

# Quick Permutation Test: feature filtering of n-gram data

Piotr Sobczyk<sup>1\*</sup>, Michał Burdukiewicz<sup>2</sup>, Chris Lauber<sup>3</sup>, Paweł Mackiewicz<sup>2</sup>  
\*Piotr.Sobczyk@pwr.edu.pl

<sup>1</sup>Wrocław University of Technology, Institute of Mathematics and Computer Science, Poland

<sup>2</sup>University of Wrocław, Department of Genomics, Poland

<sup>3</sup>Dresden 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	4	2	4	3	4	3
S2	1	1	3	2	2	2
S3	2	4	4	2	3	1

Sample sequences.

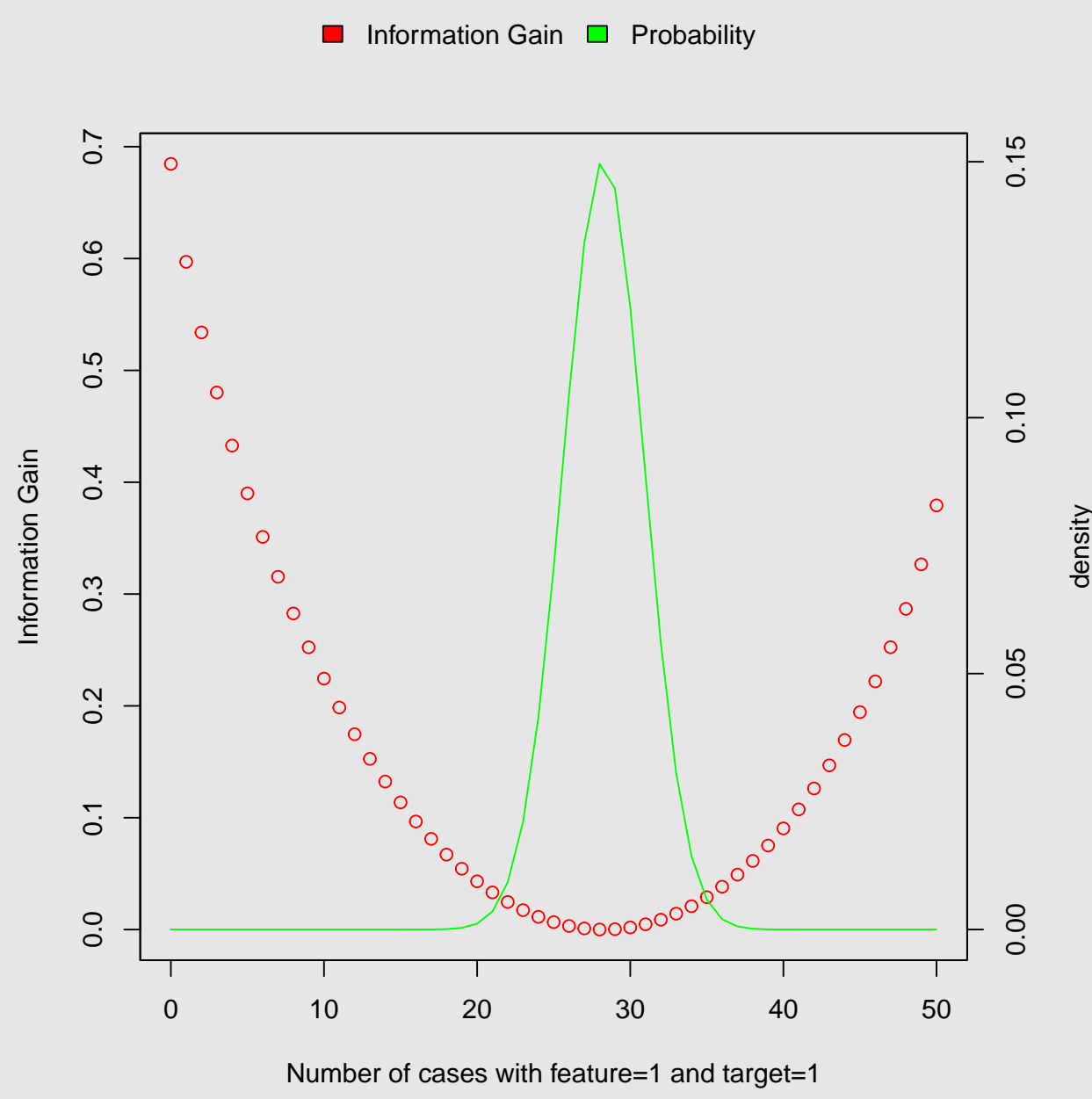
	1	2	3	4
0	1	2	3	
2	3	1	0	
1	2	1	2	

Unigram counts.

P1_1	P2_1	P3_1	P4_1	P5_1	P6_1	P1_2	P2_2	P3_2	P4_2	P5_2
0	0	0	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	0	0	1	1
0	0	0	0	0	1	1	0	0	1	0

A fraction of possible unigrams with position information.

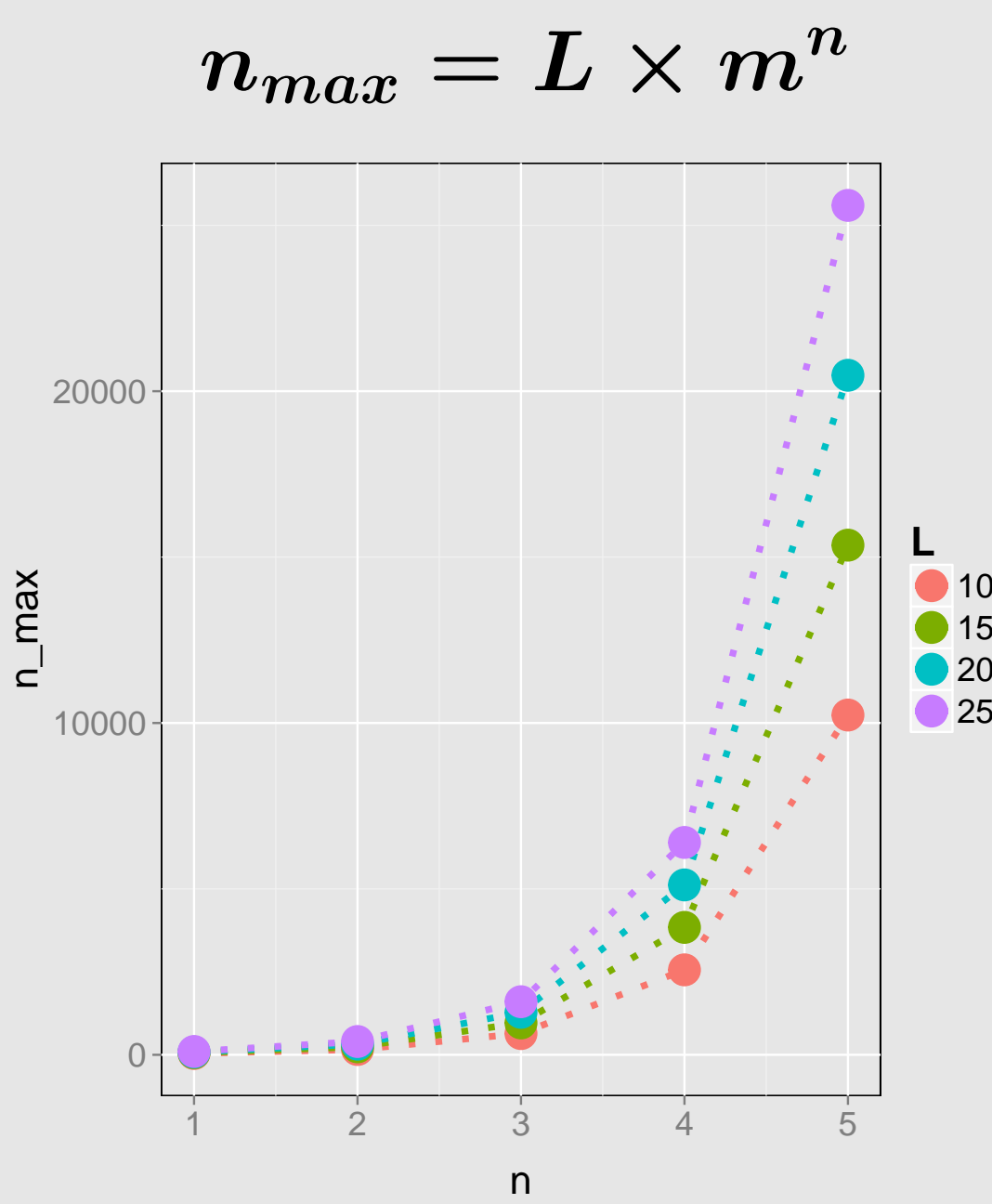
## Validation procedure



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

## Curse of dimensionality

Number of possible positioned n-grams (not taking into account distances between elemnts of n-gram):



## Permutation test

During permutation tests 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 model and statistic independent, but 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

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

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

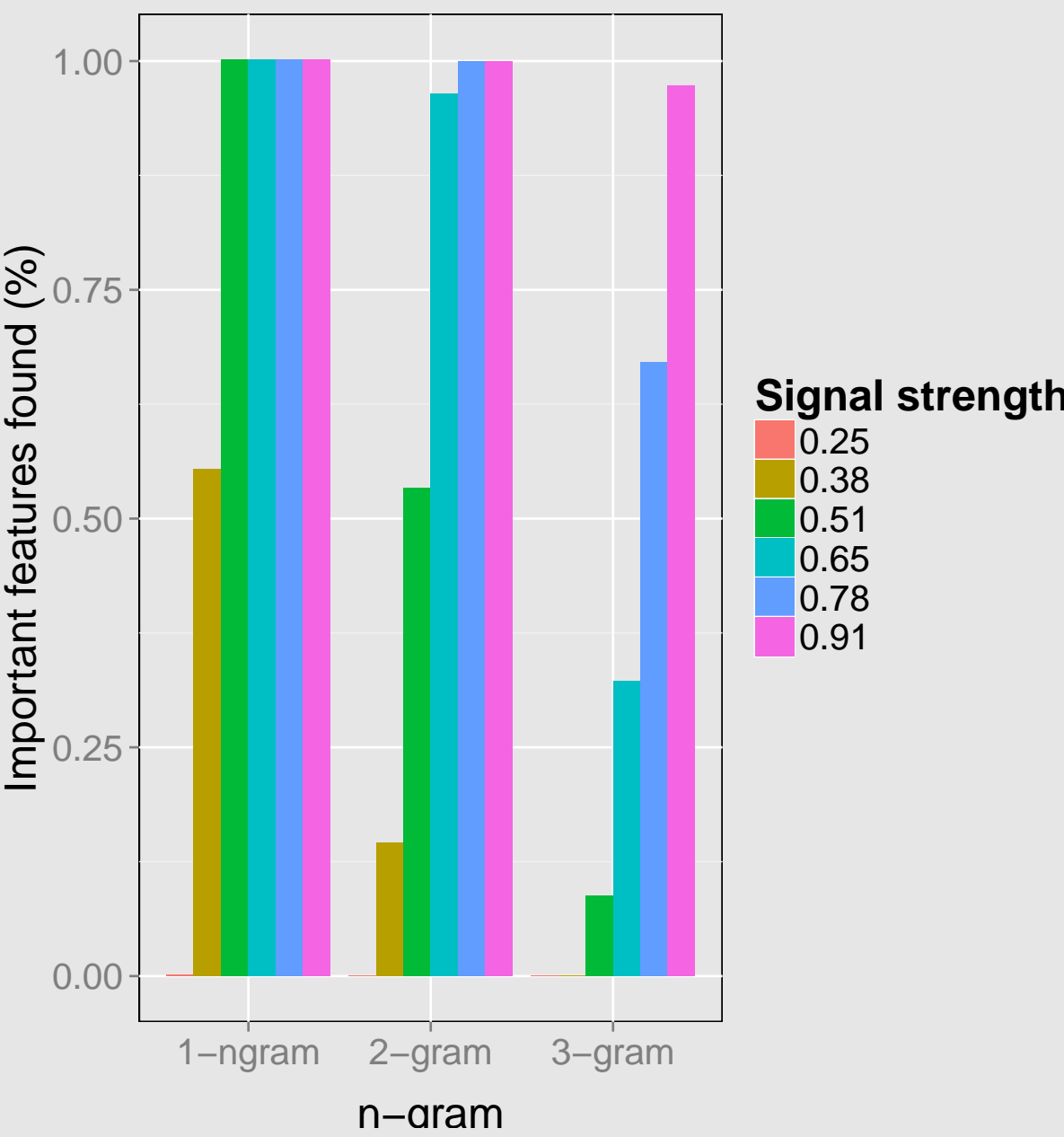
$$P(\text{Target}, \text{Feature}) = (0, 1)) = (1 - p) \cdot q$$

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

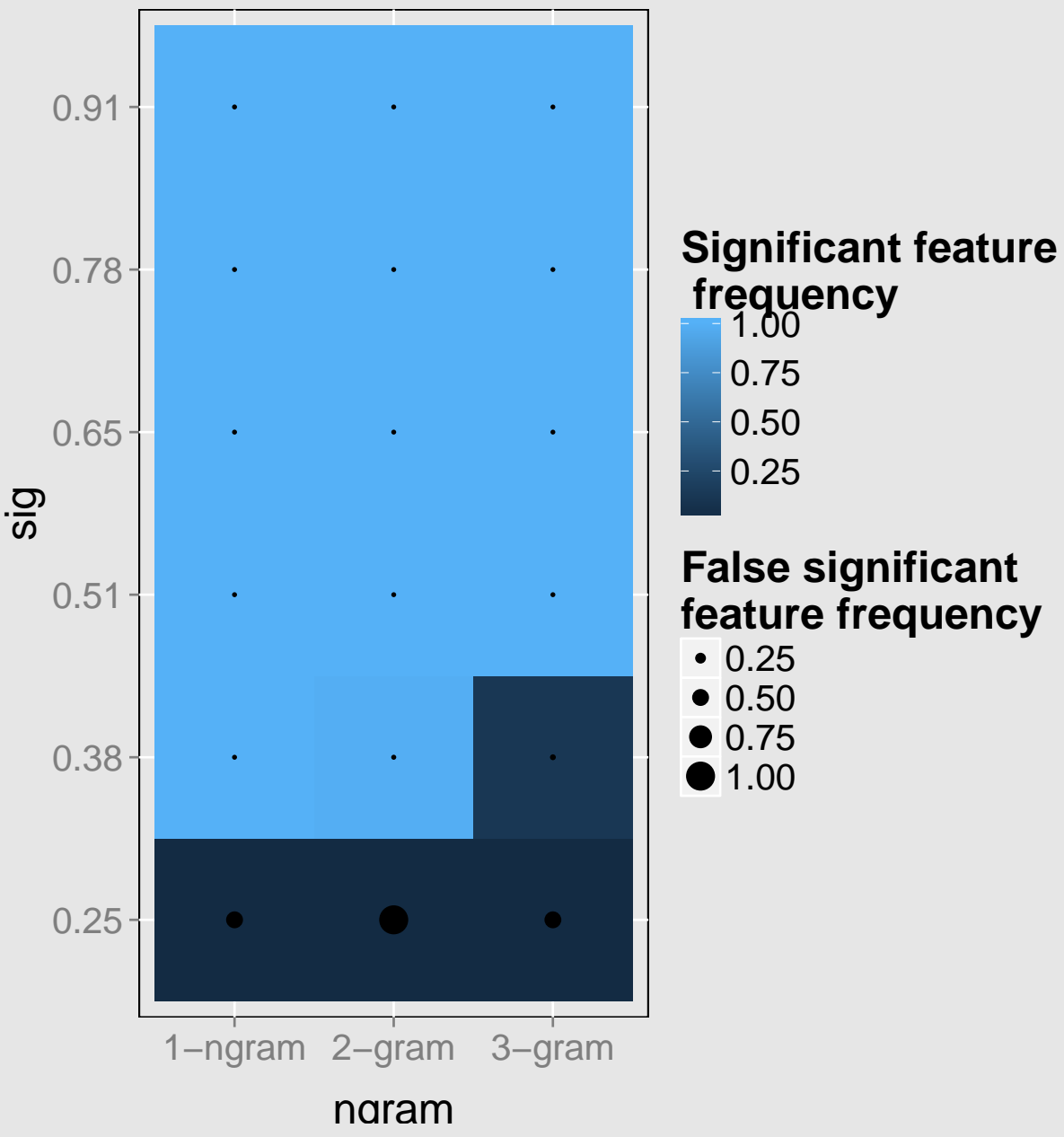
$$F(n_{1,1}, n_{1,0}, n_{0,1}, n_{0,0}) = \binom{n}{n_{1,1}} (p \cdot q)^{n_{1,1}} n - n_{1,1} \binom{n}{n_{1,0}} (p \cdot (1 - q))^{n_{1,0}} n - n_{1,1} - n_{1,0} \binom{n}{n_{0,1}} ((1 - p) \cdot q)^{n_{0,1}} n - n_{1,1} - n_{1,0} - n_{0,1} \binom{n}{n_{0,0}} ((1 - p) \cdot (1 - q))^{n_{0,0}}$$

In addition to this:  $n_{1,\cdot} = n_{1,1} + n_{1,0}$  and  $n_{\cdot,1} = n_{1,1} + n_{0,1}$  are known and fixed.

## Test power



## 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