

Comparing algorithms that search for cancer mutations

July 17 2017



Adam Godzik, Ph.D., is director of the Bioinformatics and Structural Biology Program at Sanford Burnham Prebys Medical Discovery Institute. Credit: Bil Zelman

Sanford Burnham Prebys Medical Discovery Institute (SBP) has

undertaken the first-ever comparative analysis of a newly emerging category of algorithms that mine genetic information in cancer databases by focusing on internal gene structure (subgene resolution algorithms), in contrast to classical approaches that focus on genes treated as single units. These powerful data-sifting tools are helping untangle the complexity of cancer, and find previously unidentified mutations that are important in creating cancer cells.

The study, published today in *Nature Methods*, reviews, classifies and describes the strengths and weaknesses of more than 20 algorithms developed by independent research groups.

"Despite the increasing availability of high-resolution genome sequences, a common assumption is to consider a gene as a single unit," explains Adam Godzik, Ph.D., director of the Bioinformatics and Structural Biology Program at SBP, and senior author of the study.

"However, there are a number of events, such as single site DNA substitutions and splicing variants that can occur within a gene—at the subgene level. Subgene algorithms provide a high-resolution view that can explain why different [mutations](#) in the same gene can lead to distinct phenotypes, depending on how they impact specific protein regions.

"A good example of how different subgene mutations influence cancer is the NOTCH1 gene," says Godzik. "Mutations in certain regions of NOTCH1 cause it to act as a tumor suppressor in lung, skin and head and neck cancers. But mutations in a different region can promote chronic lymphocytic leukemia and T cell acute lymphoblastic leukemia. So it's incorrect to assume that mutations in a gene will have the same consequences regardless of their location."

The study researchers applied each subgene [algorithm](#) to the data from The Cancer Genome Atlas (TCGA), a large-scale data set that includes genome data from 33 different tumor types from more than 11,000

patients.

"Our goal was not to determine which algorithm works better than another, because that would depend on the question being asked," says Eduard Porta-Pardo, Ph.D., a former postdoc in Godzik's lab and first author of the paper who is now a staff scientist in the Barcelona Supercomputer Center. "Instead, we want to inform potential users about how the different assumptions behind each subgene algorithm influence the results, and how the results differ from methods that work at the whole gene level."

"We found two important things," says Porta-Pardo. "First, we found that the algorithms are able to reproduce the list of known cancer genes established by cancer researchers—validating the subgene approach and the link between these genes and cancer. Second, we found a number of new cancer driver genes—genes implicated in the process of oncogenesis—that were missed by whole-gene approaches."

"Finding new cancer driver [genes](#) is an important goal of cancer genome analysis," adds Porta-Pardo. "This study should help researchers understand the advantages and drawbacks of subgene algorithms used to find new potential drug targets for [cancer](#) treatment."

More information: Eduard Porta-Pardo et al, Comparison of algorithms for the detection of cancer drivers at subgene resolution, *Nature Methods* (2017). [DOI: 10.1038/nmeth.4364](https://doi.org/10.1038/nmeth.4364)

Provided by Sanford-Burnham Prebys Medical Discovery Institute

Citation: Comparing algorithms that search for cancer mutations (2017, July 17) retrieved 19 April 2024 from <https://medicalxpress.com/news/2017-07-algorithms-cancer-mutations.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.