Biomolecular Detection and Quantification 13 (2017) 7–31

Contents lists available at ScienceDirect

Biomolecular Detection and Quantification

journal homepage: www.elsevier.com/locate/bdq

Research paper

*K-means and cluster models for cancer signatures

Zura Kakushadze[a,b,1,*], Willie Yu[c]

[a] Quantigic® Solutions LLC, 1127 High Ridge Road #135, Stamford, CT 06905, United States
[b] Free University of Tbilisi, Business School & School of Physics, 240, David Agmashenebeli Alley, Tbilisi 0159, Georgia
[c] Centre for Computational Biology, Duke-NUS Medical School, 8 College Road, Singapore 169857, Singapore

ARTICLE INFO

Handled by Jim Huggett

Keywords:
Clustering
K-means
Nonnegative matrix factorization
Somatic mutation
Cancer signatures
Genome
eRank
Machine learning
Sample
Source code

ABSTRACT

We present *K-means clustering algorithm and source code by expanding statistical clustering methods applied in https://ssrn.com/abstract=2802753 to quantitative finance. *K-means is statistically deterministic without specifying initial centers, etc. We apply *K-means to extracting cancer signatures from genome data without using nonnegative matrix factorization (NMF). *K-means' computational cost is a fraction of NMF's. Using 1389 published samples for 14 cancer types, we find that 3 cancers (liver cancer, lung cancer and renal cell carcinoma) stand out and do not have cluster-like structures. Two clusters have especially high within-cluster correlations with 11 other cancers indicating common underlying structures. Our approach opens a novel avenue for studying such structures. *K-means is universal and can be applied in other fields. We discuss some potential applications in quantitative finance.

1. Introduction and summary

Every time we can learn something new about cancer, the motivation goes without saying. Cancer is different. Unlike other diseases, it is not caused by "mechanical" breakdowns, biochemical imbalances, etc. Instead, cancer occurs at the DNA level via somatic alterations in the genome structure. A common type of somatic mutations found in cancer is due to single nucleotide variations (SNVs) or alterations to single bases in the genome, which accumulate through the lifespan of the cancer via imperfect DNA replication during cell division or spontaneous cytosine deamination [1,2], or due to exposures to chemical insults or ultraviolet radiation [3,4], etc. These mutational processes leave a footprint in the cancer genome characterized by distinctive alteration patterns or mutational signatures.

If we can identify all underlying signatures, this could greatly facilitate progress in understanding the origins of cancer and its development. Therapeutically, if there are common underlying structures across different cancer types, then a therapeutic for one cancer type might be applicable to other cancers, which would be a great news.[2] However, it all boils down to the question of usefulness, i.e., is there a small enough number of cancer signatures underlying all (100+) known cancer types, or is this number too large to be meaningful or useful? Indeed, there are only 96 SNVs,[3] so we cannot have more than 96 signatures.[4] Even if the number of true underlying signatures is, say, of order 50, it is unclear whether they would be useful, especially within practical applications. On the other hand, if there are only a dozen or so underlying signatures, then we could hope for an order of magnitude simplification.

To identify mutational signatures, one analyzes SNV patterns in a cohort of DNA sequenced whole cancer genomes. The data is organized into a matrix G_{is}, where the rows correspond to the $N = 96$ mutation categories, the columns correspond to d samples, and each element is a

* Corresponding author at: Quantigic® Solutions LLC, 1127 High Ridge Road #135, Stamford, CT 06905, United States.
E-mail addresses: zura@quantigic.com (Z. Kakushadze), willie.yu@duke-nus.edu.sg (W. Yu).

[1] Disclaimer: This address is used by the corresponding author for no purpose other than to indicate his professional affiliation as is customary in publications. In particular, the contents of this paper are not intended as an investment, legal, tax or any other such advice, and in no way represent views of Quantigic® Solutions LLC, the website www.quantigic.com or any of their other affiliates.

[2] Another practical application is prevention by pairing the signatures extracted from cancer samples with those caused by known carcinogens (e.g., tobacco, aflatoxin, UV radiation, etc).

[3] In brief, DNA is a double helix of two strands, and each strand is a string of letters A, C, G, T corresponding to adenine, cytosine, guanine and thymine, respectively. In the double helix, A in one strand always binds with T in the other, and G always binds with C. This is known as base complementarity. Thus, there are six possible base mutations C>A, C>G, C>T, T>A, T>C, T>G, whereas the other six base mutations are equivalent to these by base complementarity. Each of these 6 possible base mutations is flanked by 4 possible bases on each side thereby producing $4 \times 6 \times 4 = 96$ distinct mutation categories.

[4] Nonlinearities could undermine this argument. However, again, it all boils down to usefulness.

http://dx.doi.org/10.1016/j.bdq.2017.07.001
Received 13 June 2017; Received in revised form 18 July 2017; Accepted 18 July 2017
Available online 02 August 2017

nonnegative occurrence count of a given mutation category in a given sample. Currently, the commonly accepted method for extracting cancer signatures from G_{is} [5] is via nonnegative matrix factorization (NMF) [6,7]. Under NMF the matrix G is approximated via $G \approx W H$, where W_{iA} is an $N \times K$ matrix, H_{As} is a $K \times d$ matrix, and both W and H are nonnegative. The appeal of NMF is its biologic interpretation whereby the K columns of the matrix W are interpreted as the weights with which the K cancer signatures contribute into the $N = 96$ mutation categories, and the columns of the matrix H are interpreted as the exposures to the K signatures in each sample. The price to pay for this is that NMF, which is an iterative procedure, is computationally costly and depending on the number of samples d it can take days or even weeks to run it. Furthermore, it does not automatically fix the number of signatures K, which must be either guessed or obtained via trial and error, thereby further adding to the computational cost.[5]

Some of the aforesaid issues were recently addressed in [8], to wit: (i) by aggregating samples by cancer types, we can greatly improve stability and reduce the number of signatures;[6] (ii) by identifying and factoring out the somatic mutational noise, or the "overall" mode (this is the "de-noising" procedure of [8]), we can further greatly improve stability and, as a bonus, reduce computational cost; and (iii) the number of signatures can be fixed borrowing the methods from statistical risk models [9] in quantitative finance, by computing the effective rank (or eRank) [10] for the correlation matrix Ψ_{ij} calculated across cancer types or samples (see below). All this yields substantial improvements [8].

In this paper we push this program to yet another level. The basic idea here is quite simple (but, as it turns out, nontrivial to implement – see below). We wish to apply clustering techniques to the problem of extracting cancer signatures. In fact, we argue in Section 2 that NMF is, to a degree, "clustering in disguise". This is for two main reasons. The prosaic reason is that NMF, being a nondeterministic algorithm, requires averaging over many local optima it produces. However, each run generally produces a weights matrix W_{iA} with columns (i.e., signatures) not aligned with those in other runs. Aligning or matching the signatures across different runs (before averaging over them) is typically achieved via nondeterministic clustering such as k-means. So, not only is clustering utilized at some layer, the result, even after averaging, generally is both noisy[7] and nondeterministic! I.e., if this computationally costly procedure (which includes averaging) is run again and again on the same data, generally it will yield different looking cancer signatures every time!

The second, not-so-prosaic reason is that, while NMF generically does not produce exactly null weights, it does produce low weights, such that they are within error bars. For all practical purposes we might as well set such weights to zero. NMF requires nonnegative weights. However, we could as reasonably require that the weights should be, say, outside error bars (e.g., above one standard deviation – this would render the algorithm highly recursive and potentially unstable or computationally too costly) or above some minimum threshold (which would still further complicated as-is complicated NMF), or else the noncompliant weights are set to zero. As we increase this minimum threshold, the matrix W_{iA} will start to have more and more zeros. It may not exactly have a binary cluster-like structure, but it may at least have some substructures that are cluster-like. It then begs the question: are there cluster-like (sub)structures present in W_{iA} or, generally, in cancer signatures?

To answer this question, we can apply clustering methods directly to the matrix G_{is}, or, more, precisely, to its de-noised version G'_{is} (see below) [8]. The naïve, brute-force approach where one would simply cluster G_{is} or G'_{is} does not work for a variety of reasons, some being more nontrivial or subtle than others. Thus, e.g., as discussed in [8], the counts G_{is} have skewed, long-tailed distributions and one should work with log-counts, or, more precisely, their de-noised versions. This applies to clustering as well. Further, following a discussion in [11] in the context of quantitative trading, it would be suboptimal to cluster de-noised log-counts. Instead, it pays to cluster their normalized variants (see Section 2 hereof). However, taking care of such subtleties does not alleviate one big problem: nondeterminism![8] If we run a vanilla nondeterministic algorithm such as k-means on the data however massaged with whatever bells and whistles, we will get random-looking disparate results every time we run k-means with no stability in sight. We need to address nondeterminism!

Our solution to the problem is what we term **K-means*. The idea behind *K-means, which essentially achieves determinism *statistically*, is simple. Suppose we have an $N \times d$ matrix X_{is}, i.e., we have N d-vectors $\mathbf{X}_i$. If we run k-means with the input number of clusters K but initially unspecified centers, every run will generally produce a new local optimum. *K-means reduces and in fact essentially eliminates this indeterminism via two levels. At level 1 it takes clusterings obtained via M independent runs or samplings. Each sampling produces a binary $N \times K$ matrix Ω_{iA}, whose element equals 1 if $\mathbf{X}_i$ belongs to the cluster labeled by A, and 0 otherwise. The aggregation algorithm and the source code therefor are given in [11]. This aggregation – for the same reasons as in NMF (see above) – involves aligning clusters across the M runs, which is achieved via k-means, and so the result is nondeterministic. However, by aggregating a large number M of samplings, the degree of nondeterminism is greatly reduced. The "catch" is that sometimes this aggregation yields a clustering with $K' < K$ clusters, but this does not pose an issue. Thus, at level 2, we take a large number P of such aggregations (each based on M samplings). The occurrence counts of aggregated clusterings are not uniform but typically have a (sharply) peaked distribution around a few (or manageable) number of aggregated clusterings. So this way we can pinpoint the "ultimate" clustering, which is simply the aggregated clustering with the highest occurrence count. This is the gist of *K-means and it works well for genome data.

So, we apply *K-mean to the same genome data as in [8] consisting of 1389 (published) samples across 14 cancer types (see below). Our target number of clusters is 7, which was obtained in [8] using the eRank based algorithm (see above). We aggregated 1000 samplings into clusterings, and we constructed 150,000 such aggregated clusterings (i.e., we ran 150 million k-means instances). We indeed found the "ultimate" clustering with 7 clusters. Once the clustering is fixed, it turns out that within-cluster weights can be computed via linear regressions (with some bells and whistles) and the weights are automatically positive. That is, we do not need NMF at all! Once we have clusters and weights, we can study reconstruction accuracy and within-cluster correlations between the underlying data and the fitted data that the cluster model produces.

We find that clustering works well for 10 out the 14 cancer types we study. The cancer types for which clustering does not appear to work all that well are Liver Cancer, Lung Cancer, and Renal Cell Carcinoma. Also, above 80% within-cluster correlations arise for 5 out of 7 clusters. Furthermore, remarkably, one cluster has high within-cluster correlations for 9 cancer types, and another cluster for 6 cancer types. These

[5] Other issues include: (i) out-of-sample instability, i.e., the signatures obtained from non-overlapping sets of samples can be dramatically different; (ii) in-sample instability, i.e., the signatures can have a strong dependence on the initial iteration choice; and (iii) samples with low counts or sparsely populated samples (i.e., those with many zeros – such samples are ubiquitous, e.g., in exome data) are usually deemed not too useful as they contribute to the in-sample instability.

[6] As a result, now we have the so-aggregated matrix G_{is}, where $s = 1, \ldots, d$, and $d = n$ is the number of cancer types, not of samples. This matrix is much less noisy than the sample data.

[7] By "noise" we mean the statistical errors in the weighs obtained by averaging. Typically, such error bars are not reported in the literature on cancer signatures. Usually they are large.

[8] Deterministic (e.g., agglomerative hierarchical) algorithms have their own issues (see below).