Quick Permutation Test: feature filtering of n-gram data

Piotr Sobczyk[1]*, Michał Burdukiewicz[2], Chris Lauber[3], Paweł Mackiewicz[2]
*Piotr.Sobczyk@pwr.edu.pl

[1]Wrocław University of Technology, Department of Mathematics, Poland
[2]University of Wrocław, Department of Genomics, Poland
[3]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. 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.
Originally developed for natural language processing, n-grams are also used in genomics (Fang et al., 2011), transcriptomics (Wang et al., 2014) and proteomics (Guo et al., 2014).

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

Sample sequences. S - sequence, P - postion.

	1	2	3	4
S1	2	0	2	2
S2	0	2	2	2
S3	0	3	1	2

Unigram counts.

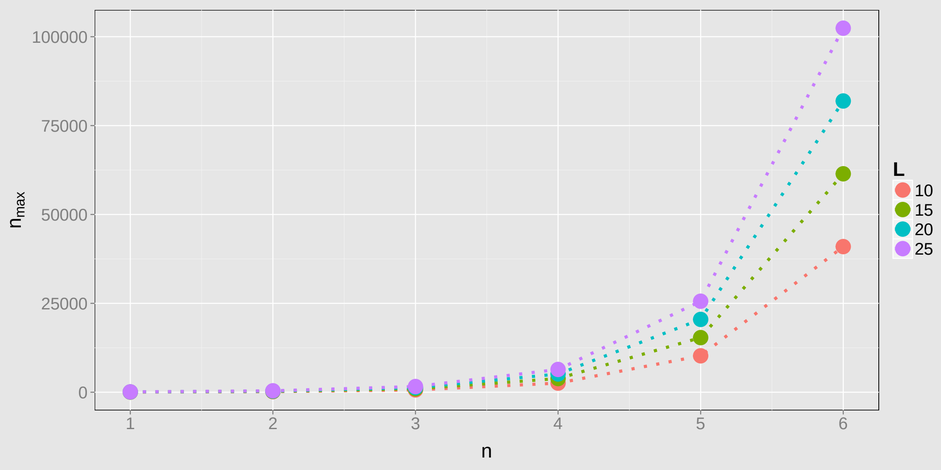
	P1_1	P2_1	P3_1	P4_1	P5_1	P6_1	P1_2	P2_2	P3_2	P4_2	P5_2	P6_2	P1_3
S1	0	0	0	0	1	1	0	0	0	0	0	0	1
S2	0	0	0	0	0	0	0	1	1	0	0	0	1
S3	0	0	0	0	0	0	0	0	1	1	1	0	0

A fraction of possible unigrams with position information.

Curse of dimensionality

Even when we limit ourselves to only continuous positioned n-grams build on $\boldsymbol{m}$ possible characters, feature space growths rapidly with the number of elements in n-gram ($\boldsymbol{n}$) and the length of the sequence ($\boldsymbol{L}$). The number of possible positioned n-grams:

$$\boldsymbol{n}_{\text{max}} = \boldsymbol{L} \times \boldsymbol{m}^{\boldsymbol{n}}$$



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 a significance statistic. p-values are defined as:

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

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

QuiPT concept

In each permutation, for every observation, there are four possible results:

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

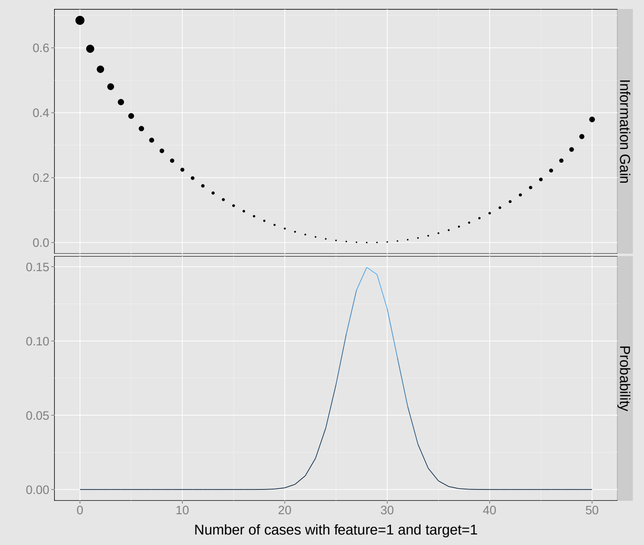
Where $\boldsymbol{p}$ and $\boldsymbol{q}$ are fractions of positive observations in target and feature respectively. An another view at permutation test is therefore that we get a contingency table, which is to be tested for independence. Computing probability of a such table with two constraints, $\boldsymbol{n_{1,\cdot} = n_{1,1} + n_{1,0}}$ and $\boldsymbol{n_{\cdot,1} = n_{1,1} + n_{0,1}}$, and conditioning on $\boldsymbol{n_{1,1}}$, leads to hypergeometric distribution. $\boldsymbol{n_{i,j}}$ denotes number of observations for which $\boldsymbol{(Target, Feature) = (i,j)}$
This is in fact exact two-sided Fisher's test (Lehmann, 1986).

Computational cost

The cost of performing QuiPT is equal to computing a test statistic and probability of occurrence for $\boldsymbol{n_{1,1} + n_{0,1}}$ contingency tables.
Suppose we consider 6-grams build on sequences of length 25 build of four characters. Then there are around 100,000 n-grams (features) to test. This means that for Benjamini-Hochberg procedure, we need to calculate p-values with accuracy of $\boldsymbol{0.05 \times 10^{-5}}$. This requires at least 2 million permutations. Each permutation, apart from reshuffling labels, requires computation of a test statistic. Since n-gram features are very sparse vectors, QuiPT needs to evaluate only few contingency tables.
The relative difference in speed between QuiPT and normal permutation tests depends on several factors, as a number of permutations and input data. For example, for simulation scheme presented below, QuiPT was on average 93 times faster than normal permutation test with $\boldsymbol{10^5}$ permutations.

Distribution of Information Gain for given contingency table

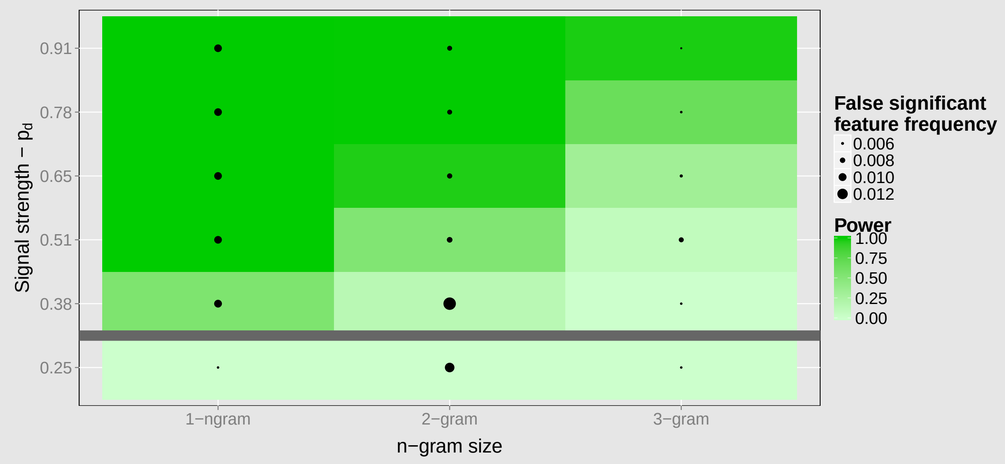
Given constraint on $\boldsymbol{n_{1,1} + n_{0,1}}$, probability distribution on contingency tables, which permutations might produce, can be computed exactly.



Simulation scheme - genomics

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 $\boldsymbol{p_d = 0.25}$. Other nucleotides have probability of occurrence $\boldsymbol{p_o = (1-p_d)/3}$.
4. Perform QuiPT (Information Gain as test statistic) and choose significant features (with p-value $\boldsymbol{< 0.001}$).
5. Iterate steps 1-4 over other values of $\boldsymbol{p_d}$ - 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 a binary feature – binary target testing scenario. It is especially useful when precisely computed p-values are required and features are sparse vectors.

Avaibility

QuiPT is a part of **biogram** R package devoted to the analysis of n-grams extracted from biological sequences:
`http://cran.r-project.org/web/packages/biogram/`

Bibliography

Fang, Y.-C., Lai, P.-T., Dai, H.-J., and Hsu, W.-L. (2011). Meinfotext 2.0: gene methylation and cancer relation extraction from biomedical literature. *BMC Bioinformatics*, 12(1):471.

Guo, S.-H., Deng, E.-Z., Xu, L.-Q., Ding, H., Lin, H., Chen, W., and Chou, K.-C. (2014). inuc-pseknc: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition. *Bioinformatics*, 30(11):1522–1529.

Lehmann, E. (1986). *Testing statistical hypotheses*. Wiley series in probability and mathematical statistics: Probability and mathematical statistics. Wiley.

Wang, Y., Liu, L., Chen, L., Chen, T., and Sun, F. (2014). Comparison of metatranscriptomic samples based on k–tuple frequencies. *PLoS ONE*, 9(1):e84348.