Optimization Algorithms for Identification and Genotyping of Copy Number Polymorphisms in Human Populations

Gökhan Yavaş[1] Mehmet Koyutürk[1,2] Thomas LaFramboise[2,3]
Presented by: Matthew Ruffalo[1]

Case Western Reserve University, Cleveland, OH, USA

[1]Department of Electrical Engineering and Computer Science

[2]Department of Genetics

[3]Center for Proteomics and Bioinformatics

PRIB 2010

Biological Basics

Definition

Copy Number: Quantity of a certain segment or allele in a person's genome (usually 2)

Definition

Copy Number Variation (CNV): Genome segment of at least 1kb in length that varies in copy number from person to person.

Definition

Copy Number Polymorphism (CNP): CNV observed in at least 1% of the population

Justification

- Significance: various diseases are associated with CNPs, such as
  - HIV acquisition and progression
  - lupus glomerulonephritis etc.
- Algorithms that are specifically designed for common CNP discovery are needed!

CNP Identification Framework: POLYGON

- **POLYGON**: a novel optimization based method for identifying common CNPs
- Uses output of existing CNV detection algorithms

Objective

Assign a copy number to all genome markers in all samples such that the copy number assignment is:

- smooth across all markers
- consitent across all samples

Problem Definition

- M markers defined on each of N samples
- $C = \{0, 1, 2, 3, 4\}$ set of copy number classes
- seeking a set of mappings $S : N \times M \rightarrow C$

Input

- a set of CNVs: $V = \{v_1, v_2, \ldots, v_K\}$ identified by any single-sample CNV detection algorithm (each $v \in V$ is a pair (s_v, e_v): start position, end position)
- $R_{n,m}$: the raw copy number estimate for each sample marker $(n, m) \in N \times M$

Our CNV Identification Framework: POLYGON

Two phases:

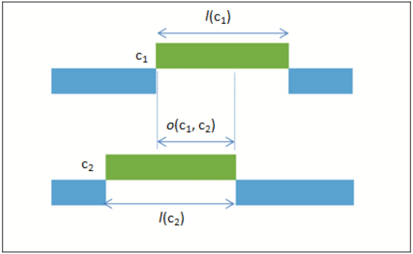
1. Clustering CNVs to obtain an initial set of *candidate CNPs* (clusters of CNVs that potentially correspond to the same event)
2. Fine tuning of the boundaries of candidate CNPs (M_w) and precise estimation of copy number (S_w) in each sample

CNV Similarity Measure

Minimum Reciprocal Overlap

Used to decide whether two CNVs c_1 and c_2 in two different samples correspond to the same event

$$MRO\,(c_1, c_2) = \min\left(\frac{o(c_1, c_2)}{l(c_1)}, \frac{o(c_1, c_2)}{l(c_2)}\right)$$



CNV Cluster Similarity

- *Minimum Reciprocal Overlap* for CNV clusters ρ_i and ρ_j:

$$MRO(\rho_i, \rho_j) = \min_{v_q \in \rho_i, v_p \in \rho_j} \{MRO(v_q, v_p)\}$$

Agglomerative Clustering Process

- Each cluster initially contains a single CNV
- At each iteration, two clusters with maximum overlap are merged
- Clustering stops when the MRO between any two clusters drops below 0.5
- After completion, all CNVs in the same cluster ∴ have at least 50% mutual overlap

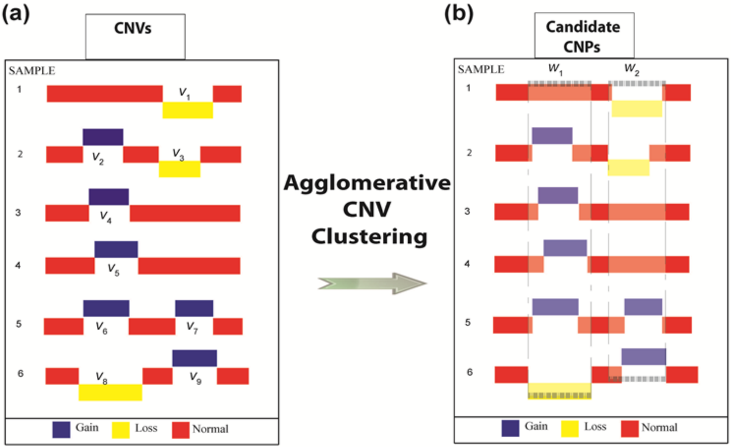
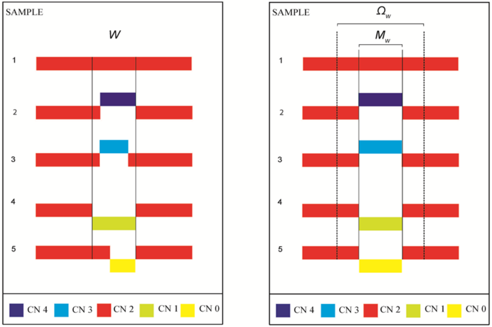


Figure: CNV Clustering Result

CNP Boundary Adjustment

- For each CNP region w spanning a set of markers M_w, select a window Ω_w where M_w is allowed to be enlarged or shrunk such that $l(\Omega_w) = 2l(M_w)$ (with lengths defined in terms of the number of genome markers).



How to find the best S_w and M_w?

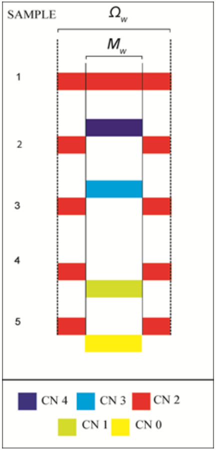
Find S_w and M_w that minimize the following objective function:

$$f(M_w, S_w) = k_\sigma \sigma\left(M_w, S_w\right) + k_\chi \chi\left(M_w, S_w\right) + k_\lambda \lambda\left(M_w\right)$$

$\lambda\left(M_w\right) = \frac{1}{2^{l_w}}$ defines the reliability of a CNP in terms of its length.

In-class Variation Component σ

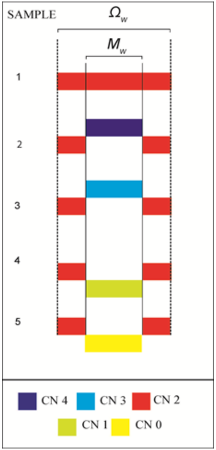
- Variation in raw copy numbers within each copy number class should be minimized.
- $\mu(\square)$ denotes the mean raw copy number for the corresponding class in window w



$\sigma\,(M_w, S_w)$: Σ|🟪−μ(🟪)|+Σ|🟦−μ(🟦)|+Σ|🟥−μ(🟥)|+Σ|🟩−μ(🟩)|+Σ|🟨−μ(🟨)|

Inter-class Variation Component χ

- Variation in raw copy numbers across different copy number classes should be maximized.
- $\mu(\square)$ denotes the mean raw copy number for the corresponding class in window *w*



$$\chi\,(M_w, S_w)\colon\; 2^{1/(\mu(\text{🟪})-\mu(\text{🟦}))}+2^{1/(\mu(\text{🟦})-\mu(\text{🟥}))}+2^{1/(\mu(\text{🟥})-\mu(\text{🟩}))}+2^{1/(\mu(\text{🟩})-\mu(\text{🟨}))}$$

Algorithm for CNP Genotype Optimization

Overview

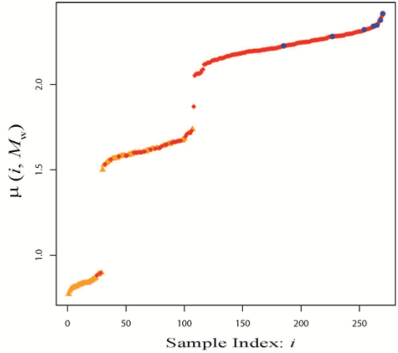
- Solution: marker boundaries M_w and copy number genotype $S_w(n)$ for each sample $n \in N$.
- To find an optimal solution, find an optimal S_w for each possible M_w and choose the best among all possible assignments of M_w.
- Each CNP region is limited to a fixed window Ω_w, which makes this exhaustive search feasible.

Optimal CNP genotyping for fixed boundaries

We define the mean raw copy number of markers within M_w in sample n as:

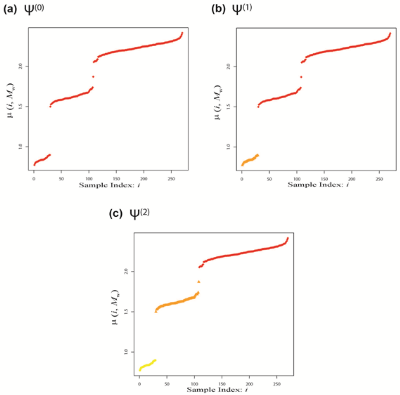
$$\mu\left(n, M_w\right) = \frac{\sum_{m \in M_w} R_{n,m}}{l_w}$$

First, order samples w.r.t. $\mu(i, M_w)$



- Each point represents the mean raw copy value of a sample in region M_w.
- In the figure, the initial class assignments done by a single-sample method are shown.

- Genotype all with copy number class 2
- Next, use a **split & ripple shift strategy** until no more valid splits are left or $f(M_w, S_w)$ does not improve.


(a) $\Psi^{(0)}$
(b) $\Psi^{(1)}$
(c) $\Psi^{(2)}$

- Use the optimal CNP genotyping algorithm on each possible boundary in Ω_w.
- Optimal boundaries of the CNP are set to the coordinates of minimum value in the heat map, and optimal genotype is assigned as before.

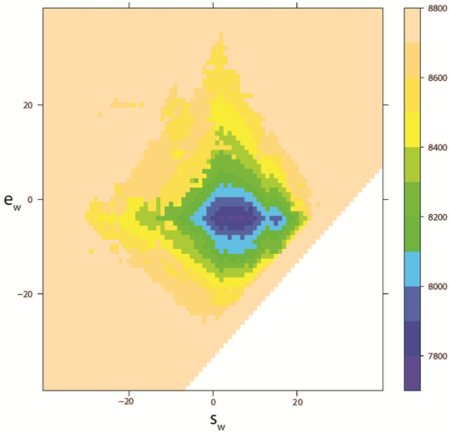


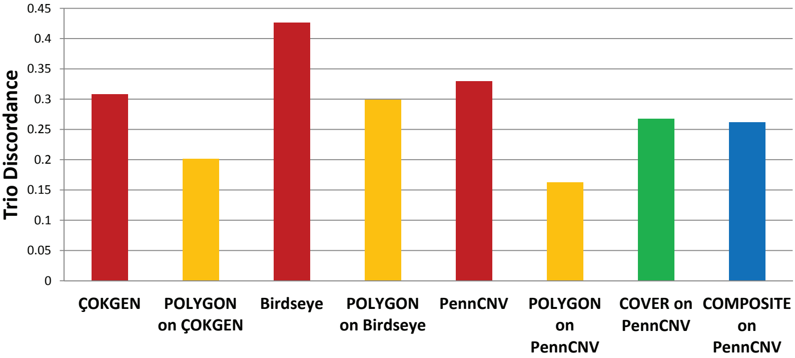
Figure: Example heat map of $f\left(M_w^{(a,b)}, S_w^{(a,b)}\right)$ at the optimal genotype solution for each candidate boundary (a, b), recentered to $(0, 0)$ for demonstration purposes

Results

Performance of POLYGON in Comparison to Existing Software:

- COMPOSITE & COVER (Mei et al., 2010)
- POLYGON performance evaluation used the following single-sample CNV tools:
  - ÇOKGEN (Yavaș et al., 2009)
  - PennCNV (Wang et al., 2007)
  - Birdseye (Korn et al., 2008)

Trio Discordance Performance



Sensitivity[1] Performance

	ÇOKGEN	PennCNV	Birdseye
Initial sensitivity	86%	88.6%	84.7%
Sensitivity by POLYGON	88.3%	88.6%	89.9%
Sensitivity by COMPOSITE	N/A	62.8%	N/A
Sensitivity by COVER	N/A	40.2%	N/A

[1] Sensitivity on a previously reported set of CNVs (Pinto et al., 2007)

Sensitivity Performance

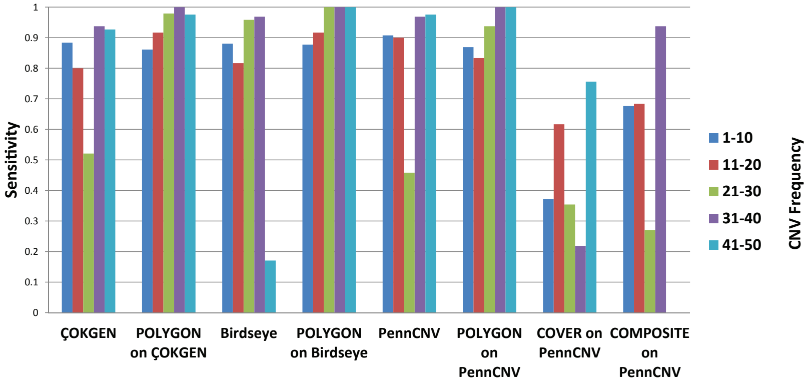


Figure: Sensitivity vs. CNV frequency across different tools

Acknowledgments

Figure: Gökhan Yavaș

Figure: Mehmet Koyutürk

Figure: Tom LaFramboise

- Supported in part by National Science Foundation Award IIS-0916102
- Dr. Meral Özsoyoğlu, EECS department, Case Western Reserve University