

Quick Permutation Test: feature filtering of n-gram data

Piotr Sobczyk^{1*}, Michał Burdukiewicz², Chris Lauber³, Paweł Mackiewicz²
*Piotr.Sobczyk@pwr.edu.pl

¹Wrocław University of Technology, Department of Mathematics, Poland

²University of Wrocław, Department of Genomics, Poland

³Dresden University of Technology, Institute of Medical Informatics and Biometry, Poland

Introduction

N-grams (k-tuples) are vectors of n characters derived from input sequence(s). They may form continuous sub-sequences or be discontinuous. Another important n-gram parameter is its position. Instead of just counting n-grams, one may want to count how many n-grams occur at a given position in multiple (e.g. related) sequences.

	P1	P2	P3	P4	P5	P6
S1	2	2	1	3	3	4
S2	2	1	1	2	4	4
S3	4	2	2	4	1	2

Sample sequences.

1	2	3	4
1	2	2	1
2	2	0	2
1	3	0	2

Unigram counts.

P1	1	P2	1	P3	1	P4	1	P5	1	P6	1	P1	2	P2	2	P3	2	P4	2	P5	2	P6	2	P1	3
0	0	0	1	0	0	0	0	1	1	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1	1	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0

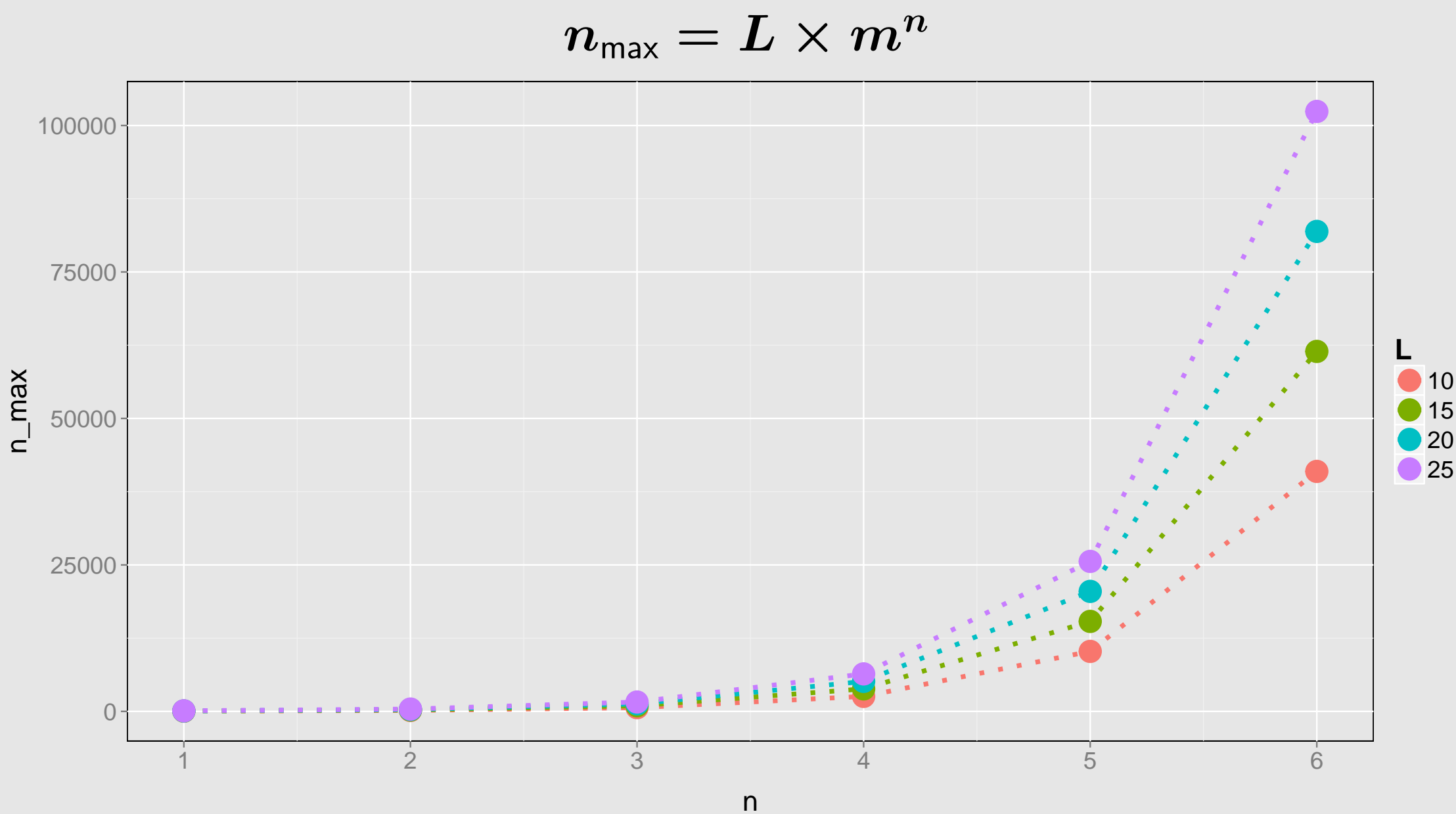
A fraction of possible unigrams with position information.

Originally developed for natural language processing, n-grams are also used in proteomics, genomics and transcriptomics.

Curse of dimensionality

Even when we limit ourselves to only continuous positioned n-grams, feature space grows rapidly with the number of elements in n-gram (n) and length of the sequence (L).

Number of possible positioned n-grams:



Feature selecting permutation tests

Model and statistic independent permutation tests can be used to filter features obtained through counting n-grams. During a permutation test class labels are randomly exchanged during computation of significance statistic. p-values are defined as:

$$\text{p-value} = \frac{N_{T_P > T_R}}{N}$$

where $N_{T_P > T_R}$ is number of times when T_P (permuted test statistic) was more extreme than T_R (test statistic for non-permuted data). Permutation tests are computationally expensive (especially precise estimation of low p-values, because the number of permutations is inversely proportional to the interval between p-values).

QuiPT algorithm

If probability that target equals 1 is p and probability that feature equals 1 is q and feature and target are independent then each of them has the following probabilities

$$P(\text{Target}, \text{Feature}) = (1, 1)) = p \cdot q$$

$$P(\text{Target}, \text{Feature}) = (1, 0)) = p \cdot (1 - q)$$

$$P(\text{Target}, \text{Feature}) = (0, 1)) = (1 - p) \cdot q$$

$$P(\text{Target}, \text{Feature}) = (0, 0)) = (1 - p) \cdot (1 - q)$$

Therefore another view is that we get a contingency table that needs to be tested for independence.

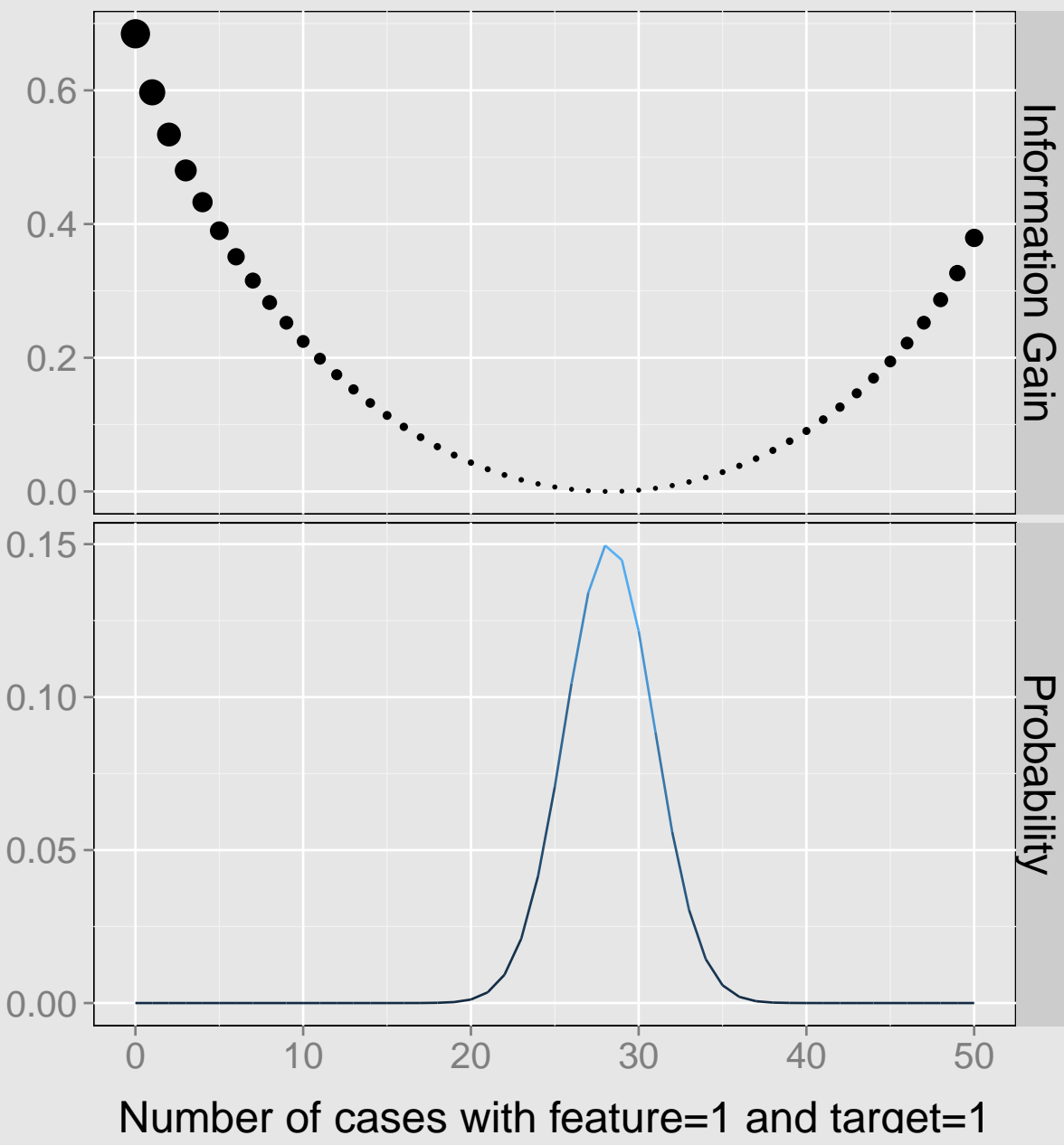
Independence test

$$F(n_{1,1}, n_{1,0}, n_{0,1}, n_{0,0}) = \binom{n}{n_{1,1}} (p \cdot q)^{n_{1,1}} \binom{n - n_{1,1}}{n_{1,0}} (p \cdot (1 - q))^{n_{1,0}} \binom{n - n_{1,1} - n_{1,0}}{n_{0,1}} ((1 - p) \cdot q)^{n_{0,1}} \binom{n - n_{1,1} - n_{1,0} - n_{0,1}}{n_{0,0}} ((1 - p) \cdot (1 - q))^{n_{0,0}}$$

This distribution comes with two constraints: $n_{1,\cdot} = n_{1,1} + n_{1,0}$ and $n_{\cdot,1} = n_{1,1} + n_{0,1}$. Thus, conditioning on $n_{1,1}$, we get hypergeometric distribution.

This is in fact exact two-sided Fisher's test. Information Gain is used here as a way of deciding which contingency tables are more extreme.

Validation procedure

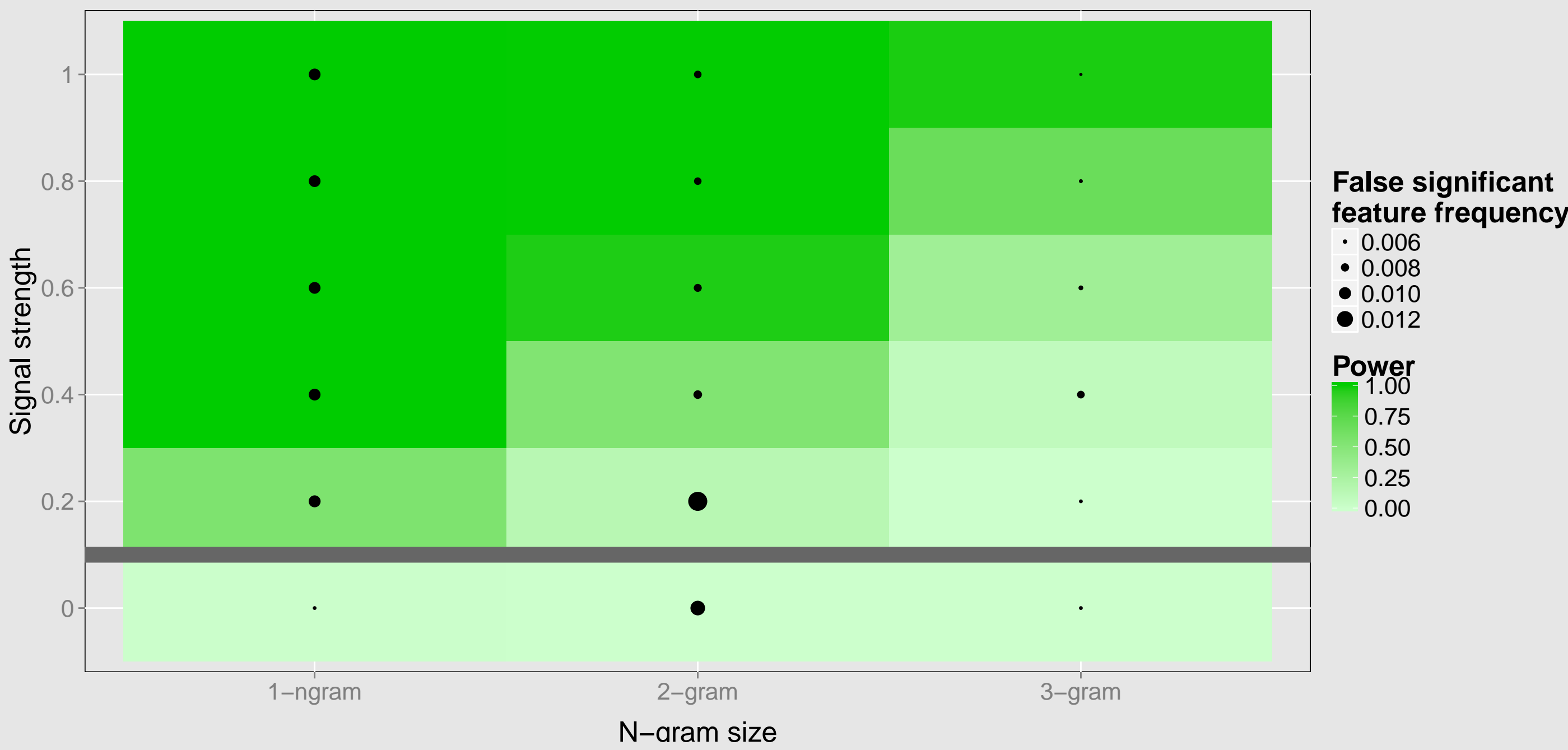


	Target	Feature	Freq
1	0	0	40
2	1	0	10
3	0	1	25
4	1	1	40

Simulation scheme

1. Random 4000 sequences (20 nucleotides each). The half of the sequences has label 0.
2. Choose a single position between 3 and 18 (to avoid border cases).
3. Resample nucleotides at chosen position. The dominant nucleotide has probability of occurrence $p_d = 0.25$. Other nucleotides have probability of occurrence $p_o = (1 - p_d)/3$.
4. Perform QuiPT (Information Gain) and choose significant features (with p-value < 0.001).
5. Iterate steps 1-4 over other values of p_d - 0.38, 0.51, 0.65, 0.78, 0.91.
6. Repeat steps 1-5 200 times.

Power and False discoveries



Summary

Quick permutation test is a powerful and quick equivalent of permutation test in binary feature-binary target testing scenario.

Availability

biogram R package:
<http://cran.r-project.org/web/packages/biogram/>

Bibliography