Personal genomics: new concepts for future community data banks.

Guy Dodin

Université Paris Diderot, Laboratoire ITODYS-CNRS UMR 7086, 15 rue Jean-Antoine de Baïf 75005 Paris

guydodin@orange.fr

dodin@paris7.jussieu.fr

INTRODUCTION

The advent of fast, low-cost sequencing techniques is boosting the number of sequenced genomes both human and non-human.[1] Numerous voices advocate the necessity for defining a fair 'serve-and-protect' rule in order to guarantee both easy access to genomic data and, at the same time, preservation of privacy [2,3]. For example, it has been shown that re-identification of anonymous donors can be achieved through crossing freely available partial genomic information and family names.[4] In order to preserve privacy, scores of algorithms have been devised to ensure protection of both textual (name of a person, his or her attached medical files, etc..) and sequence data.[5,6] Access to data banks restricted to acknowledged research institutions complying with ethical guidelines and data sharing protocols also provide efficient safeguards against abusive exploitation of information. Encryption, at the price of encryption-decryption keys management, provides a supplementary level towards better security. [7,8,9]

It is predictable that in a very near future millions of individuals will have access to their personal genomic data. Reasserting the motto 'Your data are not a product',[9] and adding 'your data are your property' a person will likely want to personally exert control on how his or her personal details are dispatched and utilized

The primary concern when it comes to banks is the resilience of the stored information to breaches in confidentiality and integrity. Data is generally stored as a whole (social security numbers, genomic sequences like ATGC.., etc.). Even encrypted, at the price of heavy storage and retrieval operations, information remains accessible to intruders due to the intrinsic limits of cipher protocols and to progress in computer technology.

In order to sum up the above observations and to reconcile security of data, control of its utilization and availability to the scientific community, a new type of genomic data bank is proposed here. The concept is grounded onto two main new ideas.

On the one hand, genomic data are split into several parts independently stored. The information content of each piece should be low enough so that breaking the confidentiality and the integrity of one will not allow the full information to be retrieved. In this context, data no longer need to be encrypted thus allowing easier handling. Split information is also the basis for a controlled dispatching of data.

On the other hand, the new bank will be a community bank, belonging to consenting individuals, and organized along a scheme borrowing from the blockchain technology. Free mining will be permitted (provided certain conditions are fulfilled) under the permanent control of the bank's community.

SPLIT GENOMIC DATA

The prior requirement of the new concept is that access to one piece of the split data (or several up to a certain limit) will never deliver enough information to unveiling a genomic sequence stored in the bank. How genomic information can be made incomplete surely? A genomic sequence can be 'vertically' chopped into **n** subsequences each being subsequently stored separately. Although it assures confidentiality (access to one or several subsequences does not reveal the full information), this approach is unable to readily detect breaches of integrity stemming from addition, deletion, substitution of nucleotides. Another concern may arise when a subsequence is handed out by the bank to an outside investigator since the latter is likely to check open-access genomic banks for similar sequences through alignment algorithms [11]. However, this is a minor issue in the context where individual's information would now exclusively be stored in the community bank, and even if a successful alignment is observed, confidentiality is not at stake since similarity does not mean exact matching with the genuine sequence.

Alternatively, a genomic sequence can also be split 'horizontally' that is, a sequence like ATTCAGC can be represented as 4 files (subsequently referred to as Base-files): ANNNANN, NTTNNNN, NNNNNGN, NNNCNNC. The literal Base-files can be replaced by binary sequences, called bin (Base) sequences, where, for example, the above A-file leads to bin(A)=1000100, the T-file gives bin(T) =0110000, etc. This operation is not simply swapping symbols but rather it permits to transform letter sequences into numerical objects now amenable to processing by formal tools like Boolean algebra, linear algebra, Fourier Transform. It will be shown how Boolean OR and XOR (exclusive OR) operations, and binary array representations readily allow detecting breaches in data integrity and easy reconstruction of the literal genomic sequence. The binaries can also be compressed as decimal or hexadecimal numbers, thus making their storage easier. The bin(Base) representation of genomic sequences will be used throughout the rest of this contribution.

Properties of the binary files

Confidentiality

Is a single binary sequence or the union of two of them enough to reconstruct the full genomic sequence or at least to provide sufficient information to break confidentiality? An illustration is provided by the following example (see also Results). In a 100 base-long nucleotide sequence consisting of equal proportion of A, T, G, C, each binary sequence has about 25 '**1**' and 75 '**0**'. The number of possible nucleotide sequences with the same bin(Base)(that is the number of colliding sequences) would be equal to 3^{75}, and hence the probability on finding the true sequence by chance is $(3^{75})^{-1}$ The union of two bin files, given their orthogonality (if there is **1** at a given position in any binary, the same position in the other files can only be occupied by **0**) results in a sequence with approximately 50 '**1**' and 50 '**0**'. Hence, in this case the number of colliding sequences is 2^{50} and the probability of finding the genuine 100-base remains vanishingly low with a value of (9.10^{-16}). Supplying a third binary sequence makes this probability jump to 1 (1^{25}). In a view burrowed from cryptography a third sequence can be seen as acting as a public decryption key.

Interestingly, even if its probability becomes significant (for short sequences), finding the true nucleotide arrangement by chance is unlikely because of the absence of a criteria to decide if the choice of one particular sequence among several others is the right one. If the ciphertext 'Xc12VF' is decrypted as 'genome', speakers in numerous languages will immediately recognize this word as the true plaintext. Even if the ciphertext is only partially decrypted as 'g-no-e', an observer, familiar with the Latin alphabet of 26 letters, will make the reasonable choice, that among the 26^2 possibilities, most of them leading to meaningless words, 'genome' is likely to be the true plaintext. Such a criteria for filling blanks does not exist for a partially decrypted sequence such as 'AT-GC-' since, in most cases, choosing ATTGCT, for instance, does not make more sense than selecting ATCGCT or any other sequence occurring with the same probability. Even a brute force approach where all blanks are replaced by any of the 4 nucleotides will not lift the indetermination. Hence, protection will still be assured for short genomic sequences. (Figure 1).

Hence, since single bin(Base) files or the union of 2 of them can never reveal the true sequence, they provide a secure representation of nucleotide sequences.

Integrity

As mentioned above if data is stored as literal subsequences, corruption (replacement, deletion or addition of bases) remain undetected because comparison with a copy is not possible within the bank.

In contrast, the decomposition protocol, in many cases, gives an inner warning when corruption of some bin(Base) file has occurred. Modification in the length of a binary file (insertion or deletion) is simply detected from comparison with the length of other (uncorrupted) binary files.

There are 2 ways for detecting breaches of integrity both of them stemming from the properties of the binary files.

First, because of the orthogonality property, union of all binaries should give the complete file 111111111….When dealing with uncorrupted files, both Boolean term by term OR and XOR operations yield the same binary. If now a 0 is replaced by a 1 in a corrupted file, one of the binaries resulting from ORing must be different from that from XORing (one extra 0 appears in the XORed file, with respect to the ORed one). If a 0 replaces a 1, the overall union has one 0 (it should have none). (Figure 2)

Second, any genomic sequence can be represented as a unique rectangular binary array of 4 lines whose lengths are that of the genomic sequence. Because of the orthogonality of the bin(Base) files, each column in the array can be represented only as one of the decimal digits 1 (2^0), 2 (2^1), 4 (2^2), 8 (2^3). Thus, occurrence of any digit different from those in the decimal sequence indicates file corruption. In passing it is worth noting that the resulting unique decimal sequence (when uncorrupted) allows straightforward reconstruction of the nucleotide arrangement. (Figure 2 and Supplementary Information).

It must be stressed that failed integrity (even left undetected) has no consequence on confidentiality of the sequences.

ORGANISATION OF A GENOMIC DATA COMMUNITY BANK

The bank imposes the condition that investigations should be conducted by at least two independent research groups. Multi-party collaboration, as it implies mutual control, has proved to be efficient in concluding fair and honest deals. If parties accept to comply with this requirement they will be granted free access and if they do not they will incur a penalty and access will be denied.

Among many possible organization schemes, the bank could have two entries: chromosomes and individuals (CHR(Y)(10) means chromosome Y of person 10). The sequences (or parts of them) will be represented by their 4 bin(Base) files (or by the equivalent hexadecimal files for storage convenience). A blockchain organization of the bank is illustrated with the following example (Figure 3). Ian wants to analyze the sequences of chromosome(1) throughout all the community. Starting with Alice's Ian gets the approval of the administrator to buy Chr1(A) and Chr1(T) for 1 monetary unit (MU). The price for buying extra bin(Base) sequences has been put prohibitively high by the administrator so that investigators have to resort to other means to complete their knowledge. The transaction is then recorded in a ledger together with the time when it was performed, the balance of the bank at this time and more parameters if wished. The ledger is subsequently added to the chain which represents the history of all transactions. The high price getting more than 2 bin(Base) sequences imposes collaboration between investigators and it has the consequence of forbidding investigations to be conducted by a unique group as required by the bank regulation (see above). If Isabel has already acquired bin (G) and bin(T) from Alice along the same procedure, she will agree with Ian for a charge-free mutual exchange where Ian gets a copy of Chr1(G) and she gets a copy of Chr1(A). Hence both investigators have now a complete knowledge of the chromosome sequence. If Ian and Isabel inform the bank that they have in their hands 3 bin(Base) files (<u>and giving evidence for this</u> by handing copies of the 3 files back to the bank), the bank, in order to reward their good will and as a prove of work, will return 1 MU to each of them (transaction not shown on graph). The ledger is then updated, and the community acknowledges that the bank balance is now 0 and that two identified investigators possess the sequence of Alice's chromosome(1). Generalization of the procedure to multiple chromosomes, individuals, investigators gives a thorough account of all transactions in the data bank. Fine-tuned examination of the blockchains may help detecting spurious operations. For instance, the bank balance is expected to be generally close to zero since investigators are likely wanting to be quickly refunded. If Isabel buys 2 bin(Base) sequences with the prospect of selling one of them to Ian, at a price well below that of the bank, she cannot claim refunding and hence the balance of the bank will remain positive in the long run. Isabel's malicious operation will be readily traced back.

RESULTS

Data security

Security, that is confidentiality and integrity, of genomic data storage is satisfactorily assured using the bin(Base) representation of genomic sequences. This was illustrated above with the example of a 100-base long sequence. It was also mentioned that the risk of unveiling the true sequence by alignment with existing sequences in an open access library should not exist since, since, as agreed by the community members, sequences of their genomes should not be found outside the new bank. However, will security be questioned if data has also been deposited in an open bank as a consequence of erroneous or malevolent operations? Sequences from human chromosome 21 and

chromosome Y have been borrowed from a public bank (NCBI). Arbitrary long subsequences are decomposed into their Base-files subsequently checked for alignment with sequences in the bank. A single Base-file (as a reminder, the A-file from sequence ATGTA is ANNNA or A---A) has approximately 75% undetermined positions and that of the union of 2 Base-files has about 50%. As expected the sequence comparison algorithm fails to detect any significant alignment with the original chromosome in the bank. The high level of security offered by the Base-files or bin(Base) representation is confirmed even when sequences are accessible from outside the community library. The file resulting from the union of 3 Base-files successfully aligns with the genuine chromosome sequence .

The probabilities of finding the genuine sequence of the first 1000 nucleotides from HS21 by chance from bin(A) and bin(A + T) are $(3^{679})^{-1}$ and $(2^{346})^{-1}$ respectively.

Would a more in-depth inspection of a single or 2 bin(Base) files reveal some additional insights that might allow identification of the true genomic sequence?

It is expected, since they show the position within the sequence of a particular nucleotide, the bin(Base) files capture part of the biological information from the nucleotide arrangement. For instance, from bin(G) and bin(C) one can localize CpG islands supposedly more frequent in gene-rich regions of human chromosomes. According to the <u>second</u> Chargaff's rule, the number of A and T (and G and C<u>) along a single strand</u> are roughly equivalent, and thus this might help inferring the overall number of '**0**' or '**1**' in the complementary binary sequence. Long range correlations observed in many genomes (including human), with additional short range triplet correlation in bacterial genomes, indicate periodic patterns in genomic sequences. In this respect, the Fourier spectrum of the bin(Base) files reveals, in the low-frequency domain, a long-range correlation [10]and, in the case of bacterial genomes an additional triplet correlation arising from intronless, densely packed protein-coding genes. The triplet correlation, revealed by the 1/3 frequency in the Fourier spectrum, is the signature of bacterial genomes which can be recognized as contaminants in complex, unidentified samples. All these periodical properties are global and they do not grant access to information at the single nucleotide level and hence they are not likely to undermine bin(Base) decomposition as a safe tool to protect confidentiality.

<u>A community bank</u>

A community owned genomic bank, yet to come, based on storage of split information and on a blockchain-like organization of the bank traffic would offer a good trade-off between free data mining and protection of individual's genomic data. The implementation of blockchchain-based genomic banks is expected to largely benefit from this quickly developing technology that emerges as a powerful tool in many areas (like the bank industry) for monitoring transactions.[12]

Python snippets are available for Base-files and bin(Base) files generation, sequence reconstruction from binary arrays, Boolean XOR and OR operations, Fourier Transform of binary files and bank constitution. (Supplementary information)

REFERENCES

1-Green, E.D., Rubin, E.M. & Olson, M.V. *Nature* **550**, 179-181(2017)

2-Lauter,K., Boss, J.W.& Naehrig, M. *J Biomed Inform* **50** 234-243 (2014)

3-Gentry,C. *in Proceedings of the 41st ACM Symposium on Theory of Computing (ACM),* 169-178 (2009).

4 Gymrek, M.,McGuire, A.,Golan,D., Halperin, E.& Erlich, Y. *Science* **339**, 321-324 (2013).

5-Gkoulalas_Divanis, A., Loukides, G.& Sun, J. *J Biomed Inform* **50**,4-19 (2014).

6- Baldi,P.,in *Proceedings of the 18th ACM Conference on Computer and Communication Security*. 61-702 (2011).

7-Mailman, M.D. et al *Nature Genetics* ***39***, 1181-1186 (2007)

8- Jagadeesh,K.A. *et al. Science* **357**, 692-695 (2017).

9-*Nature Genetics* **44**, 35, 357 (2012).

10-Peng, *et al. Nature* **356**, 168–170 (1992).

11-Stephen F. *et al* Nucleic Acids Res. 25:3389-3402 (1997)

12-https://www.ibm.com/blockchain/platform/

FIGURES

Figure 1: Partial information; a) vertical splitting; b) horizontal splitting; the number of collisions, col, is = (length of the residual alphabet)$^{\text{number of } \mathbf{0}}$. c) sequence reconstruction from the binary rectangular array; the columns of the array represent decimal numbers

a) ATTCAAGCGTA→ ATTCA|AGC|GTA

b) ATTCAAGCGTA

bin(A):10001100001(col=3^7)

bin(G):00000010100(col=3^9)

Alice has bin(A)+ bin(G):10001110101(col=2^5) and gets bin(C) from Bob:10011111101(col=1^3); bin(T) is hence defined.

bin(T): 01100000010(col.=3^8)

bin(C):00010001000(col=3^9)

Bob has bin(T)+binc(C): 01110001010(col=2^6) and gets bin(G) from Alice:01110101110(col=1^4); bin(A) is hence defined

c)reconstruction.

bin(A)	10001100001		
bin(G)	00000010100		
bin(C)	00010001000		
bin(T)	01100000010		
decimal sequence:	81128842418	→	ATTCAAGCGTA

Figure 2: detection of failed integrity of binary files 1) illustrated with bin(A) and uncorrupted bin(G); Sigma is the union of all bin(Base) and is binary (11111111…..).;2) illustrated with the overall binary array

Truth tables of Boolean **OR** and **XOR**

A	B	A **OR** B
0	0	0
1	0	1
0	1	1
1	1	1

A	B	A **XOR** B
0	0	0
1	0	1
0	1	1
1	1	0

1-a) uncorrupted bin(A):

bin**(** bin(A) OR bin(N)**)**=bin**(** bin(A) XOR bin(N)**);** N: T,G,C

b) corrupted bin(A):

addition or deletion: length (bin(A))≠length (bin(N))

replacement of **0** by **1**: one bin(bin(A) OR bin(other base))≠bin(bin(A) XOR bin(other base))

replacement of **1** by **0**: Sigma has one **0**

2-a) Uncorrupted array**:**

A 10001100001

G 00000010100

C 00010001000

T 01100000010

Decimal: 81128842418

2-b Corrupted array:

A 10001100001

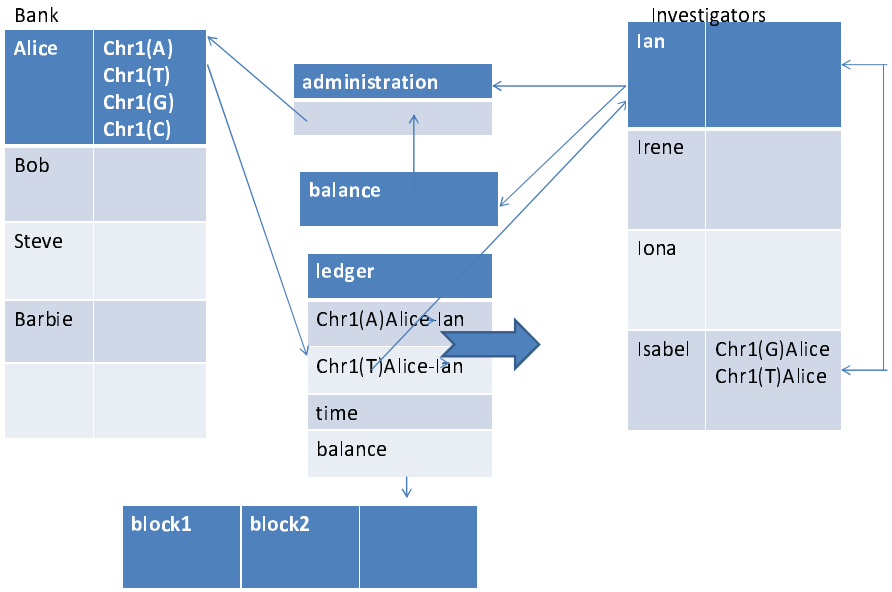
G 0**1**000010100

C 00010000000

T 01100000010

Decimal: 8**5**12884**0**418

Figure 3 : scheme of principle of the blockchain bank. Example of the management of one investigator's query on one person's chromosome.(see text)


Bank
Alice
Chr1(A)
Chr1(T)
Chr1(G)
Chr1(C)
Bob
Steve
Barbie
administration
balance
ledger
Chr1(A)Alice-Ian
Chr1(T)Alice-Ian
time
balance
block1
block2
Investigators
Ian
Irene
Iona
Isabel
Chr1(G)Alice
Chr1(T)Alice