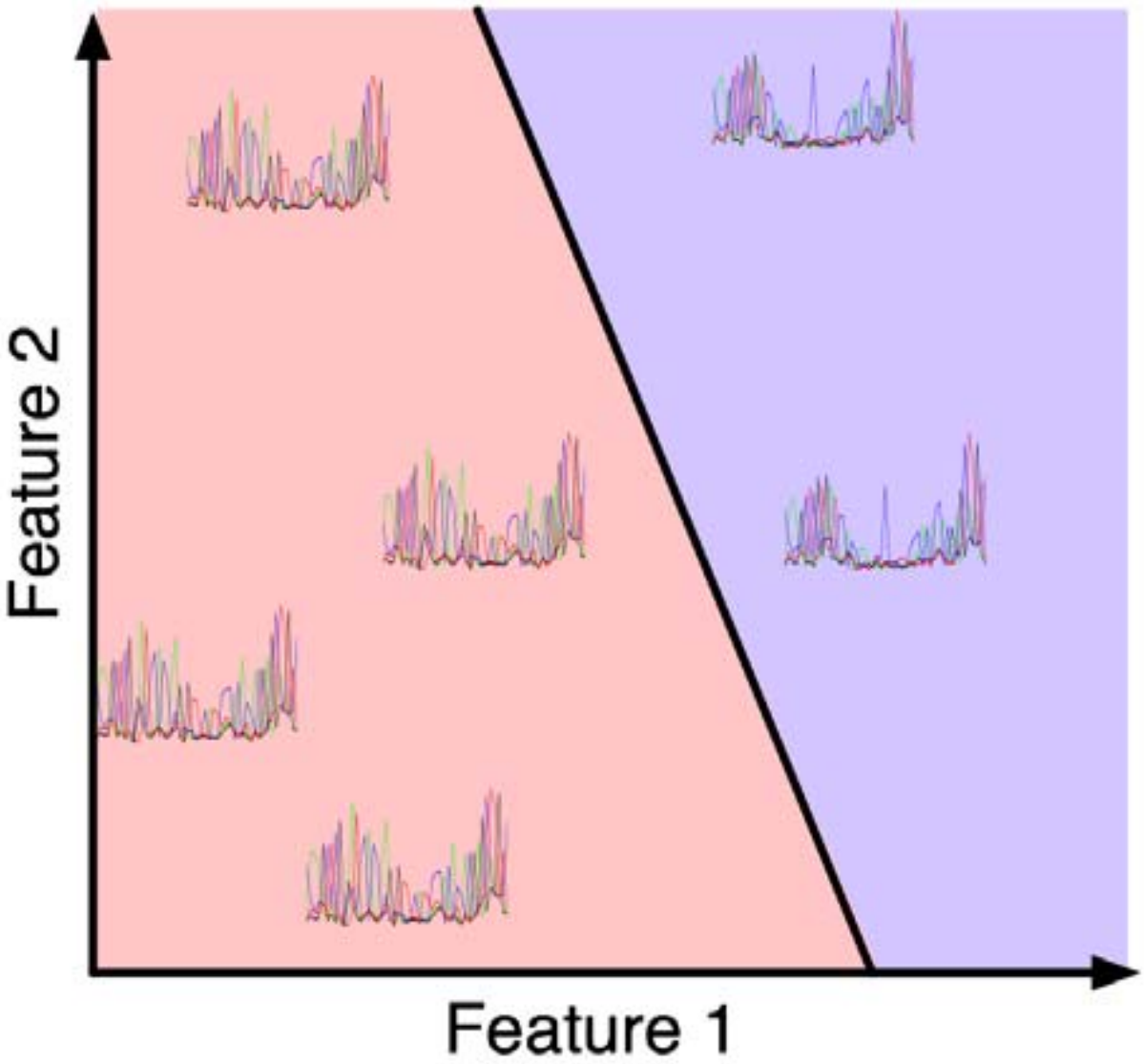
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
  - many features
  - small training sets
- Elegant theory
- "Support Vector Machines"

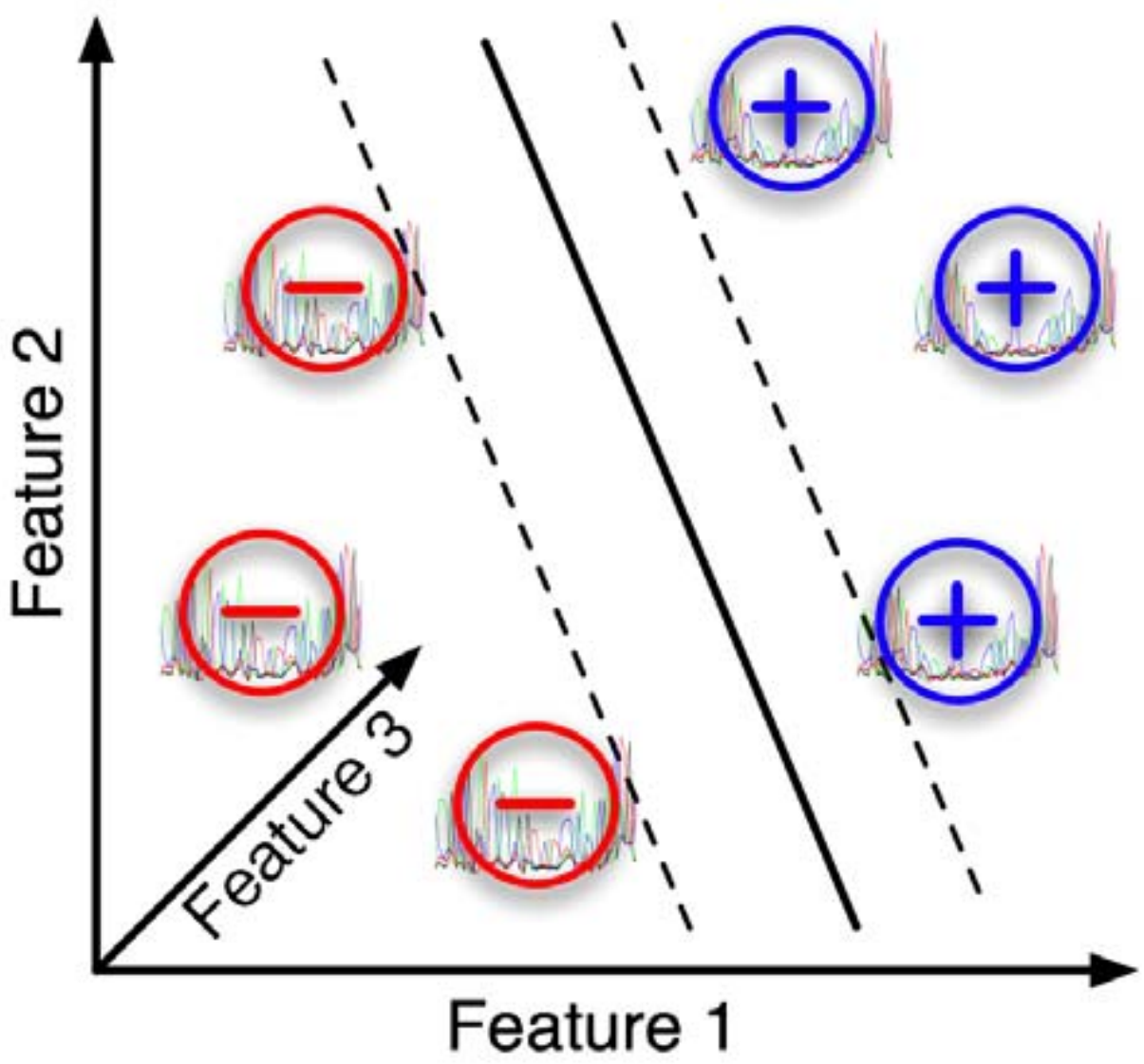
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
  - many features
  - nonlinearities
  - using kernels
- Elegant theory
- "Support Vector Machines"

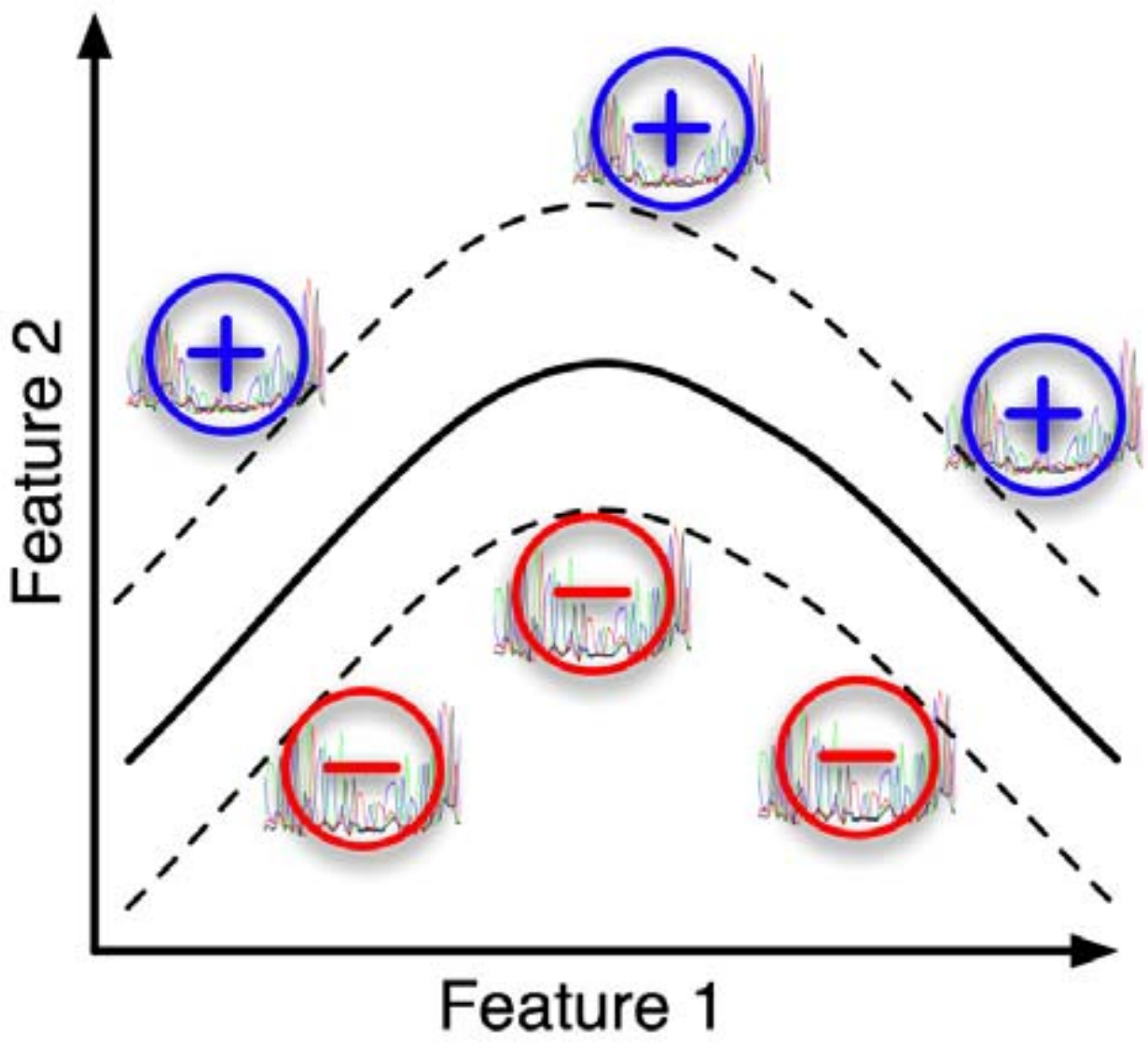
A Brief Excursion to Machine Learning I



Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
  - many features
  - nonlinearities using kernels
- Elegant theory
- "Support Vector Machines"

A Brief Excursion to Machine Learning I

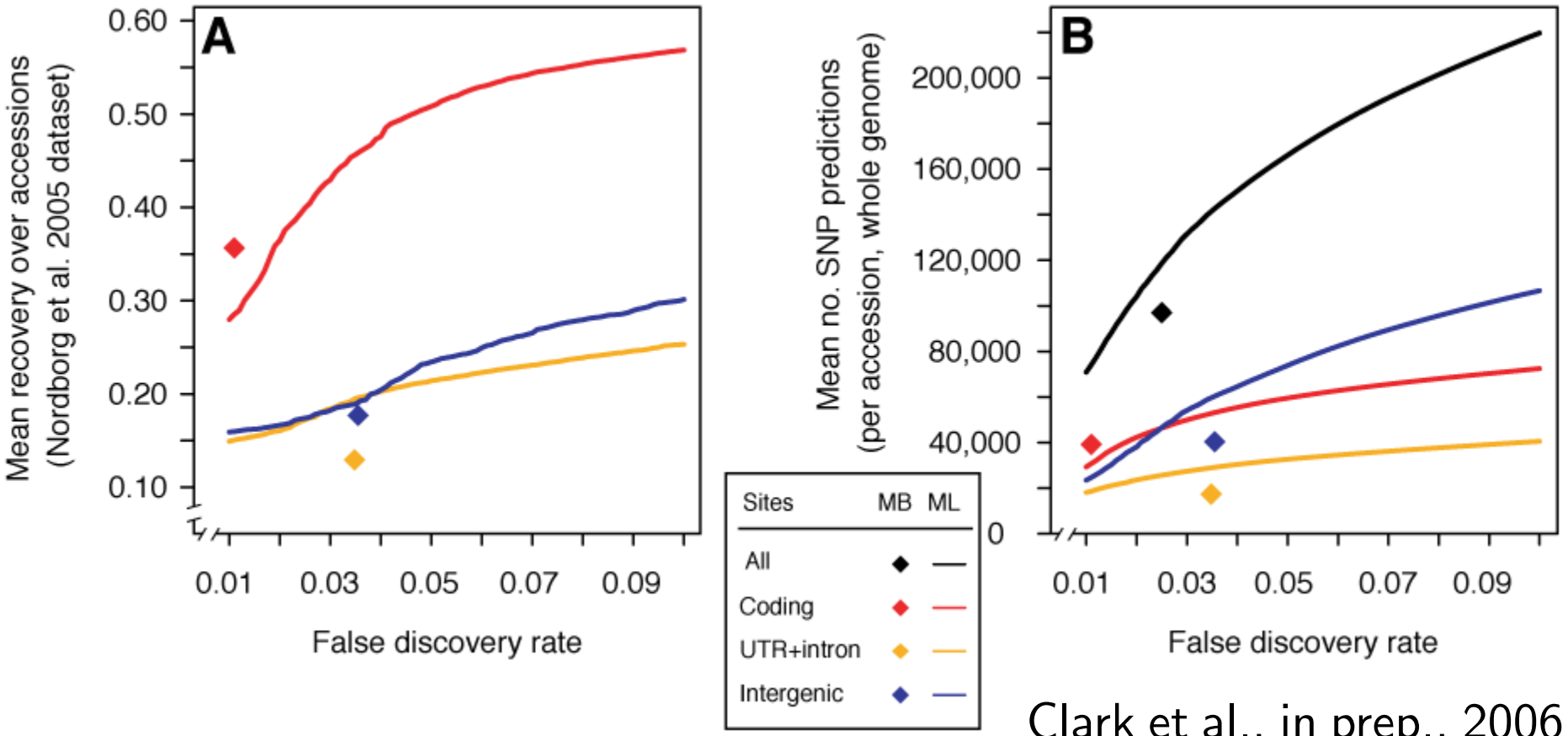


Large Margin Learning

- Extract features
- Find linear separation
- With large margin
- Classify new points
- Works with
  - many features
  - nonlinearities using kernels
- Elegant theory
- "Support Vector Machines"

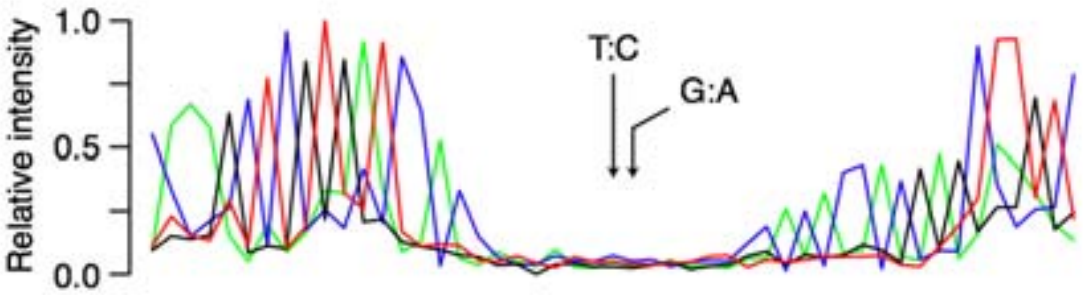
Application to SNP discovery

- Classification using Support Vector Machines with 302 features
- ≈2,400 known SNPs/ecotype (Nordborg et al., PLoS Biol., 2005)
- Out-of-sample evaluation and prediction on whole genome
- Comparison with Perlegens model based method (Hinds et al., Science, 2005)



Clark et al., in prep., 2006.

Identification of Highly Polymorphic Regions



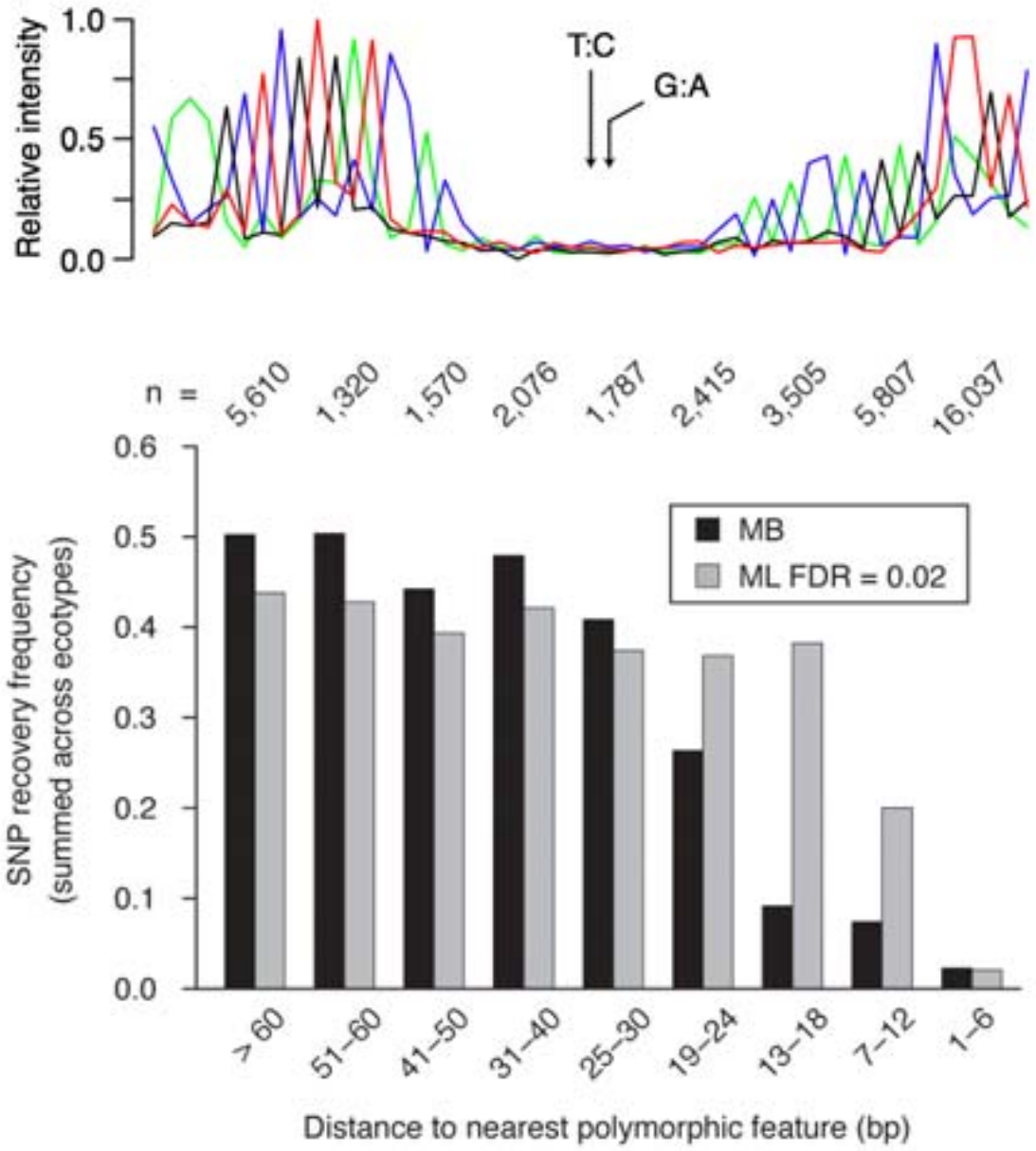
Results

- Performance drops, when other SNPs are in vicinity (1-20nt)
- Least predicted SNPs in highly polymorphic regions!
- ML more sensitive

New Approach

- **P**olymorphic **R**egion **P**rediction (PRP)
- Novel ML techniques for predicting complex properties

Identification of Highly Polymorphic Regions



Results

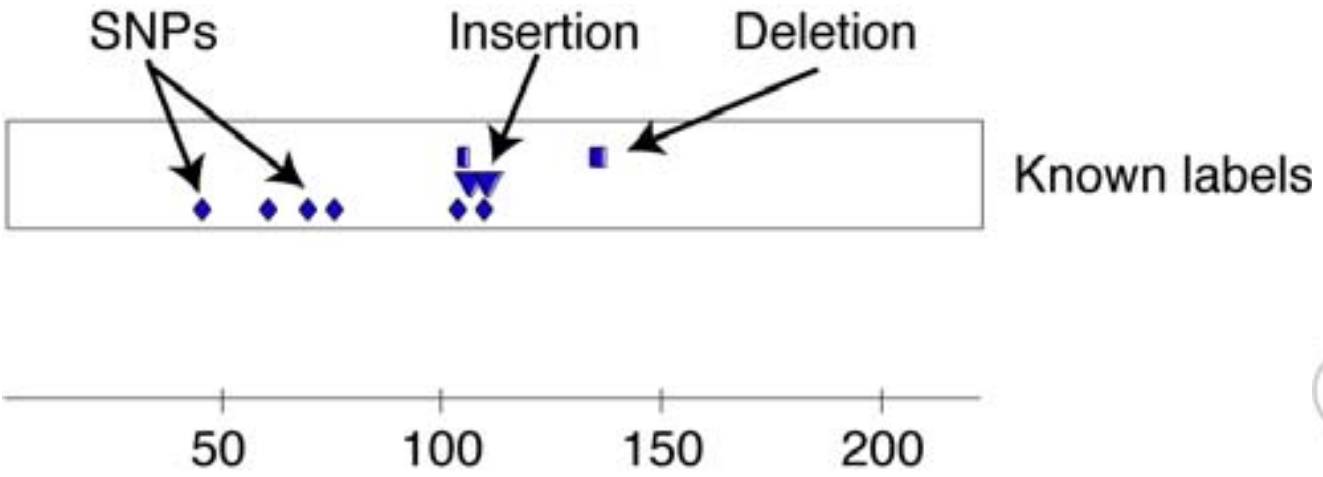
- Performance drops, when other SNPs are in vicinity (1-20nt)
- Least predicted SNPs in highly polymorphic regions!
- ML more sensitive

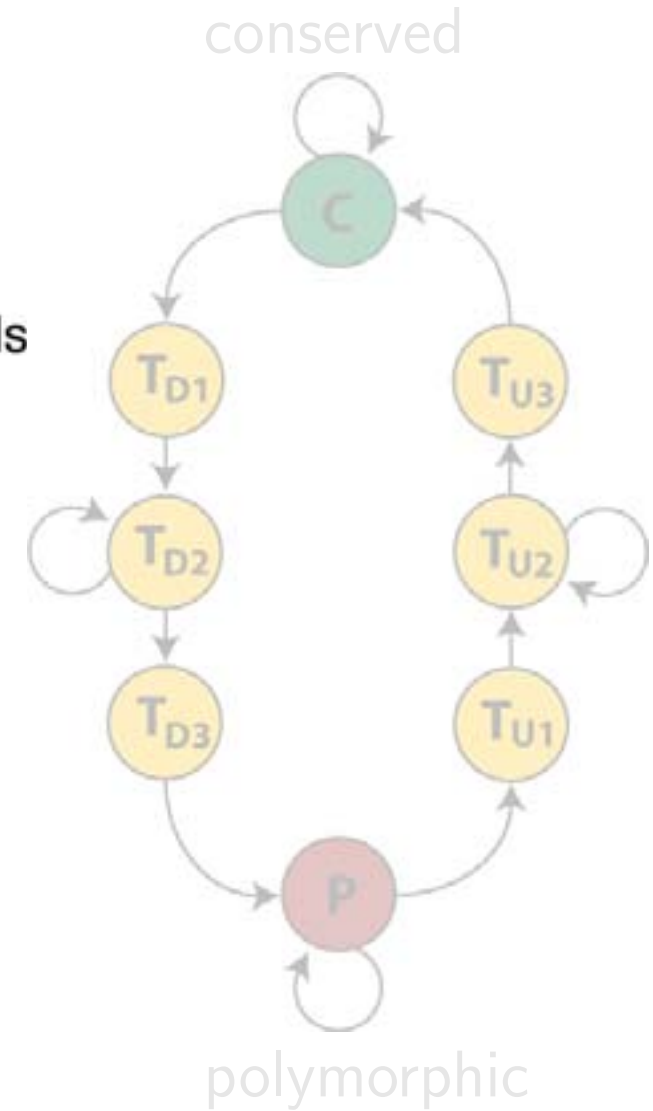
New Approach

- **P**olymorphic **R**egion **P**rediction (PRP)
- Novel ML techniques for predicting complex properties

Modeling polymorphic regions

Example sequence from Bor-4





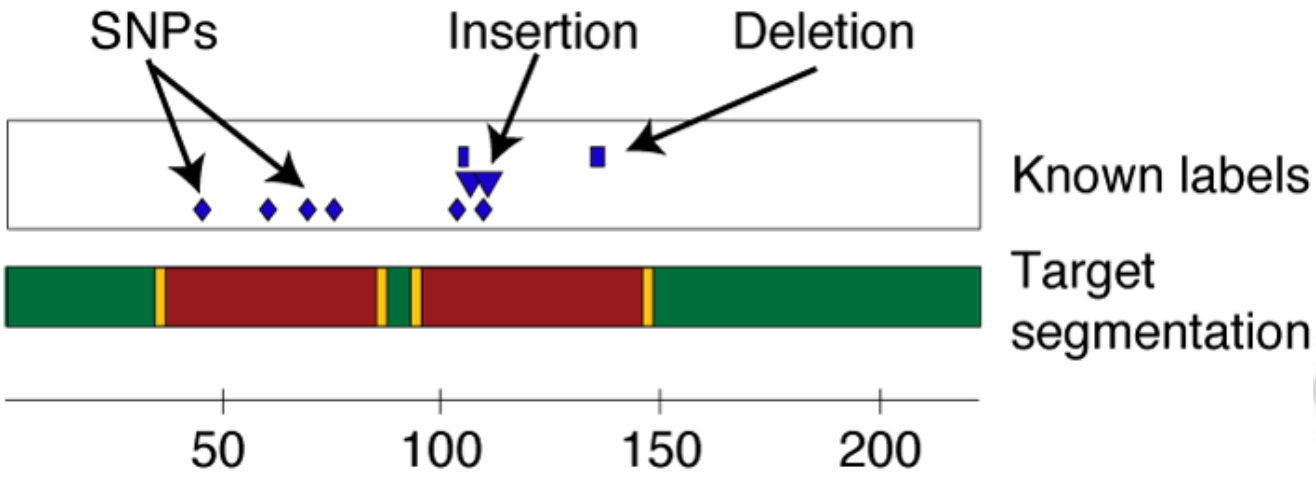
Create segmentation into

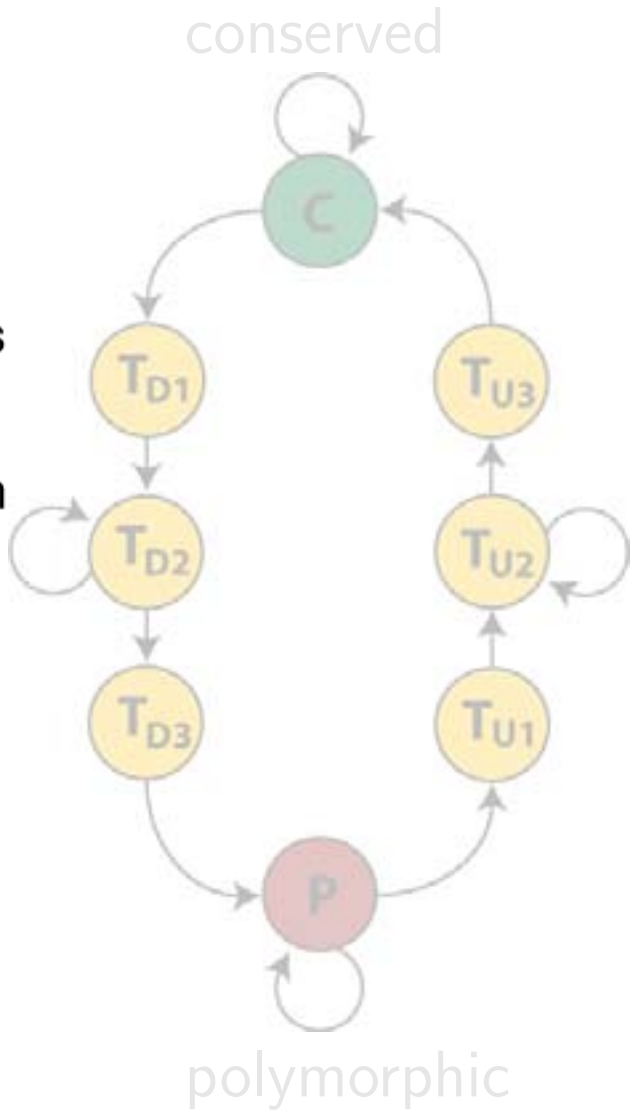
- conserved and
- polymorphic regions (distance <25nt)

Predict segmentation using state-model based approach

Modeling polymorphic regions

Example sequence from Bor-4





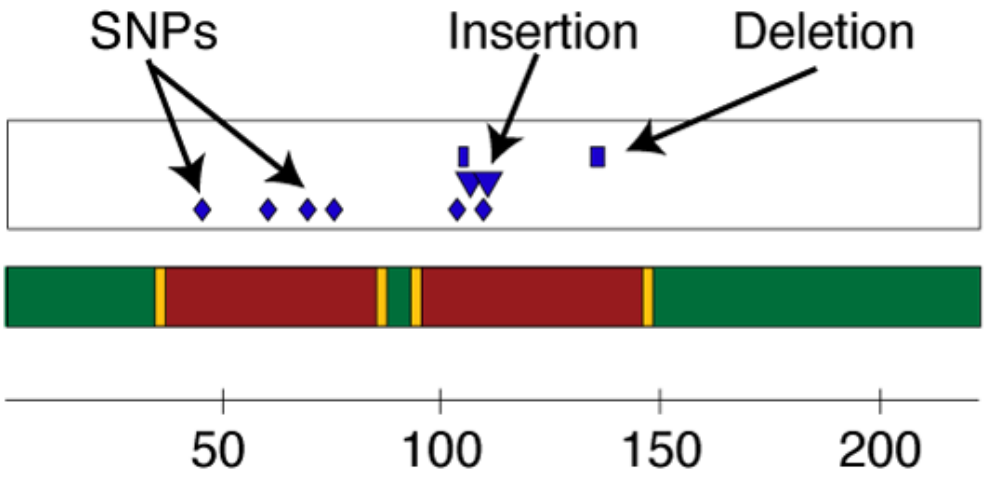
Create segmentation into

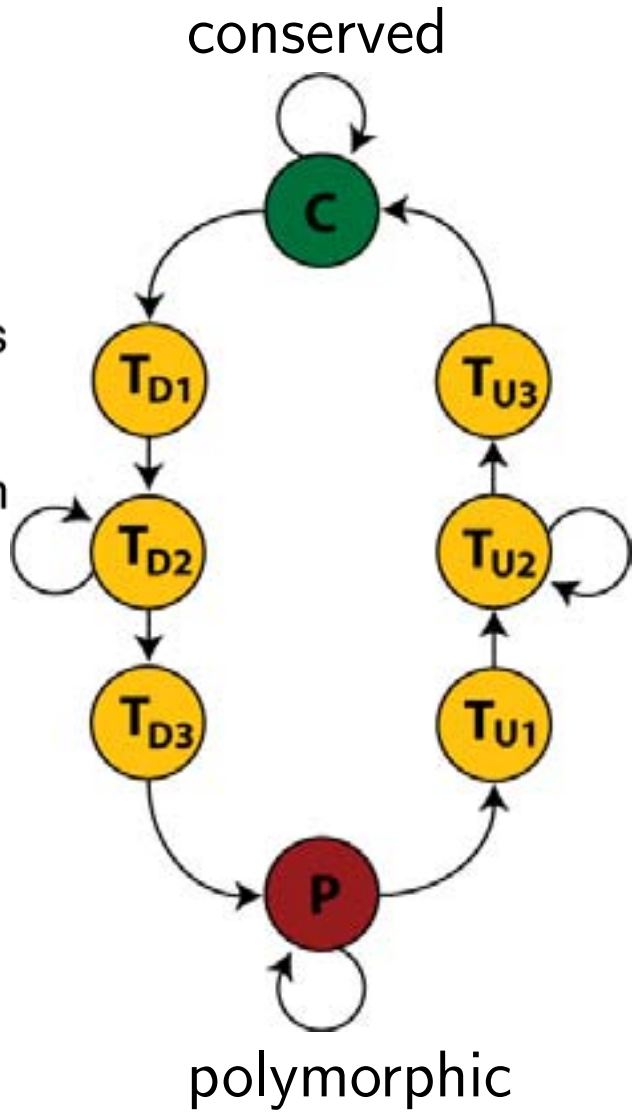
- conserved and
- polymorphic regions (distance <25nt)

Predict segmentation using state-model based approach

Modeling polymorphic regions

Example sequence from Bor-4



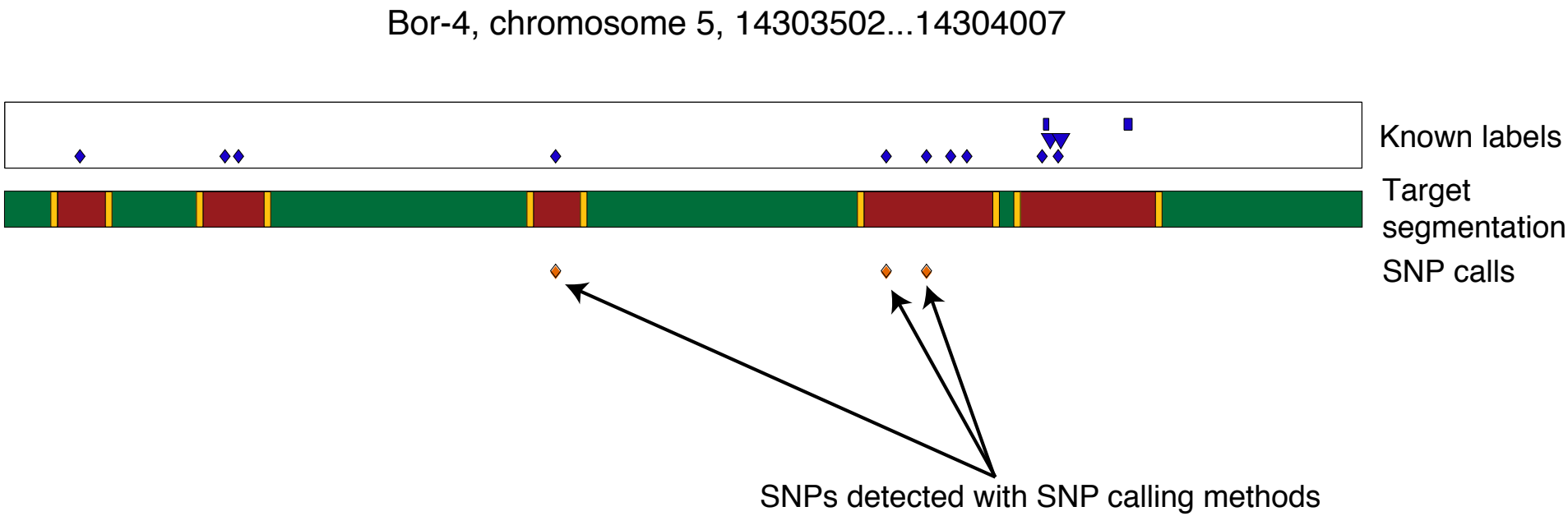


Create segmentation into

- conserved and
- polymorphic regions (distance <25nt)

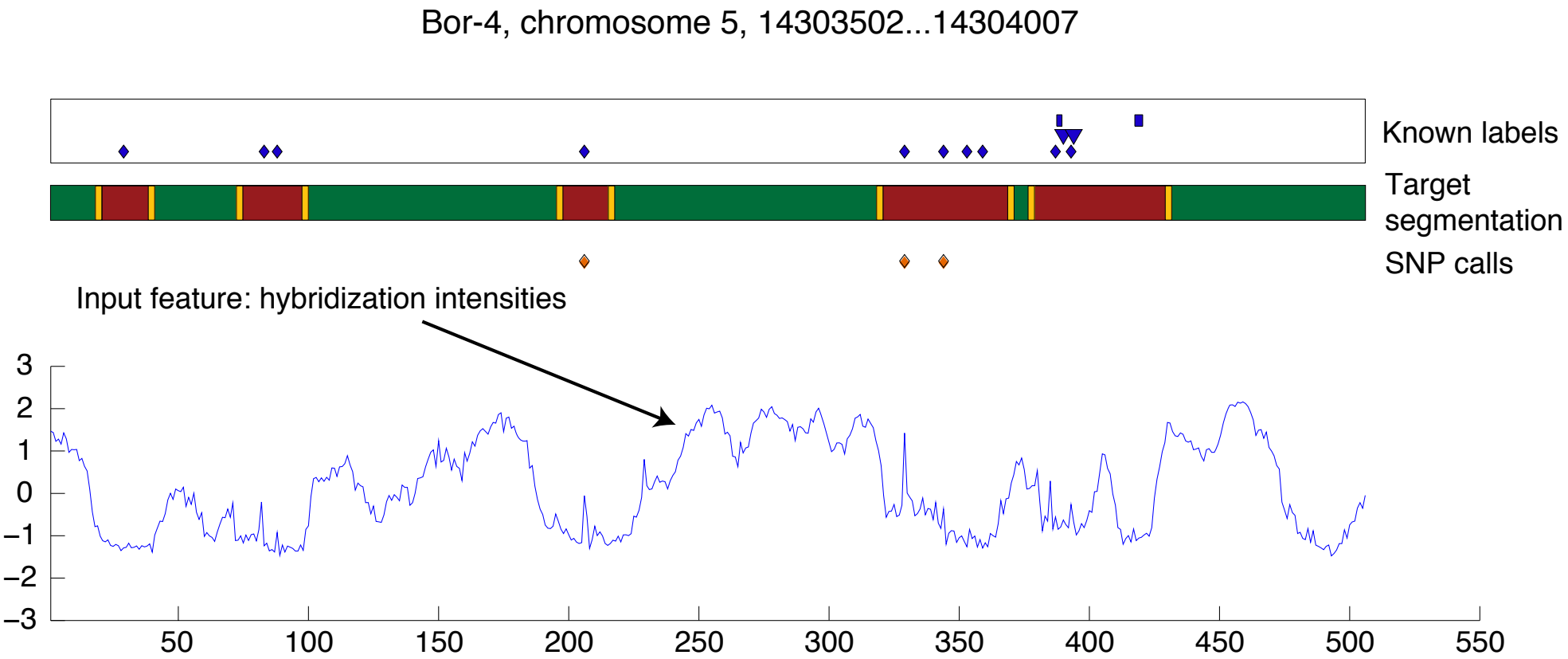
Predict segmentation using state-model based approach

Example



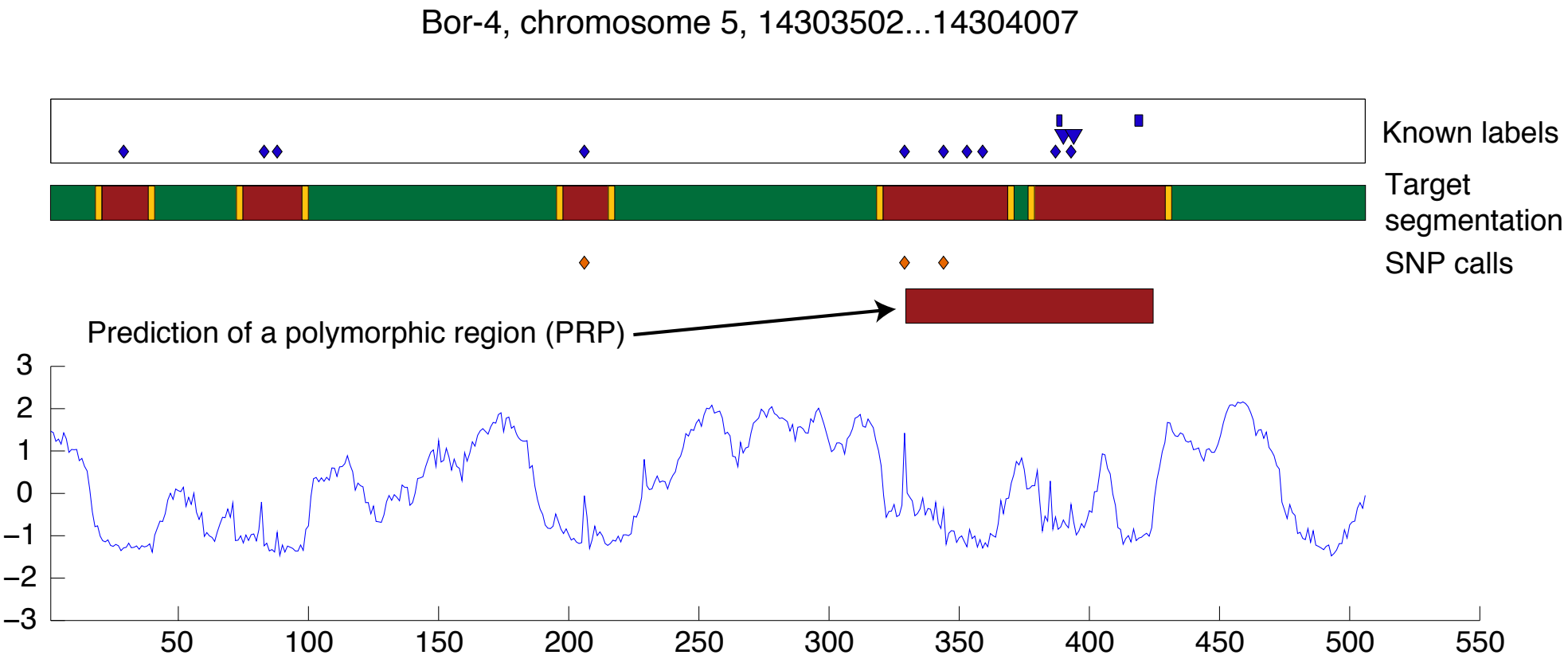
- Count prediction as correct if it 75% overlaps
- 55% Sensitivity, 90% Specificity (excluding isolated SNPs)
- Previously: Heuristic method with much lower sensitivity

Example



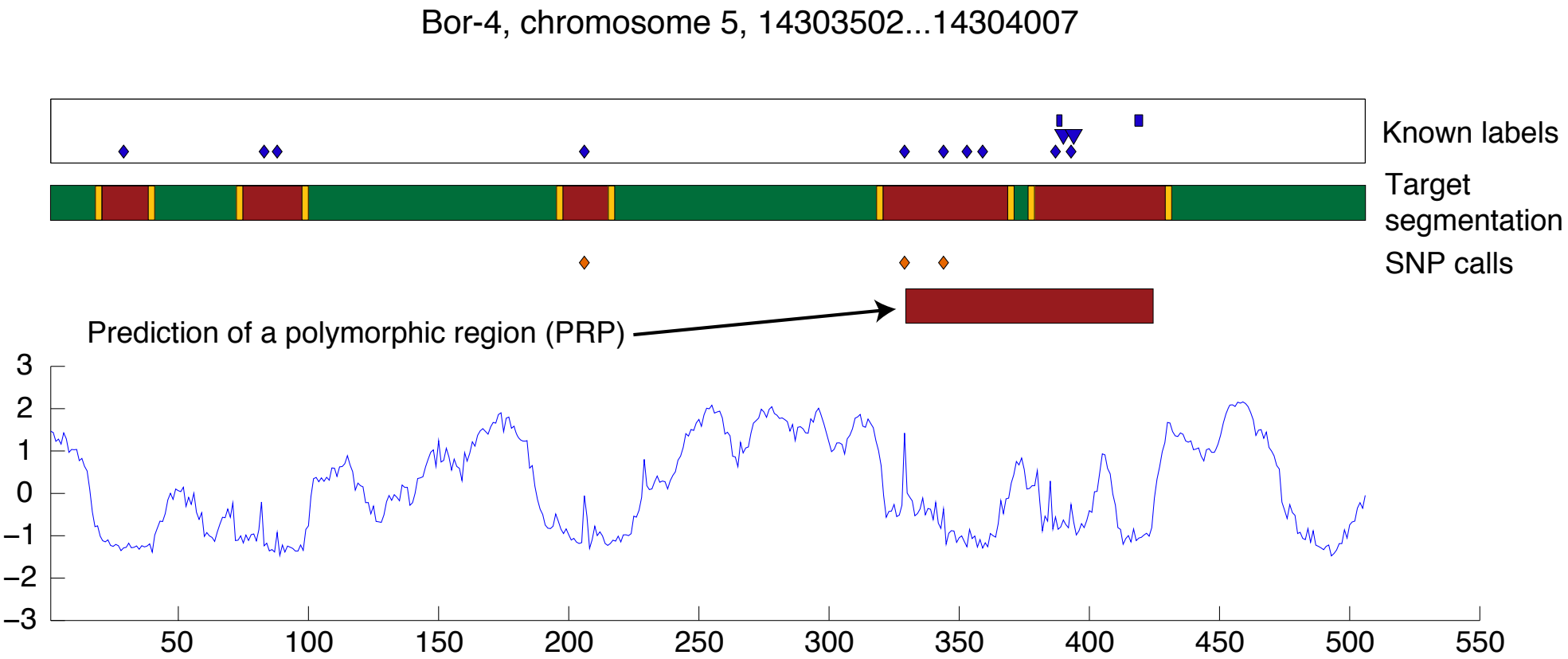
- Count prediction as correct if it 75% overlaps
- 55% Sensitivity, 90% Specificity (excluding isolated SNPs)
- Previously: Heuristic method with much lower sensitivity

Example



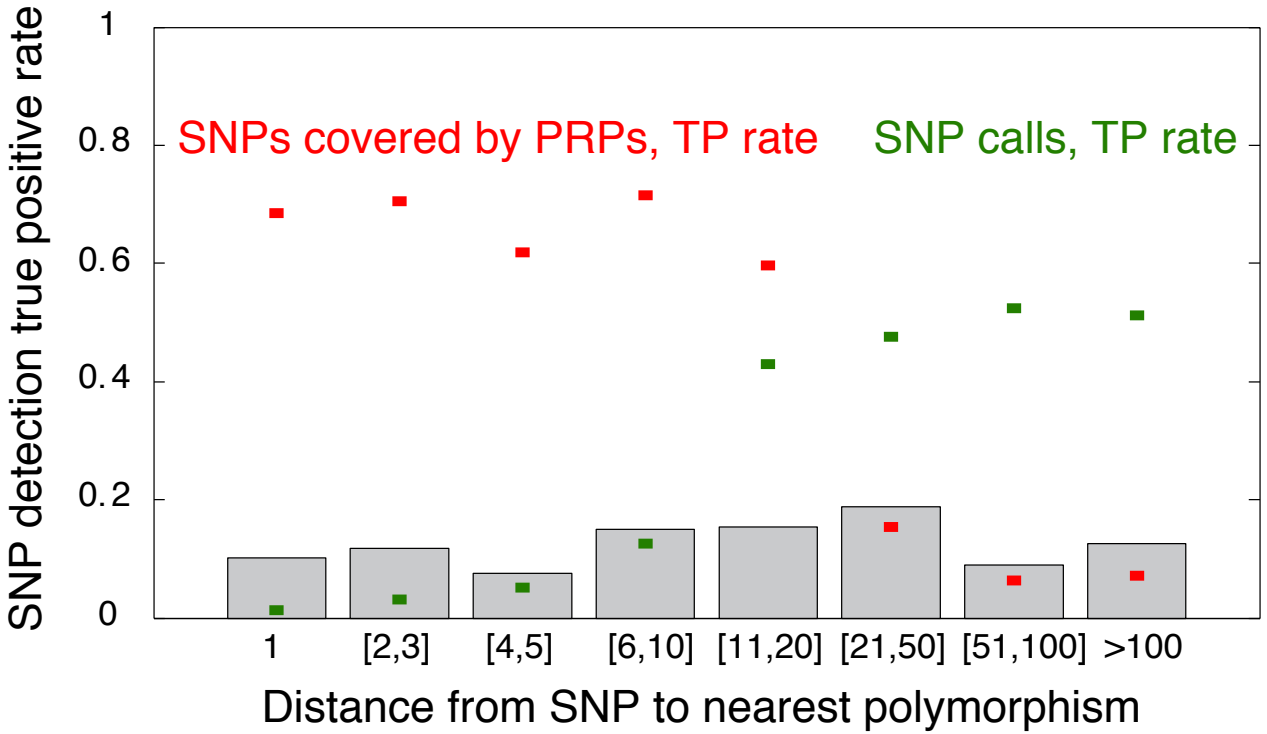
- Count prediction as correct if it 75% overlaps
- 55% Sensitivity, 90% Specificity (excluding isolated SNPs)
- Previously: Heuristic method with much lower sensitivity

Example



- Count prediction as correct if it 75% overlaps
- 55% Sensitivity, 90% Specificity (excluding isolated SNPs)
- Previously: Heuristic method with much lower sensitivity

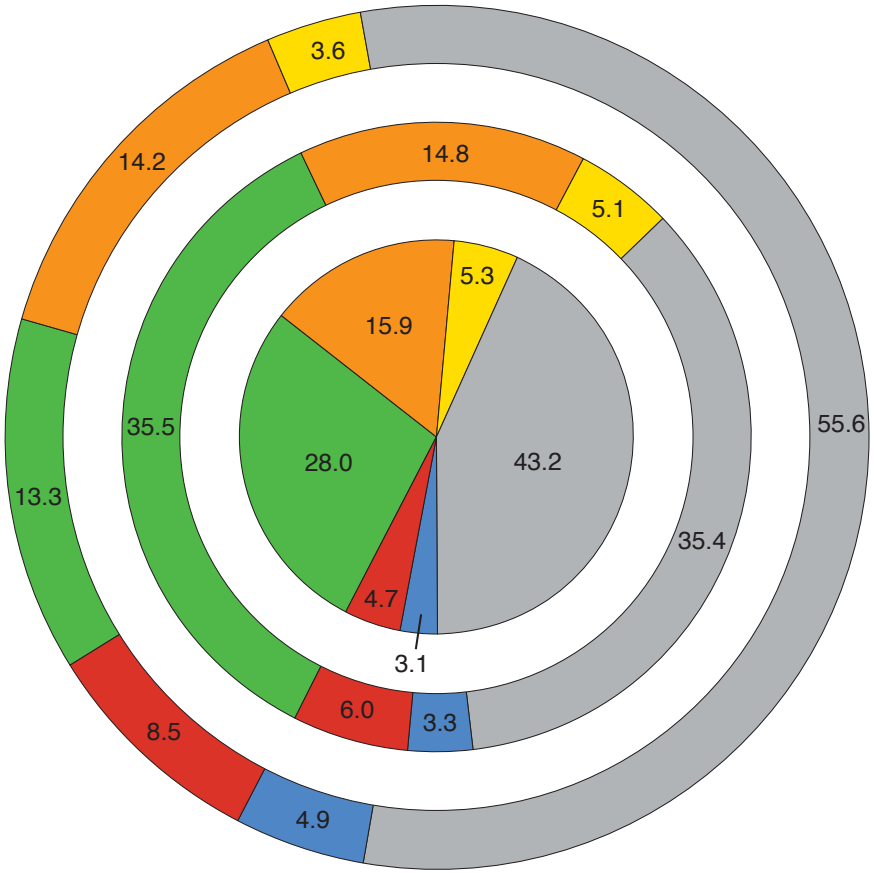
Complementing SNP Calls



Fraction of called/covered polymorphisms (test set):

	SNP calling (MB+ML)	Region predictor
SNPs	∼32%	∼41%
SNPs both methods	∼65%	
Deletions (per base)		∼71%
Insertions		∼39%

Polymorphism Distribution



Sequence type	SNP no.	bp in PSPs
Coding	230,186	3,828,824
Intron	96,069	4,090,306
Untranslated RNA	32,771	1,029,948
Intergenic	229,335	16,046,419
Pseudogene	21,552	1,413,369
Transposon	38,657	2,462,213

Inner circle: genomic bases
Inner ring: SNP predictions
Outer ring: bases in PRPs

Coding regions are underrepresented for PRPs while transposons are overrepresented.

Clark et al., Zeller et al., in preparation, 2006

Predicted Effects on Gene Products

SNPs may interfere with signal recognition

- Transcription start & stop
- Splice sites
- Translation start & stop
- Regulatory elements

Consequences

- Modification of consensus sequence (e.g. splice site dinucl.)
  - Permanent change

  $\Rightarrow$ Easy to check
- Weaker or stronger recognition of signals
  - Different expression levels
  - (Alternative) splicing pattern may change

  $\Rightarrow$ Needs ab initio gene finding $\Rightarrow$ build on our previous work

Predicted Effects on Gene Products

SNPs may interfere with signal recognition

- Transcription start & stop
- Splice sites
- Translation start & stop
- Regulatory elements

Consequences

- Modification of consensus sequence (e.g. splice site dinucl.)
  - Permanent change

  $\Rightarrow$ Easy to check
- Weaker or stronger recognition of signals
  - Different expression levels
  - (Alternative) splicing pattern may change

  $\Rightarrow$ Needs ab initio gene finding $\Rightarrow$ build on our previous work

Effects on Genes

SNPs (MB+ML) leading to consensus changes

- 109,976 amino acid changes
- 1,227 premature stops
- 156 initiation methionine changes
- 435 splice site changes
- Major changes in 573 genes validated by dideoxy sequencing*

Polymorphic Regions Predictions

- Overlap coding regions of 16,692 genes in at least one ecotype
- 50% overlap in 1,910 genes, 95% overlap in 743 genes
- 122 coding sequence deletions validated by dideoxy sequencing*

* Verified in collaboration with laboratory of Joseph Ecker

Effects on Genes

SNPs (MB+ML) leading to consensus changes

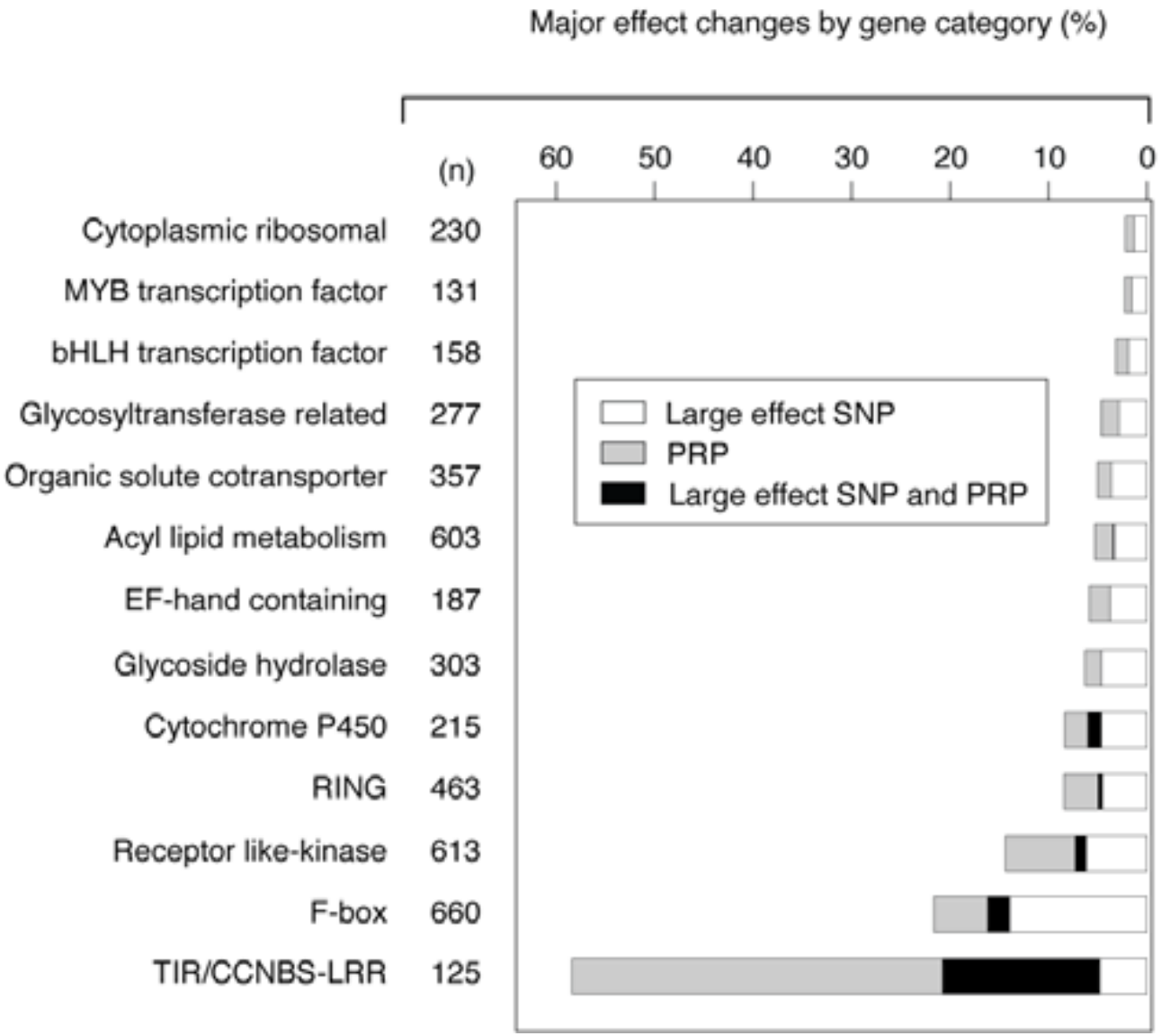
- 109,976 amino acid changes
- 1,227 premature stops
- 156 initiation methionine changes
- 435 splice site changes
- Major changes in 573 genes validated by dideoxy sequencing*

Polymorphic Regions Predictions

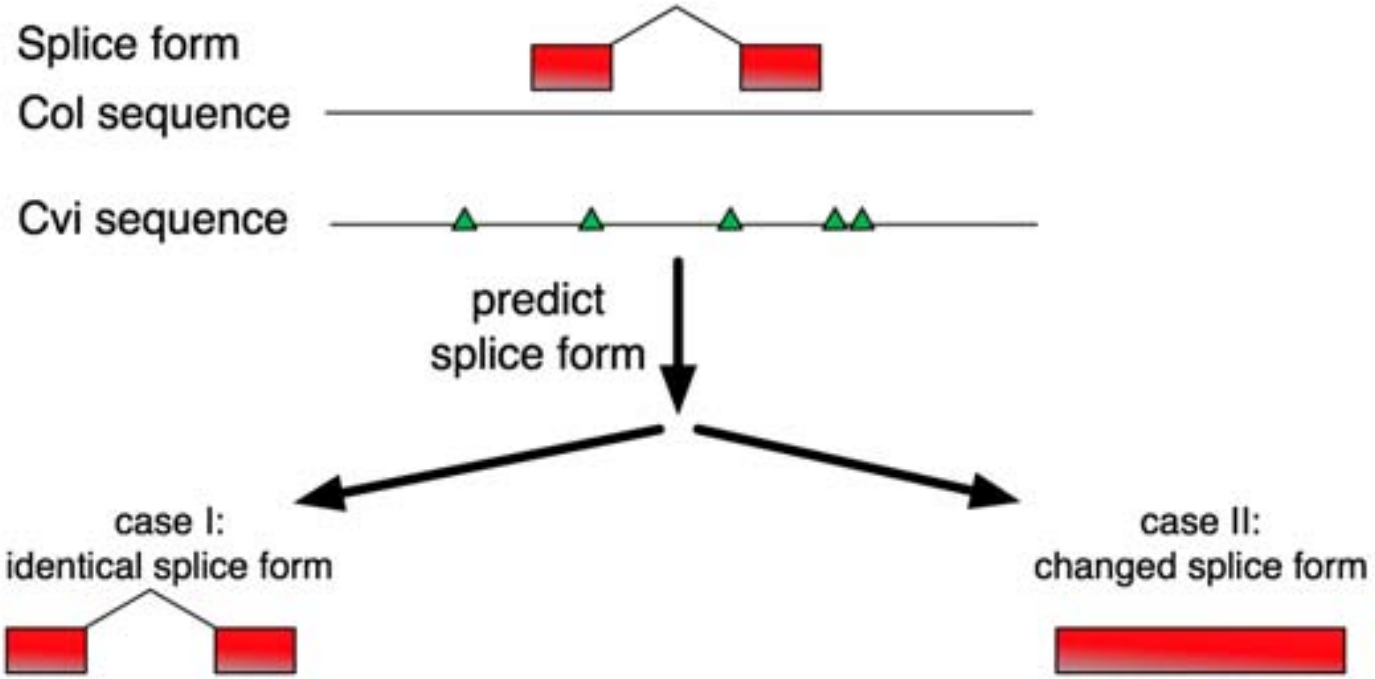
- Overlap coding regions of 16,692 genes in at least one ecotype
- 50% overlap in 1,910 genes, 95% overlap in 743 genes
- 122 coding sequence deletions validated by dideoxy sequencing*

* Verified in collaboration with laboratory of Joseph Ecker

Major Effects Distribution



Predicted Effects by Gene Finding (Preliminary)



16664 genes

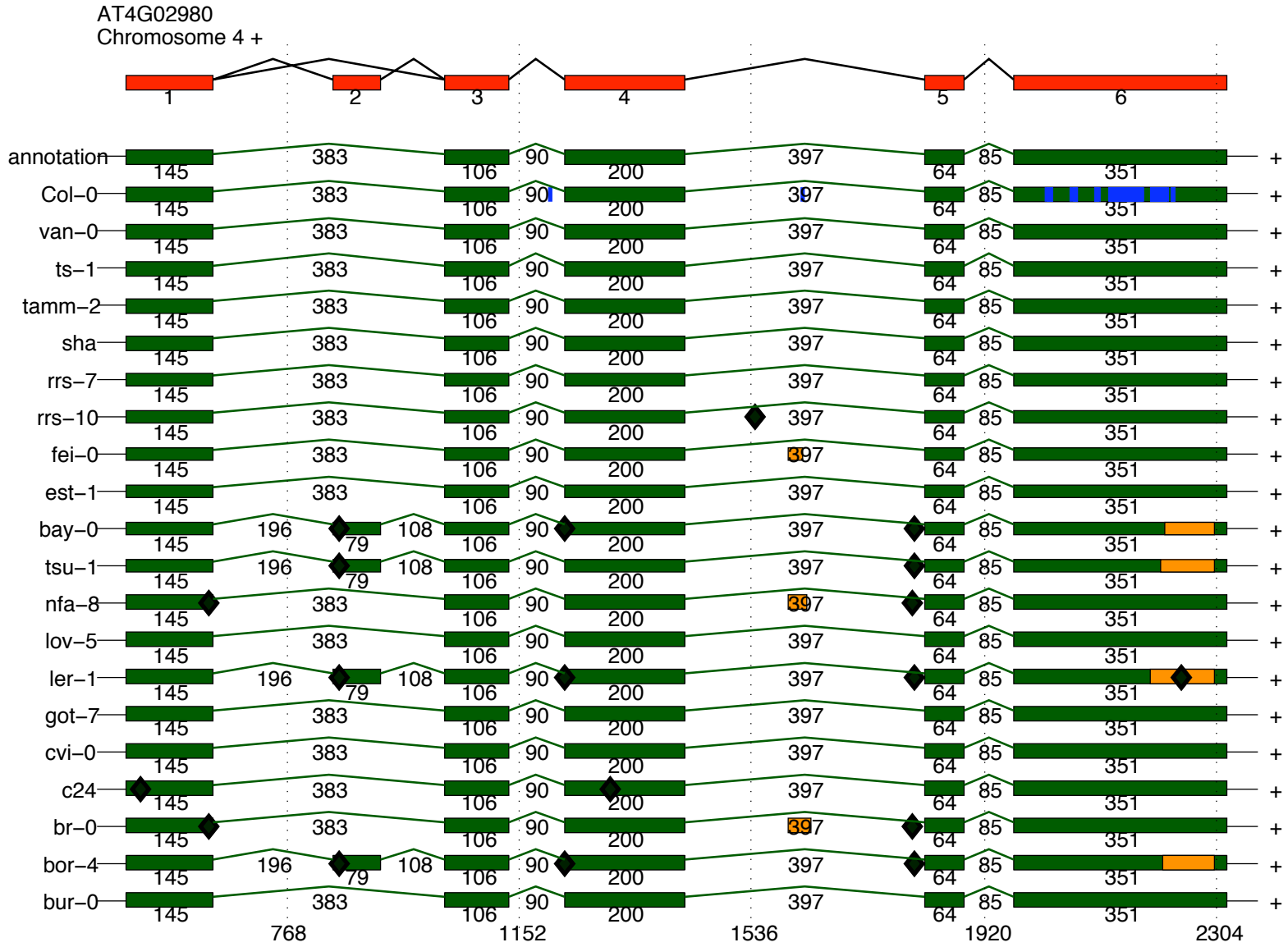
~92 %

1366 genes

~8 %

intron retentions:	**771**
single exon skips:	**380**
alt. 5 prime sites:	**325**
alt. 3 prime sites:	**477**
multiple exon skips:	**23**

Example of Predicted Splice Form Change



Conclusions

- Created inventory for polymorphisms in *Arabidopsis thaliana*
  - Highly polymorphic: ≈0.5% in SNPs, ≈25% in PRPs
- New method for SNP calling; more accurate than Perlegen's
- Accurate polymorphic region predictions
  - Important for further analysis (e.g. dideoxy sequencing)
- Large number of major effect changes
  - Overrepresented in R genes, F-box genes, Receptor-like kinases
- More predicted changes by *ab initio* gene finding
- Application to other genomes (rice, human?)
- Study variations using mRNA tiling arrays
  - Expression levels, splicing

Conclusions

- Created inventory for polymorphisms in *Arabidopsis thaliana*
  - Highly polymorphic: ≈0.5% in SNPs, ≈25% in PRPs
- New method for SNP calling; more accurate than Perlegen's
- Accurate polymorphic region predictions
  - Important for further analysis (e.g. dideoxy sequencing)
- Large number of major effect changes
  - Overrepresented in R genes, F-box genes, Receptor-like kinases
- More predicted changes by *ab initio* gene finding
- Application to other genomes (rice, human?)
- Study variations using mRNA tiling arrays
  - Expression levels, splicing, ...

Conclusions

- Created inventory for polymorphisms in *Arabidopsis thaliana*
  - Highly polymorphic: ≈0.5% in SNPs, ≈25% in PRPs
- New method for SNP calling; more accurate than Perlegen's
- Accurate polymorphic region predictions
  - Important for further analysis (e.g. dideoxy sequencing)
- Large number of major effect changes
  - Overrepresented in R genes, F-box genes, Receptor-like kinases
- More predicted changes by *ab initio* gene finding
- Application to other genomes (rice, human?)
- Study variations using mRNA tiling arrays
  - Expression levels, splicing, ...

Conclusions

- Created inventory for polymorphisms in *Arabidopsis thaliana*
  - Highly polymorphic: ≈0.5% in SNPs, ≈25% in PRPs
- New method for SNP calling; more accurate than Perlegen's
- Accurate polymorphic region predictions
  - Important for further analysis (e.g. dideoxy sequencing)
- Large number of major effect changes
  - Overrepresented in R genes, F-box genes, Receptor-like kinases
- More predicted changes by *ab initio* gene finding
- Application to other genomes (rice, human?)
- Study variations using mRNA tiling arrays
  - Expression levels, splicing, ...

Acknowledgments

Friedrich Miescher Laboratory

- Gabriele Schweikert
- Georg Zeller

The Salk Institute, CA

- Paul Shinn
- Joseph Ecker

MPI for Biological Cybernetics

- Bernhard Schölkopf

MPI for Developmental Biology

- Richard Clark
- Stephan Ossowski
- Norman Warthmann
- Detlef Weigel

ZBIT, University of Tübingen

- Daniel Huson

Perlegen Sciences, CA

- Kelly Frazer

Summary Of Lectures

- Statistical Learning Theory
- Support Vector Machines
- Convex Optimization
- Introduction to (String) Kernels
- Fast SVMs using String Kernels
- Marginalized kernels and Graph kernels
- Multiple Kernel Learning
- Structured Output Learning
- Semi-Supervised Learning
- Probabilistic Approaches
- Principal Component Analysis
- Applications

Probabilistic Learning Model

Assumption

All data is generated by the same hidden probabilistic source!

Formally

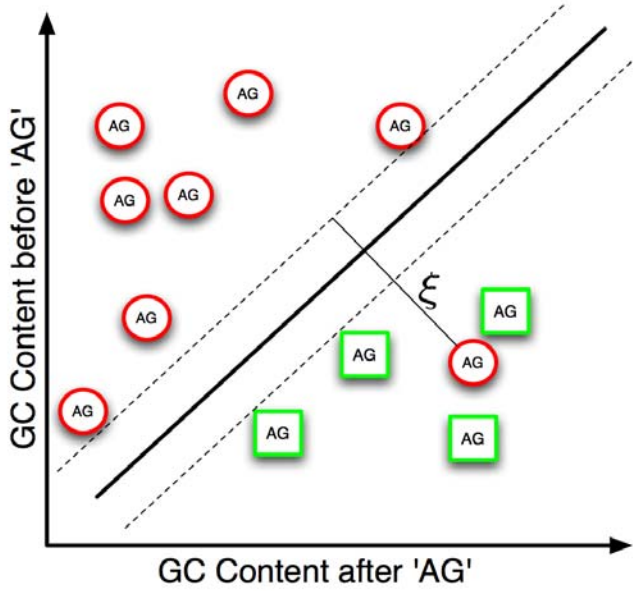
- ρ is an unknown joint probability distribution over $\mathcal{X} \times \mathcal{Y}$
- Training data $((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n))$ is iid $\sim \rho$
- Aim: find best $f^* \in \mathcal{F}$ that minimizes risk

$$\mathbf{R}(f) = \int \ell(f(\mathbf{x}), y) d\rho.$$

- ERM: find best $f_n \in \mathcal{F}$ that minimizes empirical risk

$$\mathbf{R}_{emp}(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(f(\mathbf{x}_i), y_i).$$

SVMs: Geometric View



Minimize

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i$$

Subject to

$$y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) \geqslant 1 - \xi_i$$
$$\xi_i \geqslant 0$$
for all $i = 1, \ldots, N$.

- The examples on the margin are called support vectors [Vapnik, 1995]
- Called the soft margin SVM or the C-SVM [Cortes and Vapnik, 1995]

Convex Optimization

Constrained Optimization (generally hard)

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leqslant 0 \text{ for all } i \\ & g_j(x) = 0 \text{ for all } j \end{aligned}$$

Convex Optimization (generally easy)

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leqslant 0 \text{ for all } i \\ & a_j^\top x = b_j \text{ for all } j \end{aligned}$$

$f_0, f_1, \ldots, f_m$ are convex, and the equality constraints are affine Boyd and Vandenberghe [2004].

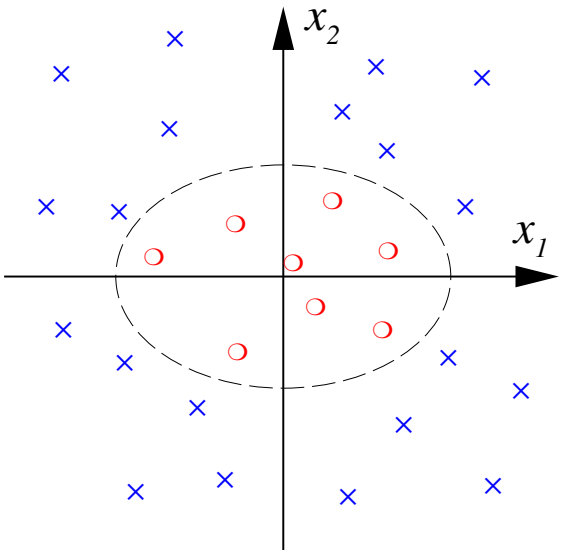
Nonlinear Algorithms in Feature Space

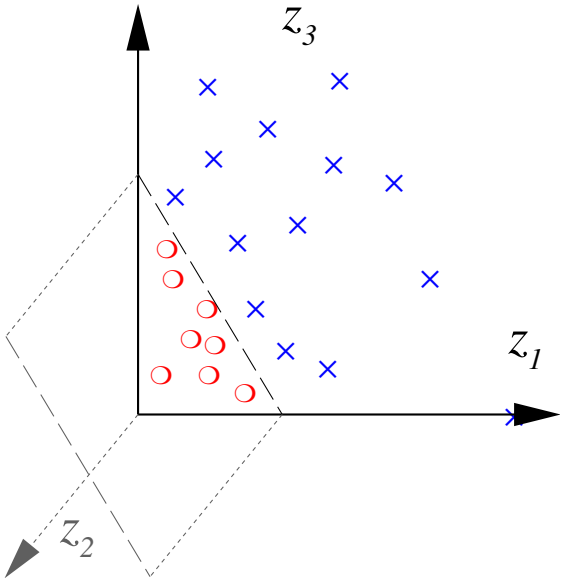
Linear separation might be not sufficient!

$\Rightarrow$ Map into a higher dimensional feature space

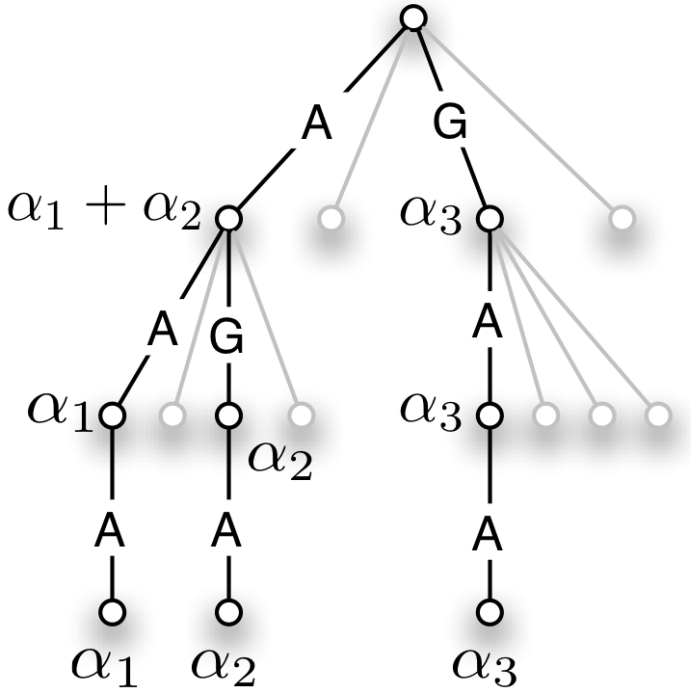
Example: all second order monomials

$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$





Example: Trees & Tries



Tree (trie) data structure stores sparse weightings on sequences (and their subsequences).

Illustration: Three sequences AAA, AGA, GAA were added to a trie (α's are the weights of the sequences).

- Building tree: $\mathcal{O}(Q \cdot L \cdot D)$
- Compute all $f(\mathbf{x}_i)$: $\mathcal{O}(N \cdot L \cdot D)$
- Memory: $\mathcal{O}(Q \cdot L \cdot D \cdot |\Sigma|)$
- Works for any D

Definition of Diffusion Kernel

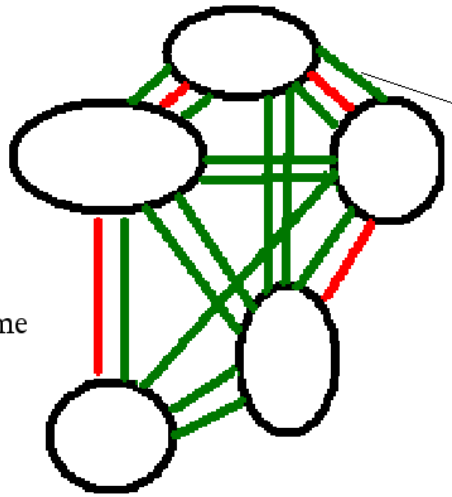
- **A: Adjacency matrix,**
- **D: Diagonal matrix of Degrees**
- **L = D-A: Graph Laplacian Matrix**
- ***Diffusion kernel matrix***

$$K = \exp(-\beta L)$$

  - β : **Diffusion paramater**

- **Matrix exponential, not elementwise exponential**

Protein graph model



Node attributes
- hydrophobicity
- polarity
- polarizability
- van der Waals volume
- length
- helix, sheet, loop

Edge attributes
- type (sequence, structure)
- length

Karsten Borgwardt et al. - Classifying proteins into functional classes via graph kernels 11

Summary

MKL Algorithm

- Automatically computes best convex combination of kernels

$$k(\mathbf{x}, \mathbf{x}') = \sum_{p=1}^{M} \beta_p k_p(\mathbf{x}, \mathbf{x}'), \quad \sum_{p=1}^{M} \beta_p = 1, \ \beta_p \geq 0$$

- SILP formulation makes large scale training and evaluation possible.

Possible Applications

- Heterogeneous data.
- Improving interpretability.

Max-Margin Structured Output Learning

- Learn function $f(\mathbf{y}|\mathbf{x})$ scoring segmentations $\mathbf{y}$ for $\mathbf{x}$
- Maximize $f(\mathbf{y}|\mathbf{x})$ w.r.t. $\mathbf{y}$ for prediction:

$$\operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}^*} f(\mathbf{y}|\mathbf{x})$$

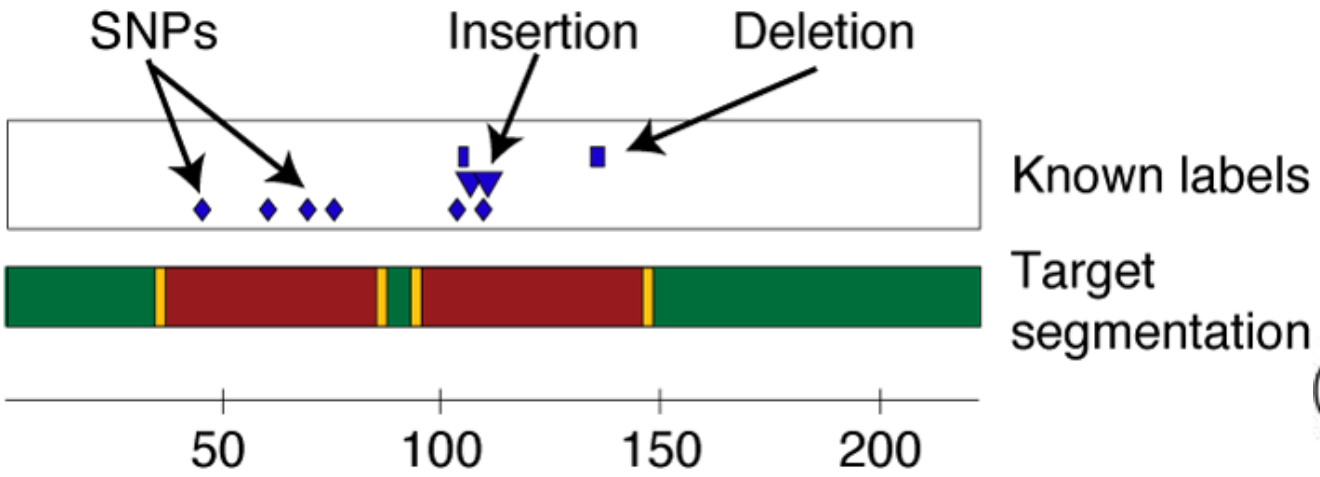
- Given N sequence pairs $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)$ for training
- Determine f such that there is a large margin between true and wrong segmentations

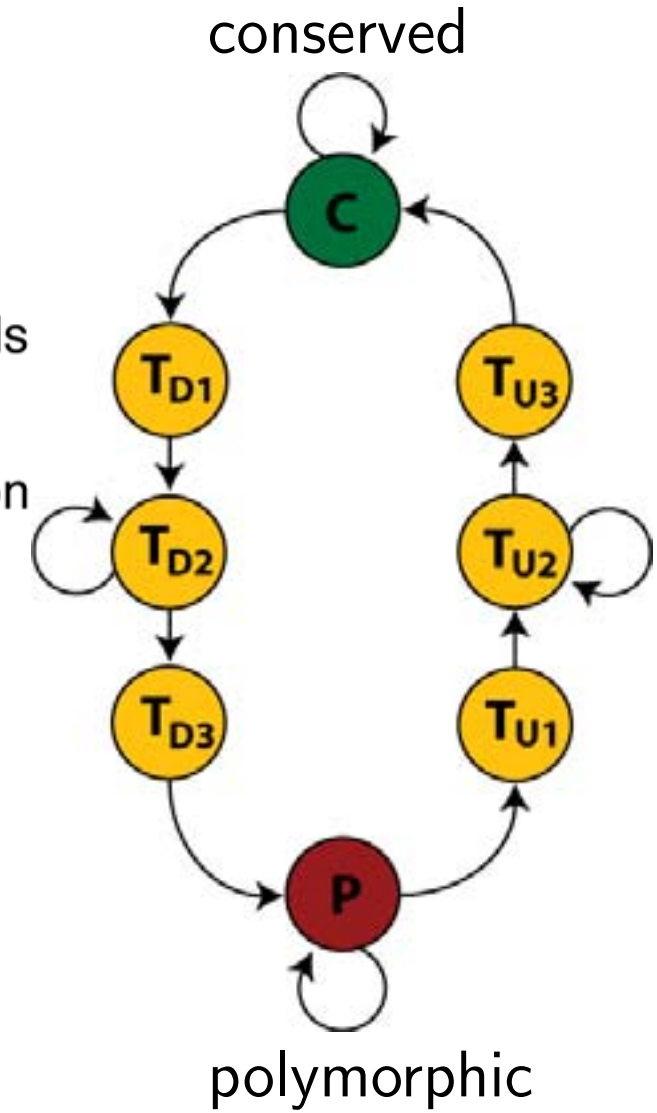
$$\begin{aligned} \min_f \quad & C\sum_{n=1}^{N}\xi_n + \mathbf{P}[f] \\ \text{w.r.t.} \quad & f(\mathbf{y}_n|\mathbf{x}_n) - f(\mathbf{y}|\mathbf{x}_n) \geq 1 - \xi_n \\ & \text{for all } \mathbf{y}_n \neq \mathbf{y} \in \mathcal{Y}^*, n = 1, \ldots, N \end{aligned}$$

- Exponentially many constraints!

Modeling polymorphic regions

Example sequence from Bor-4



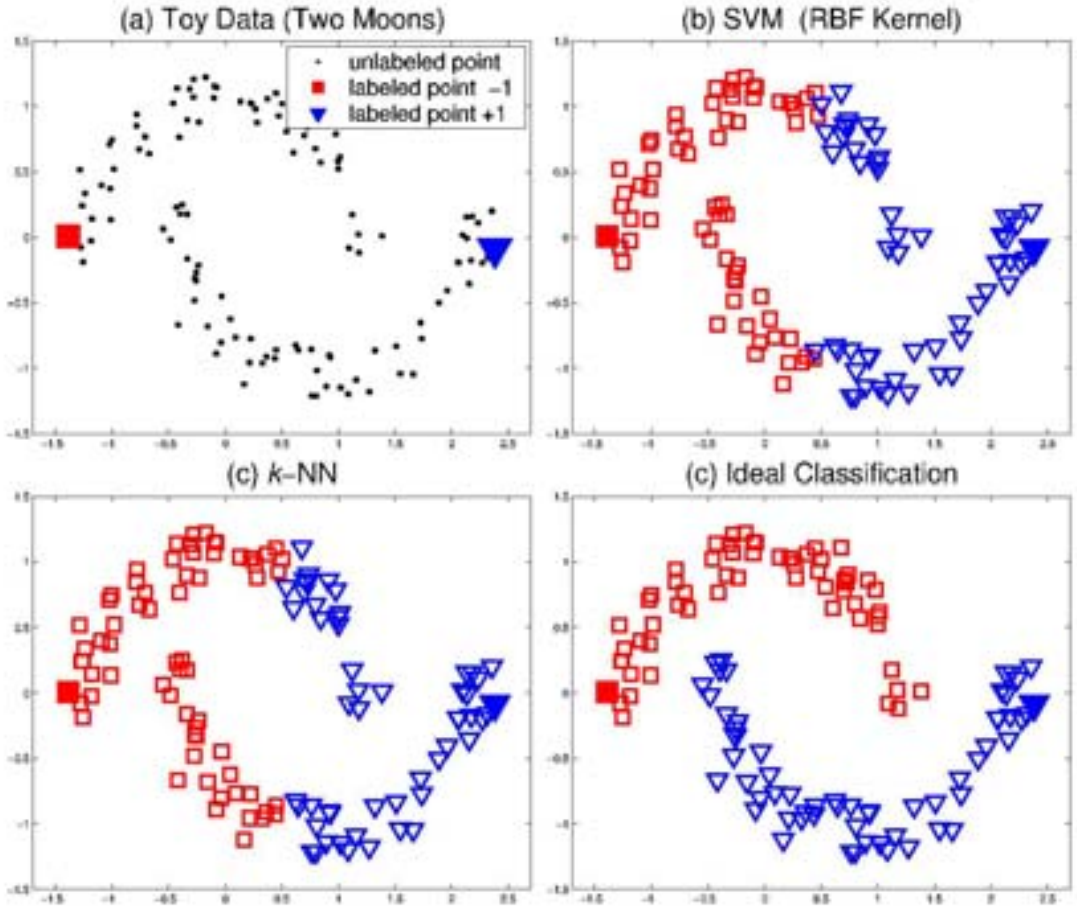


Create segmentation into

- conserved and
- polymorphic regions (distance <25nt)

Predict segmentation using state-model based approach

Called **Label Propagation**, as the same solution is achieved by iteratively propagating labels along edges until convergence

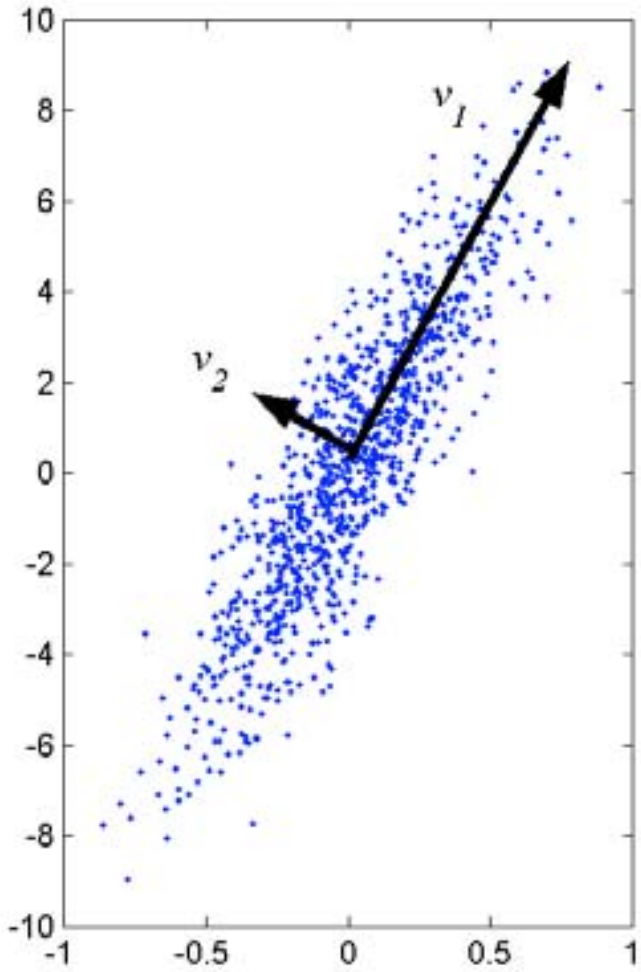


Note: here color $\stackrel{\triangle}{=}$ classes

[images from "Learning with Local and Global Consistency", Zhou, Bousquet, Lal, Weston, Schölkopf; NIPS 2004]

Summary PCA

Keep in mind the following picture:

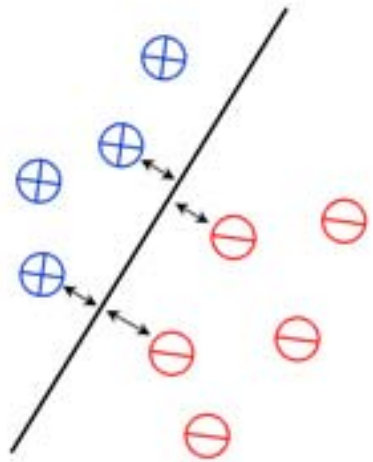


A Brief Excursion to Machine Learning II

Problems with Mature Solutions

Machine Learning works well for relatively simple objects with simple properties:

- Classification
- Regression
- (Novelty detection)



Current Research

- Large scale problems (>10 million examples)
- Classification of structured objects, e.g. sequences, networks etc.
- Improving interpretability of learning results
- Prediction of complex properties

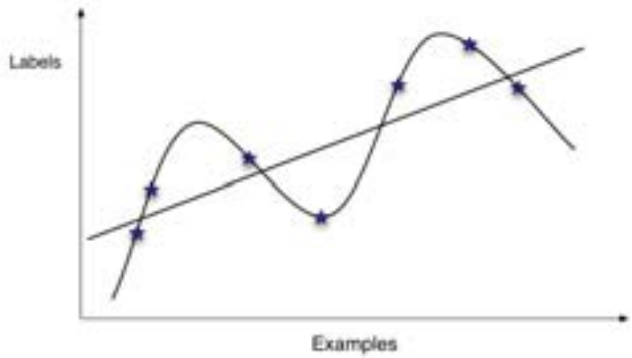
Sonnenburg et al., Journal of Machine Learning Research, 2006:
`http://www.fml.mpg.de/raetsch/projects/shogun`.

A Brief Excursion to Machine Learning II

Problems with Mature Solutions

Machine Learning works well for relatively simple objects with simple properties:

- Classification
- Regression
- (Novelty detection)



Current Research

- Large scale problems (>10 million examples)
- Classification of structured objects, e.g. sequences, networks etc.
- Improving interpretability of learning results
- Prediction of complex properties

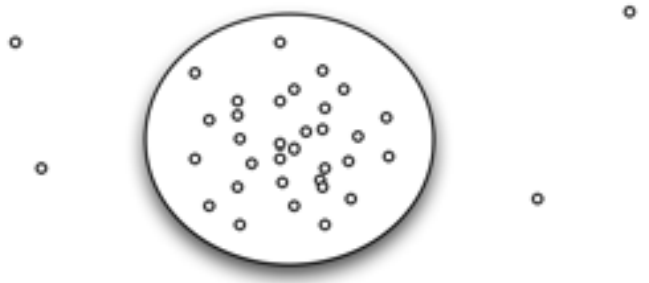
Sonnenburg et al., Journal of Machine Learning Research, 2006:
`http://www.fml.mpg.de/raetsch/projects/shogun`.

A Brief Excursion to Machine Learning II

Problems with Mature Solutions

Machine Learning works well for relatively simple objects with simple properties:

- Classification
- Regression
- (Novelty detection)



Current Research

- Large scale problems (>10 million examples)
- Classification of structured objects, e.g. sequences, networks etc.
- Improving interpretability of learning results
- Prediction of complex properties

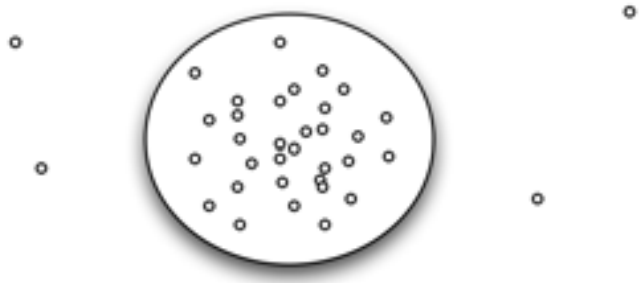
Sonnenburg et al., Journal of Machine Learning Research, 2006:
`http://www.fml.mpg.de/raetsch/projects/shogun`.

A Brief Excursion to Machine Learning II

Problems with Mature Solutions

Machine Learning works well for relatively simple objects with simple properties:

- Classification
- Regression
- (Novelty detection)



Current Research

- Large scale problems (>10 million examples)
- Classification of structured objects, e.g. sequences, networks etc.
- Improving interpretability of learning results
- Prediction of complex properties

Sonnenburg et al., Journal of Machine Learning Research, 2006:
`http://www.fml.mpg.de/raetsch/projects/shogun`.

References

L. Florea, G. Hartzell, Z. Zhang, G.M. Rubin, and W. Miller. A computer program for aligning a cdna sequence with a genomic dna sequence. *Genome Research*, 8:967–974, 1998.

W. J. Kent. BLAT–the BLAST-like alignment tool. *Genome Res*, 12(4):656–664, April 2002.

R. Mott. EST_GENOME: a program to align spliced dna sequences to unspliced genomic dna. *Comput. Appl. Biosci.*, 13:477478, 1997.

G. Rätsch and S. Sonnenburg. Accurate splice site detection for *Caenorhabditis elegans*. In K. Tsuda B. Schoelkopf and J.-P. Vert, editors, *Kernel Methods in Computational Biology*. MIT Press, 2004.

Gunnar Rätsch, Bettina Hepp, Uta Schulze, and Cheng Soon Ong. PALMA: Perfect alignments using large margin algorithms. In *German Conference on Bioinformatics*, 2006.

Sören Sonnenburg, Gunnar Rätsch, Christin Schäfer, and Bernhard Schölkopf. Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, 7:1531–1565, July 2006.

S.J. Wheelan, D.M. Church, and J.M. Ostell. Spidey: a tool for mrna-to-genomic alignments. *Genome Research*, 11(11):1952–7, 2001.

M. Zhang and W. Gish. Improved spliced alignment from an information theoretic approach. *Bioinformatics*, 22(1):13–20, January 2006.