Life Histories of Myeloproliferative Neoplasms Inferred from Phylogenies

Gal Cesana
gal.cesana@mail.huji.ac.il

Noa Margulis
noa.margulis@mail.huji.ac.il

Eitan Samson
eitan.samson@mail.huji.ac.il

Yoel Marcu
yoel.marcu@mail.huji.ac.il

Adi Yefroimsky
adi.yefroimsky@mail.huji.ac.il

March 19, 2025

Introduction

Driver mutations are genetic alterations that confer a selective growth advantage to cancer cells, promoting tumor growth and invasiveness. These mutations can be gain-of-function in proto-oncogenes or loss-of-function in tumor suppressor genes. In contrast, passenger mutations do not contribute to cancer progression and are often found alongside driver mutations due to the clonal expansion of cells containing the driver mutation[1]. Little is known about the ages at which driver mutations occur, the timelines of clonal expansion over an individual's lifetime, or how these relate to clinical presentation with cancer[2].

MPNs (myeloproliferative neoplasms) are a group of clonal hematologic malignancies, morphologically characterized by the expansion of terminally differentiated myeloid cells (white blood cells, erythrocytes, and platelets). There are multiple different types of MPN, but all share similar pathobiological and clinical features. MPNs have a complex and incompletely understood pathogenesis that includes systemic inflammation, clonal hematopoiesis, and constitutive activation of the JAK-STAT pathway[3]. In patients with blood cancers, and specifically MPN patients, the observation of normal blood counts months to years before diagnosis suggested that tumor development occurs quickly. Therefore, driver mutations must occur later in life, closer to diagnosis. However, the presence of driver mutations in normal tissues — including blood from healthy individuals with clonal hematopoiesis, some of whom subsequently develop malignancies — supports a longer, multi-process view of cancer. Some of these mutational processes accumulate at a steady rate across life, representing a 'molecular clock'. Finding the tissue-specific rate of mutation accumulation might enable broad estimates for the timing of mutations and estimate disease progression[2].

Phylogenetic trees can be a powerful tool for studying driver mutations in cancer by explaining tumor evolution and the order of mutation acquisition. By sequencing multiple regions of a tumor or single cells, a phylogenetic tree could be constructed, showing the evolutionary relationships between different tumor subclones. The leaves represent different subpopulations, and the inner nodes represent key mutational events. The resulting tree topology and inner states could point at possible driver mutations and — under the 'molecular clock' assumption — when did they occur relatively to the patient's lifetime[4].
Recent works used this approach to analyze the phylogeny of clones taken from MPN patients. The researchers conducted a whole genome sequencing of over 1,000 single-cell derived colonies from 12 MPN patients, and processed it into SNV data. From the SNV profile of each clone, phylogenetic trees were built to trace each clone's ancestry. Their study revealed surprisingly early origins for driver mutations. For example, the $JAK2^{V617F}$ mutation was acquired in some patients during gestation or childhood. The authors hypothesized that early driver mutation acquisition combined with life-long clonal growth underpins adult MPN, raising the possibility of very early detection and intervention.

Our hackathon project aimed to replicate and understand existing research on MPN patients presented by Williams et al.[2]. The project focused on reconstructing phylogenies from genomic data published as part of the original paper, and attempting to partially recreate its results. This allowed us to investigate the identity and timing of cancer-driving mutations acquired in MPN patients. These questions tested the hypothesis that MPN-driving mutations originate early in life.

Methods

The project's goal was recreating the phylogenetic trees constructed in the paper and estimating driver mutation timing, based on the SNV data the researchers gathered. We followed these steps:

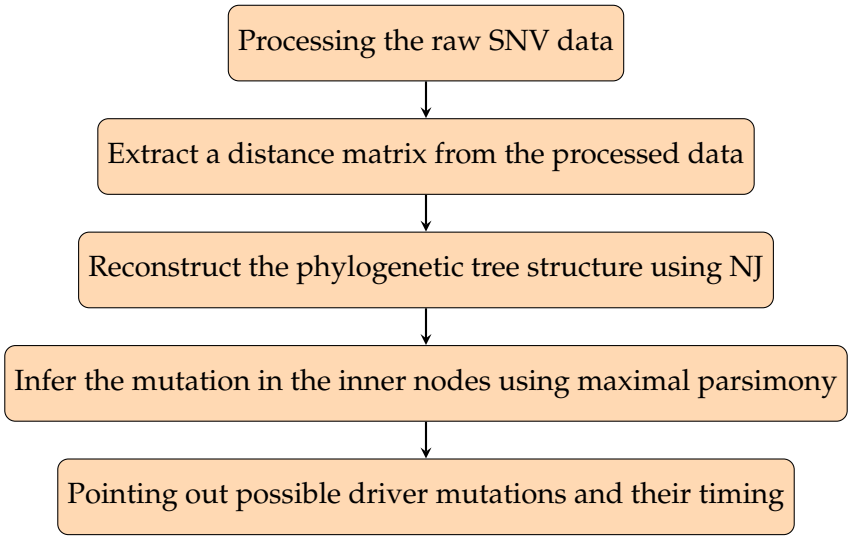


Figure 1: The suggested phylogenetic tree reconstruction workflow.

Data Processing — VCF files processing and filtration

First, we had to process the data published in the article. Initially, the researchers conducted a whole genome sequencing of 1088 in vitro expanded single-cell-derived haematopoietic colonies. Colonies were sampled from 12 patients of ages 20-81 with different types of MPN, and were sequenced to approximately 16.7× mean depth across 17 time points. After filtering for low sequencing coverage and cross-colony contamination, 1,013 colonies were used in subsequent analyses. Among them, 560,978 single nucleotide variants (SNV) and 19,155 small insertions and deletions were identified. This SNV data was recorded in a VCF format, which is a common file format for storing DNA sequence variations. The file's header contains metadata describing custom fields of each SNV entry[5]. In this case, these fields were:

- Mutation type description — SNV / Indel.
- The gene that the variation occurred in.
- A binary field indicating if the encoded protein has changed as a result of the variation.
- A binary field indicating whether the variation is included in the calculation. Variations in copy-number aberrations, loss of heterozygosity or sex chromosomes were excluded from the analysis.
- A field indicating whether the variation was confidently found in the clone. This field was calculated in the original paper by comparing the number of times the variant appeared compared to the reference.

Additionally to the custom information fields, the default VCF fields describe the chromosome and position the variant was found on, the variation ID, the reference and alternative bases, and the quality score given to it (and whether it passed the required quality).
The raw VCF file was processed using SAMtools' 'BCFtools' module. Entries were filtered according to the quality filter and the custom exclusion made by the researchers. Each clone could now be represented as a binary vector, indicating the presence of each variation in the clone.

Tree Topology Reconstruction

After obtaining the presence matrix, we continued with estimating the phylogenetic tree's topology. In this case, the tree needs to be reconstructed from the bottom-up approach.
The initial distance matrix between each clone was calculated using hamming distances between each binary vector representation. This metric represents the number of changed SNVs between each clone pair. There are multiple methods for tree structure estimation. Here we used the Neighbor Joining approach. It assumes a lighter 'additivity' criterion, which could fit the data in a more precisely matter.

Inner Tree States prediction — Maximum Parsimony Method

The maximum parsimony method of phylogenetic tree reconstruction is a method based on the assumption that a tree is more likely to be correct if it includes less mutation events during its levels.
This assumption is based on the relative rarity of mutations in nature and on the concept of Occam's razor, which states that an explanation needs to be as simple as possible. The maximum parsimony problem for phylogenetic trees has two versions. The first version (called 'large parsimony') includes reconstructing the entire tree topology given the leaves of a tree, with minimal number of mutation events. However, it was proven that this is a NP-hard problem[6].

Therefore, we first used the Neighbor-Joining algorithm for the tree reconstruction, as mentioned above, and made use of an algorithm that solves the second version of the problem. The second version of the maximal parsimony problem, called small parsimony, gets the leaf sequences as well as the tree topology as its input, and aims to assess the sequences of the ancestral inner nodes in the tree, under the same aspiration to minimize the number of mutations. Here, we used Fitch's algorithm[7] (see Algorithm 1) for parsimonious assignment of sequences to nodes of a phylogenetic tree. This algorithm assumes that the mutations are independent on previous mutations anywhere in the sequence. The algorithm includes two phases. In the first phase, a possibility set is constructed from each node in postorder tree walk. It contains the intersection between the possibility sets of the children nodes if this intersection is not empty, and their union otherwise.

Algorithm 1 Fitch's Algorithm for the Small Parsimony Problem

Require: A rooted binary tree T with leaf nodes labeled by some letters from sequence alphabet.
Ensure: The minimum number of substitutions required for the most parsimonious tree.

Phase 1: Bottom-up pass (Set Computation)
for all leaf nodes v in T **do**
  $S(v) \leftarrow$ observed state at leaf v
end for
for all internal nodes v, in postorder (from leaves to root) **do**
  Let u, w be the children of v
  if $S(u) \cap S(w) \neq \varnothing$ **then**
    $S(v) \leftarrow S(u) \cap S(w)$
  else
    $S(v) \leftarrow S(u) \cup S(w)$
    Increment parsimony score by 1
  end if
end for

After the first phase the final assignments are decided by inorder walk, where for each node, if the final decision for its parent is in its possibility set, it is decided to be the same, and else an arbitrary choice from the possibility set is made.

Phase 2: Top-down pass (State Assignment)
Assign root node r any state from $S(r)$
for all internal nodes v, in preorder (from root to leaves) **do**
  Let u, w be the children of v
  for all child x of v **do**
    if $S(x) \cap S(v) \neq \varnothing$ **then**
      Assign any state from $S(x) \cap S(v)$ to x
    else
      Assign any state from $S(x)$ to x
    end if
  end for
end for

In our case, the sequences of the nodes are not continuous DNA sequences. Instead, they are binary lists, where each index corresponds to a specific variation. A value of 1 indicates that the phenotype at that position exhibits the mutational variation, while a value of 0 signifies the absence of a mutation at that locus. The loci are not necessarily adjacent and may be located far apart in the genome, a fact that makes the independence assumption in the algorithm reasonable.

The runtime of the Fitch's algorithm is $\mathcal{O}(V)$ where V in the number of vertices. The memory complexity is $\mathcal{O}(V)$ as well, since the temporary sets for the nodes in the first phase need to be saved.

During the hackathon, we also tried a slightly different algorithm, which utilizes the fact that the sequences in our case are binary. We call it 'Bitwise-AND Assignment'(see Algorithm 2). In this method, we calculate the assignment for an inner node using the 'AND' operator between its children. The logic behind it is that for a single locus of the parent where the children differ at that locus, the parent had one of the genotypes of its children. Since we have no additional information, we opt for the scenario in which the parent had the nonmutant variation (0 for the null hypothesis), and the mutation occurred in one of its children over the scenario of reversed mutation in one of the children. This method is local; that is, it does not take into account information from further nodes. As a result, it may yield less parsimonious results.

Algorithm 2 Bitwise-AND Assignment

Require: A rooted binary tree T with leaf nodes labeled by binary vectors of length n.
for all internal nodes v, in postorder (from leaves to root) **do**
  Let u, w be the children of v with states $S(u), S(w)$
  $S(v) \leftarrow AND(S(u), S(w))$
end for

However, the simplified algorithm performs similarly to Fitch's algorithm in the case of binary alphabet, and it takes only one operation for each inner node to compute, and no additional memory.

Pointing Out Possible Driver Mutations

After constructing the tree and inferring the variations represented by each inner node, we attempt to point out a possible driver mutation by observing the tree topology.
Driver mutations are characterized by long branches that lead to large subclades. A driver mutation gives a growth or survival advantage to the cell that acquires it. This means that the subclone carrying the driver mutation will expand faster than other subclones that do not have it. As a result, descendants of this cell will outcompete other cells, leading to a large subclade in the phylogenetic tree. Additionally, acceleration in evolutionary rate creates longer branches in the tree as more mutations accumulate along that lineage in a given time period.

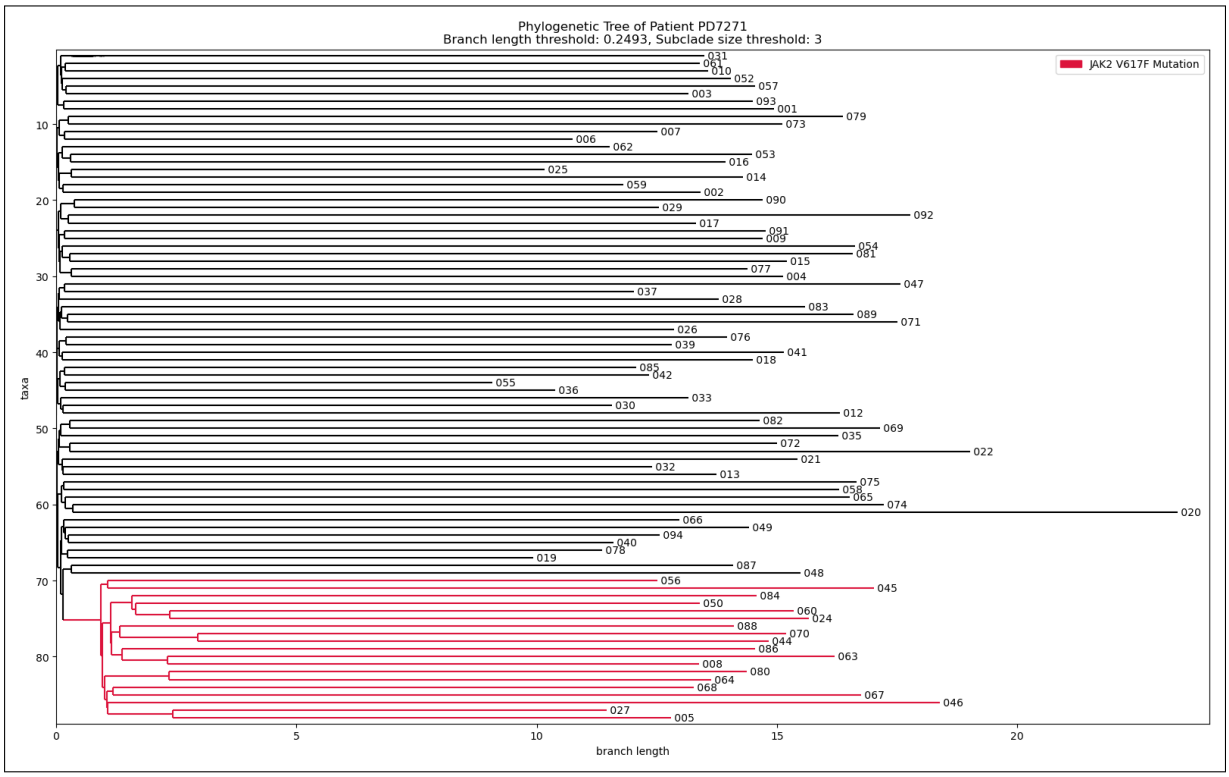
We tried to identify possible driver mutation sites by looking for branches with a large subclade and length, relative to the current tree. First, distributions of all branch lengths in the tree (excluding terminals and the root) and the number of terminal nodes within each subclade are collected. These distributions are used to set dynamic thresholds based on percentiles. After that, we traverse each nonroot and nonterminal clade and sort them by their branch length. Branch that exceeds both percentile thresholds (length and subclade size) is considered to be a possible driver mutation site.

Determining the Timing of Driver Mutations

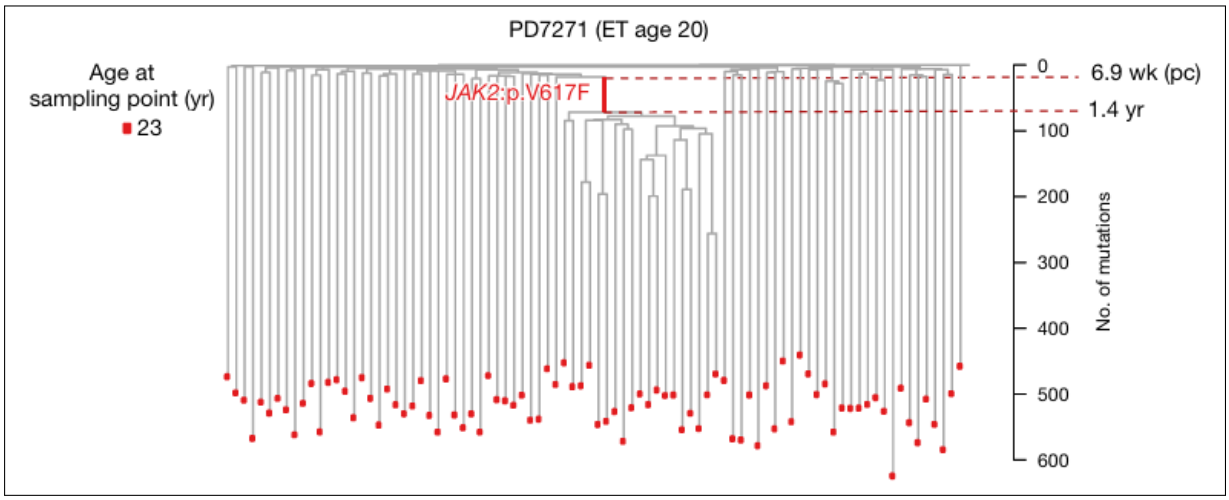
One of the goals of our project was to investigate the identity and timing of cancer-driving mutations acquired in MPN patients. However, up until now, the trees' branches didn't necessarily represent the time that passed between each event, but rather a metric of evolutionary distance between variants.
One importnat assumption we made during the project is the linear rate of mutation acquisition. This means that the number of acquired mutations is relative to the time that have passed up until that point in time. Therefore, it allows us to translate the constructed trees to the time domain, while still keeping their structure. The branches were normalized using min-max normalization such that the furthest leaf's distance from the root (0 years), will be the age of the patient at sampling.

Results

The procedure's results for patients PD5163 and PD7271 are presented below. The results present the constructed trees' topology, as well as the suspected driver mutation branches. Below our results, we show the results published by Williams et al. for both patients. The original paper uses the number of mutations as their branch length scale. In our trees, branch lengths represent the age of the patient during the mutational event, which we assume is proportional to the number of mutations in the branch. In both patients, the top branch forming the highlighted subtree is suspected as a driver mutation event.

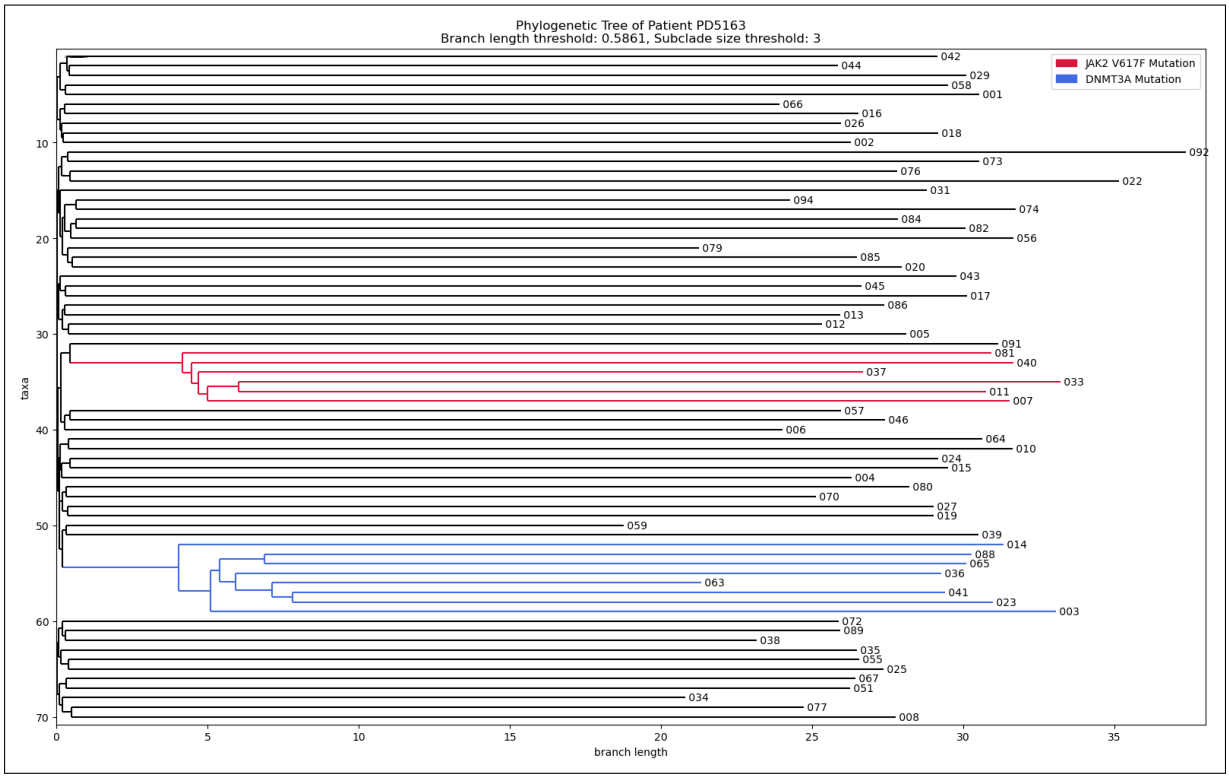


(a) Tree constructed for patient PD7271. Branch forming the highlighted subtree was suspected to contain a driver mutation, and indeed represents a $JAK2^{V617F}$ mutation.

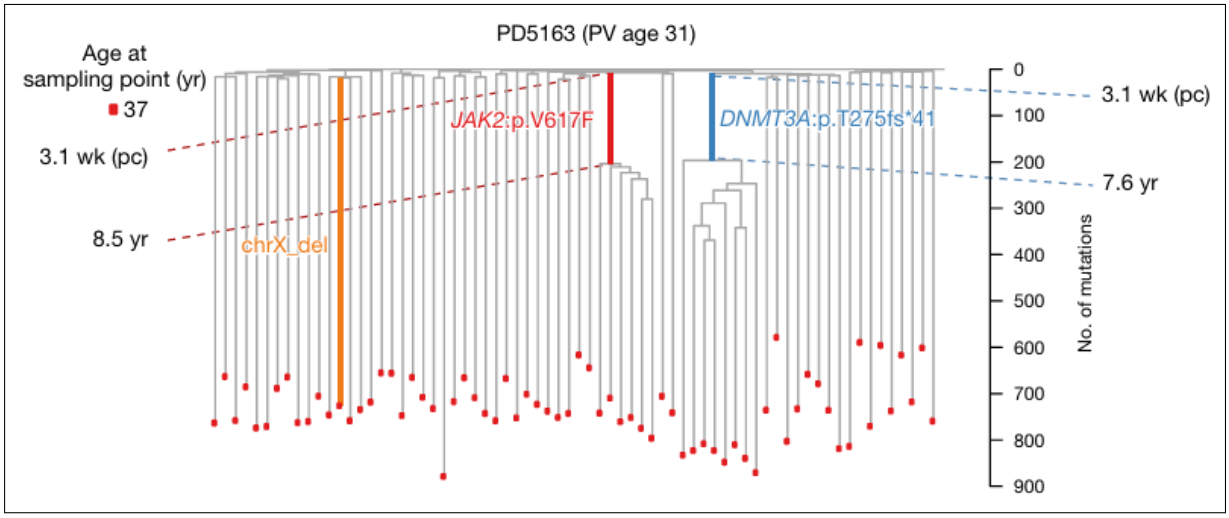


(b) Tree constructed by Williams et al. for patient PD7271.

Figure 2: Comparison between our results and the paper's results on patient PD7271.



(a) Tree constructed for patient PD5163. Branches forming the subtrees highlighted in red and blue were suspected to contain a driver mutation, and indeed represent a $JAK2^{V617F}$ mutation and a DNMT3A mutation respectively.



(b) Tree constructed by Williams et al. for patient PD5163.

Figure 3: Comparison between our results and the paper's results on patient PD5163.

Discussion

Our hackathon project used phylogenetic analysis of clonal sequence variations to identify driver mutations. We processed the raw data into an SNV presence matrix, creating a binary vector indicating each clone's variations. We then calculated pairwise Hamming distances, generating a distance matrix. Applying Neighbor-Joining to the distance matrix produced an estimated tree topology. We then tried to estimate variation presence of inner nodes using maximum parsimony reconstruction algorithms. Assuming a linear rate molecular clock model, we could align the tree with each patient's lifetime. Finally, we identified longer branches leading to large sub-clades as potential driver mutations, determining their identity by comparing the connected nodes.

In this project, we attempted to replicate part of the work by Williams et al. on clonal sequencing of MPN patients, using the data provided by the authors. Overall, the data contained clonal sequence variations from 12 patients. Each patient's clones could be used for the construction of their own phylogenetic tree. Some patients' sequencing results had shown more complicated cases of chromosomal aberrations and deletions in some subpopulations, which had an effect on tumor progression. In order to accommodate these complex cases, the authors of the original paper adopted a more robust and complex model to estimate the timing of events. In this project, due to the restricted time we had, we decided to focus on simpler examples. That allowed us to consider only the presence of the variations themselves in our analysis, and to not integrate the greater biological context of complicated examples into our model.

The phylogenetic trees we generated from clones taken from 'simpler' patients were similar to those presented by Williams et al. After obtaining the primary tree topology using Neighbor Joining, it could be seen that the trees' general structures closely resemble those presented in the paper. The Maximum Parsimony analysis and driver event identification showed key driver mutations in the same locations as shown in the paper. The main example being the location of $JAK2^{V617F}$ mutation, a well documented driver mutation which is very typical of MPN patients. Similarly to the original paper, $JAK2^{V617F}$ has been found to form long branches that lead to large sub-clades. Another driver mutation DNMT3A was also correctly identified and placed. The assumption of linear mutation rate relatively to the patients' lifetime provided driver mutation timings similar to those reported in the paper.
The results gathered in our project suggest that founding driver mutations might occur early in life, with the first bifurcation in many trees marking the appearance of a driver mutation. The long branch from the root to the driver mutation node implies years, and even decades, of silent clonal evolution before clinical disease onset.

Our model closely resembled the results of the original paper, in the scope of the 'simpler' patient cases. However, when trying to run our procedure on other examples, the resulting phylogenetic trees are inaccurate. This is expected, since our model assumes a relatively sim-

ple setting of linear mutation accumulation rate, and does not specially consider more complicated chromosomal events. In order to make a more robust model, the authors suggest more advanced techniques to estimate branch lengths, as well as incorporation of chromosomal aberration and telomere timing analysis. Future work could use such tools for more accurate tree estimates. Other tree annotation methods, such as maximum likelihood or other Bayesian approaches could be used instead of maximum parsimony, and other forms of information, except for the pure variations, could be integrated.

Code Availability

All the python scripts that were used during the project can be found on the project's GitHub repository in the link: `https://github.com/galcesana/CBIO_Hackathon`.

Division Of Labor

- Gal Cesana — Graph algorithm planning and application. Validation of results.
- Noa Margulis — Presentation making, summary, data organisation.
- Yoel Marcu — Data processing, topology reconstruction, tree parsimony.
- Eitan Samson — Data processing, topology reconstruction, tree parsimony.
- Adi Yefroimsky — Data processing, tree parsimony, driver mutation location extraction.

References

[1] Morjaria, S. (2020). Driver mutations in oncogenesis. International Journal of Molecular and Immuno Oncology, 6, 100–102. https://doi.org/10.25259/ijmio_26_2020

[2] Williams, N., Lee, J., Mitchell, E., Moore, L., Baxter, E. J., Hewinson, J., Dawson, K. J., Menzies, A., Godfrey, A. L., Green, A. R., Campbell, P. J., & Nangalia, J. (2022). Life histories of myeloproliferative neoplasms inferred from phylogenies. Nature, 602(7895), 162–168. https://doi.org/10.1038/s41586-021-04312-6

[3] Tremblay, D., Yacoub, A., & Hoffman, R. (2021). Overview of myeloproliferative neoplasms. Hematology/Oncology Clinics of North America, 35(2), 159–176. https://doi.org/10.1016/j.hoc.2020.12.001

[4] Coorens, T. H. H., Chapman, M. S., Williams, N., Martincorena, I., Stratton, M. R., Nangalia, J., & Campbell, P. J. (2024). Reconstructing phylogenetic trees from genome-wide somatic mutations in clonal samples. Nature Protocols, 19(6), 1866–1886. https://doi.org/10.1038/s41596-024-00962-8

[5] The Variant Call Format (VCF) Version 4.2 Specification. (2024). The variant call Format (VCF) Version 4.2 specification. In https://github.com/samtools/hts-specs. https://samtools.github.io/hts-specs/VCFv4.2.pdf

[6] R. L. Graham and L. R. Foulds. Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computational time. Math. Biosci., 60:133–142, 1982

[7] Walter M. Fitch, Toward Defining the Course of Evolution: Minimum Change for a Specific Tree Topology, Systematic Biology, Volume 20, Issue 4, December 1971, Pages 406–416, https://doi.org/10.1093/sysbio/20.4.406