
GAZİ ÜNİVERSİTESİ
Gazi
1926

AN EFFICIENT APPROACH FOR DETECTING MUTATED SUBNETWORKS IN BIOLOGICAL NETWORKS

Lionel NGOBESING ALANGEH

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN COMPUTER ENGINEERING DEPARTMENT

GAZİ UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

JANUARY 2024

ETHICAL STATEMENT

I at this moment declare that in this thesis study, I prepared the thesis writing rules of Gazi University Graduate School of Natural and Applied Sciences;

- All data, information, and documents presented in this thesis have been obtained within the scope of academic rules and ethical conduct,
- All information, documents, assessments, and results have been presented by scientific ethical conduct and moral rules,
- All material used in this thesis that is not original to this work has been exhaustively cited and referenced,
- No change has been made in the data used,
- The work presented in this thesis is original,

or else, I admit all loss of rights to be incurred against me.

Lionel NGOBESING ALANGEH

09/01/2024

AN EFFICIENT APPROACH FOR DETECTING MUTATED SUBNETWORKS IN BIOLOGICAL NETWORKS

(M. Sc. Thesis)

Lionel NGOBESING ALANGEH

GAZI UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

January 2024

ABSTRACT

The search for a cure for cancer is one of the most challenging problems faced by medical experts, computational biologists, and experts in bioinformatics today. With the vast amounts of genomic data generated daily, evaluating and investigating this data have become increasingly difficult. The heterogeneity of cancer data, consisting of passenger genes that do not contribute to oncogenesis and driver genes that directly lead to it, makes identifying these driver genes from passenger genes in large datasets a daunting task. In this research, we propose two methods to solve this problem: LBSA-Driver, a network-based method that uses a novel List-based simulated annealing implementation, and ARO, a bio-inspired method that integrates a mutation phase. ARO merges the survival behavior of rabbits through exploration and exploitation with a gene interaction network to improve the accuracy of discovering cancer-driver genes. In this study, both methods were applied to four types of cancers: breast, brain, prostate, and ovarian. The results showed that our proposed models can identify well-known labeled canonical driver genes while prioritizing them over unknown cancer driver genes.

Science Code	:	92410
Key Words:	:	Cancer, driver genes, gene ontology, List-based Simulated Annealing, Artificial Rabbits Optimization, meta-heuristic algorithms, biological-interaction networks
Page Number	:	87
Supervisor	:	Asst. Prof. Dr. Yılmaz ATAY

BİYOLOJİK AĞLARDA MUTASYONLU ALT AĞLARIN TESPİTİ İÇİN ÖNERİLEN ETKİN YAKLAŞIM

(Yüksek Lisans Tezi)

Lionel NGOBESING ALANGEH

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Ocak 2024

ÖZET

Sağlık görevlilerinin, hesaplamalı biyologların ve biyoinformatik uzmanlarının kansere çare bulma mücadelesi, bugün dünyanın en zor sorunlarından biridir. Günlük olarak üretilen büyük miktarda genomik veri göz önüne alındığında, bu verilerin değerlendirilmesi ve araştırılması giderek zorlaşmaktadır. Bunun nedeni, kanser verilerinin heterojen olması, onkogeneze doğrudan yol açan sürücü genlerin yanı sıra onkogeneze katkıda bulunmayan yolcu genlerden oluşmasıdır. Bu nedenle, bu büyük veri yığınlarındaki yolcu genlerinden bu sürücü genlerini belirlemek giderek daha zor bir görev haline geliyor. Bu sorunu çözmek için geliştirilmiş önceki yöntemleri göz önünde bulundurarak, bu araştırmada hem LBSA-Sürücü adı verilen ağ tabanlı bir yöntem hem de kullanılacak bir mutasyon aşamasını entegre eden Yapay Tavşan Optimizasyonu (ARO) adlı biyo-ilhamlı bir yöntem öneriyoruz. kanser sürücü genlerini belirleme sorununu çözer. İlk yöntem, yeni bir Liste tabanlı simüle edilmiş tavlama uygulamasını kullanırken, ikinci yöntem, kanser sürücü genlerini keşfetmenin doğruluğunu artırmak için bir gen etkileşim ağı ile sırasıyla hem küresel hem de yerel aramayı ele almak için keşif ve sömürü yoluyla tavşanların hayatta kalma davranışını birleştirir. Modeller meme kanseri, beyin kanseri, prostat kanseri ve yumurtalık kanseri olmak üzere 4 farklı kanser türüne uygulanmaktadır. Sonuçlar, önerilen modelin, iyi bilinen etiketli kanonik sürücü genlerini tanımlayabildiğini ve bunları bilinmeyen kanser sürücü genlerine göre önceliklendirdiğini göstermektedir.

1. Giriş

Kanser araştırmaları, genomik verileri kullanarak kanser sürücü genleri tanımlamak için sürekli bir evrim geçirmiştir. Frekans temelli ve ağ temelli yöntemler olmak üzere iki temel yaklaşım, kanser başlangıcından ve ilerlemesinden sorumlu mutasyonları ortaya çıkarmak için kullanılmıştır. MutsigCV ve ContrastRank gibi frekans temelli yöntemler, mutasyonların hasta grupları arasındaki yaygınlığına odaklanırken, HotNet gibi ağ temelli yöntemler biyolojik ağları keşfetmek için kullanılır ve sürücü genleri tanımlamayı amaçlayan bir iletim sürecini uygular. Ancak, her iki metodoloji de sınırlamalara sahiptir ve daha nuanslı yaklaşımlara olan ihtiyacı ortaya koymaktadır. Kanser sürücü mutasyonlar, tümör başlangıcında ve ilerlemesinde temel bir rol oynar, bu da onları yolcu mutasyonlardan ayırır. Sürücü mutasyonlar kanser gelişimine doğrudan katkı sağlar, yolcu mutasyonlar ise tümörün büyümesini sürdürür ve kolaylaştırır. Zorluk, bu mutasyonları geniş bir genom veri havuzunda ayırt etmektedir. Var olan frekans temelli yöntemler, özellikle daha az yaygın ancak kanser ilerlemesi için eşit derecede önemli olan mutasyonları ele alırken zorlanmaktadır.

Kanserle ilgili genlerin büyük bir miktarı, yolcu ve sürücü genler olarak iki ana gruba ayrılmıştır. Yolcu mutasyonlar yaygındır, sürücü mutasyonlar ise tümör başlangıcında temel öneme sahiptir. Sürücü mutasyonlarını belirlemek, yolcu mutasyonlarının ezici yaygınlığı nedeniyle zorlayıcıdır. Var olan yöntemler, genellikle frekans temellidir ve daha az yaygın ancak önemli sürücü mutasyonları ile başa çıkmakta zorlanır, bu da daha gelişmiş optimizasyon tekniklerine olan ihtiyacı ortaya çıkarır. Bu araştırmanın temel hedefi çift katlı bir yaklaşımı benimsemektir. İlk olarak, biyolojik etkileşim ağını kullanarak kanser sürücü genlerini doğru bir şekilde tanımlamak için güçlü bir optimizasyon tekniği geliştirmektir. Bu yaklaşım, sadece mutasyonların frekansını değil, aynı zamanda biyolojik ağ içindeki ilişkilerini ve etkileşimlerini de dikkate alır. İkinci olarak, araştırma, doğruluk ve verimlilik açısından mevcut en iyi yöntemlere üstünlük sağlamayı hedefler. Potansiyel sürücü genleri sıralayarak ve sonuçları hassasiyet, hatırlama, F-puanı ve doğruluk hesaplamalarıyla doğrulayarak, önerilen modelin üstünlüğünü göstermeyi amaçlar.

Bu araştırmanın önemi, erken kanser tespiti ve önleme konusundaki potansiyeline dayanmaktadır. Ulusal Kanser Enstitüsü'nün vurguladığı gibi, erken teşhis, kanser yönetimi için hayati öneme sahiptir. Erken aşamalardaki sürücü mutasyonları tanımlamak ve önlemek, hastalık önlemede önemli bir katkı sağlayabilir. Önerilen yöntem, sadece sürücü mutasyonları tanımlamak için yeni bir yaklaşım sunmakla kalmaz, aynı zamanda araştırmacılara bilgi sağlamak için bir bilgi platformu olarak da hizmet eder, bu da kanser araştırmasının geniş anlayışına katkıda bulunur. Özetle, bu araştırma, geniş bir genom veri havuzunda kanser sürücü genlerini doğru bir şekilde tanımlamanın zorlu bir sorununu ele almaktadır. Biyolojik etkileşim ağını kullanarak geliştirilen optimizasyon tekniği, mevcut metodolojilerin sınırlamalarını aşmak için umut verici bir çözüm sunmaktadır. Bu araştırmanın potansiyel etkisi, sadece bilişsel biyoloji alanını değil, aynı zamanda erken kanser teşhisi ve önleme konusundaki geniş anlayışı artırarak kanser araştırma ve tedavi çabalarına değerli bir katkı sağlamaktadır.

2. Literatür tarama/ilgili çalışmalar

Gen ifade değişimindeki ilgili değişikliklerle genetik olaylar arasındaki ilişkiyi bilimsel olarak kullanarak kanserde mutasyona uğramış sürücü genleri bulma yöntemlerinde bir eksiklik yaşanmıştır. Mevcut yaklaşımlar ve algoritmaların çoğu, belirli bir gendeki mutasyonların sürücü geni belirleme kriteri olarak mutasyonların sıklığına tamamen bağımlıdır. Ancak, sıklık analizi, düşük sıklıklı ancak yine de çok önemli olan sürücü genleri belirlemede başarılı olmadığı için büyük bir eleştiriyle karşı karşıyadır (Bashashati vd., 2012). 2012 yılında yaptıkları araştırmada, somatik sürücü genlerin kanserdeki transkripsiyon ağlarındaki etkisini ortaya çıkarmak amacıyla DriverNet adlı bir model geliştirmiştir. Bu nedenle, bu araştırma, genetik sapmaların (transkripsiyonel) biyolojik ağ üzerindeki etkilerini ve potansiyel kanser sürücü genlerini belirlemek için modifiye bir biyo-ilham algoritması olan Yapay Tavşanlar Optimizasyonu (ARO) algoritması ile birleştirilmiş bir çerçeve olan LBSA-DRIVER'ı sunmaktadır. Bu araştırma, aşağıda bahsettiğimiz diğer araştırmacılar tarafından yapılan bazı çalışmalara benzer zeminler bulmaktadır.

DriverNet, 2012 yılında kanser transkripsiyon ağlarında bulunan sürücü mutasyonları belirlemek için bir hesaplama çerçevesi kullanma amacıyla geliştirilen bir yaklaşımdı (Bashashati vd., 2012). Bu çalışma, bir bipartit grafik ve bir biyolojik etkileşim ağı kullanmıştır. Bu araştırmadaki bipartit grafik, bir katmanındaki mutasyonların gen mutasyonlarını, diğer katmandaki mutasyonların ise aykırı mutasyonları temsil ettiği şekilde geliştirilmiştir. Bu araştırmadaki önerilen algoritma, minimum küme örtme problemi kullanarak gerçekleştirilen açgözlü bir algoritma idi. Yani, hasta-gen olayları, ana evrenin elemanları olarak kabul edildi ve her mutasyon, mutasyonla ilişkilendirilmiş hasta-gen olaylarını içeren bir küme içeren bir seti temsil etti. Bu araştırmadaki orijinal veri setleri 500 kez permüte edildi ve ardından algoritma rastgele veri setlerinde uygulandı ve algoritmanın anlamlılığını test etmek için sonuçlar elde edildi. Bu nedenle, bu araştırmada önerilen DriverNet algoritması, Cancer Gene Census (CGC) (Futreal vd., 2004) tarafından tanımlanan birkaç geni, MDM2, BCR, ERBB2, DDIT3, FGFR1, BRCA2, MET ve PDGFRA dahil, keşfetmiştir.

Malebary vd., büyük veri setlerinde sürücüleri tanımlamak için etkili araçların olmaması sorununu ele aldı, çünkü geliştirilen yöntemlerin çoğu sıklık temelliydi. Bu çalışmada önerilen model, genlerin önemini göz önünde bulundurarak sıralanmış verilere dayalı olarak sürücü genleri tanımlamak üzere tasarlanmıştı. Model, aşağıdaki adımlarla geliştirildi: İlk olarak, eğitim ve test için bir benchmark veri seti alındı; sorunla ilgili organik bir düzenleme testi tanımlandı; üçüncü olarak, tahmin algoritması geliştirildi, ardından sonuçlar üzerinde doğrulama gerçekleştirildi ve ardından genel kullanım için bir çerçeve sağlandı. Etkinlik ve değerlendirme için bu modele farklı doğrulama yöntemleri uygulandı. Doğrulama yöntemleri, Matthew korelasyon katsayısı, doğruluk, özgüllük ve duyarlılık gibi metrikler elde etmede yardımcı oldu. Ek deneylerle, sonuçlar modelin kendine tutarlılık oranının %91,08, bağımsız set oranının %87,26 ve çapraz doğrulama oranının %92,48 olduğunu gösterdi (Malebary and Khan, 2021).

Han vd. bir model olan DriverML'yi önerdi. Bu model, gerçekten kanser gelişimine dahil olan kanser sürücü genlerini tanımlamayı amaçlamaktaydı. Bu model, Gözetimli Makine Öğrenimi ve Rao'nun puan testi yöntemlerinin entegrasyonuna dayanmaktaydı. Puan testi, ilgili mutasyonların ilgili proteinler üzerindeki fonksiyonel etkiyi nicel olarak temsil eden ağırlık parametrelerini içeriyordu. Çalışma, Cancer Genome Atlas'tan (TCGA) elde edilen 31 veri kümesi üzerinde gerçekleştirildi ve 20 diğer mevcut yöntemle karşılaştırma yapıldı. DriverML modeli, çoklu mutasyon tipi entegrasyonu için bir puan istatistiği kullanımı, somatik mutasyonların fonksiyonel sonuçlarını nicel olarak analiz eden bir makine öğrenimi (ML) algoritması kullanımı ve nihayetinde puan istatistiği için bir tahmin dağılımı elde etmek için somatik mutasyonların Poisson sürecini taklit eden bir süreç olan simüle edilen daha gelişmiş yöntem ve malzemeler kullanmıştır. DriverML, diğer mevcut yöntemlere göre sağlam ve etkili olduğunu kanıtladı, çünkü örneklerin %87'sinin en az 1 sürücü geni taşıdığı ortaya çıktı (Han vd., 2019).

2018'de Bailey vd., kapsamlı bir sürücü gen karakterizasyonu gerçekleştirdi. Bu, PanCancer ve PanSoftware'u sistematik olarak analiz eden ve 9423 tümörü kapsayan bir çalışmaydı. Bu çalışma, 26 hesaplama aracı kullanarak sürücü genlerin ve mutasyonlarının bir kataloğunu gerçekleştirdi. Deneylerde, anatomik siteye bağlı olarak 299 sürücü gen belirlendi ve yapı ve dizi tabanlı analizle 3400'den fazla olası sürücü mutasyonu belirlendi. Yazarlara göre, bu araştırma 2018'e kadar kanser mutasyonlarının ve kanser genlerinin en kapsamlı tanımlamasını sağladı (Bailey vd., 2018). Erten vd., kanser sürücü genlerini belirlemek için genomik verileri birleştiren ağ merkezli bir model olan BetweenNet'i uyguladı. Bu önerilen model, hastaya özgü ağlar üzerindeki merkezlik derecesini ölçen bir yöntem kullanarak disfonksiyonel genleri, yani aykırı genleri belirledi. Bu model daha sonra, mutasyonlarla aykırı genler arasında bir ilişki kurarak oluşturulan bir bipartit grafikte rastgele yürüyüşler uygulayarak sürücü genleri önceliklendirdi. Bu çalışmada üç büyük ölçekli veri kümesi kullanıldı: Meme Kanseri veri kümesi, Akciğer Kanseri veri kümesi ve Pan-Kanser veri kümesi. Bu verilerin deneysel analizi, BetweenNet'in diğer yöntemlere göre daha iyi performans gösterdiğini ortaya koydu, özellikle gen sayısı N 100 ile 300 arasında olduğunda BetweenNet diğer yöntemlere kıyasla daha iyi performans gösterdi (Erten Houdjedj, and Kazan, 2021).

Song vd., gen dokusuna özgü ifade, düzensiz ifade ve gen-gen ağı ile ilgili kanser sürücü mutasyonlarını belirledi. Bu araştırma, sürücü genlerin, altta yatan genlerin ifadesi üzerinde göreceli bir etki yapma eğiliminde olduklarını dikkate almaktadır. Bu, olası etkileşim aracılığıyla, aynı dokular için benzer özelliklere sahip ifade edilebilen fonksiyonel modüllerin oluşmasına eğilimli oldukları anlamına gelir. Bu araştırmadaki önerilen model DyTiDriver adını taşıyor. DyTidriver, doku özgü ifade, gen-düzensiz ifade ve varyasyon frekanslarını insan fonksiyonel etkileşim ağı ile entegre etmek için kullanıldı ve kanserle ilgili sürücü genleri keşfetmek üzere tasarlandı. Bu modelin deneysel analizi ve değerlendirmesi, modelin Precision, F-score ve Recall değerlerini dikkate aldığında iyi durumda olan modelleri aştığını gösterdi. Ayrıca, modelin zenginleştirme analizi, bazı bilinmeyen sürücü genleri de keşfetme yeteneğini gösterdi. Bu çalışma, en büyük ifade verisi platformlarından biri olan Gene Expression Omnibus (GEO) veritabanını kullanarak gerçekleştirildi. Bu çalışmada kullanılan ana kısıtlama, bu veri setlerinin TCGA ve GEO dahil olmak üzere farklı projelerden elde edilmiş olması ve farklı hastalar arasında heterojenlik bulunması nedeniyle belirsiz veriler içeriyor olabilmesiydi (Song, Peng, Wang, and Wang, 2019a).

2020'de Akhavansafar ve Teimourpour, bir ağ tabanlı yaklaşım yöntemi ile gen anomalilerinin emilim ve yayılması sırasında göreceli gen öneminin birleşimi kullanarak kanser sürücü genlerini keşfetmeyi amaçladı. Bu çalışmada önerilen model, bir düğümün bir ağdaki göreceli etkisini hesaplamak için Katz ölçümünü kullanıyordu. Bu araştırmada, Katz, bir düğüm ile herhangi iki düğüm arasındaki tüm kenarları hesaplayarak göreceli etkiyi ölçtü. Direkt komşu prensibini kullanarak, düğüm ile ağdaki diğer düğümler arasında doğrudan bağlantılı komşuların sayısını ölçerek bu göreceli etkiyi ölçtü. Bu araştırmadaki önerilen yaklaşım KatzDriver adını taşıyor. Bu algoritma, gen etkileşim ağı ve gen ifade verisi kullanarak orijinal Katz algoritmasını değiştirmek üzere geliştirildi. Bu, her bir gen için iki skor üretti: bir skor, genin düzenleyici ağı üzerindeki biyolojik etkinin temsilini yapan etkileşim tabanlı biyolojik etkiyi temsil ederken, diğer skor, genin çıkan etkileşimlere dayalı etkisini temsil eder. KatzDriver'ın diğer ağ tabanlı modellere kıyasla üstün performans gösterdiği görülmüştür, çünkü diğer yöntemler tarafından keşfedilmemiş 22 benzersiz geni keşfetmiştir (Akhavansafar and Teimourpour, 2020).

3. Malzemeler ve yöntemler

Bu bölüm, bu araştırmanın önerilen modellerini, veri setlerini, biyolojik etkileşim ağının analizini ve farklı önemli değerlendirme parametrelerini detaylı bir şekilde incelemektedir. Bu bölümün yapısı, uygulanan veri setlerini, LSBA-Driver modelini ve geliştirilmiş bir optimizasyon algoritmasına dayanan ikinci bir modeli ayrıntılı bir şekilde ele alarak üç ayrı alt bölüme ayrılmıştır.

Veri Setleri

Deney aşamasının temeli, The Cancer Genome Atlas (TCGA) veri setlerinden alınan ve cBioPortal for Cancer Genomics (2022) kaynaklı veri setlerine dayanmaktadır. Bu araştırmanın temelini oluşturan dört ayrı veri seti, her biri gen mutasyon verilerini, kopya sayısı değişikliği (CNA) ve mRNA sekanslama ifade verilerini içermektedir. Glioblastoma Multiforme (GBM): 585 hastanın gen mutasyonu, CNA ve mRNA-seq ifade verilerini içeren kapsamlı bir veri seti, toplamda 68,802 geni içeriyor. Ovarian Serous Cystadenocarcinoma (OV): 585 hastadan elde edilen bu veri seti, 53,204 geni içeren gen mutasyonları, CNA ve mRNA-seq ifade verilerini içermektedir. Prostate Adenocarcinoma (PRAD): 334 hastayla oluşturulan PRAD veri seti, 34,192 geni içeren gen mutasyonları, gen ifade verileri ve CNA verilerini içermektedir. Breast Cancer (BRCA): 112 hastadan elde edilen bu veri seti, 17,272 geni içeren gen mutasyonu verilerini, CNA ve mRNA-seq ifade verilerini içermektedir. Veri setlerini güçlendiren biyolojik etkileşim ağı, REACTOME'den elde edilen, 518,302 geni içeren bir gen-gen etkileşim ağı olarak ortaya çıkmaktadır.

LBSA-Driver Modeli

Bu araştırmanın kalbinde, listeye dayalı simüle edilmiş birleştirme (LSBA-Driver) modeli bulunmaktadır. Bu model, orijinal simüle edilmiş birleştirme (SA) algoritmasının geliştirilmiş bir versiyonu olan güçlü bir çerçevedir. Bu alt bölüm, orijinal SA algoritmasına kısa bir girişle başlar ve araştırma metodolojisini şekillendirmedeki temel rolünü açıklar. SA, yerel minima sınırlarını aşma amacı güden bir optimizasyon algoritmasıdır ve Hill-Climb algoritmasının içinde bulunduğu sınırlamalardan evrilmiştir. SA algoritması, metalin anneleşmesini simüle etmek için sıcaklığın sistemli bir azalmasını yönlendiren dinamik ve tekrarlayan bir süreçtir. Her iterasyon, sıcaklığa bağlı bir olasılık dağılımı ile mevcut çözüme yaklaşımı değerlendiren yeni bir çözüm sağlar. Kabul veya reddetme işlemine Metropolis kabul kriteri hakimdir ve yeni çözümleri daha yüksek veya daha düşük amaç fonksiyonu değerlerine sahip olma esnekliğini sunar. Bu esneklik, yerel minima sıkışma riskini önlemede etkili olup, çözüm uzayının daha kapsamlı bir şekilde keşfedilmesini sağlar.

Belirtilen olasılık hesaplaması, çözüm kabulüne yönelik nuanslı bir yaklaşım sunar. Sıcaklık, yönlendirici bir parametre olarak hareket eder ve çözüm kabul olasılığını belirler, optimizasyon sürecinde keşif ve sömürü arasında denge kurar. Algoritmanın iteratif doğası, sıcaklığın sistemli azalmasıyla birleşerek çözüm uzayında kapsamlı bir şekilde dolaşmanın esaslarını içerir. SA algoritmasının başarılı bir şekilde uygulanabilmesi için sıcaklık, soğuma faktörü ve mahalle yapısı

gibi kritik parametrelerin tanımlanması gerekir. Ancak, bu parametrelerin optimal yapılandırması bir zorluk oluşturur, bu da adaptif soğutma stratejilerinin keşfini teşvik eder. Özellikle amaç fonksiyonunun hesaplamasının karmaşık olduğu durumlarda, algoritma genellikle sabit bir Markov zinciri uzunluğu kullanır ve sonlandırma koşullarını belirli iterasyon sürelerine dayandırır. Özetle, LSBA-Driver modeli, listeye dayalı simüle edilmiş birleştirmenin yeteneklerini kullanarak, yerel ve genel minima karmaşıklıklarını gezme yeteneğini optimize eder. Bu adaptasyon, çözüm uzayının kapsamlı bir şekilde keşfi aracılığıyla kanser sürücü genlerin tanımlanmasını kolaylaştırır. Bu önerilen modellerin, veri setlerinin ve algoritmaların kapsamlı keşfi, araştırma metodolojisinin karmaşıklıklarını anlama açısından güçlü bir temel oluşturur. İlerleyen bölümler, yürütülen araştırma çabasının sonuçlarına ve tartışmalarına dalacak, gerçekleştirilen çalışmanın etkilerini ve katkılarını aydınlatarak derinlemesine incelemeler sunacaktır.

Geliştirilmiş ARO Modeli

Yapay Tavşanlar Optimizasyonu (ARO), tavşanların hayatta kalma stratejilerinden türetilmiş benzersiz bir biyo-ilhamlı problem çözme yaklaşımı sunar. Doğada "sapma taraması" ve "rastgele gizlenme" stratejilerine dayanmaktadır ve keşif ve sömürü mekanizmalarını birleştirerek adaptabilitesini ve etkinliğini artırmaktadır. Sapma taraması, temel bir keşif mekanizması olup, tavşanların yuvalarından uzakta yiyecek arama davranışını taklit eder. ARO modelinde, popülasyondaki her tavşan pozisyonunu rastgele diğer tavşanların yuvasına yakın bir şekilde günceller. Bu, matematiksel olarak, tavşanın t+1 zamanındaki aday pozisyonunu, mevcut pozisyonu ve başka bir rastgele seçilmiş tavşanın pozisyonu temel alarak belirler. R,vhesaplanan bir parametre, sapma taramasının etkisini belirler. Ayrıca, Denklem, keşif stratejisini iterasyonlar ilerledikçe uyarlayan çalışma uzunluğu (L) kavramını tanıtır. ARO'daki sömürü, tavşanların avcılardan kaçmak için kullandığı rastgele gizlenme stratejisi tarafından temsil edilir. Algoritma, her iterasyonda arama alanı boyunca bir dizi tünel oluşturur ve bunlardan biri rastgele seçilerek saklanır. Hiding parametresi (H), iterasyonlar ilerledikçe lineer olarak azalır, arama alanını daraltır ve belirli bir bölgeye odaklanır. Tavşanların rastgele gizlenme sırasında yeni aday pozisyonlarını üretme ve tavşanın pozisyonunu güncelleme süreçlerini ayrıntılandırır.

Tavşanlar enerji seviyelerine bağlı olarak sapma taraması ve rastgele gizlenme arasında geçiş yaparlar. ARO, bu enerji odaklı davranışı, sömürü ve keşif arasındaki geçişi modellemek için enerji faktörü (A) uygulayarak yakalar. A(t), gösterildiği gibi iterasyon sayısının bir fonksiyonudur ve keşif veya sömürü kararı üzerinde etkilidir. Büyük bir A(t) değeri, yüksek enerji seviyelerini gösterir ve keşfi teşvik ederken, küçük bir A(t) değeri sömürüyü çağırır. Bu, ARO'nun performansını ve adaptabilitesini artırır. ARO, sapma taraması ve rastgele gizlenmeden önce mutasyon uygulayarak arama alanına çeşitlilik katma stratejisi ile geliştirilmiştir. Her iterasyonda, mevcut çözümdeki bireyler rastgele seçilir ve yeni rastgele seçilmiş pozisyonlar verilir. Mutasyon, popülasyona çeşitlilik enjekte ederek erken yakınsamayı engeller ve sürekli keşfi sağlar. İyileştirilmiş ARO algoritmasının pesudokodu açıklanmıştır, başlangıç, mutasyon ve sapma taraması ile rastgele gizlenme aşamalarını göstererek ana adımları gösterir. Sonuç olarak, ARO algoritması sapma taraması, rastgele gizlenme ve enerjiye bağlı uyum mekanizmalarını entegre ederek dinamik ve etkili bir optimizasyon tekniği sunar. Tavşanların hayatta kalma stratejilerinden esinlenerek, keşif ve sömürüyü başarılı bir şekilde dengeleyerek karmaşık problem alanlarında etkili bir şekilde gezinir. Algoritmanın mutasyon adımı, yakınlaşmayı önler ve sürekli keşfi garanti ederek popülasyona çeşitlilik katar. ARO algoritmasının akış şeması, sıralı adımlarını gösterir ve onun tekrarlayan ve uyarlanabilir doğasını sergiler. Gösterildiği gibi, ARO, biyolojik dünyadan elde edilen bilgileri kullanarak yenilikçi hesaplama çözümleri için ilham kaynağı olan vaat edici bir optimizasyon yaklaşımı sunar.

4. Bulgular

Bu bölümde, her iki önerilen modelin sonuçları değerlendirilir, sunulur ve diğer önde gelen yöntemlerle karşılaştırılır. Değerlendirme, doğruluk, hassasiyet ve duyarlılık hesaplanarak gerçekleştirilir. Önce LBSA modelinin sonuçları analiz edilecek ve sonra ARO modelinin sonuçları analiz edilecektir.

Algoritmalar, rastgele bir üst N sıralı gen sayısını alarak ardından sonuçların hassasiyetini, duyarlılığını, F-skorunu ve doğruluğunu hesaplayarak değerlendirilir. Denklemlerde, doğru pozitifler TP tarafından temsil edilir, doğru negatifler TN tarafından temsil edilir, yanlış pozitifler FP tarafından temsil edilir ve yanlış negatifler FN tarafından temsil edilir. TP için değer, algoritma tarafından keşfedilen genlerin toplam sayısını hesaplayarak elde edilir ve aynı zamanda NCG7.0 tarafından kanonik sürücü genler olarak etiketlenmiştir. TN için değer, NCG7.0 tarafından kanonik sürücü genler olarak etiketlenmemiş ve ayrıca algoritma tarafından en üst N gen arasında sıralanmamış gen sayısını temsil eder (N farklı populasyon boyutlarını temsil eder). FP için değer, algoritmamız tarafından tanımlanan ancak NCG7.0 tarafından kanonik sürücü genler olarak etiketlenmemiş genlerin toplam sayısını hesaplayarak elde edilir. Son olarak, FN değeri, algoritma tarafından keşfedilmemiş ancak NCG7.0 veritabanı tarafından sürücü genler olarak etiketlenmiş genlerin toplam sayısını hesaplayarak elde edilir. Aşağıdaki Tablo 4.1, bu araştırmada kullanılan farklı veri kümeleri için NCG7.0 tarafından kanonik gen olarak etiketlenen gen sayısını göstermektedir. Beyin kanseri için Glioblastoma Multiforme veri seti, 24 etiketli kanonik kanser sürücü gen içerir. Dünya genelinde en popüler kanser türlerinden biri olan Meme Karsinomu, 110 farklı kanonik sürücü gen içerir. Over Kanseri, 11 kanonik sürücü geni etiketler ve Prostat Adenokarsinomu, 43 kanonik sürücü geni etiketler.

Meme kanseri veri seti için performans karşılaştırma analizinde, algoritma, Meme Kanser Genleri ve Sağlıklı Sürücüler Ağı (NGC7.0) tarafından kabul edilen TP53'ü en üst sıradaki gen olarak belirleyerek meme kanserinde 32 kanonik sürücü geni tanımlayabildi. Algoritmamızın sonuçlarını DriverNet, DawnRank, IntDriver ve SubDyquency sonuçları ile karşılaştırdıktan sonra, algoritma, yukarıda bahsedilen herhangi bir yöntem tarafından tanımlanmayan beş kanonik sürücü geni tanımlayabildi. Bunlar AKT1, KMT2C, CBFB, ARID1A ve CTCF'dir. Yöntem aynı zamanda DriverNet tarafından da tanımlanan ancak diğer yöntemler tarafından tanımlanmayan üç diğer kanonik geni keşfetti. DriverNet tarafından tanımlanmış olsalar da, DriverNet sonuçlarında çok daha düşük bir sıraya sahip oldukları belirlendi. Bu genler, NOTCH1, algoritmamız tarafından 7. sırada sıralandı ve DriverNet algoritması tarafından 1243. sırada sıralandı, NF1, algoritmamız tarafından 19. sırada sıralandı ve DriverNet algoritması tarafından 235. sırada sıralandı ve RB1, DriverNet tarafından 58. sırada, DawnRank tarafından 35. sırada, Subdyquency tarafından 22. sırada sıralandı ve IntDriver tarafından keşfedilmedi.

NGC7.0 tarafından listelenen 11 kanonik sürücüden, bu araştırmadaki önerilen algoritma, Over Kanseri Veri Seti'nden yedi geni tanımlayabildi. Bu yedi genin altısı, bu algoritma tarafından en iyi 30 sıralamasına girdi. Bunlar: TP53, BRCA1, RB1, CTNNB1, FAT3 ve KRAS. Bu yaklaşımın sonuçlarını DriverNet algoritmasının sonuçlarıyla karşılaştırdıktan sonra bulgular, bu algoritmanın DriverNet algoritması tarafından keşfedilmemiş olan FAT3'ü keşfettiğini gösterdi. Bu algoritma tarafından keşfedilen ancak en iyi 30 sıralamasına girmeyen 1 gen, 571. sırada yer alan KREMEN1'dir.

Bu algoritmanın GBM veri kümesi ile yapılan performans analizinde, algoritma NGC sürüm 7 tarafından listelenen 124 kanonik ve aday sürücü arasından 23 sürücü gen keşfetmeyi başardı. Keşfedilen 23 genin içinde, algoritma DriverNet Algoritması tarafından keşfedilmemiş olan beş geni keşfetmeyi başardı. Bu beş gen sırasıyla PTPN11, STAG2, ATRX, COL1A2 ve IGF1R'dir. Bu genlerin algoritmamızdaki sıralamaları sırasıyla 13, 16, 24, 27 ve 29'dur. PRAD veri kümesi için yapılan performans analizi sonuçları, dört farklı yöntemle (DriverNet, DawnRank, Subdyquency ve IntDriver) karşılaştırıldı. Bu çalışmada önerilen yöntem, NGC veritabanı tarafından kanonik sürücü genler olarak listelenen 13 geni keşfetmeyi başardı. Bu 13 gen içinde, TP53, CTNNB1, FOXA1, PIK3CA, ATM, PTEN ve AKT1 sırasıyla 1, 3, 4, 5, 6, 8 ve 15 sırada yer aldı. AKT1 geni, bu algoritma tarafından keşfedilen ve diğer algoritmalar tarafından keşfedilmeyen bir gen oldu. Performans analizi, bu algoritmanın mevcut en iyi yöntemlerden daha iyi bir performans sergileyebildiğini kanıtladı. Bu araştırmada önerilen yaklaşım tarafından belirlenen ve karşılaştırılan yöntemler tarafından belirlenmeyen en iyi sıralı kanonik sürücü genler şunlardır: Meme Kanseri veri kümesi için bu yaklaşım, AKT1, KMT2C, CBFB, ARID1A ve CTCF'yi keşfetti; Over Kanseri veri kümesi için bu yaklaşım, FAT3 ve KREMEN1'i keşfetti; Beyin kanseri olan Glioblastoma

Multiforme veri kümesi için bu yaklaşım, PTPN11, STAG2, ATRX, COL1A2 ve IGF1R'yi keşfetti; son olarak, Prostat Adenokarsinoma veri kümesi için bu yaklaşım, AKT1 sürücü genini keşfetti. Glioblastoma multiforme beyin kanseri veri kümesinin (GBM) performans analizi karşılaştırması, toplamda 24 kanonik gen içeren NCG7.0 tarafından sunulan 68,802 genle yapıldı ve 15 kanonik gen belirlendi. Bu 15 genin dokuzu, ARO algoritması tarafından en iyi 30 gen arasına yerleştirildi. Bu genler arasında; TP53, PTEN, PIK3R1, PIK3CA, RB1, NF1, PDGFRA, PTPN11 ve STAG2 bulunmaktadır. Triple-negatif meme kanseri veri kümesi (BRCA) için, toplam 110 kanonik sürücü genin içinden bu çalışmada 55 BRCA sürücü geni belirlendi. Belirlenen 55 sürücü genin 25'i, ARO tarafından en iyi 30 gen arasına yerleştirildi. Over kanseri (OV) için, toplamda 11 kanonik kanser geninden altısı bu çalışmada belirlendi ve bu genlerden 4'ü (TP53, BRCA1, RB1, FAT3) en iyi 30 gen içinde yer aldı. Son olarak, prostat adenokarsinoma (PRAD) veri kümesi için, 43 kanonik sürücü genin içinden ARO deneyleri ile 19'u belirlendi ve bu genlerin 12'si en iyi 30 gen içinde sıralandı.

Algoritma, sonuçların doğruluğunu, duyarlılığını, f-skorunu ve doğruluğunu hesaplayarak değerlendirildi. Burada, gerçek pozitifler TP olarak, gerçek negatifler TN olarak, yanlış pozitifler FP olarak ve yanlış negatifler FN olarak temsil edilmektedir. TP için elde edilen değer, algoritma tarafından bulunan ve aynı zamanda NCG7.0 tarafından kanonik sürücü genleri olarak etiketlenen genlerin toplam sayısını hesaplayarak elde edildi. TN için değer, NCG7.0 tarafından kanonik sürücü genleri olarak etiketlenmemiş ve ayrıca algoritma tarafından en iyi N geni arasında sıralanmamış genlerin sayısı olarak alındı (N, farklı populasyon büyüklüklerini temsil etmektedir). FP için değer, algoritmamız tarafından belirlenen ancak NCG7.0 tarafından kanonik sürücü genleri olarak etiketlenmemiş genlerin toplam sayısını hesaplayarak elde edildi. Son olarak, FN için değer, algoritma tarafından keşfedilmemiş ancak NCG7.0 veritabanı tarafından sürücü genleri olarak etiketlenmiş genlerin toplam sayısını hesaplayarak elde edildi.

Büyük biyolojik veri setlerinde eğilimlerin belirlenmesi gerektiğinde, genellikle kullanılan en ünlü teknik, ilaç mekanizmalarını ve hastalıkları keşfetmek amacıyla tıbbi olarak uygulanabilen fonksiyonel zenginleştirme analizidir (Wijesooriya, Jadaam, Perera, Kaur, and Ziemann, 2022). Bu teknik gen setleri üzerine uygulandığı için Gen Seti Zenginleştirme Analizi (GSEA) olarak da bilinir. Tanımlama süreci, genlerin ya azalmış ya da zenginleşmiş bir grup olup olmadığını keşfetmek için istatistiksel yöntemler ve yaklaşımlar kullanılarak gerçekleştirilir. GSEA'nın altındaki yapıda birkaç prosedür bulunmaktadır. Bunlar, gen düzeyi istatistikleri, global testler, dönüşüm, gen seti istatistikleri, null hipotezleri ve anlamlılık değerlendirmeyi içerir. Gen düzeyi istatistikleri, kat değişimini, sinyal-gürültü oranını, t-istatistiği, regresyon katsayısını, log-olabilirlik oranını ve korelasyon katsayısını içeren parametreleri analiz eder. Global testler için küresel Ancova ve Hotelling's T2 testleri kullanılır. Dönüşüm, ikili, sıralı, p-değeri, yerel FDR, mutlak değer, kare tabanlı dönüşüm veya bazı durumlarda uygulanmayan bir dönüşüm olabilir. Gen seti istatistikleri, Wilcoxon sıralama testi, MaxMean istatistiği, ortalama, toplam ve medyan, zenginleştirme skoru, işaret testi ve koşullu FDR'ı içeren çeşitli testleri içerir. Anlamlılık değerlendirme testleri, örnek etiketi permütasyonu, gen örnekleme ve tekrar standartlaştırma içerir. Zenginleştirme analizi gerçekleştirmek için farklı araçlar bulunmaktadır. Bu araçlar arasında NASQAR (Yousif, Drou, Rowe, Khalfan, and Gunsalus, 2020), Enrichr (Laboratory Ma'ayan, 2014), AmiGO 2 (Consortium Gene Ontology) ve Blast2GO (Conesa et al., 2005) bulunmaktadır. Bu araştırmada, Gen Ontolojisi (GO) analizi, Gene Ontology Resource (GOR) içinde PANTHER Overrepresentation testi kullanılarak gerçekleştirilir (The Gene Ontology Resource, 2022). Bu, 43.648 GO terimini, 7.510.543 açıklamayı, 1.473.200 gen ürününü ve 5.183 türü içeren GOR'un en son sürümüdür.

5. Tartışma

Bu araştırmada iki yöntem önerilmiştir. İlk yöntem LBSA-DRIVER olarak adlandırılmıştır ve kanser sürücü genlerini tanımlamak için bir transkripsiyon ağı kullanılan ağ tabanlı bir modeldir. Bu alanda tek başına genler yerine alt ağları tanımlamak için benzer çalışmalar yapılmıştır (Vandin, Upfal, and Raphael, 2011; Bashashati et al., 2012; Bokhari, Alhareeri, and Arodz, 2020; Xi, Wang, and Li, 2018; Sarkar A. et al., 2020). Bu çalışmada, gen-gen etkileşim ağı, ilgili genlerin etkisini doğrulamak için kullanılmaktadır. Burada önerilen model, DriverNet algoritmasında kullanılan bipartit grafikle

benzer bir şekilde kullanır. Bu bipartit grafik, sol tarafındaki düğümlerin gen mutasyonları olduğu, sağ tarafındaki düğümlerin ise aykırı ifade mutasyonları olduğu bir şekilde oluşturulur. Bipartit grafiği oluşturduktan sonra, optimizasyon problemi oluşturulur; bu problem, grafikte sol tarafta bulunan genleri, biyolojik bir ağdaki etkilerine göre sağ tarafta bulunan genlere eşleştirmeye çalışır (Bu durumda bir gen-gen etkileşim ağı). Bu optimizasyon problemini çözmek için, (Zhan, Lin, Zhang, and Zhong, 2016) tarafından önerilen List-tabanlı Simüle Edilmiş Tavlama (LBSA) adlı yeni bir yöntem kullanılmaktadır. Bu yöntem, simüle edilmiş tavlama yapmak için özel olarak oluşturulmuş bir sıcaklık listesini kullanır. Ardından, TCGA'nın dört karşılaştırmalı veri setinde deneysel analiz yapılır. Bu veri setleri, beyin kanseri ile ilgili TCGA Glioblastoma Multiforme (GBM) veri seti, meme kanseri için TCGA Breast Cancer (BRCA) veri seti, prostat kanseri için TCGA Prostate Adenocarcinoma (PRAD) veri seti ve over kanseri için TCGA Ovarian Serous Cystadenocarcinoma (OV) veri setini içerir. Deneylerden elde edilen sonuçlara yönelik olarak karşılaştırmalı analiz zenginleştirme analizi yapılmış ve sonuçlar, algoritmamızın NCG7.0 veritabanında temsil edilen kanonik kanser genlerini etkili bir şekilde tanımlayabildiğini göstermiştir.

GBM veri setinin analizinde, 23 sürücü geni tanımlayabildik. Bunlardan bu yaklaşım, DriverNet algoritması tarafından tanımlanmayan PTPN11, STAG2, ATRX, COL1A2 ve IGF1R'yi tanımlayabildi. Benzer şekilde, OV veri setinin karşılaştırmalı analizinde, algoritma FAT3 gibi diğer son teknoloji yöntemlerle keşfedilmeyen TP53, BRCA1, RB1, CTNNB1, FAT3 ve KRAS'ı tanımlayabildi. PRAD veri setindeki sonuçlar, OV veri setiyle benzerdi, çünkü yöntem diğer algoritmalar tarafından bulunmayan AKT1 gibi bir ek geni tanımlayabildi. Son olarak, BRCA veri setindeki performans, diğer son teknoloji yöntemler tarafından tanımlanmayan AKT1, KMT2C, CBFB, ARID1A ve CTCF'yi tanımladı. Sonuçların değerlendirilmesi, doğruluk, f-score, hatırlama ve doğruluk hesaplanarak ve DriverNet, SubDyquency, DawnRank ve IntDriver gibi state-of-the-art yöntemlerin sonuçlarıyla karşılaştırılarak gerçekleştirildi. Bu algoritmanın bir dezavantajı vardır; işleme süresidir. Simüle edilmiş algoritma tekrarlayıcı olduğu için burada geliştirilen model, doğruluk için çalışma süresini riske atmaktadır. Veri seti ne kadar büyük olursa, algoritmanın sonuçları üretmesi o kadar uzun sürer. Bu nedenle, işleme süresi sorununu ele almak için daha hızlı çalışan ancak yaklaşımın doğruluğunu koruyan modern bir meta-sezgisel algoritma olan Yapay Tavşanlar Optimizasyon Algoritması'nı uygulayarak bu sorunu çözmeye çalışıyoruz. Bu, modelin çok daha büyük olan PanCancer veri setlerinde etkili bir şekilde kullanılmasına olanak tanır.

Bu araştırmadaki optimizasyon sorununu çözmek için etkili bir şekilde çalışacak başka bir algoritma örneği de Gri Kurt Optimizasyon algoritmasıdır (2022) (Besharatnia, Talebpour, and Aliakbary, 2022).
Bu araştırmadaki ikinci yöntem, 2022'de Wang L ve ark. tarafından önerilen biyolojiden esinlenmiş Yapay Tavşanlar Optimizasyon algoritmasının geliştirilmiş bir versiyonudur. Bu algoritma, ARO'nun detour foraging ve random hiding aşamalarından önce mutasyon aşaması eklenerek iyileştirilmiştir. Bu çalışmadaki model şu şekildedir: İlk olarak, TCGA veri deposundan elde edilen kanser verileri ve biyolojik etkileşim ağı kullanılarak, DriverNet ve QuaDMutNetEx'te sunulan gibi bir bipartit grafik oluşturuldu. Daha sonra, ARO bu bipartit grafiğe uygulandı ve kanser sürücü genleri tanımlanmaya çalışıldı. Bu geliştirilmiş ARO algoritması şu şekilde uygulanır: Rastgele bir çözüm, komşuluk uzayında başlatıldı. Daha sonra bu çözüme mutasyon uygulandı ve arama uzayını kapsayan daha çeşitli bir çözüm oluşturuldu. Sonraki adımda, enerji faktörü A, mevcut tavşan çözümünde detour foraging veya random hiding'in uygulanıp uygulanmayacağını belirlemek için hesaplandı. Bu, tavşanların hayatta kalma için kullandığı bir enerji küçültme mekanizmasıdır. Eğer enerji faktörü 1'den büyükse, o zaman detour foraging (keşif) uygulanır ve tavşanın yeni konumu oluşturulur. Eğer enerji faktörü 1'den küçük veya eşitse, o zaman random hiding uygulanır ve mevcut tavşanın yeni konumu oluşturulur. Her iterasyonda, mevcut çözümün fitness'ı Schwefel 2.22 fonksiyonu kullanılarak hesaplandı ve aday çözümün fitness'ı ile karşılaştırılarak yeni mevcut çözüm seçildi. En iyi çözüm, iterasyonların sonunda elde edildi. Daha sonra bu modelin sonuçlarını değerlendirmek için performans analizi uygulandı. Performans analizi, dört benchmarking veri seti üzerinde gerçekleştirildi: GBM beyin kanseri veri seti, PRAD prostat kanseri veri seti, BRCA veri

seti ve OV veri seti. Sonuçlar, kanser sürücü mutasyonlarını tanımlamak için kullanılan dört diğer state-of-the-art modelle karşılaştırıldı: Dendrix, DriverNet, HotNet2 ve QuaDMutNetEx. Bu

çalışmada önerilen ARO'nun, bu yöntemlere kıyasla kanser sürücü mutasyonlarını tanımlamada oldukça etkili olduğu kanıtlanmıştır. ARO, NCG7.0 tarafından kanonik genler olarak etiketlenen şu sürücü genleri tanımlayabildi. GBM için: TP53, PTEN, PIK3R1, PIK3CA, RB1, NF1, PDGFRA, PTPN11 ve STAG2, en üst 30 sıra içinde tanımlananlar arasında yer aldı; OV için: TP53, BRCA1, RB1 ve FAT3, en üst 30 sıra içinde tanımlananlar arasında yer aldı; PRAD için: TP53, CTNNB1, FOXA1, PIK3CA, ATM, PTEN, SPOP, HRAS, KMT2D, NCOR1, APC ve KMT2C, en üst 30 sıra içinde tanımlananlar arasında yer aldı; BRCA için: TP53, PIK3CA, EP300, AKT1, CDH1, KMT2C, GATA3, PIK3R1, MAP3K1, NOTCH1, ERBB3, NCOR1, ERBB2, NF1, CBFB, EGFR, ARID1A, BRCA1, CTCF, SMAD4, ATR, RB1, RUNX1 ve PTEN, en üst 30 sıra içinde tanımlananlar arasında yer aldı. Bu çalışma için ARO'nun uygun bir algoritma olmasının birkaç nedeni vardır. ARO'da detour foraging, küresel arama yapmada yardımcı olurken, random hiding, yerel arama yapmada yardımcı olur. Denklem 3.6'da sunulan R parametresi, iterasyon sayısı arttıkça adaptif olarak ayarlanabilir ve keşiften sömürüye doğru gidilen kademeli değişimi teşvik edebilir. Iterasyonların yaklaşık yarısı sömürüye ($A \leq 1$) ve diğer yarısı keşfe ($A > 1$) atanır, bu da enerji faktörü A'nın bir sonucu olarak ortaya çıkar. Bu enerji faktörü, sömürü ve keşif arasında etkin bir şekilde geçiş yapabilen ve bu aşamaları geliştiren zamana bağlı bir faktördür. ARO, sömürü ve keşfi dengeleme yeteneği sayesinde elde edilen hata teşhisini taşıma yeteneğine sahiptir. ARO'nun hesaplama karmaşıklığı lineerdir ve şu şekilde verilir: O(ARO) = O(1+ n + Tn + 0.5*Tnd + 0.5 Tnd) = O(Tnd + Tn + n). Üstün bir performansa sahip olmasına rağmen, ARO'nun birkaç eksikliği vardır. ARO, belirsiz değişken türlerine sahip problemlerin optimizasyonu sırasında çoklu arama mekanizmalarını içermemektedir. Ayrıca, çoklu ekstremumları olan unimodal problemleri işleme konusunda zayıflıkları bulunmaktadır. Bu nedenle, ARO, NP-Zor problemleri çözmekte daha az etkilidir. Bu sorunlar, algoritmanın çoklu hedefli ve ikili sürümlerini uygulayarak çözülebilir. Son olarak, her veri seti için en iyi 30 sıralanan genler üzerinde biyolojik süreçler üzerindeki etkilerini belirlemek için GO analizi yapılmıştır.

6. Sonuçlar

Biyoenformatik uzmanları ve araştırmacılar, yolcu mutasyonlarından gen sürücü mutasyonlarını belirlemek için yöntemler geliştirmişlerdir. Sürücü mutasyonlarını tanımlama sorunu, hangi mutasyonların yolcu mutasyonlar olarak kabul edileceğini bulmakla doğrudan ilgilidir ve bunlar kanser hastalığının gelişimine doğrudan katılmazlar. Kanser sürücü mutasyonları, kanser başlangıcını ve ilerlemesini doğrudan etkileyen mutasyonlardır, yolcu mutasyonları ise değildir. Kanser tümörlerindeki mutasyonların oldukça büyük bir yüzdesini oluşturan yolcu genlerden kanser sürücü genlerini tanımlama konusunda aşırı bir çaba harcanmıştır. İyi bir teknik bulma sorunu, bioenformatik ve hesaplamalı biyoloji topluluğunda on yıllardır var olan ve bilim adamlarının eldeki sorunu çözmek için birçok yöntem ve girişimde bulunduğu bir sorundur. Bu sorun, ilk başta yalnızca tek genleri veya mutasyonları tanımlamaktan, benzer fonksiyonel ve biyolojik özelliklere sahip bir grup geni içeren patikaları tanımlamaya doğru yavaşça kaymıştır. Başlangıçta, bunları yapmak için kullanılan tüm yöntemler frekans temelliydi, yani bir gen sadece frekansına dayanarak bir sürücü mutasyon olarak kabul ediliyordu. Yani, yüksek frekansta bulunan genler sürücü genler olarak kabul edilirken, daha düşük frekanslı genler yolcu gen mutasyonları olarak etiketlenirdi. Ancak bu önemli bir sorunu beraberinde getiriyordu, çünkü çoğu sürücü gen düşük frekansta olmasına rağmen kanser gelişiminde önemli bir role sahipti. Bu nedenle, frekansa dayanmayan, ancak genlerin bir biyolojik ağ üzerindeki etkisine dayanan ağ tabanlı yöntemler geliştirilmiştir.

Bu araştırmada önerilen yöntemler, bir optimizasyon problemi çözmek için yeni bir liste tabanlı simüle edilmiş tavlama yaklaşımını kullanan ağ tabanlı bir yöntem ve kanser sürücü genlerini tanımlamak için kullanılan geliştirilmiş bir biyolojiden ilham alan ARO algoritmasıdır. Algoritma, önceki önerilen yöntemlere kıyasla sunduğu avantajlar nedeniyle seçilmiştir, özellikle arama alanını tüketme yeteneği ve etkinliği ile öne çıkmaktadır. Benchmarking veri setlerinin deneysel analiziyle, bu yöntemin kanser sürücü genlerini tanımlamada etkili olduğu kanıtlanmıştır. Yaklaşımlar, zaman içinde geliştirilmiş olan DriverNet, DawnRank, IntDriver ve Subdyquency gibi state-of-the-art yöntemlerle karşılaştırılmıştır. Bu yaklaşımlar, açıkça kanonik sürücü genleri tanımlamada bu yöntemlerden daha iyi performans göstermektedir.

Kaynaklar

Akhavansafar, M., and Teimourpour, B. (2020). A network Based method to predict cancer causal genes in GR Network. *bioRxiv*.(3),24-27.

Bailey, M., Tokheim, C., Porta-Pardo, E., Sengupta, S., Bertrand, D., Weerasinghe, A., and Schein, J. (2018). Comprehensive characterization of cancer driver genes and mutations. *Cell*, 173(2), 371-385.

Bashashati, A., Haffari, G., Ding, J., Ha , G., Lui, K., Rosner, J., and Shah, P. (2012). DriverNet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer. *Genome biology*, 13(12), 1-14.

Chaudhary, M., Pham, V., Le, T., and Robinson, P. (2021). NIBNA: a network-based node importance approach for identifying breast cancer drivers. *Bioinformatics*, 37(17), 2521-2528.

Dees, N., Z hang, Q., Kandoth, C., Wendl, M., Schierding, W., and Koboldt, D. (2012). MuSiC: identifying mutational significance in cancer genomes. *Genome Research*, 22(8),1589–98.

Erten, C., Houdjedj, A., and Kazan, H. (2021). Ranking cancer drivers via betweenness-based outlier detection and random walks. *BMC bioinformatics*, 22(1), 1-16.

Malebary, S., and Khan, Y. (2021). Evaluating machine learning methodologies for identification of cancer driver genes. *Scientific reports*, 11(1), 1-13.

Shui, L., Xiaoxiao, Y., Dan, Z., Jiyao, S., Xin, W., Qingyu, W., . . . Xuewen , Z. (2018). Analysis of Transcription Factor-Related Regulatory Networks Based on Bioinformatics Analysis and Validation in Hepatocellular Carcinoma. *BioMed Research International*. (23) 128–137.

Song, J., Peng, W., Wang, F., and Wang, J. (2019a). Identifying driver genes involving gene dysregulated expression, tissue-specific expression and gene-gene network. *BMC Medical Genomics*, 12(7), 1-12.

Zhang, T., and Zhang, D. (2017). Integrating omics data and protein interaction networks to prioritize driver genes in cancer. *PubMedCentral,*. 8(35), 58050–58060.

Zhang, T., Zhang, S., and Li, Y. (2021). Identifying driver genes for individual patients through inductive matrix completion. *Bioinformatics*, 37(23), 4477-4484.

.

Bilim Kodu	:	92410
Anahtar Kelimeler	:	Kanser, sürücü genler, gen ontolojisi, Liste Tabanlı Simüle Edilmiş Tavlama, Yapay Tavşan Optimizasyonu, meta-sezgisel algoritmalar, biyolojik etkileşim ağları
Sayfa Adedi	:	87
Danışman	:	Dr. Öğr. Üyesi Yılmaz ATAY

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Asst. Prof. Dr. Yılmaz ATAY, for his unwavering guidance on every aspect of this thesis. I would also like to thank all my friends and family who continued to show their support throughout the preparation of this thesis.

CONTENTS

Page

ABSTRACT iv

ÖZET v

ACKNOWLEDGEMENTS xv

CONTENTS xvi

LIST OF TABLES xviii

LIST OF FIGURES xix

SYMBOLS AND ABBREVIATIONS xxi

1. INTRODUCTION 1

2. LITERATURE REVIEW/RELATED WORKS 17

3. MATERIALS AND METHODS 29

3.1. Datasets 29

3.2. LBSA-Driver Model 30

3.2.1. Simulated Annealing (SA) problem 30

3.2.2. The proposed LBSA algorithm 36

3.3. Improved ARO Model 46

3.2.1. General concept 46

3.2.2. Model and algorithm 47

4. RESULTS 53

4.1. Results Analysis for LBSA-Driver 54

4.1.1. Benchmarking analysis for BRCA 63

4.1.2. Benchmarking analysis for Ovarian Cancer (OV) dataset 64

4.1.3. Benchmarking analysis for Glioblastoma Multiforme (GBM) dataset 64

4.1.4. Benchmarking analysis for Prostate Cancer (PRAD) dataset 68

Page

4.2. Results for Improved ARO 68

4.2.1. Performance benchmarking analysis 70

4.2.2. Performance evaluation 71

5. DISCUSSION 75

6. CONCLUSION 79

REFERENCES 81

RESUME 91

LIST OF TABLES

Table **Page**

Table 3.1. Details of TCGA datasets obtained from the cBioPortal 30

Table 3.2. Pseudocode for the Improved ARO algorithm 50

Table 4.1. Number of canonical driver genes identified by the NCG7.0 for different datasets..... 53

Table 4.2. Comparison of accuracy for BRCA and PRAD datasets..... 54

Table 4.3. Comparison of Precision, Recall and F-Score for GBM, OV and PRAD datasets..... 55

Table 4.4. Benchmarking analysis of the BRCA dataset in comparison with DriverNet, DawnRank, Int_Driver and Subdyquency..... 65

Table 4.5. Benchmarking analysis of the OV and GBM datasets in comparison with DriverNet..... 66

Table 4.6. Benchmarking analysis of the PRAD dataset in comparison with DriverNet, DawnRank, Int_Driver and Subdyquency..... 67

Table 4.7. Top-ranked Canonical driver genes identified by the proposed LBSA-DRIVER model and not discovered by DriverNet, DawnRank, IntDriver and Subdyquency..... 68

Table 4.8. Top 30 ranked genes discovered by ARO from experiments on GBM, BRCA, PRAD and OV 69

Table 4.9. Driver genes discovered by ARO ranked amongst the top 30..... 71

Table 4.10. Canonical driver gene count in the NCG7.0 for datasets..... 72

Table 4.11. Comparison of canonical genes with HotNet2, DriverNet, Dendrix and QuaDMutNetEx..... 73

LIST OF FIGURES

Figure **Page**

Figure 1.1. Precision Medicine, An overview. 2

Figure 1.2. Improving cancer treatment and detection through cancer biomarkers. (Bourré, 2019)..... 4

Figure 1.3. Driver mutation vs passenger mutation frequency found in cancer patients. 8

Figure 1.4. Layers of a bipartite graph..... 12

Figure 1.5. Analysis of Transcription Factor-Related Regulatory Networks Based on Bioinformatics Analysis and Validation in Hepatocellular Carcinoma (Shui et al. 2018)..... 13

Figure 1.6. Example of a gene expression network in the blood of a Schizophrenia patient (Simone de Jong, 2012) 14

Figure 3.1. A display of the Simulated Annealing algorithm on both local and global minima/maxima 31

Figure 3.2. Flowchart for the Simulated Annealing (SA) algorithm. 33

Figure 3.3. A graphical summary of the processes involved in identifying cancer driver mutations using a biological interaction network, a bipartite graph and the LBSA optimization algorithm..... 34

Figure 3.4. Constructing the bipartite graph using gene mutation information and a biological interaction network. 37

Figure 3.5. Graphical examples of the different properties of biological networks. 38

Figure 3.6. Some of the interactions found in the Reactome_FlslnGene GGI network used in this research..... 40

Figure 3.7. Flow chart describing the processes involved in List-based Simulated Annealing (LBSA) algorithm 43

Figure 3.8. Flow chart for creating the initial temperature list for LBSA 45

Figure 3.9. Behavior of the energy factor, A, over 1000 iterations. 49

Figure 3.10. Flow chart of Artificial Rabbits Optimization (ARO) algorithm. 51

Figure 4.1. Comparing values of precision between LBSA-Driver and DriverNet for the GBM dataset 58

Figure 4.2. Comparing values of precision between LBSA-Driver and DriverNet for the PRAD dataset..... 58

Figure **Page**

Figure 4.3. Comparing values of precision between LBSA-Driver and DriverNet for the OV dataset. 59

Figure 4.4. Comparing values of precision between LBSA-Driver and DriverNet for the BRCA dataset 59

Figure 4.5. Comparing values of recall between LBSA-Driver and DriverNet for the GBM dataset 60

Figure 4.6. This curve represents the comparison of recall for the PRAD dataset between LBSA-Driver and DriverNet 60

Figure 4.7. This curve represents the comparison of recall for the OV dataset between LBSA-Driver and DriverNet 61

Figure 4.8. This curve represents the comparison of recall for the PRAD dataset between LBSA-Driver and DriverNet 61

Figure 4.9. Curve comparing F-score values between LBSA-Driver and DriverNet for GBM dataset 62

Figure 4.10. Curve comparing F-score values between LBSA-Driver and DriverNet for PRAD dataset 62

Figure 4.11. Curve comparing F-score values between LBSA-Driver and DriverNet for OV dataset 63

Figure 4.12. Curve comparing F-score values between LBSA-Driver and DriverNet for OV dataset 63

Figure 4.13. Settings for performing GO using PANTHER Overrepresentation Test 74

SYMBOLS AND ABBREVIATIONS

The abbreviations used in this study are presented below, along with their explanations.

Abbreviations	Explanations
ARO	Artificial Rabbits Optimization
AUC	Area under the Curve
BRCA	Breast Cancer
CGC1	Cancer Gene Census
CGC2	Cancer Genome Consortium
CML	Chronic Myelogenous Leukemia
CNA	Copy Number Alteration
CRC	Colorectal Cancer
CTC	Circulating Tumor Cells
DNA	Deoxyribonucleic Acid
ELBSA	Enhanced List-based Simulated Annealing
FAP	Familial Adenomatous Polyposis
FDR	Frequency Discovery Rate
FN	False Negative
FP	False Positive
GBM	Glioblastoma Multiforme
GCN	Gene Co-expression Network
GEO	Gene Expression Omnibus
GGI	Gene-Gene Interaction
GO	Gene Ontology
GSEA	Gene Set Enrichment Analysis
GWO	Grey Wolf Optimization
HGS	High Grade Serous Ovarian Cancer
KEGG	Kyoto Encyclopedia of Genes and Genomes
LBSA	List-based Simulated Annealing
LBTA	List-Based Threshold Accepting
NCG	Network of Cancer Genes and Healthy Drivers
NCI	National Cancer Institute

Abbreviations	Explanations
OV	Ovarian Cancer
PNC	Personalized Network Control
PRAD	Prostate Adenocarcinoma
RNA	Ribonucleic Acid
SA	Simulated Annealing
TCGA	The Cancer Genome Atlas
TN	True Negative
TN BRCA	Triple Negative Breast Cancer
TP	True Positive
TSP	Traveling Salesman Problem
VMCL	Variable Markov Chain Length

1. INTRODUCTION

The medical and technological worlds constantly battle to sustain human life against various diseases. While some diseases have a cure, others do not, and pose a severe threat to humanity. One of such a disease is cancer. Scientists from different fields, such as bio-science, mathematics, and computer science, have collaboratively worked together for decades to find a cure for cancer. This collaborative effort is generally classified under the field of Precision medicine, which is a critical phenomenon based on the analysis of the roles that genes and proteins play in the therapeutic treatment of cancer. As defined by the National Cancer Institute (NCI) (NIH, 2022), precision medicine is a form of medicine that leverages information about an individual's proteins and/or genes to diagnose, prevent, or treat a disease.

Precision medicine is composed of various elements. With time, these components have been updated or upgraded and their definitions and terms altered as more and more research is carried out in the area of study. According to the University of California, San Fransisco, (Yamamoto, 2022) it is composed of elements, such as Basic Discovery, Building the knowledge network, Clinical Discovery, Computational Health Sciences, Digital Health, Ethics and Engagement, Imaging, Omics' Medicine and Population Sciences. Basic discovery can be regarded as the underlying criteria for biological research. It forms the main pillars of medicine and disease whereby it is required for being able to acquire a good understanding and good knowledge of precision medicine. It has a direct collaboration with clinical discovery and the spectrum of omics data generated by the help of computational health scientists and the available facilities. Concerning ethics and engagement, the gains and positives of precision medicine cannot be recognized if it does not serve the benefit of everyone in the community. Furthermore, with the development in technology, engagement with the community has been made much easier and much more realizable, and the same goes for the collection of important data in underrepresented communities. As concerns digital health, it is an aspect whereby patients tend to collaborate with researchers and clinicians. This could be realized by patients uploading data to research labs using FDA-approved applications and devices.

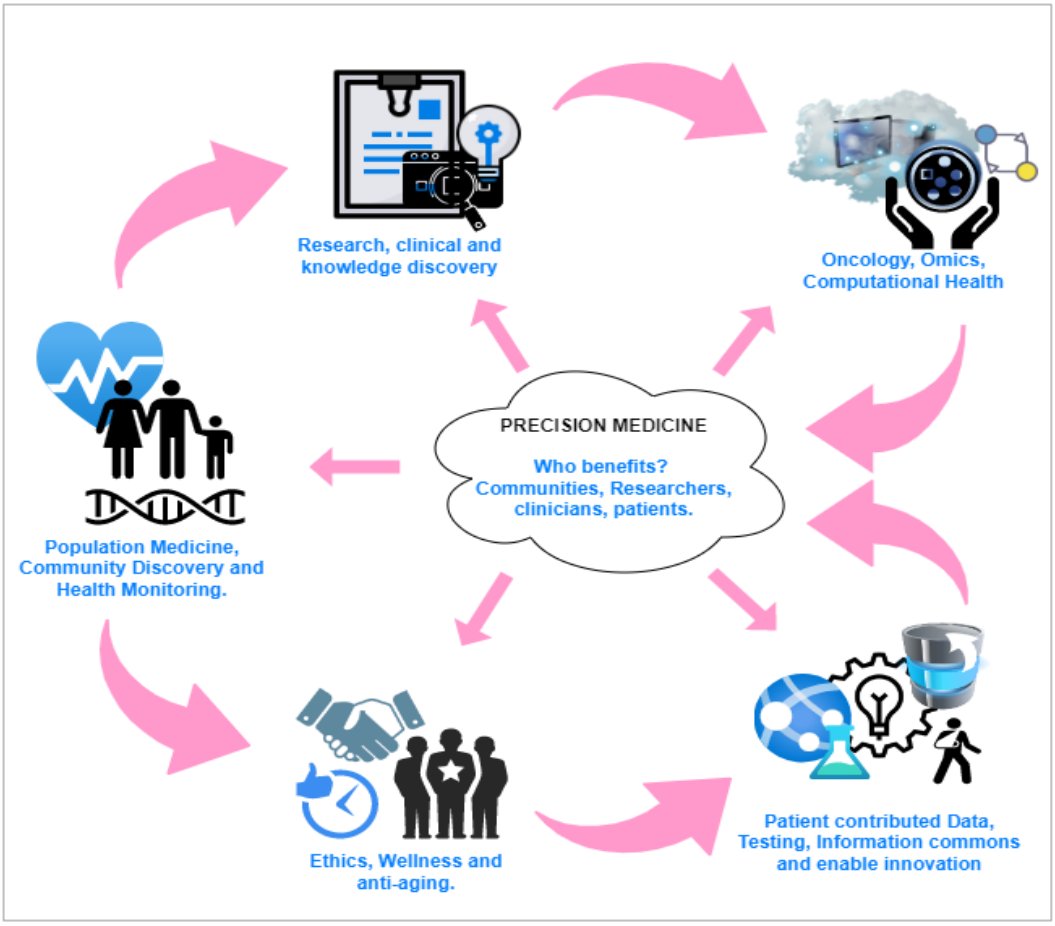


Figure 1.1. Precision Medicine, An overview.

This kind of treatment uses data about the lifestyle and the environment of the patient, integrates this information into machine learning prediction algorithms, and produces effective strategies for treatment. For example, concerning cancer, precision medicine will pick an individual considered a cancer patient, and extract specific or directed information related to that individual's tumor, to perform diagnosis, plan treatment, and also track the progress of the treatment. The image in Figure 1.1 shows a general view of precision medicine whereby the beneficiaries of precision medicine include academic, social and medical researchers, members of communities, clinicians, and most importantly, the patients. The science of precision medicine is considerably population-based, where the study of population medicine is considered when trying to identify communities and as well perform health monitoring. This is in a direct relationship with research as understanding the communities can help researchers make significant discoveries, hence clinical and knowledge discovery. It is important to note that monitoring the population and obtaining data from the communities requires some level of ethics to go with. Therefore, precision

medicine considers ethical procedures and the wellness of the population on which the research is going to be performed. With patient-contributed data and novel technology, methods, and algorithms, it is possible to perform testing and foster innovation. The recent improvement in computational health and bioinformatics has played quite a significant role in fostering testing and deduction of accurate results by researchers. All of these components are what make up the concept of precision medicine.

As another example, let us consider a patient who is in a situation where he or she needs a blood transfusion. Concerning precision learning, the blood that will be given to the patient will be obtained according to the blood type of the donor rather than randomly selecting any person as a donor. This helps to prevent complications where the receptor body might be rejecting the donor's blood.

Just like precision medicine, another important medical phenomenon known as Biomarker testing is used for cancer treatment. This is a method of finding genes, proteins, and other biological components that can provide useful information about cancer. Looking at a simple demonstration, such as that shown in Figure 1.2, it can be seen that biomarkers are used to track how much a disease develops and responds to therapy. This evolves about four main groups of patients: the drug responders, the patients who go through a poor prognosis, patients who are exposed to severe side effects, and those who have a high risk of being carriers of the disease. For assessing cancer biomarkers, levels of biological components, such as proteins, biochemicals, peptides, RNA, and DNA are measured.

Depending on the application and use, biomarkers can be categorized into three main types. These are Predictive Biomarkers, Prognostic Biomarkers, and Diagnostic Biomarkers. Predictive biomarkers are used to guide medical experts in making useful treatment decisions by indicating the responses of specific therapy techniques. For example, HER2 activation present in breast cancer gives a response to trastuzumab; KRAS activation shows resistance carriers of EGFR present in Colorectal Cancer (CRC); there is also the BCR-ABL1 responding to carriers in Chronic Myelogenous Leukemia (CML). Prognosis Biomarkers, unlike predictive biomarkers, deal with the aftermath of therapy. That is, they are concerned with assessing the risk of clinical outcomes, for example, in cases where there could be disease progression or recurrence. For example, the rise or fall of Prostate-specific antigen (PSA) levels give information about the progression of prostate cancer; the presence of mutations, such as BRCA1 and BRCA2, that have an added risk of women experiencing

breast cancer and men experiencing prostate cancer; recurrent mutations in the APC gene which can expose a person to the Familial Adenomatous Polyposis (FAP), alterations in DNA methylation in tumors, and the availability of Circulating Tumor Cells (CTC) found in peripheral blood. As for diagnostic biomarkers, they are useful in understanding if a given patient has a specific disease condition. For example, the Bence-Jones protein urine test used in testing for multiple myeloma; high levels of carcinoembryonic antigen used for monitoring CRC patients; high levels of PSA that are typically more in patients who have prostate cancer; CD20 that is crucial in performing the diagnosis and treatment of relapsed lymphoma. The different classes of cancer biomarkers that exist include Genetic Biomarkers, Transcriptomic Biomarkers, Epigenetic Biomarkers, Proteomic Biomarkers, and Metabolic Biomarkers.

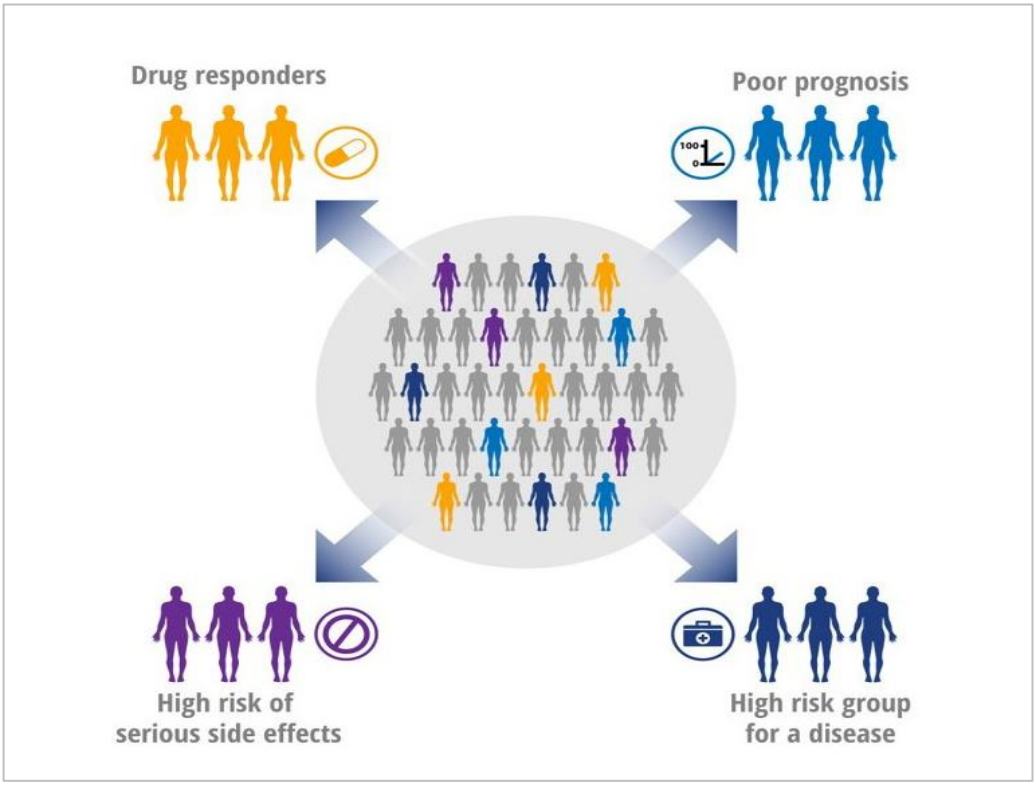


Figure 1.2. Improving cancer treatment and detection through cancer biomarker (Bourré, 2019)

For each cancer patient, the set of cancer biomarkers is unique such that finding out what biomarkers are involved in an individual's cancer helps their doctors determine what cancer treatment is best suited for that patient. This is where the role of experts in computational biology and bioinformatics becomes of most importance as they aim to identify biomarkers, also known as mutations. These mutations are identified as actionable genomic aberrations

that serve as a basis for targeted treatment of the cancer disease. With this in mind, the concept of driver gene mutation identification has become a major topic of concern for experts in the field with the constant development of novel methods, approaches, and techniques to tackle this issue. One of the most popular initiatives for this is The Cancer Genome Atlas (TCGA) (Network, 2008) initiative. This project is a large-scale sequencing initiative that simultaneously generates transcriptome interrogations as well as a whole genome for cases that have the same tumor type. Originally with the sole aim of systematically exploring (to full scale) changes found in cancer in humans, this project was able to characterize about 20 000 normal samples as well as primary cancer samples from over 33 types of cancer. As a result, the influence that somatic mutations have on functional gene expression networks can be appropriately ascribed. In this research, "cancer driver genes" and "cancer driver mutations" are used interchangeably.

Topic description

Over the years, there has been the constant development of computational methods that can be used for identifying cancer driver genes, typically based on extracted cancer genomic data (Cheng, Zhao, and Zhao, 2015). These methods can be divided into two major groups: Frequency-based and Network-based methods. The frequency-based methods assume a mutation is a driver mutation if it occurs more frequently across a cohort of many patients (Lawrence et al., 2013). Some of the popular studies which are frequency-based include: the study by (Lawrence et al., 2013) called MutsigCV, whose frequency measure is concerning the related biological profile (Dees et al., 2012) consider the mutated genes with a frequency that are more than an expected chance using a Background Mutation Rate (BMR) as significant driver genes. Also, Tian et al. proposed a method known as ContrastRank, which uses a reverse approach assuming that the variants with the least frequency have a higher impact and are considered driver mutations (Tian, Basu, and Capriotti, 2014). The main disadvantage with frequency-based methods is that they all consider that only genes with very high frequencies (typically greater than 20%) are potential driver genes. However, statistical analysis shows that cancer mutations occur more often at lower frequencies, typically between 2% and 20%. Due to this limitation, researchers had to develop a solution to this by introducing network-based methods. This was based on the concept that genes act together in biological processes and form biological networks. One of the most popular network-based methods is the HotNet (Vandin, Upfal, and Raphael, 2011) developed by Vandin et al. in 2011 that extracts significantly mutated subnetworks by applying it to a

mutated gene network, a propagation process that aims to identify driver genes. Just as frequency-based methods have their limitations, network-based methods also have another limitation whereby the methods extensively rely on the biological network. This means that if the biological network contains false information or contains errors, it significantly affects the results of the method.

Cancer studies describe two main types of mutations. These are passenger mutations and driver mutations. The problem of identifying driver mutations is directly related to finding out what mutations qualify as passenger mutations and are not directly involved in the development of the cancer disease. Cancer driver mutations are mutations that directly drive cancer initiation and progression, while passenger mutations do not (Wodarz, Newell, and Komarova, 2018; Kumar et al., 2020). Excessive work has been conducted in trying to identify cancer driver genes from passenger genes as passenger genes consist of quite a large percentage of mutations in cancer tumors. This research presents a method for discovering cancer-driver genes which evidently performs quite better than other state-of-the-art methods as proven through the experimental results. Here, the model identifies canonical driver genes that have been labelled by cancer databases. Identifying these driver genes is significant in cancer studies and research, and the race to find a potential cure for the disease.

Problem statement

There is considerable amount of cancer genes found in cancer patients. These genes are generally divided into two main groups - Passenger Genes and Driver Genes (Aparisi et al., 2019). Careful studies have proved that only a very small percentage of the genes found in cancer carriers are actually driver mutations. These kinds of mutations are called “driver mutations” because they are considered to be the initiators of tumors – otherwise known as tumor-genesis. For example, more or less, the most popular driver mutation present in most types of cancer, including breast cancer, lung cancer, brain cancer, and ovarian cancer, is the TP53. Another example is the APC mutation found in colon cancer (Li , Ishikawa, Oshima, and Taketo, 2005). Passenger mutations, on the other hand, are not involved in tumor-genesis but rather are involved in sustaining the tumor after it has been created and also foster rapid growth and development. This process of tumor growth is also known as metastasis. An example of a passenger mutation is the TWIST1 passenger mutation (Benjamin and Keshet, 1997). As mentioned earlier, the percentage of passenger mutations in a cancer tumor is far more than the percentage of driver mutations in that tumor. It is for

this reason that finding these driver mutations is a really challenging problem faced by medical scientists and technology experts (similar to the well-known phrase “finding a needle in a haystack”). A demonstration of the existence of driver mutations and passenger mutations in a cancer patient is shown in Figure 1.3.

Finding these driver mutations requires solving a very hard optimization problem, as the amount of available data is also growing rapidly. For example, let us consider a data set extracted from cancer patients with about 50,000 mutations. Among these mutations, on average, 30 of these mutations are effectively detectable as cancer driver mutations. Finding these 30 mutations among the 50,000 will require a really good optimization algorithm that can accurately sort out these mutations from the rest of the mutations which are only passenger mutations. Developing such an optimization technique is a problem on which researchers all over the world spend time and effort relentlessly. With that said, it is important to acknowledge the already developed techniques for addressing the aforementioned problem. Generally, there have been frequency-based approaches developed and implemented in this area of research. However, as carefully observed, these methods have not been able to effectively pick out the driver mutations/genes. The reason for this is that these methods are solely and totally based on the frequency of these mutations (as the name implies). This means that the methods consider the number of times a single gene appears in the large bulks of data and consider it as a driver mutation if this number of appearances (frequency) is greater than a certain value. As much as this method is reasonable and produces good results, it faces quite a setback, considering that there actually exist cancer driver mutations that are not very frequent in the available data (with a frequency that is typically less than 2%) but are highly important cancer driver genes. Now, as clearly expected, a method that is based solely on frequency will not be able to detect such a mutation.

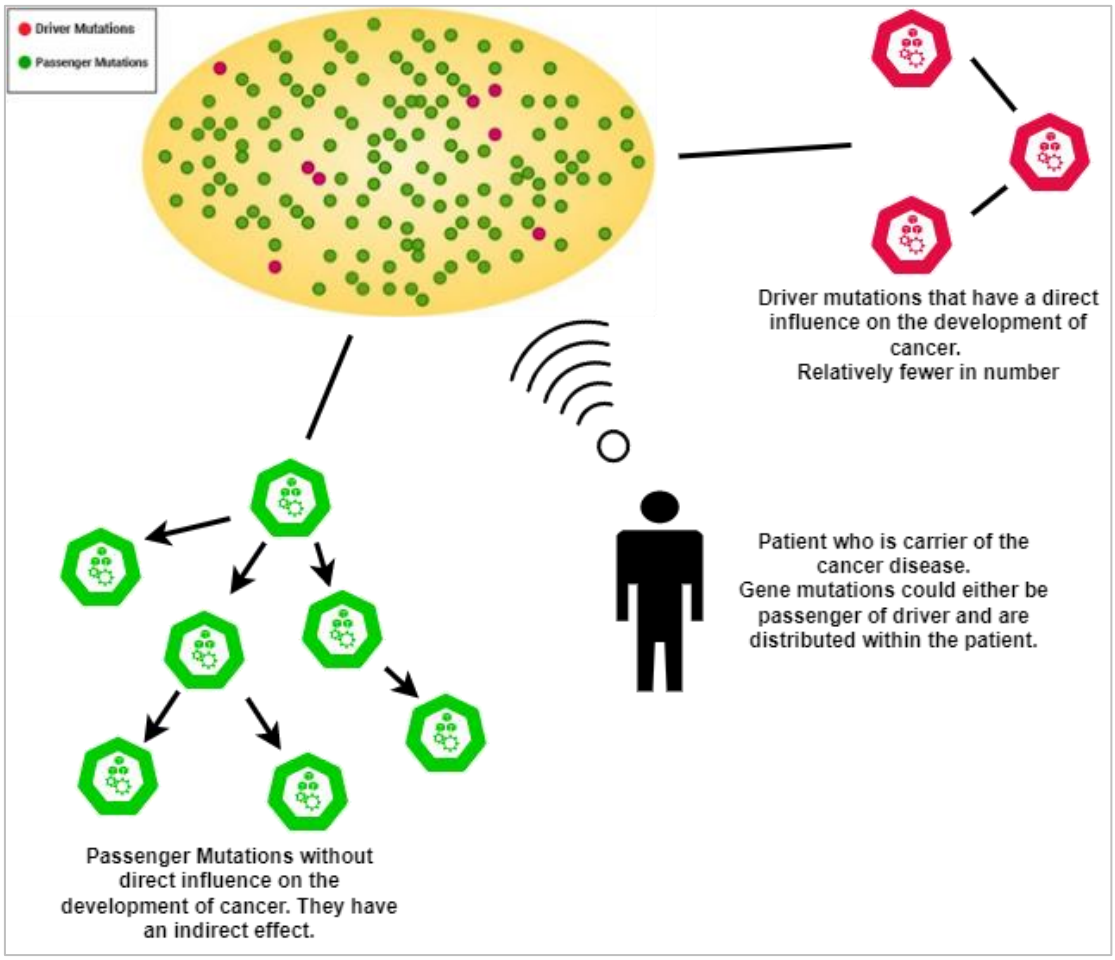


Figure 1.3. Driver mutation vs passenger mutation frequency found in cancer patients.

The problem of finding a good technique to identify cancer driver genes has existed in the bioinformatics and computational biology society for decades and has seen many methods and attempts by scientists to solve the problem at hand. This problem has also gradually moved from identifying only single genes or mutations to identifying pathways that consist of a group of genes sharing similar functional and biological properties. The different great studies that have been conducted in this area of research will be mentioned in detail in the literature review section of this research.

Aim of research

The base aim of this research is to be able to identify or detect cancer driver genes from large sets of data using an accurate optimization technique as well as a biological interaction network. Here, the biological interaction network plays a very important role in the proposed model. This is because, by using a biological interaction network, the model can match genes not only concerning their frequency but also with respect to their relationships and

interactions found in this interaction network. The interaction network consists of mutations that are linked to each other according to some biological and functional characteristics. This research then aims to create an accurate bipartite graph (a graph that consists of two layers) made up of mutations on one layer of the graph and gene expression on the other layer of the graph, which will then be merged with the biological interaction network and used as input for the developed optimization algorithm.

The second aim of this research is to be able to produce results that have an upper hand when compared to state-of-the-art methods that have been developed with the similar aim of finding cancer driver genes, with respect to the accuracy and efficiency of the results derived. This is to be done by ranking the detected potential driver genes in ascending order whereby the gene with the rank of 1 is considered the most eligible to be a driver mutation. After these genes get ranked, a number, N, of genes, is defined and the top N genes will be obtained and a detailed comparison will be made with those obtained from the other methods or techniques. It is important to note that only identifying the top-ranked genes is not enough to acknowledge that the proposed model is working efficiently. That is why the research aims to validate the results derived by performing a calculation of the Precision, Recall, F-score, and Accuracy of the output. The process of doing this calculation will inquire access to data from the Network of Cancer Genes and Healthy Drivers (NCG, 2022). With the data obtained from NCG, the values for False Positives (FP), False Negatives (FN), True Positives (TP) and True Negatives (TN) will need to be derived with respect to the mutations that have been labeled as Canonical driver mutations on by the platform. Canonical driver mutations are mutations that have been confirmed to be driver mutations involved in carcinogenesis. This study aims to perform this on four different popular types of cancer. These are brain cancer, breast cancer, lung cancer, and ovarian cancer. These types of cancer are very popular and occur in a large number of humans; therefore, it has a rich database. After performing the comparison of results, the next target is to perform what is known as a functional enrichment analysis process, otherwise known as Gene Ontology, on the driver mutations. This analysis shows the role that the discovered genes play in affecting biological processes in the human homo-sapiens species.

Importance of this research

One of the greatest questions that is asked in the world today when talking about cancer is how it can be detected early and prevented. This is not only discussed by patients or people

who know or are related to cancer patients but is also discussed by the big important organizations, such as the National Cancer Institute (NCI). The detection part here is a phenomenon that can be addressed scientifically using detection techniques. Given the fact that an eligible cure for this disease has not yet been developed by medical experts in the world, preventing the disease firsthand is the best medicine for it. That is why detecting the potential of a person becoming a cancer patient is of great importance, as the person can work well to prevent it. As mentioned earlier, cancer starts with the availability of cancer driver mutations and develops over time with the help of passenger mutations. Given the fact that cancer genes mutate at a very rapid pace in the late stages means that detecting the driver mutations that foster carcinogenesis in the early stages and preventing them is the best way of actually treating the disease.

This research is crucial as it aims to bring forward an accurate method for detecting these driver mutations which will help prevent the disease. as Also, it gives other researchers in the field a lifting point for their research as well as also serving as a knowledge platform for information about cancer research through the literature review section, which discusses quite a number of the different studies that have been conducted in this area. Different researchers and scientists typically have different points of view when tackling the driver mutation identification problem. These different views are what help medical experts administer their different forms of therapy to cancer patients concerning the information and data obtained from the experiments carried out using proposed techniques. The technique proposed in this research not only adds to the available pool of methods and approaches for handling cancer therapy, but its great results give a great standpoint on which genes are more important and should be paid more attention to as compared to those found in the other methods.

Research assumptions

When conducting this research, some very important assumptions that are related both to the datasets and the performed proposed algorithm are taken into consideration. These assumptions permit carrying the research better while considering some plausible constraints. The assumptions made in this research are as follows:

- The assumption that the downloaded cancer datasets used in this research were accurately collected by the third-party organization from which the data was downloaded (In this case, the Cancer Genome Atlas).

- The assumption that the NCG7.0 database provides accurate information about what genes are labelled as canonical driver genes. This is important because this information is used to perform the calculation of the accuracy of the proposed algorithm which is used to compare with the other state-of-the-art algorithms and prove that it can produce better results.
- This research assumes that the results provided by the other research methods with which this research is compared are accurate and not fabricated.
- The research assumes that the web platform that is used to perform functional enrichment analysis on identified driver genes provides accurate information about the biological process related to these genes.
- The research assumes that the selected algorithm (List-based Simulated Annealing Algorithm), which was proposed by Zhan et al., was accurately defined by the related authors as this algorithm is the base technique used in this research for identifying and ranking driver mutations (Zhan, Lin, Zhang, and Zhong, 2016).

Research limitations

The major limitation faced in this research is the problem of computational time when using the List-based Simulated Annealing algorithm. As much as it is important to derive accurate results, it is also very important when these and how these results are obtained. The computational time of an algorithm involved in cancer driver mutation identification is important because the size of data used in this field of study increases by the day, given the number of cancer cases also increases rapidly. This means that in the near future, the available data to work with are actually going to be of much larger sizes and this data will need to be efficiently used. A slow algorithm will take days and even weeks to complete processing on such data. This research forfeits computational time for accuracy of results, and therefore, the computational time of the algorithm turns out to be a little less efficient. In future works, the research aims to integrate efficiency with accuracy to be able to provide the best results in the shortest possible time.

Another limitation of this research is that this research used data which are generated by third-party platforms. For example, the biological gene-gene interaction network used in this research is one generated by a third-party research database. This implies that the accuracy of the results is also affected by how good these generated data are. If the generated data are faulty, it could affect the performance of the algorithm negatively. Therefore, the research

makes the assumption that the information provided is accurate enough to minimize the chances of error.

Definitions

The following are some important terms used in this research:

Biological Interaction Network: This is a network consisting of nodes and edges whereby the nodes in the network represent biological entities, such as genes, proteins, or enzymes. These entities are linked to each other through edges, and the edges represent the relationship between them. In a biological interaction network, one bio-entity can be linked to more than one other bio-entity and also can have more than one other entity linked to it (typically a many-to-many relationship).

Bipartite Graph: A bipartite graph is a graph made up of two layers such that the elements of one layer are linked to the elements of the other layer and vice versa. Bipartite graphs are used in bioinformatic research to represent biological entities of different types. For example, one layer consists of gene mutations while the other layer consists of proteins, as shown in Figure 1.4 below.

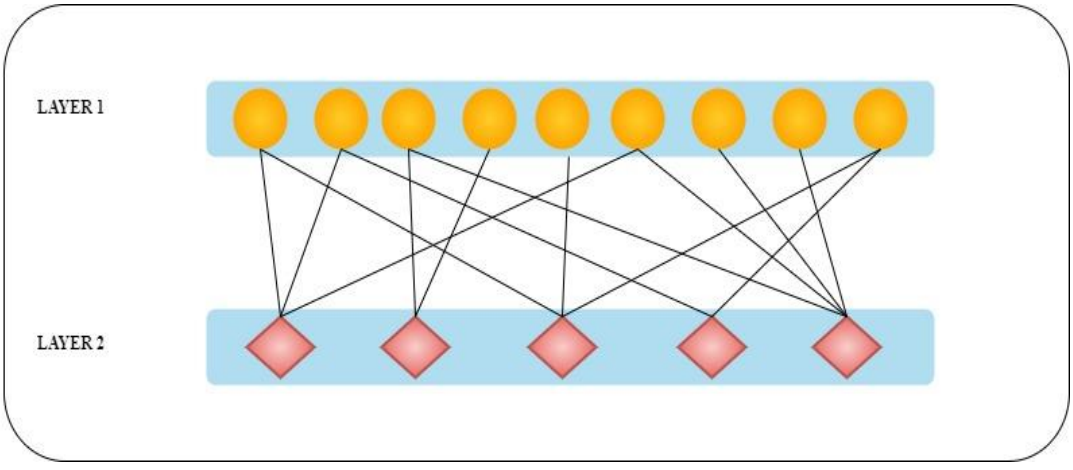


Figure 1.4. Layers of a bipartite graph

Gene Ontology: Gene Ontology is a method of functional enrichment analysis whereby biological components are analyzed to understand their usefulness in both functional and biological processes.

Carcinogenesis: This is the process of cancer initialization whereby the cancer disease is first identified in the patient. It is also known as tumor-genesis.

Canonical Drivers: These are drivers that have been accepted worldwide as driver mutations that drive the cancer process and lead to carcinogenesis.

Somatic Mutations: Somatic mutations (Irving, 2017) are changes in the DNA sequences of somatic cells of multicellular organisms with their own germ cells. These are mutations that occur in cells that are not intended to receive mutations. Somatic mutations, unlike dedicated germline mutations, are not passed on from generation to generation. According to the National Cancer Institute, somatic mutations are changes in DNA that occur after conception. These types of mutations can occur in all somatic cells except gametes (sperm and egg cells) and are, therefore, not passed on to children. Somatic mutations are genetic changes passed on to offspring during cell division. Somatic mutations are often caused by external factors, such as UV exposure.

Transcription Networks: These are networks in which individual components of the system between transcription factors and regulatory interactions, such as proteins or genes interact. For example, Figure 1.5 below is a demonstration of a transcription factor-related regulatory network.

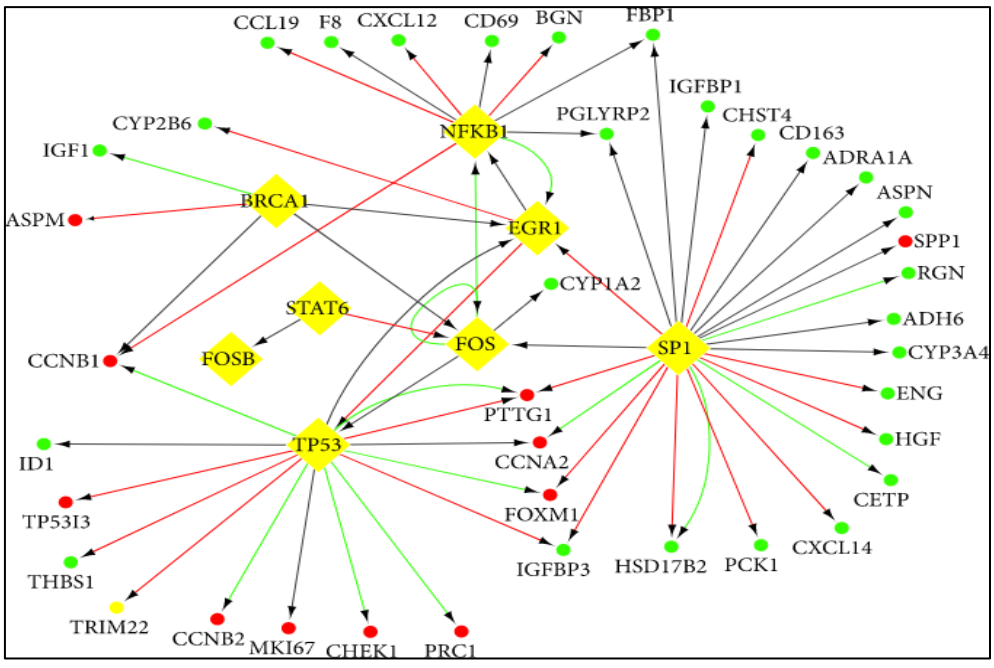


Figure 1.5. Analysis of Transcription Factor-Related Regulatory Networks Based on Bioinformatics Analysis and Validation in Hepatocellular Carcinoma (Shui et al., 2018)

Transcriptome: A transcriptome refers to the protein-coding (Annelies et al., 2022) portion of an organism's genome. It includes a range of RNA molecules, such as messenger RNA (mRNA), transfer RNA (tRNA), ribosomal RNA (rRNA), and other non-coding RNA molecules present in cells. It is a combination of the words transcript and genome related to the process of transcript production during the biological transcription process. Data derived from transcriptomes are used in research to gain insight into processes, such as cell differentiation, oncogenesis, transcriptional regulation, and biomarker discovery.

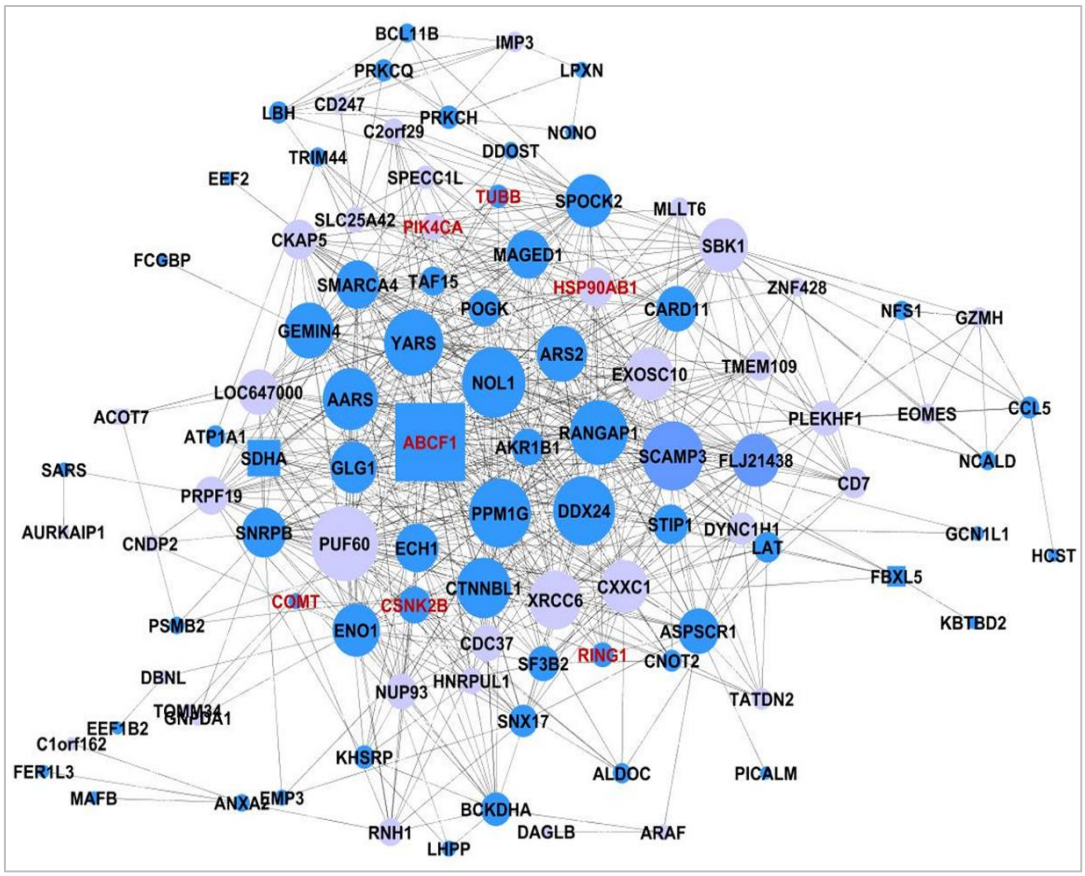


Figure 1.6. Example of a gene expression network in the blood of a Schizophrenia patient (Simone de Jong, 2012)

Gene Expression Network: This is a diagram showing that each node corresponds to a gene, and pairs of genes connected to the same edge have significant co-expression relationships (Example shown in Figure 1.6). Gene expression networks are commonly referred to as gene co-expression networks (GCNs) (Wangxiao, Hui, Huifang, Yaowen, and Xingchun, 2022). These types of networks are of great interest in biology because co-expressed genes are under the control of the same transcriptional regulatory program. Gene expression networks are gene-rich but have the disadvantage of being able to handle very small samples for cost

reasons. Transcription is the first step in gene expression, where a gene's DNA is copied into RNA. A GCN is a transcript-transcript association network commonly reported as an undirected graph.

Oncogenic Interaction Network: Oncogenes are genes that encode proteins that can transform cells in culture or induce cancer in animals. Most oncogenes appear to be derived from normal genes and are highly evolutionarily conserved. The involvement of proto-oncogenes in signal transduction leads to unregulated growth regulation. Oncogene activation and tumor suppressor gene inactivation are two processes believed to be involved in the oncogenic process. Oncogenic interactions are interactions between oncogenes in an oncogenic interaction network.

Metabolic Network: Metabolic networks are the complete set of metabolic and physical processes that determine the physiological and biochemical properties of cells. A metabolic network consists of metabolic chemical reactions, metabolic pathways, and the regulatory interactions that control those reactions. These networks can be used to identify comorbidity patterns in sick patients. Examples include the Kyoto Encyclopedia of Genes and Genomes (KEGG), EcoCyc, BioCyc, and metaTIGER. Metabolic networks represent the relationships between small biomolecules (metabolites) and the enzymes (proteins) interacting with them to catalyze biological processes. They can also be defined as interconnected pathways of biochemical reactions within living cells that assemble or degrade building blocks or compounds necessary for cellular function.

Copy Number Alterations: This is a phenomenon in which parts of the genome are repeated, and the number of repeats in the genome varies from individual to individual. Copy number alterations (Ried et al., 2019) contribute to the development and pathogenesis of cancer in such a way that the germline DNA of an individual differs in copy number from the DNA of a clonal subpopulation of cells. This type of number change is called Copy Number Alteration (CNA). The difference between copy number variations and copy number alterations is that CNAs are copy number alterations originating in somatic cells, whereas copy number variations arise from germline copy number alterations. Somatic copy number alterations are a ubiquitous feature of human cancers and contribute to tumorigenesis by influencing the dosage of multiple genes simultaneously.

Malignancy: This is the term used for diseases in which abnormal cells divide uncontrollably and can invade nearby tissues. Malignant cells can also spread to other parts of the body

through the blood and lymphatic system. There are several main types of malignant tumors. Cancer that begins in the skin or tissue that covers internal organs.

Pathogenic: A pathogen is anything (a bacterium, virus, or microbe) that causes disease. Therefore, pathogenic cancer driver mutations are mutations that cause carcinogenesis.

Evolutionary Algorithm: An evolutionary algorithm is an algorithm that simulates the behavior of living organisms to provide solutions to real-life problems. They are, in other words, known as bio-inspired algorithms.

Organization of Research Document

This research document is organized as follows: Chapter 1 is the introductory section whereby a general overview of what the research is all about is discussed, including the aim of the research, the problem to be solved, how the problem is addressed as well as some short definitions of the various terms used in the document. Chapter 2 provides a literature review of the different related research works and studies that have been conducted in this area. This review includes studies carried out over the last decade about cancer driver gene identification using both frequency-based methods and network-based methods and identification through pathways rather than single genes. In Chapters 3 and 4, the proposed models (List-based Simulated Annealing and Artificial Rabbit's Optimization, respectively) are discussed, with an analogy of the methodologies used, experiments implemented, and algorithms applied to achieve the research goals. After applying the experiments, the results have to be evaluated and analyzed. Therefore, Chapter 5 talks about the results of this research, presenting clear information and data about the research outputs, and also performing comparisons with other selected state-of-the-art methods that have the same purpose. This Chapter further performs an analysis of results through functional enrichment. Chapter 6 of this research is the discussion section of this research whereby a summary of the crucial aspects seen in this research is mentioned, and problems faced during the research and the future works that the research aims to cover. Finally, the last section Chapter 7 is the conclusion and, as the name implies, concludes the research.

2. LITERATURE REVIEW/RELATED WORKS

There has been an underdevelopment in methods that make scientific use of the relationship between related changes in gene expression and genomic events to find mutated driver genes in cancer. Most of the available approaches and algorithms are totally dependent on the frequency of the mutations of a given gene as the criteria for identifying what gene is the driver gene. However, frequency analysis faces tremendous backlash as it is not very successful in identifying driver genes that are of low frequency but still happen to be very important driver genes. (Bashashati et al., 2012), in their research in 2012, developed a model known as DriverNet to uncover the effect that somatic driver genes have on transcriptional networks featured in cancer. As such, this research presents an integrated framework called LBSA-DRIVER, which considers the effects of genomic aberrations on a transcriptional (biological network) and a modified bio-inspired algorithm known as Artificial Rabbits Optimization (ARO) algorithm to identify potential cancer driver genes. The research finds similar grounds as some studies conducted by other researchers in the field which we mention below.

DriverNet was an approach developed in 2012 with the main purpose of using a computational framework to identify driver mutations that are found in cancer transcriptional networks (Bashashati et al., 2012). This study implemented a model that made use of a bipartite graph as well as a biological interaction network. The bipartite graph in this research was developed such that the mutations on one layer of the graph represented gene mutations and those on the other layer represented outlier mutations. The proposed algorithm in this research implemented was a greedy algorithm that used the minimum set cover problem. That is, patient-gene events were considered elements of the main universe and every mutation represented a set that contains the patient-gene events linked to the mutation. The original datasets in this research were permuted 500 times, and then the algorithm was then implemented on random datasets to test the significance of the algorithm. As a result, the proposed DriverNet algorithm in this research identified several genes which were identified by the Cancer Gene Census (CGC) (Futreal et al., 2004) was also discovered here including MDM2, BCR, ERBB2, DDIT3, FGFR1, BRCA2, MET, and PDGFRA.

Malebary et al. addressed the problem of having efficient tools for identifying drivers on large datasets as most of the developed methods were frequency-based. The model proposed in this study was designed to identify driver genes from passenger genes based on sequenced data, considering the significance of the genes. The model was developed in various steps as follows: first, a benchmark dataset was obtained for training and testing; an organic arrangement test was defined concerning the problem at hand; third, the algorithm for prediction was then developed, and then validation is carried out on results after which a framework was provided for public use. For effectiveness and evaluation, different validation methods were then applied to this model. The validation methods assisted in obtaining metrics, such as Matthew's correlation coefficient, accuracy, specificity, and sensitivity. With further experiments, results showed that the model's self-consistency ratio was 91.08%, the independent set ratio was 87.26%, and the cross-validation ratio was 92.48% (Malebary and Khan, 2021).

Han et al. proposed a model called DriverML. This model aimed to identify cancer-driver genes that are truly involved in cancer development. This model was based on an integration of Supervised Machine Learning and Rao's score-testing methods. The score test consisted of weight parameters, which were a quantitative representation of the functional impact of the related mutations on related proteins. The work was performed on 31 datasets all of which were obtained from the Cancer Genome Atlas (TCGA) and a comparison was done with 20 other existing methods. The DriverML model made use of more advanced methods and materials, including the use of a score statistic for multiple mutation type integration, the use of an algorithm of machine learning (ML) that quantitatively analyzed the functional consequences of somatic mutation, and, finally, simulated a process known as the Poisson process of somatic mutations to get an estimate distribution for the score statistic. DriverML proved robust and efficient compared to other existing methods, as an average of 87% of samples were discovered to carry somatic mutations in at least 1 driver gene (Han et al., 2019).

In 2018, Bailey et al. performed a comprehensive driver gene characterization. This was a study that systematically analyzed PanCancer and PanSoftware, covering over 9423 tumors obtained from TCGA and performing a catalog of driver genes and mutations using 26 computational tools. In experiments, 299 driver genes related to their anatomical site were identified, and over 3400 putative driver mutations were also identified through structure and sequence-based analysis. According to the authors, this research served as the most

comprehensive identification of cancer mutations and cancer genes up until 2018 (Bailey et al., 2018).

Erten et al. implemented a model based on network betweenness and random-walk, whereby a protein-protein interaction network was integrated with genomic data to identify cancer driver genes. This proposed model, known as BetweenNet, identified dysfunctional genes, also known as outlier genes, by implementing a method that measured the betweenness centrality on patient-specific networks. This model then prioritized driver genes by implementing random walks on a bipartite graph developed by creating a relationship between mutations and outlier genes. In this work, three large-scale datasets were used. They included the Breast Cancer dataset, Lung Cancer dataset, and Pan-Cancer datasets. Experimental analysis of these data showed that the proposed model performed quite well compared to its alternative methods where BetweenNet performs better than the other methods with margins when the number of genes N is between 100 and 300 (Erten Houdjedj, and Kazan, 2021).

Song et al. identified cancer driver mutations involving gene tissue-specific expression, dysregulated expression, and gene-gene network. This research took into consideration the fact that driver genes impose a relative effect on the expression data of their downstream genes. This is because through possible interaction, they are prone to the formation of functional modules that can be expressed with similar properties for the same tissues. The proposed model in this research was called DyTiDriver. DyTidriver made use of integrating tissue-specific expression, gene-dysregulated expression, and variation frequencies with a human functional interaction network to be able to discover cancer-related driver genes. Experimental analysis and evaluation of this model showed that the model outperformed good state-of-the-art models when considering Precision, F-score, and Recall values. Furthermore, enrichment analysis on the model proved its ability to also discover some unknown driver genes. With the Gene Expression Omnibus (GEO) database being one of the largest expression data platforms, this study made use of this database to obtain tissue-specific profiles. The main limitation here was found in the used datasets. The datasets used in this study were obtained from different projects, including the TCGA and GEO, which may have consisted of ambiguous data, given that heterogeneity exists amongst different patients (Song, Peng, Wang, and Wang, 2019a).

Another research in 2020 by Akhavansafar and Teimourpour aimed to discover cancer-driver genes using a network-based approach method in combination with the relative gene importance during absorption and propagation of gene anomalies in a network. The proposed model here made use of the Katz measure to calculate a node's relative effect in a network. In this research, Katz measured relative effect by calculating all the edges that existed between any given two nodes. It used the direct neighbor principle whereby the count of directly connected neighbors between the node and other nodes in the network, and nodes in different networks, is measured. The proposed approach in this research was called KatzDriver. This algorithm was developed using gene interaction weights and gene expression data to modify the original Katz algorithm. This resulted in two scores for each gene. One of the scores represented the interaction-based biological impact of the gene on the regulatory network and the other score represented the impact of the gene based on outgoing interactions. Comparison of KatzDriver with other network-based models showed its outstanding performance as compared to the other methods by discovering up to 22 unique genes that were not discovered by the other methods (Akhavansafar and Teimourpour, 2020).

In a study by Song et al., an integration of subcellular localization information, variation frequency, and the interaction experienced with neighboring dysregulated genes, was implemented to predict the accuracy of driver genes. The research put more focus on transcriptional expression information and somatic mutation information for Lung, Prostate, and Breast cancer. The model here, first of all, developed a bipartite graph that was weighted, after which a random walk method was then applied to calculate mutated gene scores for every gene present on the bipartite graph. In experiments, the model was able to discover 110 samples from breast cancer, 58 samples from lung cancer, and 52 samples from prostate cancer that contain both gene expression and normal information. It is worth noting that, unlike other methods, the approach presented in this study hypothesizes that mutation of driver genes occurs in variable samples and has a more significant impact on downstream genes (Song, Peng and Wang, 2019b).

In 2021, a method for detecting breast cancer driver genes was developed by (Chaudhary, Pham, Le, and Robinson, 2021) using an important-based network analysis framework. The work used the ideology that cancer driver mutations have a significant function in the formation of cancer community structures, and identifying and taking out these cancer driver genes considerably affects the formation of these networks. The proposed model in the study

was implemented in three general steps. In the first step, a network was built by integrating a gene mutation network with gene expression data. In the second step, this network was used in the estimation of possible community structures that were found in the network. Finally, in the third step, the importance of every node in the network was calculated using a centrality-based metric. Application of this model was made on real cancer datasets and epithelial-mesenchymal transition drivers, and results showed that it was able to predict up to 265 miRNA drivers and 8coding-13miRNA drivers, respectively.

In the race of researchers in bioinformatics to find good and accurate methods for discovering cancer-driver genes, as well as new techniques for discovering the smallest number of driver nodes in complex networks, these have been derived through the exploitation of structure-based control principles. However, due to the existence of unknown variables and network dynamics, these structure-based techniques and methods could be directly implemented in discovering personalized driver genes. With that respect, this study proposed a model known as the Personalized Network Control model (PNC) (Guo et al., 2019) that used the structure-based principle with individual patients' genetic data having the aim of identifying personalized driver genes. This study was completed in different steps. First, the personalized state transition network was developed using a sample single-paired network construction method. Next, using the Feedback Vertex Sets-based control perspective, a novel structure-based network model was designed. Using the developed model, experiments were then carried out using 13 cancer datasets obtained from The Cancer Genome Atlas (TCGA). Results indicated that the PNC model could explore personalized driver genes even when there existed hidden factors in mutation and transcriptional profiles.

Zhang et al. developed a novel approach called IMCDriver which aimed at identifying driver genes found in individual and cohort patients. This approach was generally based on the similarity principle whereby multi-omics data, for example, the gene expression, somatic mutations, and a biological interaction network (such as protein-protein interaction network or gene-gene interaction network) were used alongside well-established driver gene information to calculate a similarity between genes. The IMCDriver model then uses Inductive Matrix completion to rank driver genes with respect to how much they resemble canonical driver genes. The driver genes ranked best were then extracted as personalized driver genes. The experiments in this study were performed on five cancer datasets extracted from the Cancer Genome Consortium (CGC) (Zhang, Zhang, and Li, 2021).

Another study in 2018 aimed at discovering mutated driver genes using a matrix factorization framework which is co-regularized with prior information from an interaction network and Mrna expression patterns (Xi, Wang, and Li, Discovering mutated driver genes through a robust and sparse co-regularized matrix factorization framework with prior information from mRNA expression patterns and interaction network, 2018). The authors of this study incorporated both gene mutation Mrna expression data and a biological interaction network in a sparse and robust co-regularized non-negative matrix factorization method to identify cancer driver genes found in mutation data. The model proposed in this research used a combination of WELL-known methods, such as the Frobenius norm regularization method for solving the problem of over-fitting or the Sparsity inducing penalty technique, which is used in getting sparse scores in gene representation. The genes associated with the highest sparse scores were considered cancer-driver genes. Experiments were then carried out on three benchmarking datasets from the cancer genome atlas. These were the breast cancer dataset, glioblastoma multiforme dataset, and rectal cancer dataset. These datasets were chosen because they possessed a relatively large number of benchmarking genes that could be used for evaluation. When the proposed method in this research was applied to the Glioblastoma Multiforme dataset and then the evaluation by the CGC gene list was performed, this model realized an Area under the Curve (AUC) of 28.7% as compared to the 13.7% for “Only Network,” 7.0% for “No Prior” and 17.3% for “Only expression.”

A novel approach called EntroRank, developed in 2019 by Song et al., performed driver gene identification through the integration of mutual exclusive frequency and sub-cellular localization information in the network. The EntroRank model was implemented as follows: First, the modularity of driver genes was used to cluster the mutated genes into different subgroups with respect to their locations. Every gene in the subgroups was then obtained and its indispensability was calculated using the structural entropy of the gene. Here, relative entropy played an important role as it was used to calculate the degree of gene mutual exclusivity existing between two genes concerning each mutated gene’s variation frequency. This was a calculation carried out based on the property of mutual exclusivity between driver genes in modules. The experiments were performed on three different cancers, including lung cancer, prostate adenocarcinoma, and Breast cancer. For breast cancer, seven genes were identified and generated an exclusive ratio of 0.798; for lung cancer, eight genes were identified and generated a ratio for exclusivity equal to 0.746753247; prostate cancer results identified 8 known driver genes out of those ranked in the top 30 by the proposed model in

this research. The main advantage of this model was that it could detect cancer driver genes that have a mutual exclusive property (Song, Peng, and Wang, 2019c).

(Chen, et al., 2022) in their study, aimed to detect driver genes through an integrative framework that used complementary genomic information of individuals. This framework used gene expression data, biological interaction network data, variations in somatic mutations, and a method of Bayesian network integration which is unsupervised to perform cancer driver gene discovery and also to find estimation for the disease propagation at the cohort level and/or the patient level. This was performed in three steps. First, structural information was captured, then a functional gene mutation network was constructed from the derived structural information, and finally, driver genes and driver genes controlled downstream modules, were then obtained using the minimum cover subset method.

A novel study that used molecular data integration to discover potential cancer-driving genes and which involved the integration of somatic mutations, DNA copy number variations, and gene expression data was proposed by (Zhang and Wang, 2020). This method worked by integrating the iKEEG method whereby key cancer genes were identified and these genes were considered potential driver genes depending on their changes in expression levels. The process could be generally separated into two levels. The first level, which was data processing, involves the integration of DNA copy number variation and mutation data, then mapping them to an interaction network. This was incorporated to create a mutation matrix. In the second level, key drivers were obtained using the iKEEG method, after which expression data belonging to the key genes was used in combination with the mutation matrix to construct a connection matrix. Experiments in this research were conducted on Breast Cancer data and Glioblastoma Multiforme data.

Tang et al. in their research aimed to effectively identify driver genes using a gene mutation effect function and a semi-local centrality measure. These methods were used to assess how gene mutations affect changes in the patterns of gene expression. To achieve this, a mutation score for each gene was calculated, differentially expressed genes were identified (through comparison of normal samples with samples of expression profiles), and then used in combination with mutation genes to construct a local network concerning a protein-protein interaction network. An objective function was then used to calculate the score of each mutated gene for which the genes with the highest scores were ranked higher and considered driver genes. The four datasets used in the study were GBM, BLCA, PRAD, and OVARIAN

cancer datasets. Results of experiments with OVARIAN cancer confirmed that the proposed method was able to identify rare driver genes in the top 20 ranks; for example, the EP300, EGFR and the CREBBP, which have been found by other methods, happened to be ranked higher by this method. For BLCA 18 rare genes were identified, of which 12 of these were labelled by the CGC (Tang et al., 2021).

Bokhari et al. developed a method known as QuaDMutNetEx. This method aimed to identify driver genes with low frequency using mutual exclusivity patterns of driver mutations and their connections in a human protein-protein interaction network. This is because existing tools often consider genes with high frequencies as cancer driver genes without taking into consideration other factors. This method could identify driver genes with low frequencies by relying on mutual exclusivity rather than totally on mutation frequency. The experiments were carried out on the same datasets used by the Dendrix and DriverNet algorithms. These were the Triple Negative (TN) breast cancer, the Glioblastoma multiforme (GBM) dataset, the High-grade Serous Ovarian Cancer (HGS), and the METABRIC breast cancer datasets. According to experimental results, this method performed effectively in detecting cancer-driver genes (Bokhari, Alhareeri, and Arodz, 2020).

Using an integrative genomic approach in a non-parametric Bayesian framework, Hai et al. made use of complementary information that had been systematically encoded in omics data to discover cancer driver genes. This was carried out through the integration of DNA copy number alterations, information on protein abundance, gene expressions and mutations, and DNA methylation. The proposed approach in this study discovers gene aberration clusters, with which it creates a background distribution used for the calibration of the driver genes' confidence value. It is important to note that this was conducted using multi-dimensional genomic data. Experiments in this study were carried out on TCGA cancer datasets highly enriched with the famous known canonical drivers, such as Breast Cancer (with $P < 3{,}40 \times 10^{-36}$). Using the systematic evaluation, up to 45 candidate driver genes were identified across the different four datasets, which were not previously identified as canonical driver genes (Hai, Qiang, Xue, Hushan, and Bingshan, 2017).

Michael et al. in their research, analyze somatic point mutations through about 5,000 tumor pair types found in 21 different types of cancer, with the main aim of identifying cancer driver genes. Here, a Cancer5000 set was identified, with this set containing up to 254 genes as well as a stringent set (Cancer5000-S) containing up to 219 genes. After the elimination

of genes reported in recent papers and those reported by CGC, up to 33 genes with robust functional connections to cancer were identified, with 21 of them found in the stringent set and 12 found in the larger Cancer5000 set. The discovered genes were considered important as they were found to be involved in the most important biological processes, such as genome stability, apoptosis, cell proliferation, chromatin regulation, RNA processing, and immune evasion (Michael et al., 2014).

Another study that integrated omics data with a biological network, such as a protein interaction network, to identify cancer driver genes was the study conducted by (Zhang T and Zhang D, 2017), where a computational technique was implemented. This technique was based on the effect of potential driver genes on their neighbors in a network. The proposed approach was proven to be resistant to noise and worked efficiently with three types of cancer. These three datasets were Head and Neck Squamous Cell Carcinoma, Thyroid Carcinoma, and Kidney Renal Clear Cell Carcinoma. This study was able to identify rare canonical cancer driver genes, such as PTPN6, SRC, GRB2, and MAPK1.

Liu et al. proposed a classification method that was network-based, for discovering cancer-driver genes. It achieved this by making use of multi-biological information. First, using the human protein-protein interaction network, a cancer-specific genetic network was created with the aim of combining important features and biological information, such as gene expressions of the separate genes and their mutation frequency. The proposed algorithm consistently achieved high prediction accuracy across seven different cancer types, thus outperforming other available methods. During the analysis of the predicted results, about 40% of the top candidate genes were similar to that of the Cancer Genome Census database. Interestingly, trait comparisons showed that the network-based traits were more important than biological traits, such as the mutation frequencies and the genetic differential expression (Liu, Dai, Yu , and Zhang, 2022).

MUFFINN, a novel method for discovering cancer genes was proposed by Cho et al., whereby prediction sensitivity is accurately improved using retrieved cancer genes that have a low mutation frequency. These cancer genes with low frequencies are obtained with respect to their association with cancer genes that have high frequencies in other networks. In comparison to other state-of-the-art frequency-based approaches, such as the MutSigCV technique, analysis of results on MUFFINN expressed a ROC with higher specificity and sensitivity. Notwithstanding, MUFFINN was still able to retrieve some cancer genes with a

high likelihood of being False Positives, such as TTN (which tends to accumulate so many mutations due to its large size). A network-centric analysis was performed in this research, and the normalization of mutation frequencies did not make better the prediction of known cancer genes, especially in situations where the mutational occurrence of all neighbors is used. This gave rise to the assumption that the mutation frequency perhaps was less heterogeneous in cancer pathways with respect to cancer genes. One of the good features of MUFFINN was that it had a high predictive power of cancer genes while taking samples of only a few patients (Cho et al., 2016).

Hsiehchen and Hsieh discovered that there existed a significant positive correlation that is significant between identified cancer driver genes and the sample size of over 33 TCGA-sequenced cancers. This will be normal if more sequencing leads to more identification of cancer genes, but that is not the case given the high decline rate in the discovery of driver genes with larger samples. Hence, these researchers provided an analysis on cancer type discovery relative to the effort exerted on sequencing. It was discovered that the methods of sequencing large datasets were becoming inefficient and that in the future researchers would apply their knowledge just for the mere sake of sequencing rather than striving to obtain meaningful results (Hsiehchen and Hsieh, 2018).

Yang et al. proposed a method based on deep supervised learning to identify cancer driver genes. In this method, a novel transformer network that uses an integration of associations between omics data obtained from cancer driver genes and other multi-omics data was implemented with the deep supervised learning approach. The approach was capable of identifying up to 185 cancer driver genes from analysis with a Pan-Cancer dataset, with 55% of these genes identified by the CGC and 51% of them identified by the TCGA. As a result, the proposed Trans-Driver approach outrightly reported 563 driver genes from 33 cancer types. The added advantage of the method used in this research was that it was able to identify cancer driver genes that were quite hard to discover, if only mutation data were considered (Yang et al., 2022).

Zeng et al. used deep learning for driver gene discovery in 2021. In this study, a Convolutional Neural Network (CNN) model was used for raw sequencing of tumor DNA (Zeng et al., 2021). A deep learning model was used in this study because these models were discovered to be more efficient in learning intricate patterns obtained from raw data as compared to the conventional models (Mamoshina, Vieira, Putin, and Zhavoronkov, 2016;

Mao, Yao, Pan, Luo, and Zeng, 2018; Angermueller, Pärnamaa, Parts, and Stegle, 2016). Here, the CNNs share parameters between regions to compute convolution in these regions hence permitting smooth model training on large sequences of DNA. Applications of this model were also made in DanQ (Quang and Xie, 2016), DeepBind (Alipanahi, Delong, Weirauch, and Frey, 2015), DeepCpG (Angermueller, Lee, Reik, and Stegle, 2017) and DeepSEA (Zhou and Troyanskaya, 2015).

It is a fact that meta-heuristic algorithms have become quite popular in solving optimization problems. The reason for this is that meta-heuristic algorithms are less expensive and also more efficient than the normally used numerical methods. Meta-heuristic algorithms have a random nature, which gives them an added advantage in successfully escaping local minima and exploring the entire search space. In cancer driver gene identification, some of the common meta-heuristic algorithms, such as the Genetic Algorithm and Differential Evolution, have been implemented. Although there have been many proposed algorithms, more algorithms are still being developed to solve optimization problems. The reason for this is that there does not exist an algorithm that performs best in solving all optimization problems, most algorithms are problem-specific and developed to solve specific optimization problems.

Therefore, in this study, a recently proposed metaheuristic algorithm known as the Artificial Rabbits Optimization algorithm (ARO) (Wang, Cao, Zhang, Mirjalili, and Zhao, 2022) was implemented for the discovery of cancer driver genes. This algorithm was further improved by implementing mutation, such as that implemented in the Genetic Algorithm at every iteration. The mutation step was used to help generate more optimized results. The optimization process proposed here is generally divided into the exploration step and the exploitation step. Exploration allows the algorithm to search for a new solution in the solution space found far away from the current solution while keeping the search extensive and global. Exploitation, on the other hand, aims to improve the current solution in its local neighborhood intensively. ARO is based on the mathematical modeling of the survival nature of rabbits. This nature is based on three search strategies, which will be implemented here. These are random hiding, detour foraging, and energy shrinking strategies.

3. MATERIALS AND METHODS

In this section, the proposed models of this research, the datasets, the analysis of the biological interaction network, and the different important evaluation parameters are reviewed in detail. Hence, this section is divided into three subsections. First, in section 3.1, a detailed review of the datasets used in carrying out the experiments in this research is presented. This review describes the data repository from where the datasets were obtained and goes further to give numerical statistics on each of the datasets. Furthermore, it presents a numerical statistic on the biological interaction network used in combination with the datasets. Second, in section 3.2, the first proposed model used in the research is presented. This model is called the LSBA-Driver model and is based on optimization using list-based simulated annealing. This section describes how the optimization algorithm has been used in combination with a bipartite graph and a biological interaction network, to identify cancer driver genes. Finally, in section 3.3, a second model is presented for the same purpose of identifying cancer driver genes. This model is based on an improved version of a more novel optimization algorithm known as the artificial rabbit's optimization algorithm. The improvement of this algorithm has made it more suitable for the purpose of identifying cancer driver genes, hence allowing it to produce encouraging results that are very comparable to those of state-of-the-art models. The subsections are presented as follows:

3.1. Datasets

Both models in this research were implemented on four datasets of The Cancer Genome Atlas (TCGA) datasets obtained from the (cBioPortal for Cancer Genomics, 2022). The summary of these datasets is displayed in Table 3.1 below. The first dataset was a brain cancer dataset, the Glioblastoma Multiforme (GBM), with the gene mutation data, copy number alteration (CNA), and mRNA sequencing expression data for 585 patients. The total number of genes found in these patients was up to 68,802. The second was the ovarian cancer dataset, TCGA Ovarian Serous Cystadenocarcinoma (OV), which consisted of gene mutations, copy number alteration, and mRNA sequencing expression data for 585 patients. The total number of genes found in these patients was up to 53,204. The third dataset was the TCGA Prostate Adenocarcinoma (PRAD) dataset, which contained gene mutations, gene expression data, and copy number alteration data for 334 patients. The total number of genes found for this dataset was 34,192. Finally, the fourth dataset was the the TCGA Breast

Cancer (BRCA) dataset, which contained gene mutations data, CAN, and mRNA-seq expression data for 112 patients. The number of genes found for this dataset was 17,272. The influence graph (biological interaction network) was a gene-gene interaction network obtained from REACTOME consisting of over 518,302 genes.

Table 3.1. Details of TCGA datasets obtained from the cBioPortal.

Dataset	No of Patients	No of genes	Mutations	CNA	RNA-Seq
PRAD	334	34,192	333	333	290
OV	585	53,204	523	572	300
GBM	585	68,802	397	575	160
BRCA	112	17,272	112	112	112

3.2. LBSA-Driver Model

The LBSA-Driver model is based on list-based simulated annealing, which is an improved version of the original simulated annealing algorithm. Therefore, in this section, a brief introduction to the original simulated annealing algorithm, as well as its improved list-based version, is discussed. After which the proposed model is then be explained in detail.

3.2.1 Simulated annealing (SA) algorithm

It was important to discuss the SA algorithm here because it serves as a base for the technique implemented in this research. The simulated annealing approach is one of the first optimization algorithms designed to avoid local minima in optimization problems. It is a basic upgrade to the Hill-Climb algorithm, which faces a major challenge in avoiding local minima and maxima (Talbi and Muntean, 1993). Simulated Annealing is used to solve bound-constrained and unconstrained optimization problems. This approach is based on the idea of heating an object which high temperatures and then slowly reducing the temperatures using a cooling procedure to avoid defects in the output product. SA randomly generates a new solution after every iteration of the algorithm. The distance between this newly generated solution and the current solution is then calculated concerning a probability distribution directly proportional to the temperature in question. The SA algorithm acknowledges all new solutions that are capable of setting the objective function to a lower

value and also accepts solutions that set the objective function to a higher value, but only on a certain probability. This is done to avoid the algorithm getting stuck on solutions of local minima. With this, a technique is then applied to reduce the temperature systematically. The general work logic of the simulated annealing algorithm is sometimes to accept new solutions that are worse than the previous solution depending on some given condition to dodge local minima. The acceptance probability for this new solution decreases with every iteration with respect to a parameter known as the Temperature, T. In the acceptance phase, the new solution is accepted based on the Metropolis acceptance criterion (Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller, 1953). These criteria define how a thermodynamic system changes from state to state. A new solution, y, is accepted as the current solution with respect to the previous solution, x, based on an acceptance probability, p, given by Equation 3.1.

$$p = \begin{cases} 1, & if\ f(y) \leq f(x) \\ e^{-(f(y)-f(x))/t}, & otherwise \end{cases} \tag{3.1}$$

This algorithm functions in a few simple steps: First, a random trial point is generated by the algorithm, and then, depending on the current temperature and a probability distribution, the distance from the trial point it then calculated. Next, the algorithm uncovers whether or not the new solution is worse or better than the current solution. If it is better than the current solution, then the new solution becomes the current solution. If not, then concerning a certain acceptance function the new solution still has the opportunity to get chosen as the current solution. Next, the temperature is then lowered systematically, saving the best solution at every iteration. The algorithm is then run continuously until the stopping criteria for the algorithm are met.

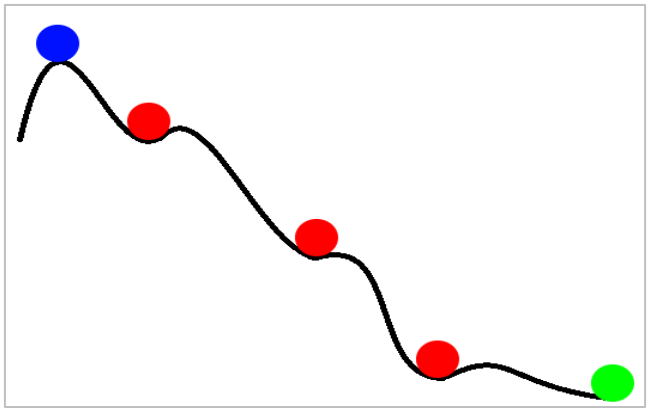


Figure 3. 1. A display of the Simulated Annealing algorithm on both local and global minima/maxima

The graph structure in Figure 3.1 is a visual representation of how the simulated annealing algorithm attempts to find global minima/maxima while avoiding local minima/maxima. The red dots on the figure represent points of local maxima, which are efficiently avoided by the use of a solution acceptance technique not used by the Hill-Climb algorithm. In the Hill-Climb algorithm, the local maxima represented by the red dots will be identified as the solutions to the optimization problem. However, since SA can efficiently avoid this, it will identify the blue dot (indicating the point of global maxima) as the correct solution to the optimization problem.

There are several parameters required for applying the Simulated Annealing algorithm to a specific problem. Just like the Temperature, a parameter known as the cooling factor must be specified just as the neighborhood structure must also be specified. This generally poses a setback to the algorithm as there is no optimal way of setting parameters to a specific problem. Generally, one way to solve the cooling factor problem is to implement an adaptive cooling strategy (Oliveira and Petraglia, 2013; Oliveira Jr and Petraglia, 2014). In practice, when the complexity of the computation of the objective function is high, the simulated annealing algorithm is likely to use a constant Markov chain length and base the termination condition on fixed iteration times.

The simulated annealing algorithm can be summarized as shown in the flow chart in Figure 3.2 above. When the algorithm starts, an initial solution, x, is randomly generated in the solution space. Next, a candidate solution, y, is generated concerning the initial solution, x, and the current neighborhood structure. If the solution, y, is better than the solution, x, then y is taken as the new solution. If y is not better than x, a probability value between 0 and 1 is randomly generated and a solution, r, is calculated to be compared with the randomly generated probability. If this solution r is less than the randomly generated probability, then x remains the current solution and moves to the next iteration. If solution r is greater than the probability value, then y is taken as the new current solution and moves to the next iteration. After the stopping criterion for the inner loop is met, the cooling technique is applied to decrease the temperature and then move to the next iteration of the outer loop. The algorithm then iterates on the outer loop until the outer loop stopping criterion is met. The stopping criterion of the outer loop is that the temperature reaches its minimum cooling value. Then, the algorithm stops and the output is extracted.

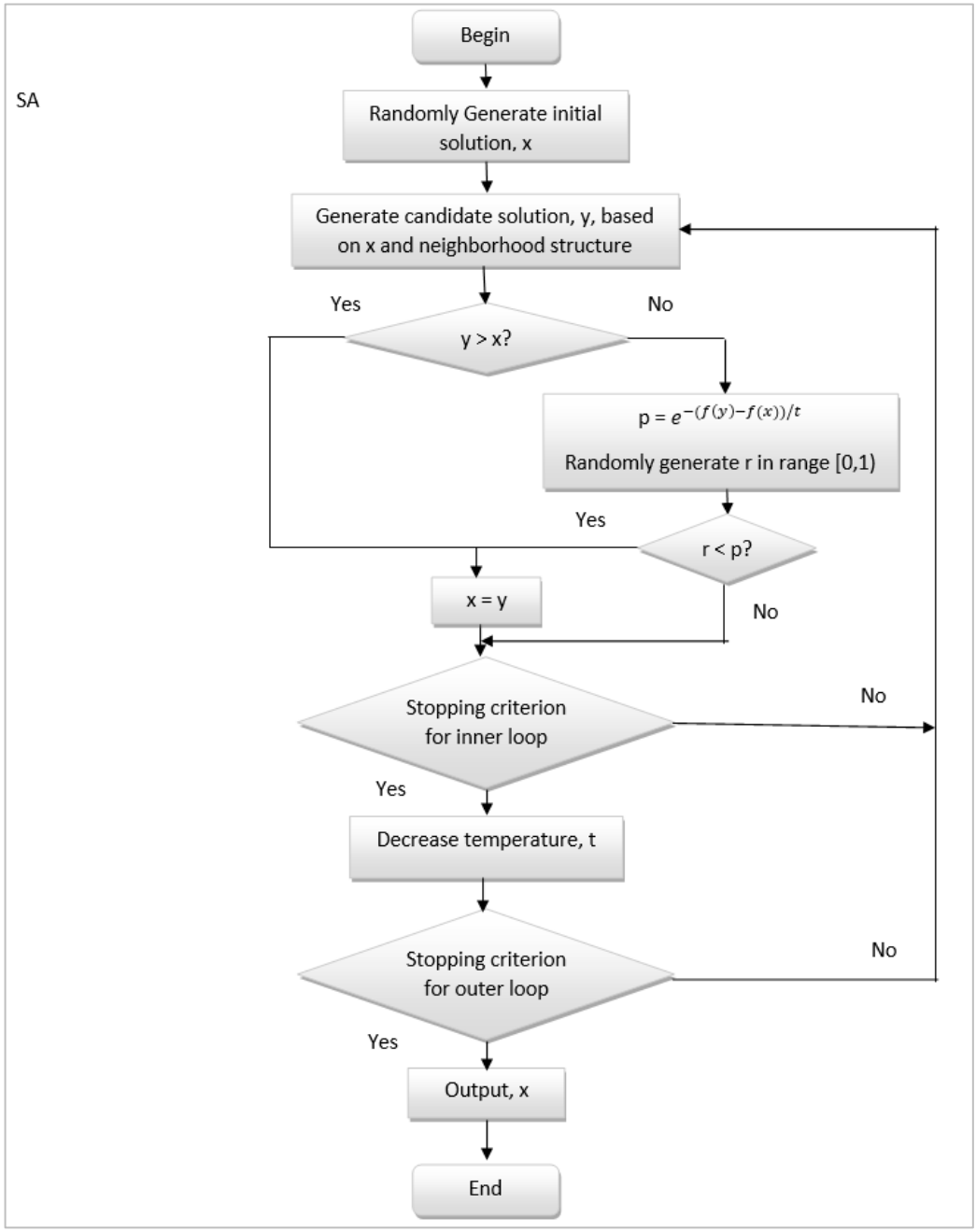


Figure 3.2. Flowchart for the Simulated Annealing (SA) algorithm.

3.2.2. The proposed LBSA model

The proposed approach in this research is called LBSA-DRIVER. The technique works, by first of all, generating a bipartite graph consisting of gene mutations and outlier expressions. This is generated such that the relationships between gene mutations and outlier expressions are defined with respect to a biological interaction network. LBSA is then implemented in

the solution to find and rank cancer driver genes. The approach is implemented in the following main steps as represented on Figure 3.3 below:

In phase (I), the datasets on which experiments will be carried out are downloaded from a well-known rich bio-database. These datasets include BRCA, GBM, OV and PRAD. A gene-gene interaction network is also used to match genes with similar biological properties. The datasets are then used in combination with the interaction network to generate the bipartite graph in the next phase.

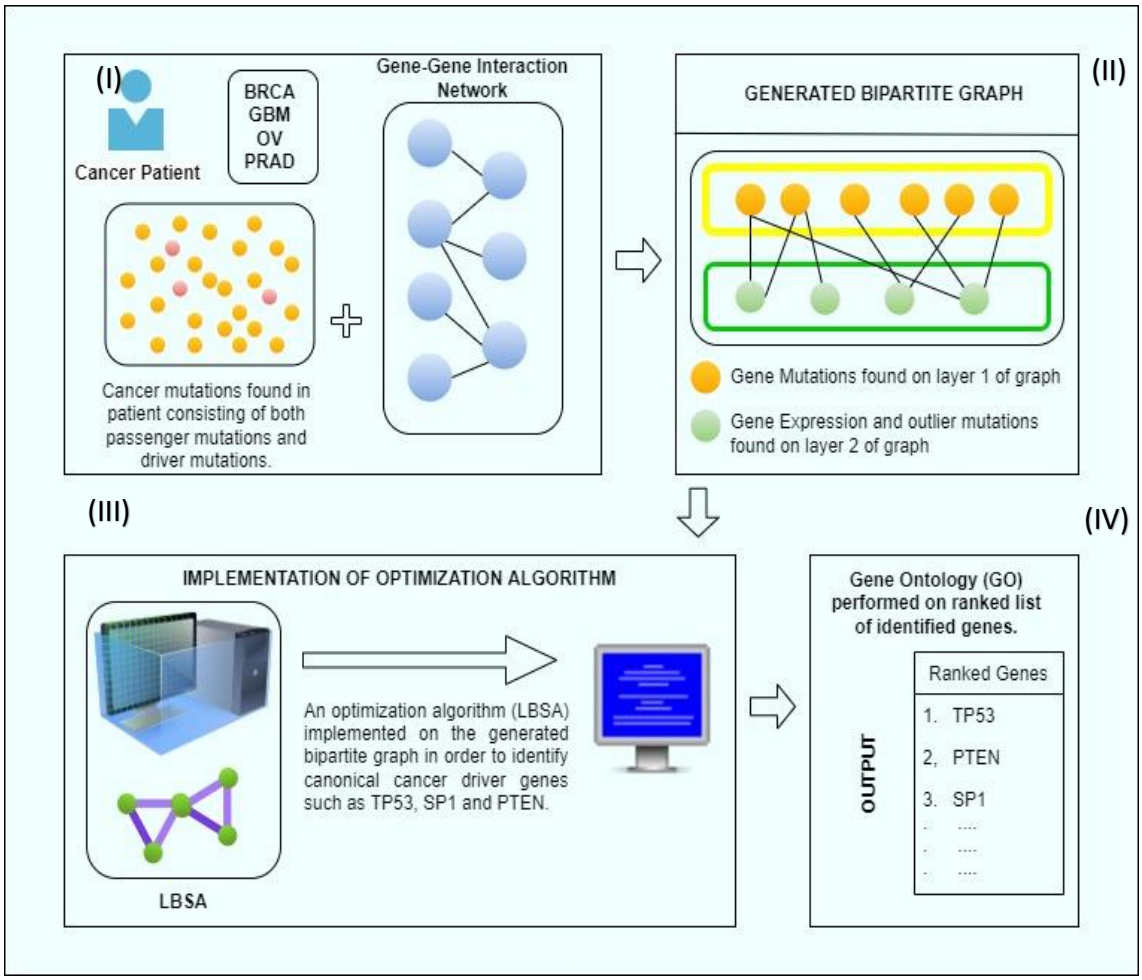


Figure 3. 3. A graphical summary of the processes involved in identifying cancer driver mutations using a biological interaction network, a bipartite graph and the LBSA optimization algorithm

Next, in phase (II), the bipartite graph is generated as follows: Let M and G represent two gene-patient matrices whereby if gene *i* of patient *j* is mutated, then *M(i,j) = 1*, and if gene *i* is not mutated, then *M(i,j) = 0*. Matrix G is an expression matrix that is transformed into an

outlier matrix G' such that, *G'(i,j) = 1* indicates that gene *i* of patient *j* is an outlier gene and *G'(i,j) = 0* otherwise. We then formulate an objective here by finding interactions on a bipartite graph consisting of points from both M and G. The optimization problem aims at finding the maximum number of connections between points on the layer 1 and layer 2 of the bipartite graph such that a layer 1 node *m* is connected to a layer 2 node *g* if *m* is a mutated and *g* is an outlier and both nodes interact with respect to the gene-gene interaction network. The objective function is created in the following manner: Suppose D represents the output of the driver mutation discovery algorithm. In D, there is a list of genes ranked by their coverage in the bipartite graph. The statistical significance of a gene, *g*, in D, along with its corresponding node coverage, COV_g, is determined by the fraction of instances where we observe driver genes having a node coverage greater than COV_g in N random runs of the algorithm. This is expressed as:

$$p - value = \frac{\sum_{i=1}^{N} \sum_{J=1}^{S} \delta[\mathrm{COV}gij > \mathrm{COV}g]}{\sum_{i=1}^{N} Si} \tag{3.2}$$

Where, S_i is the number of drivers identified in the ith run of the algorithm.

In phase (III) list-based simulated algorithm is implemented on the bipartite graph as the optimization algorithm, and the genes ranked with respect to both their frequency and their relevance in the biological interaction network.

Lastly, in phase (IV), enrichment analysis is then carried out on the ranked genes to know the importance the discovered genes have in affecting biological processes. Functional enrichment analysis is a widely used technique when identifying trends in large biological datasets, especially in the medical field to discover drug mechanisms and diseases. Here, Gene Ontology (GO) analysis is performed using the PANTHER Overrepresentation test in the Gene Ontology Resource (GOR). The latest release of GOR comprises 43648 GO terms with 7510543 annotations, 1473200 gene products, and 5183 species. Performance evaluation is also done assess the performance of the technique of which the results are compared with other state-of-the-art methods.

Generating the bipartite graph

Let M and G represent two gene-patient matrices whereby if gene *i* of patient *j* is mutated, then *M(i,j) = 1,* and if gene *i* is not mutated, then *M(i,j) = 0*. Matrix G is an expression matrix that is transformed into an outlier matrix G' such that, *G'(i,j) = 1* indicates that gene *i* of

patient *j* is an outlier gene and *G'(i,j) = 0* otherwise. We then formulate an objective here by finding interactions on a bipartite graph consisting of points from both M and G. The optimization problem aims at finding the maximum number of connections between points on the left side and right side of the bipartite graph such that a left node *m* is connected to a right node *g* if *m* is a mutated and *g* is an outlier and both nodes interact with respect to a known biological interaction network (In this case, the gene-gene interaction network obtained from REACTOME database). The process of generating the bipartite graph is demonstrated in Figure 3.4.

Biological interaction network

As a way to represent the relationship between biological entities, such as genes or proteins, biological interaction networks, are used. This is typically represented using the graph format whereby the nodes in the graph represent the biological entities while the edges on the graph are the interactions or links between these components and define how they interact with each other functionally (the graph theory). Graphs consist of very important properties that can be discovered by applying a deep analysis or walk-through on the graph. These properties can help scientists and analysts identify or discover meaningful subnetworks on the graph. There are some important definitions to take note of when discussing biological networks, including the degree of the biological network, the shortest paths in the biological network, scale-free network, transitivity, and centralities of the network. In biological networks, the degree of the network can be defined as the number of edges that are actively connected to a node. That is the number of interactions actively involved with a single biological component at the same time.

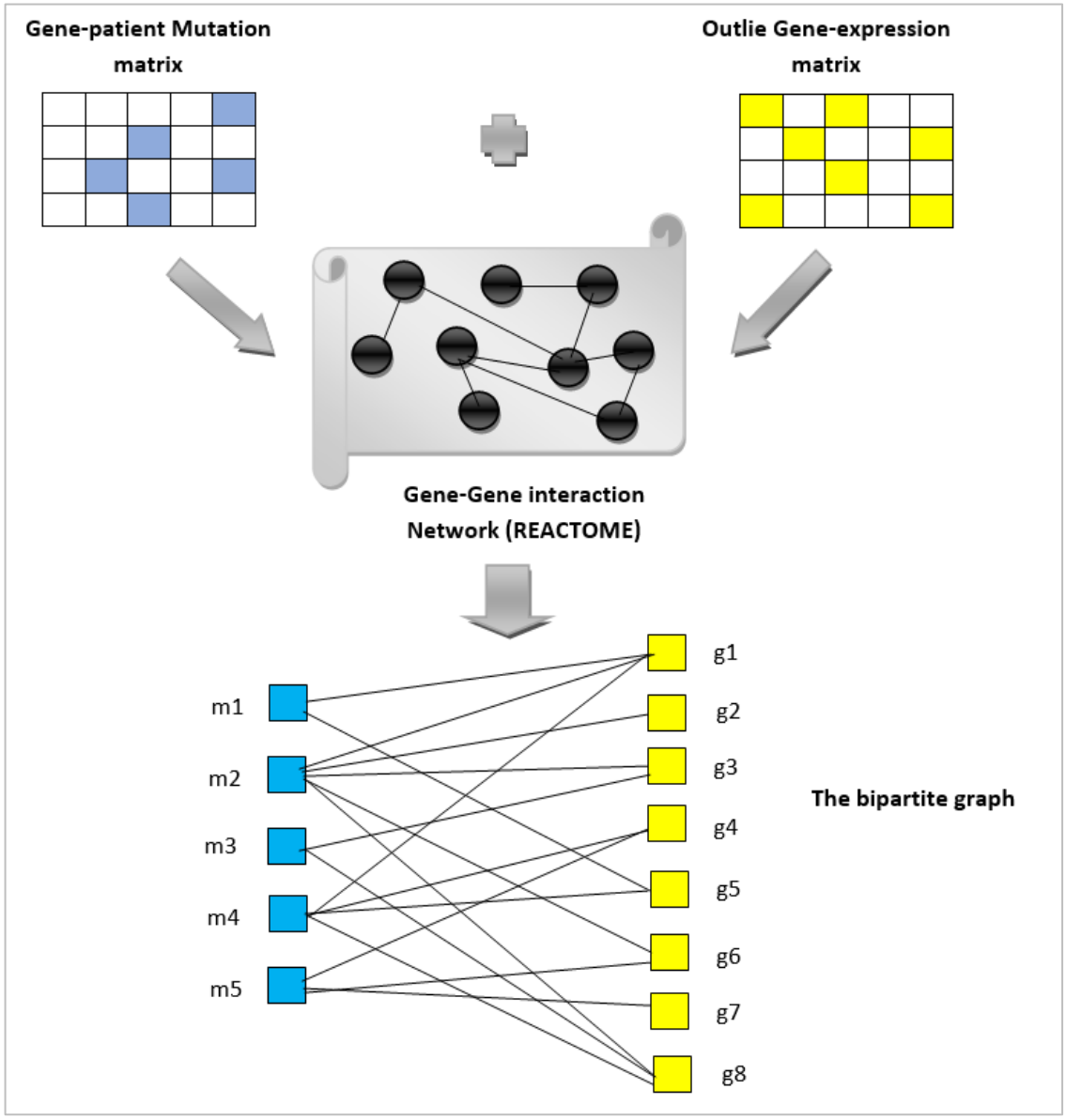
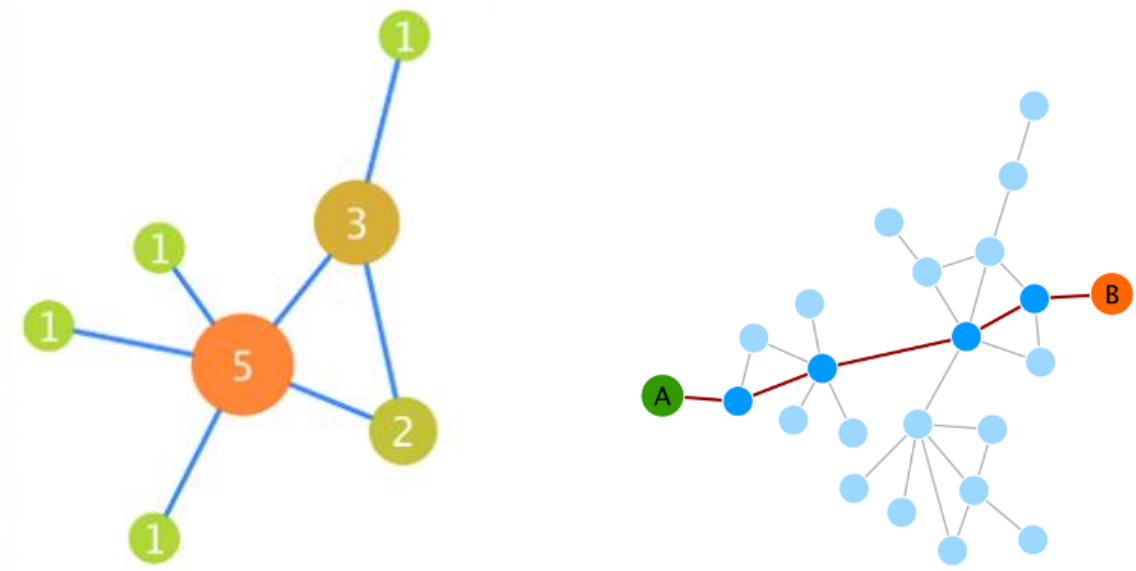


Figure 3. 4. Constructing the bipartite graph using gene mutation information and a biological interaction network.

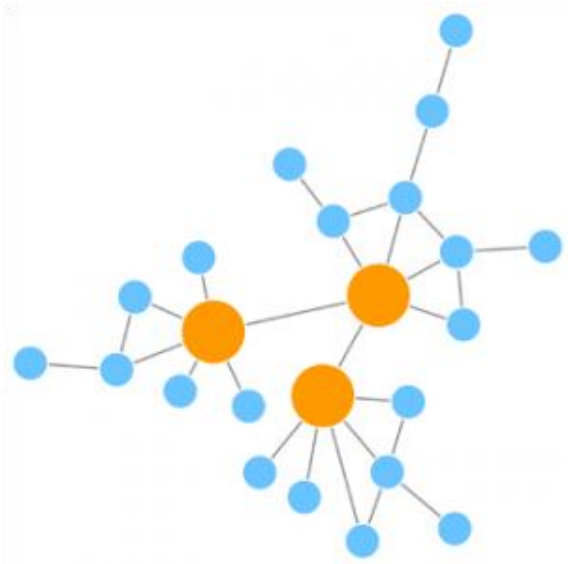
The degree of the biological network is a very important parameter because it serves as a base and influences other network parameters, such as the node centrality parameter. The node degree of a biological network can help to understand whether the network is scalable or not. Also, it is important to note that directed networks have two values for node degree. These are the incoming degree and outgoing degrees. A simple figure describing the biological network degree is shown in Figure 3.5 (a). The shortest path is another important biological network property mainly used for walking through the network. It can be defined in simple terms as "the shortest path that exists between any two points on the network, hence managing how information flows in the network." This is a very important

phenomenon in biological networks. A graphical example is displayed in Figure 3.5 (b). A scale-free biological network is a type of biological network structure whereby biological components on the network are connected to a very low number of neighbors, and at the same time, there exist few components that have high degrees, hence resulting in the existence of high connectivity to the network. Figure 3.5 (c) shows an example of a scale-free network. Transitivity is a property that represents the existence of closely connected biological components in the network known as clusters. The components found in such a network are more internally connected with each other as compared to the other network. Figure 3.5 (d) visually explains this property. Centrality, as mentioned earlier, is highly influenced by the network degree. It can be calculated for nodes as well as edges and tells how important that node or edge is on the network. There are different types of centrality measures, for example, the betweenness centrality. Centrality is shown in Figure 3.5 (e).

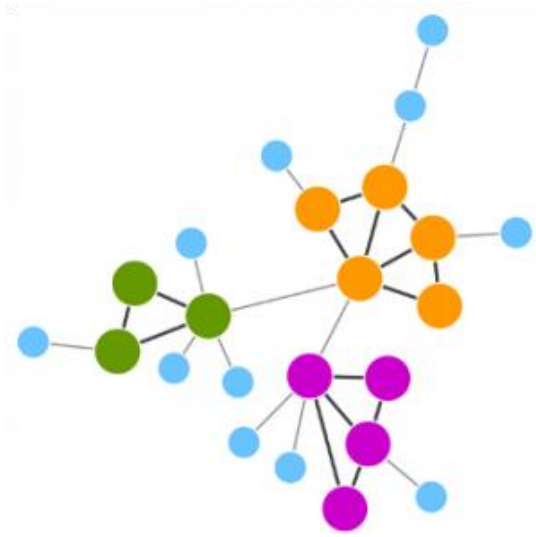


(a) (b)

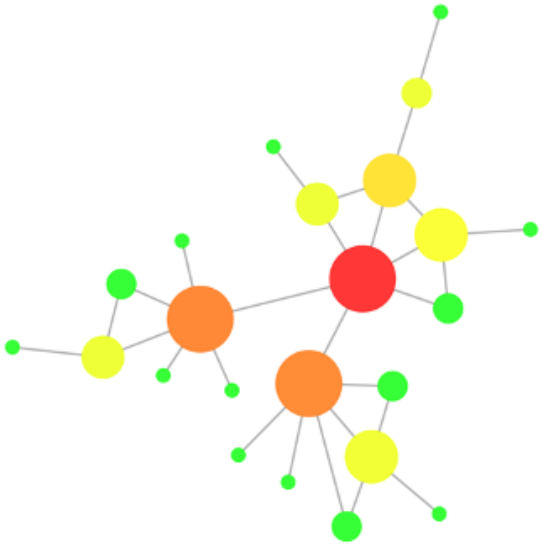
Figure 3. 5. Graphical examples of the different properties of biological networks.



(c)



(d)



(e)

Figure 3.5. (Continued) Graphical examples of the different properties of biological networks.

In (a), the red node at the center has a degree of 5 because it has 5 other nodes connected to it. (b) shows the shortest path that exists between points A and B on the network and has a cost of 5. The orange nodes on (c) represent the high-degree nodes that help connect the scale-free network to other networks. The set of purple, orange and green nodes in (d) represent the different clusters that make up transitivity. The large nodes in (e) express centrality in the network.

Biological interaction networks have played a crucial role in the new techniques that scientists have developed for identifying cancer driver genes. In the recent approaches developed for cancer driver gene discovery, these bio-networks are used as a reference graph to develop new interaction graphs whereby nodes in the new graph are considered for pairing if they show some functional similarities in the biological interaction network. There are various types of interaction networks, which could be single-layer, such as gene-gene or protein-protein interaction networks or they could be multi-layered, having interactions, such as gene-protein interactions. In this study, the biological interaction network plays a major role in creating the bipartite graph on which our model is applied. The connections between nodes on the bipartite graph are generated with respect to the roles played functionally and biologically in the biological interaction network. The network used here typically consists of connected genes (a representation of a gene-gene type biological interaction network) from the REACTOME database, hence forming a gene-gene interaction network that consists of over 259,152 gene-gene interactions (GGI). Some of the interactions found on this interaction network include TP53-SUMO2, TP53-TADA2B, UBC-SUMO2, UBC-USP22, PTEN-NF1, PTEN-GRB2, PIK3CB-INSR, SP1-INTS10 as shown on Figure 3.6 below. The different colors on this figure represent different subnetworks of genes, whereby genes on the same subnetwork share similar biological properties.

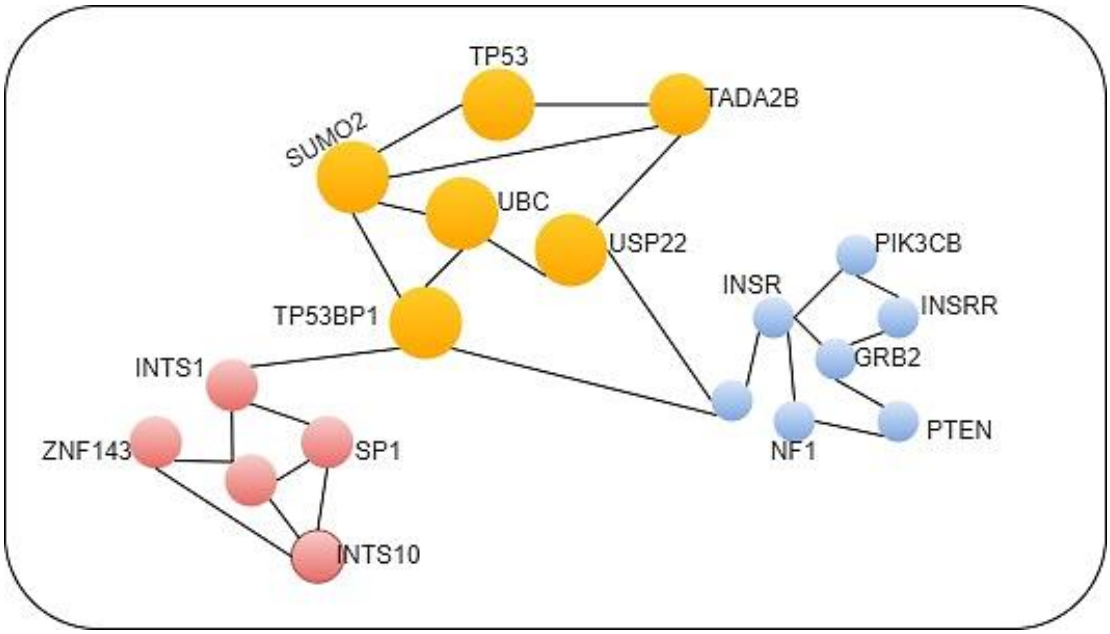


Figure 3.6. Some of the interactions found in the Reactome_FlslnGene GGI network used in this research

Applying List-based Simulated Annealing (LBSA) algorithm

List-based Simulated Annealing (LBSA) is a modified and improved version of the Simulated Annealing algorithm proposed by (Zhan, Lin, Zhang, and Zhong, 2016) in their attempt to solve the famous Traveling Salesman Problem. This method worked using a list-based type of cooling mechanism to be able to control the temperature drop rate of the SA algorithm. This method was modeled from the List-based Threshold Accepting (LBTA) algorithm (Lee, Vassiliadis, and Park, 2002; Nikolakopoulos and Sarimveis, 2007), which was a high-performance method used for solving NP-Complete optimization problems. In LBSA, the property of parameter insensitivity gives an additional advantage whereby it gives the algorithm the opportunity to be able to be applied to diverse problems without placing much effort on parameter adaptation.

In comparison with the normal SA algorithm, whereby a geometric schedule is implemented for cooling, the list-based method poses an advantage as it offers faster convergence speed and behaves better in the long term. This better performance ratio is caused by the variability in the cooling coefficient used in LBSA, as the temperature is updated in an adaptive way to fit the solution space. The LBSA consists of these main phases: definition of neighborhood structure, initializing the temperature list, and the temperature cooling procedure. The flow chart of the algorithm is shown in Figure 3.7 below. First, an initial temperature list, L, is created and the maximum temperature from the list is taken to be used in the first iteration. With this temperature, the SA algorithm is implemented to generate the new solution. In the next iteration, if the new solution is not better than the previous solution, a function is used to update the temperature list by taking out the maximum temperature value from the temperature list and reselecting a new temperature to be used for the new iterations. With this technique, the temperature cooling procedure is more dynamic and adaptive to the solution space.

There have been several improvements applied to the LBSA algorithm to try and make it better applicable. One of the latest is the Enhanced List-based Simulated Annealing (ELBSA) proposed by (Lijin, Rongying, Min, and Yiwen, 2019), whereby LBSA is upgraded with the main intention of increasing its efficiency when trying to solve very large-scale optimization problems. This improvement allows the ELBSA algorithm to be able to perform a larger number of samplings within more suitable neighborhoods and using very much better temperatures. This is performed using a Variable Markov Chain Length

(VMCL) technique. The neighborhood structures are chosen using what is known as the augmented sampling strategy. At the same time, a type of selection strategy that is systematic is implemented which guarantees that every solution has the opportunity to be improved.

The LBSA algorithm uses a temperature cooling procedure that is adaptive concerning the solution space. In the algorithm, there are two loops, there is an outer loop whose stopping criterion is given by the iteration count, *k*, and an inner loop whose stopping criterion is also given by the iteration count, m, which is given by the length of the Markov Chain, M. It is important to note that the use of the temperature list does not result in an increase or a decrease in the time complexity of the simulated annealing algorithm. LBSA works as follows: First, an initial temperature list, *L*, is generated and a maximum temperature, t_{max} is selected from the list. Next, a random neighboring solution, *y*, is created with respect to a chosen solution, *x*, and a loop counter, *m*, is initialized and incremented at every iteration. Solutions *f(y)* belonging to *y* and *f(x)* belonging to *x* are calculated and compared such that if *f(y)* is less than *f(x)*, then a probability, *p*, is calculated using *f(x)* and *f((y)*. This probability is compared with a randomly generated number, *r*, between 0 and 1. If the calculated probability is greater than the randomly generated number, *r*, then the temperature list is updated else, the original temperature list is maintained and the inner loop moves to the next iteration setting $m = m + 1$. Next, the cooling parameter is checked and if it is equal to 0, then the outer loop moves to the next iteration. If not, then the maximum value of the temperature from the temperature list is popped out of the list, and then the algorithm moves to the next iteration.

The neighborhood structure

The candidate solution is created using a greedy hybrid operator in this process. This was proposed by (Wang, Lin, Zhong, and Zhang, 2015) in 2015 in their study to solve the TSP problem using a multi-agent SA algorithm. This greedy technique uses three operators (inverse, insert, and swap) to produce three neighboring solutions, of which the best of will be selected as the candidate solution.

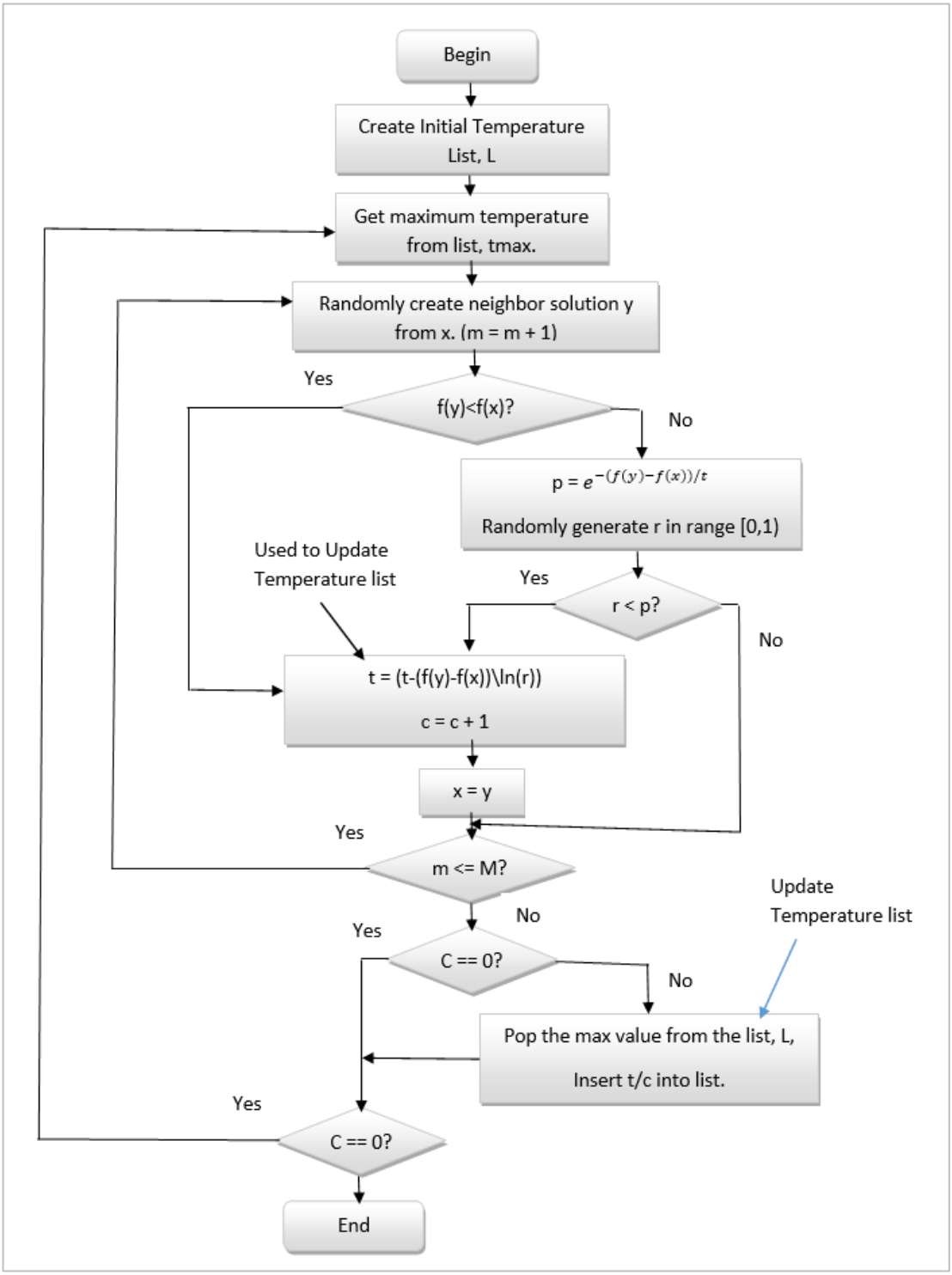


Figure 3.7. Flow chart describing the processes involved in List-based Simulated Annealing (LBSA) algorithm

For two given nodes in a population, *i* and *j*, the inverse operator will invert the positions of these nodes, creating two new nodes, the insert operator will insert node *j* into node *i* creating three new nodes and the swap operator will swap the position of *i* and that of *j* creating four new nodes. In the neighborhood structure selection analysis, there are several points that are

taken into consideration. These include effectiveness, efficiency, speed, computational effort, size, and information volume (Saed, Rebennack, and Pardalos, 2008). The effectiveness represents the ability of brute strength of the neighborhood structure in being able to cover the whole solution space. The efficiency represents the quality of the solutions when the neighborhood structure covers all of the solution space. Speed represents the total number of moves that will be used to get to any given point in the sample space. The computational effort, as the name implies, is how much computational effort is needed to complete a single move. The size represents the total number of accessible solutions in the neighborhood structure. Here, a large population is considered better because it provides a larger sample space and better accessibility. The information volume is used to be able to make more efficient moves in the neighborhood structure. It is the amount of transformed information.

The initial temperature list

Here, the LBSA algorithm produces an initial list of temperatures. Here the initial probability of acceptance, p_o, is used to generate the initial temperature list as follows: If *x* is the current solution and *y* in the new solution, and *f(x)* and *f(y)* are their corresponding objective functions, and the acceptance probability, p_o, is known, then the temperature is calculated using Equation 3.3 as follows:

$$t = \frac{-(f(y)-f(x))}{\ln(po)} \tag{3.3}$$

The flowchart for creating the initial temperature list is shown below in Figure 3.8. After the initial solution, *x* is created, an empty temperature list, *L*, is also created with a maximum length, *Lmax*. At the same time, an acceptance probability, *po*, is also defined to be used for comparing the solutions to know if to update the temperature list or not as mentioned above.

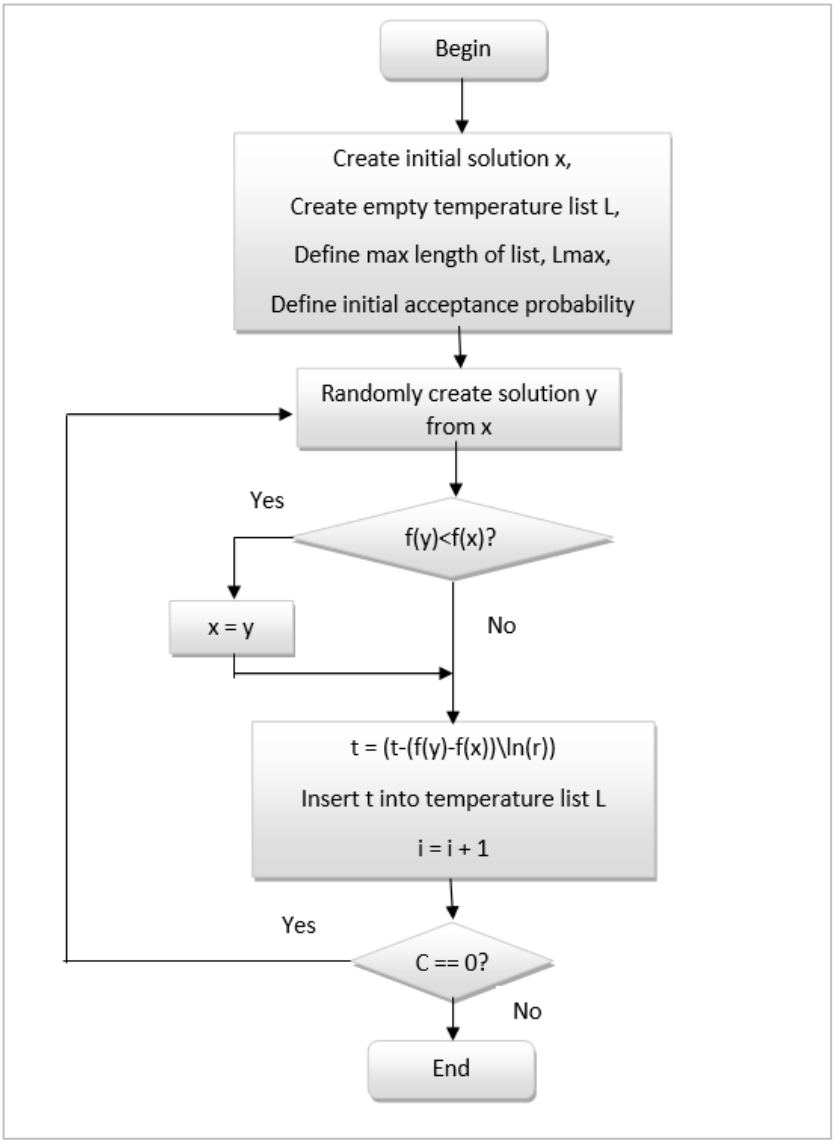


Figure 3.8. Flow chart for creating the initial temperature list for LBSA

The temperature cooling procedure

The temperature cooling procedure is implemented as follows: given the list of temperatures, *T*, the current temperature is represented by the maximum temperature, t_{max}, in the list. Given that the difference of objective function values is represented by *d*, then the acceptance probability, *p*, is given by Equation 3.4:

$$p = e^{\frac{-t}{(tmax)}} \tag{3.4}$$

For each pair of acceptance probability, *p*, and difference, *d*, a random probability value, *r*, is generated such that *r* is less than *p*. *r* is then used to calculate a new temperate as shown in Equation 3.5:

$$t = \frac{-d}{\ln(r)} \tag{3.5}$$

The cooling method of the original simulated annealing algorithm is geometric with a slower convergence speed, while that of the novel list-based simulated annealing algorithm has a quicker convergence speed as well as good long-term behavior. This is ultimately a result of using a cooling coefficient feature that is variable. This leads to the temperatures getting updated in an adaptive manner, for the structure of the given or current solution. Hence, the list-based simulated annealing algorithm can allocate more time to searching rich areas of the population in more detail. This convergence advantage of the LBSA was effectively proven by (Zhan, Lin, Zhang, and Zhong, 2016) in their research when comparing the cooling procedures of LBSA with both Geometric and Arithmetic cooling procedures.

3.3. Improved ARO Model

3.3.1. General concept

ARO is a bio-inspired algorithm derived from the strategies rabbits use to survive in nature. Rabbits are animals that feed on greens, such as leafy weeds, grass, and forbs (herbivores), and just like other evolutionary animals, they have to evolve to survive (Tůmová, Martinec, and Chodová, 2011). The survival strategy of rabbits is to eat grass which is far away from their own nests, hence preventing their nests from being discovered by predators. Given their wide arial vision, they are able to scan wide areas for food resources (Juan, 2017; Tynes, 2010). This strategy is known as “detour foraging” and will be regarded as the exploration mechanism in this study. Another strategy for survival used by rabbits is the random hiding strategy. Rabbits escape predators using burrows. Rabbits dig many burrows and randomly choose one during a chase and use it for shelter (Camp, Rachlow, Shipley, Johnson, and Bockting, 2014). Because rabbits can easily stop and change direction while being chased at high speeds, this technique has been a crucial strategy for survival. This random hiding technique is known here as exploitation. Rabbits, being on the lower level of the food chain, have a vast range of predators, meaning they have to be able to run really fast to escape danger. This affects the rabbits’ energy-wise, meaning that they have to change between random hiding and detour foraging adaptatively with respect to their energy levels.

3.3.2. Model and algorithm

In ARO, detour foraging is implemented as exploration while random hiding is implemented as exploitation. Finally, as an energy-shrinking strategy, rotation between random hiding and detour foraging is applied. The model is described as follows.

Detour Foraging (Exploration)

Detour foraging, as mentioned earlier, is a technique where the rabbits feed on food far away from their own nests. This is done by randomly selecting a location that is far away from the home location. Rabbits assume that every rabbit in the population has a nest with a given number of burrows; *d*. So, in detour foraging, the rabbits randomly update their position close to the nest of other rabbits in the population. This is demonstrated mathematically as follows:

$$\vec{v}_i(t+1) = \vec{x}_j(t) + R \cdot (\vec{x}_i(t) - \vec{x}_j(t)) + round(0.5 \cdot (0.05 + r_1)) \cdot n_1 \quad i, j = 1, \ldots, n \text{ and}$$

$$j \neq i \tag{3.6}$$

$$R = \mathrm{L} * \mathrm{c} \tag{3.7}$$

$$L = \left(e - e^{\left(\frac{t-1}{T}\right)^2}\right) * \sin(2\pi r_2) \tag{3.8}$$

$$c(k) = \begin{cases} 1 \; if \;\; k == g(l) \\ 0 \; else \end{cases} \quad \mathrm{k} = 1, \ldots, d \text{ and } l = 1, \ldots, \lceil r3 \cdot d \rceil \tag{3.9}$$

$$g = randperm(d) \tag{3.10}$$

$$n1 \sim N(0, 1) \tag{3.11}$$

In the above equations, $\vec{v}_i(t+1)$ represents the candidate position at time *t+1* of the *i*th rabbit, *n* is the population size, $\vec{x}_j(t)$ represents the position of the rabbit at time *t*, d is the problem dimension, *T* is the number of iterations, *roundperm(d)* is a function that returns a random integer between 1 and *d*, L is the running length when carrying out detour foraging, r_1, r_2 and r_3 are random numbers between 0 and 1, n_1 is the subject to standard normal deviation.

Equations 3.6-3.11 are implemented to achieve detour foraging. Equation 3.5 is used to demonstrate the random search of individuals to find a food source. This significantly contributes to exploration and gives the ARO algorithm the ability to perform global search.

Random Hiding (Exploitation)

This is the situation whereby the rabbit has to dig a couple of burrows around its nest, in which it randomly chooses and hides when it is being chased by a predator. ARO implements random hiding by generating across the search space *d* number of burrows in each iteration. At every iteration, one of the burrows is chosen for hiding to increase the probability of survival. In mathematically expressing exploitation, a hiding parameter, *H*, linearly drops from 1 to 1/T, where T is the total number of iterations (Mirjalili and Lewis, 2016). This implies that at the start of the iterations, the search space across which burrows are generated is quite large and reduces as the iteration number increases. The mathematical implementation of random hiding is demonstrated below using Equations 3.12-3.16.

$$H = \frac{T-t+1}{T} * r_4 \tag{3.12}$$

$$\vec{v}(t+1) = \vec{x}_i(t) + R * (r_4 * \vec{b}_{i,r}(t) - \vec{x}_i(t)) \quad , i=1,\ldots, n \tag{3.13}$$

$$g_\Gamma(k) = \begin{cases} 1 & if k == |r_5 \cdot d| \\ 0 & else \end{cases} \quad k = 1, \ldots., n \tag{3.14}$$

$$\vec{b}_{i,r}(t) = \vec{x}_i(t) + H * g_r * \vec{x}_i(t) \tag{3.15}$$

$$\vec{x}_i(t+1) = \begin{cases} \vec{x}_i(t) & f(\vec{x}_i(t)) \leq f(\vec{v}(t+1)) \\ \vec{v}(t+1) & f(\vec{x}_i(t)) > f(\vec{v}(t+1)) \end{cases} \tag{3.16}$$

Equations 3.6 and 3.13 denote the new candidate positions generated for each rabbit. Here, $\vec{b}_{i,r}$ is a burrow selected at random from the total *d* number of burrows. r_4 and r_5 are two randomly selected numbers between 0 and 1. If the calculated fitness score of these positions is better than that of the current position, the rabbit's position will be updated with respect to this new candidate position. Updating the position of the rabbit is shown by Equation 3.16.

Energy Shrink (Alternating between exploitation and exploration)

Rabbits tend to perform detour foraging when they have high energy levels. Later, when energy levels drop, they will switch to random hiding. This application is the same in ARO. At the start of the iterations, exploration is performed and then switches to exploitation at later stages in the iteration. This brings about an energy factor in the current rabbit used to model the alternation between random hiding and detour foraging. This energy factor is given mathematically as follows in Equation 3.17.

$$A(t) = 4(1-\frac{t}{T})\ln\frac{1}{r} \quad (3.17)$$

Where r is a random number between 0 and 1.

A large value of A(t) indicates that the rabbit has enough energy to perform exploration. On the other hand, a small value of A(t) indicates that the rabbit has less energy and has to go into hiding (exploitation). This is indicated in ARO by exploration occurs when A(t) > 1 and exploitation when A(t) ≤ 1. (Wang, Cao, Zhang, Mirjalili, and Zhao, 2022) demonstrate the behavior of the energy factor over 1000 iterations as shown on Figure 3.9 below. This shows how the value of the energy factor decreases as the number of iterations increases.

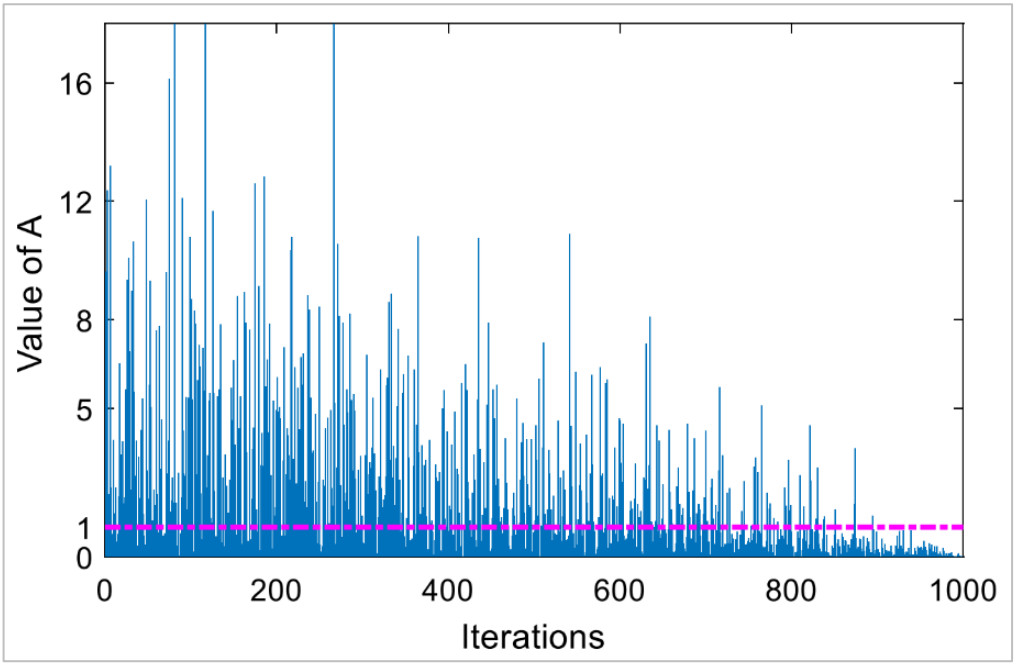


Figure 3. 9. Behavior of the energy factor, A, over 1000 iterations.

Mutation

ARO in this study is improved by implementing mutation just before detour foraging and random hiding. Mutation is implemented here in orde to variate the search space and foster evolution, hence preventing the algorithm from converging and eventually coming to a halt. To implement mutation, at every iteration, individuals in the current solution are randomly chosen and given new randomly selected positions. Exploration and exploitation are then carried out on the output solution obtained from mutation. Table 3.2 below shows the pseudocode for the proposed Improved ARO algorithm.

Table 3. 2. Pseudocode for the Improved ARO algorithm

Pseudocode for ARO

- *Begin*
- *Initialize a random solution of rabbits, X.*
- *Calculate their fitness, X_{best}.*
- *Start while (t < iteration count).*
    - *Implement mutation on the current solution.*
    - *Begin for loop.*
        - *For each solution, X, calculate the energy factor, A, using Equation 12.*
        - *If (A > 1),*
            - ✓ *Randomly choose a new rabbit from the neighborhood*
            - ✓ *Calculate R using Equations 2-6, perform detour foraging using equation 1,*
            - ✓ *Calculate the fitness and update the position of the rabbit using Equation 11.*
        - *Else If (A ≤ 1),*
            - ✓ *Generate d number of burrows and choose one of the burrows at random that will be used for hiding (Equation 10).*
            - ✓ *Use Equation 8 to perform random hiding, then calculate the fitness of the solution.*
            - ✓ *Finally, update the position of the rabbit using Equation 11.*
        - *End if.*
    - *End for a loop.*
- *End while loop.*
- *Update the solution to get the new best solution, X_{best}.*
- *End.*

The flowchart for the ARO algorithm is demonstrated in Figure 3.10.

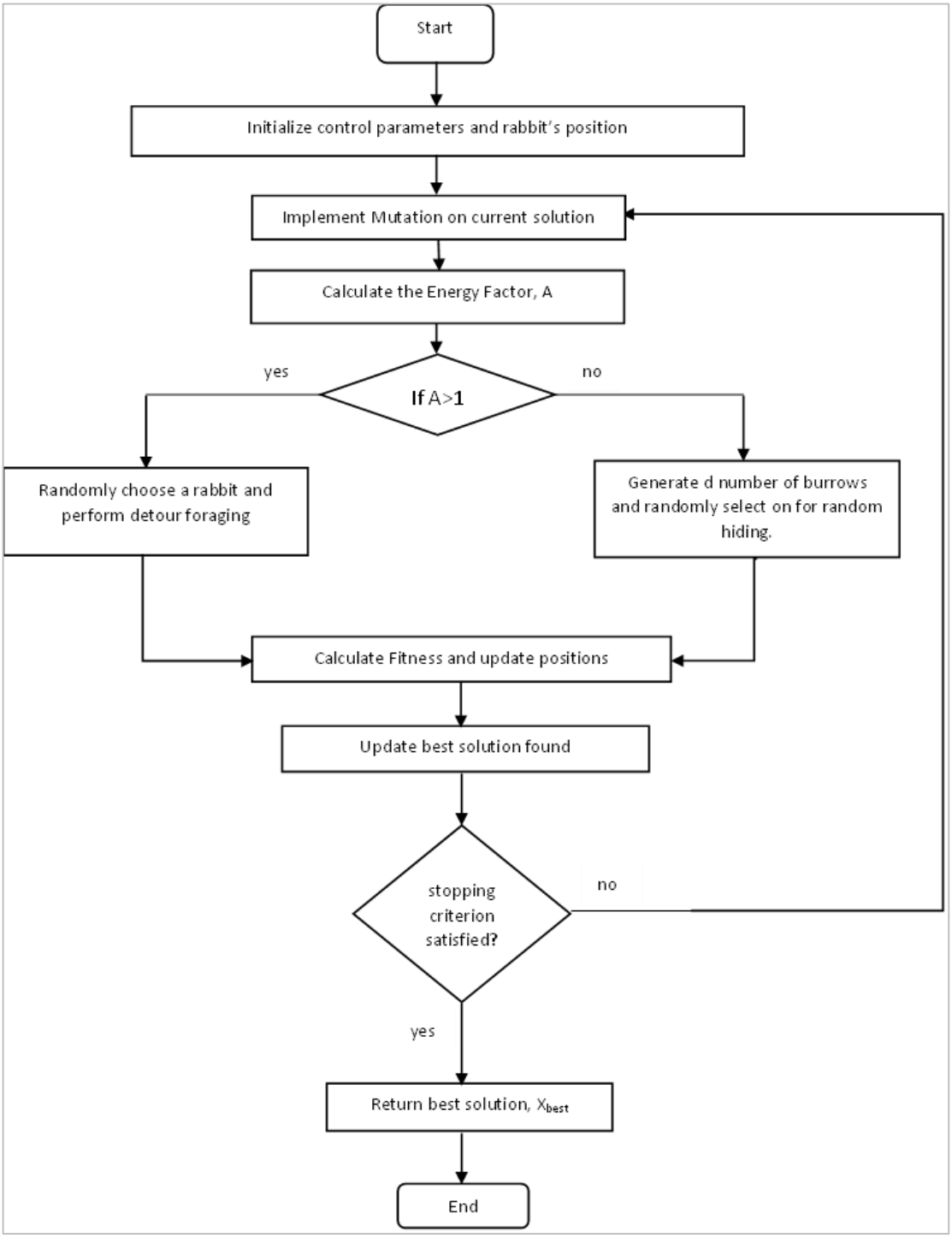


Figure 3. 10. Flow chart of Artificial Rabbits Optimization (ARO) algorithm.

4. RESULTS

In this section, the results for both proposed models are evaluated, presented, and compared with other state-of-the-art methods. The evaluation is carried out by calculating accuracy, precision, and recall. In 4.1, the results of the LBSA model will be analyzed, and in 4.2, the results of the ARO model will be analyzed.

The algorithms are evaluated by taking a random top N number of ranked genes and then calculating the precision, recall, f-score, and accuracy of the results. In the equations, true positives are represented by TP, true-negative represented by TN, false-positive represented by FP, and false-negative represented by FN. The value for TP is obtained by calculating the total number of genes discovered by the algorithm that have also been labeled as canonical driver genes by the NCG7.0. The value for TN is derived as the number of genes that have not been labeled by NCG7.0 as canonical driver genes and also have not been ranked amongst the top N genes by the algorithm (N representing different population sizes). The value for FP is derived by calculating the total number of genes identified by our algorithm but have not been labelled by the NCG7.0 as canonical driver genes. Finally, the value of FN is derived by calculating the total number of genes that have not been discovered by the algorithm but have been labeled as driver genes by the NCG7.0 database. Table 4.1 below shows the number of genes labelled as canonical genes by NCG7.0 for the different datasets used in this research. The Glioblastoma Multiforme dataset for brain cancer consists of 24 labeled canonical cancer driver genes, The Breast Carcinoma which is one of the most popular forms of cancer in the world, labelled up to 110 different canonical driver genes, Ovarian Cancer labelled 11 canonical driver genes, and Prostate Adenocarcinoma labeled 43 canonical driver genes.

Table 4.1. Number of canonical driver genes identified by the NCG7.0 for different datasets

Dataset	Number of Canonical Genes
GBM	24
BRCA	110
OV	11
PRAD	43

The formulae for calculating precision, recall, and f-score are as shown in Equations 4.1, 4.2 and 4.3, respectively, below:

$$Precision = \frac{TP}{TP+FP} \quad (4.1)$$

$$Recall = \frac{TP}{TP+FN} \quad (4.2)$$

$$F - score = \frac{Precision\ x\ Recall}{Precision+Recall} \quad (4.3)$$

4.1. Results Analysis for LBSA-DRIVER

In performing the evaluation calculations, an analysis of precision, recall, and f-score for different gene population sizes, *N*, was carried out. That is, for top *N* equal to 50, 100, 150, 200, 250, and 300. The accuracy of the algorithm was compared on two datasets, the Breast Cancer dataset and the Prostate Adenocarcinoma dataset, with the accuracy of the other four methods earlier mentioned. The results of accuracy are shown in Table 4.2. The results of precision, recall, and f-score, as shown in Table 4.3, evidently show that this algorithm performed better than the other state-of-the-art methods.

Table 4.2. Comparison of accuracy for BRCA and PRAD datasets

	LBSA_DRIVER	DRIVERNET	DAWNRANK	INT_DRIVER	SUBDYQUENCY
BRCA	0.291	0.109	0.073	0.055	0.118
PRAD	0.311	0.178	0.178	0.222	0.267
GBM	0.211	0.170	NA	NA	NA
OV	0.412	0.376	NA	NA	NA

Table 4.3. Comparison of Precision, Recall and F-Score for GBM, OV and PRAD datasets

		LBSA-DRIVER			DRIVERNET		
	N	Precision	Recall	F-score	Precision	Recall	F-score
GBM	50	0.320	0.140	0.120	0.120	0.053	0.075
	100	0.170	0.149	0.160	0.070	0.061	0.069
	150	0.133	0.175	0.153	0.046	0.061	0.065
	200	0.100	0.175	0.127	0.040	0.070	0.055
	250	0.080	0.175	0.120	0.032	0.070	0.039
	300	0.070	0.184	0.102	0.027	0.070	0.041
OV	50	0.120	0.353	0.180	0.040	0.118	0.049
	100	0.060	0.353	0.102	0.030	0.176	0.053
	150	0.040	0.353	0.071	0.020	0.176	0.036
	200	0.030	0.353	0.055	0.015	0.176	0.026
	250	0.024	0.353	0.045	0.012	0.176	0.021
	300	0.023	0.412	0.045	0.010	0.176	0.022
PRAD	50	0.200	0.167	0.188	0.100	0.083	0.093
	100	0.120	0.200	0.150	0.050	0.083	0.070
	150	0.080	0.200	0.114	0.040	0.100	0.057
	200	0.060	0.200	0.092	0.030	0.100	0.046
	250	0.048	0.200	0.077	0.024	0.100	0.040
	300	0.050	0.304	0.086	0.020	0.100	0.033
BRCA	50	0.278	0.244	0.250	0.250	0.168	0.193
	100	0.200	0.260	0.211	0.211	0.168	0.164
	150	0.125	0.266	0.149	0.100	0.197	0.126
	200	0.090	0.266	0.122	0.086	0.197	0.083
	250	0.095	0.284	0.095	0.074	0.197	0.065
	300	0.060	0.284	0.080	0.052	0.215	0.059

Figure 4.1 shows the comparison of the precision between LBSA-Driver (represented with the blue line) and DriverNet (represented with the red line). The curves are drawn with values of N, ranging from 0 to 300. It is seen clearly that LBSA-Driver dominates DriverNet with maximum precision existing within the top 50 ranked genes. That is, at N=50. Here, LBSA-Driver has a maximum precision of 0.32, while DriverNet has a maximum precision of 0.12.

Figure 4.2 is a representation of the comparison of precision between LBSA-Driver and DriverNet for the experiments carried out on the prostate adenocarcinoma dataset. LBSA-Driver proves dominant over DriveNet, having a maximum precision value of 0.2, while DriverNet has a maximum precision value of 0.1.

Figure 4.3 represents the comparison of precision for the Ovarian Cancer dataset. LBSA-Driver experiences a maximum precision value of 0.12 where N = 50, while DriverNet experiences a maximum precision value of 0.04 when N = 50.

Figure 4.4 represents the comparison of precision for the Breast Cancer dataset. Here, LBSA-Driver and DriverNet have very similar results, with LBSA-Driver only slightly dominating DriverNet. Both methods achieve high maximum precision values (0.278 for LBSA-Driver and 0.25 for DriverNet), both attained at point N = 50. As much as LBSA-Driver is dominant over DriverNet for most of the curve, at N=100, DriverNet showed to be dominant over LBSA-Driver.

Figure 4.5 represents the comparison of recall for the Glioblastoma Multiforme dataset. LBSA-Driver. The recall curve has an upward progression and is directly proportional to the value of N. Therefore, recall has a minimum value when N = 0 and a maximum value when N = 300. Here, the recall curve for LBSA-Driver is evidently dominant over that of DriverNet. LBSA-Driver has a maximum recall value of 0.184, while DriverNet has a maximum recall value of 0.070.

The recall curve for prostate adenocarcinoma, shown in Figure 4.6 above, indicates that LBSA-Driver performs better than DriverNet. The curve for LBSA-Driver experiences a steep jump in the recall value when the value of N = 50. Between N = 50 and N = 250, the recall value is more or less stable and experiences only slightly changes. Furthermore, from values of N = 250 to N = 300 the recall value experiences another quick jump. The maximum recall value for LBSA-Driver here is 0.30 while that of DriverNet is 0.10.

For ovarian cancer, the value of recall (represented in Figure 4.7 above) is more or less stable between values of N = 50 and N = 300 for LBSA-Driver. The recall increases steadily from 0.0 to 0.353 at N = 50 and then experiences a constant value at 0.353. When N = 250, the recall curve rises again from 0.353 to 0.412. In the case of DriverNet, between values of N equal to 0 and 50, the value of recall increases steadily from 0.00 to 0.118. From N equal to 50 to N equal to 100, the recall value increases at a slower pace from 0.118 to 0.176. From N equal to 100 to N equal to 300, the recall is constant.

The recall curve shown in Figure 4.8 above, for comparing LBSA-Driver and DriverNet, has similar progressions, although LBSA-Driver exhibits a larger area under the curve than DriverNet. Both algorithms rise quickly from N=0 to N=50, after which they undergo a gradual steady rise from N=50 to N=300. LBSA-Driver has a maximum recall value of 0.244, while DriverNet has a maximum recall value of 0.168.

The f-score curves for LBSA-Driver and DriverNet exhibit different progressions. As shown in Figure 4.9, for LBSA-Driver, the curve rises steadily for values between 0 and 100. And then steadily falls from N = 100 to 300. The maximum F-score value of 0.16 is experienced when N = 100. For DriverNet, the f-score curve rises steadily between N = 0 and N = 50 and then drops from there till N = 300. The maximum f-score value at N = 50 is 0.075.

In Figure 4.10, the f-score curves shown for the PRAD dataset for DriverNet and LBSA-Driver are similar, although LBSA-Driver shows a better area under the curve. Both curves rise steadily to values of N = 50 and then fall steadily to N = 300. The maximum f-score value for LBSA-Driver is 0.188, while that for DriverNet is 0.093.

For the ovarian cancer dataset, the f-score for LBSA-Driver initially spikes up to 0.18 when the value of N = 50 and falls steadily to 0.045 when N = 300. For DriverNet, the maximum value of f-score is experienced when N = 100. The maximum f-score value for DriverNet is 0.53. This is demonstrated in Figure 4.11.

In comparing the f-scores on the breast cancer dataset for LBSA-Driver and DriverNet (Figure 4.12), it is shown that the curves initially rise at a quick rate to their maximum f-score values, each attained at point N=50, after which they steadily fall back. The area under the curve shows LBSA-Driver's dominance over the DriverNet algorithm.

A - Glioblastoma Multiforme (GBM)

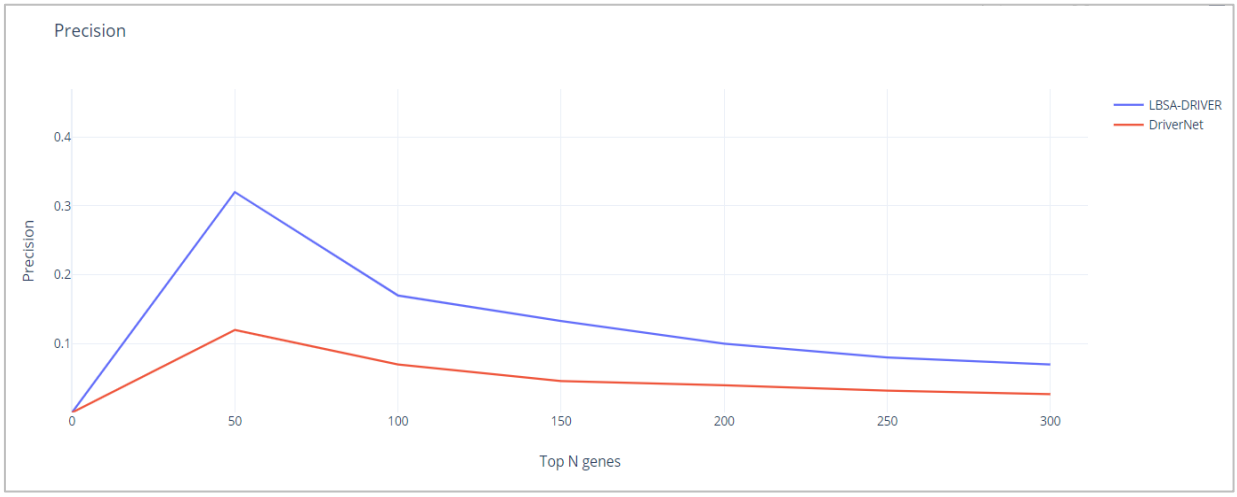


Figure 4.1. Comparing values of precision between LBSA-Driver and DriverNet for the GBM dataset

B - Prostate Adenocarcinoma (PRAD)

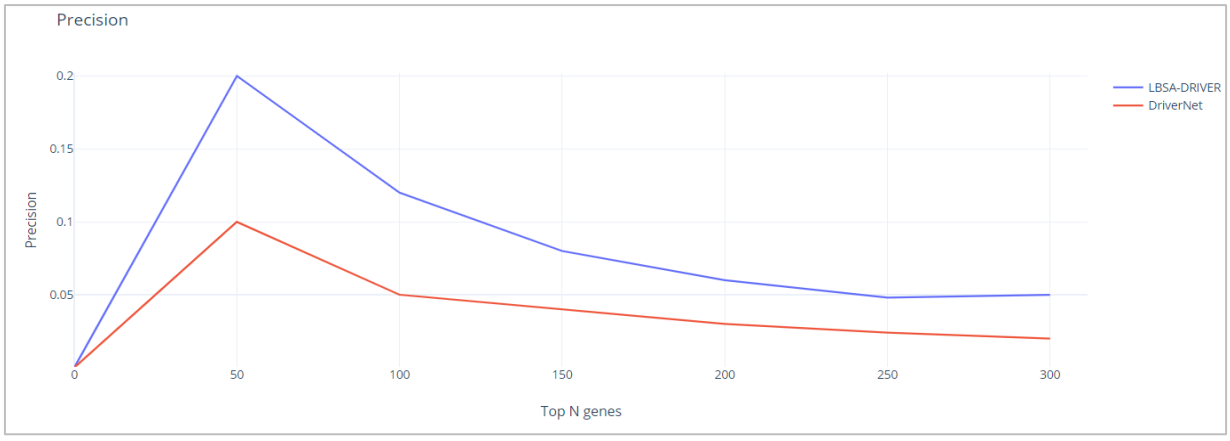


Figure 4.2. Comparing values of precision between LBSA-Driver and DriverNet for the PRAD dataset

C – Ovarian Cancer (OV)

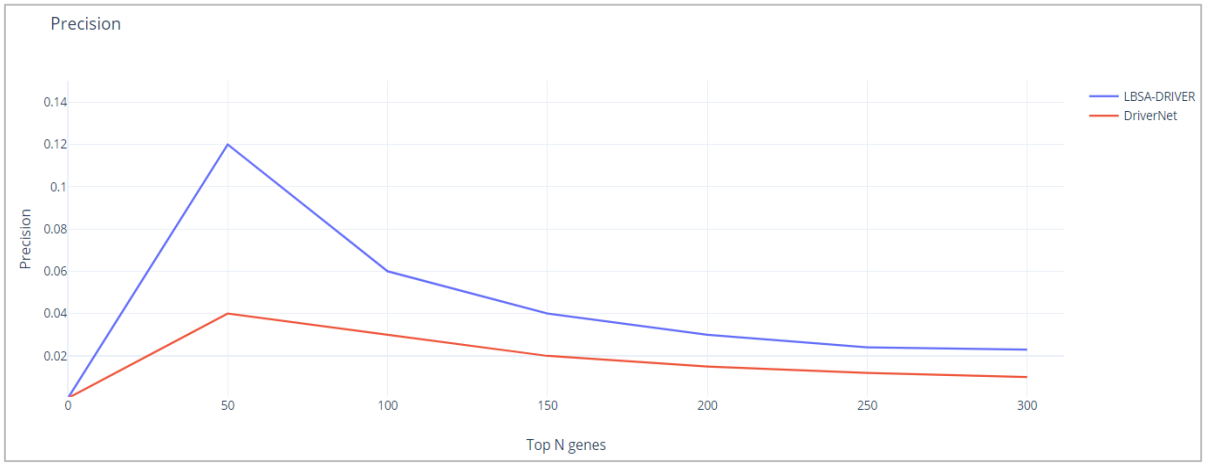


Figure 4.3. Comparing values of precision between LBSA-Driver and DriverNet for the OV dataset.

D – Breast Cancer (BRCA)

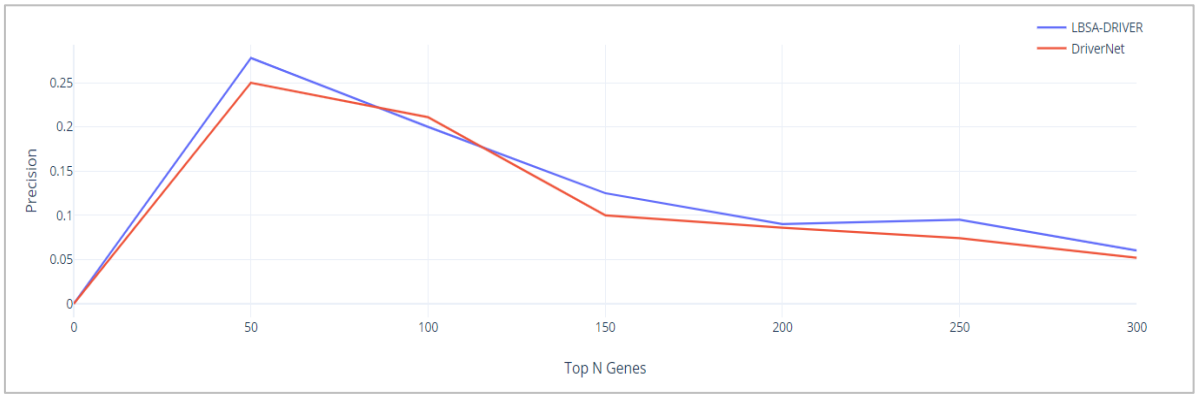


Figure 4.4. Comparing values of precision between LBSA-Driver and DriverNet for the BRCA dataset

E - Glioblastoma Multiforme (GBM)

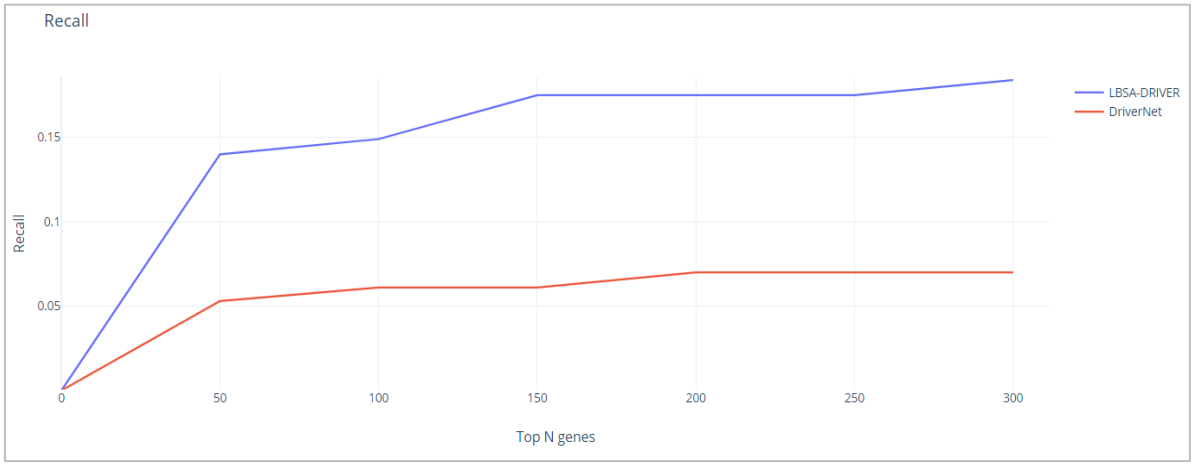


Figure 4.5. Comparing values of recall between LBSA-Driver and DriverNet for the GBM dataset

F - Prostate Adenocarcinoma (PRAD)

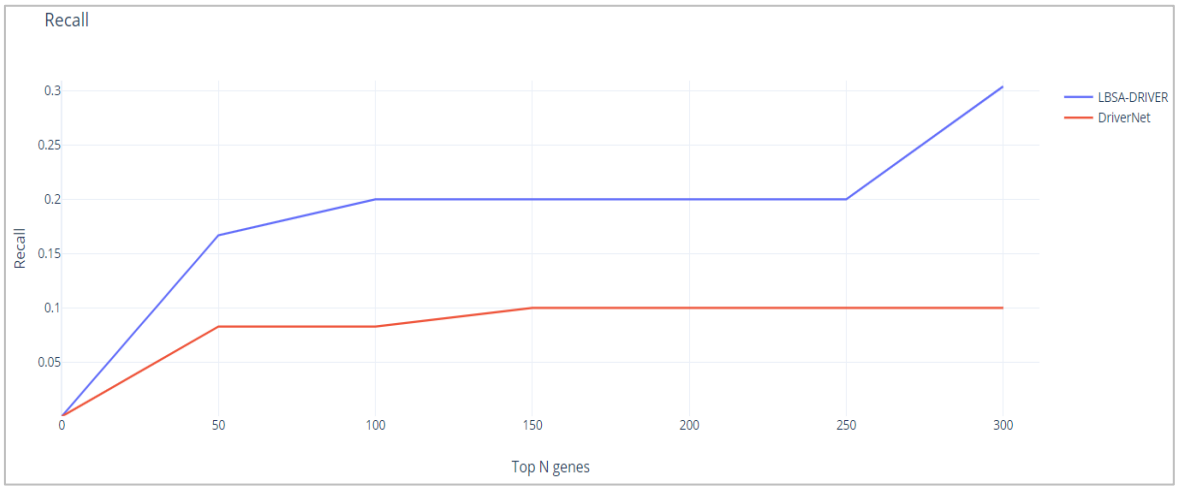


Figure 4.6. This curve represents the comparison of recall for the PRAD dataset between LBSA-Driver and DriverNet

G – Ovarian Cancer (OV)

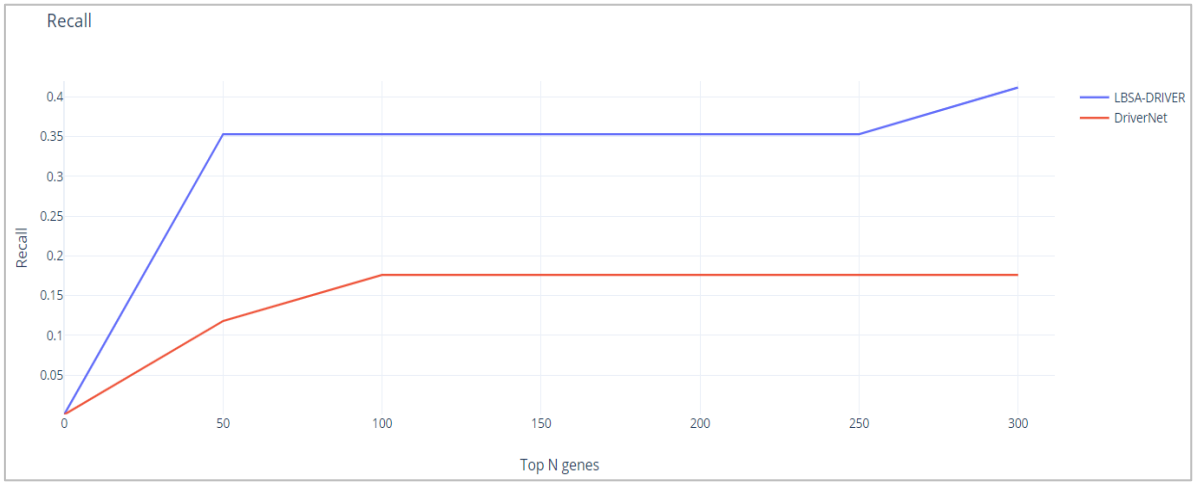


Figure 4.7. This curve represents the comparison of recall for the OV dataset between LBSA-Driver and DriverNet

H – Breast Cancer (BRCA)

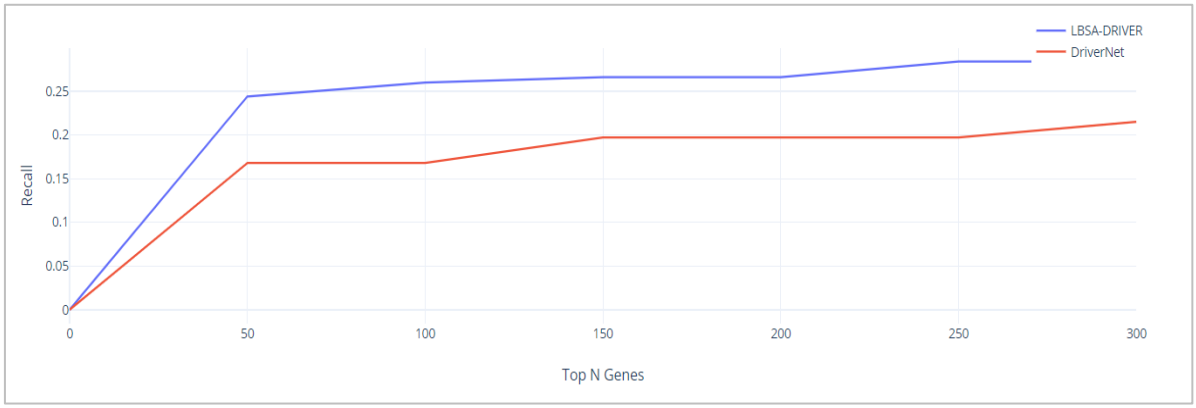


Figure 4.8. This curve represents the comparison of recall for the PRAD dataset between LBSA-Driver and DriverNet

I - Glioblastoma Multiforme (GBM)

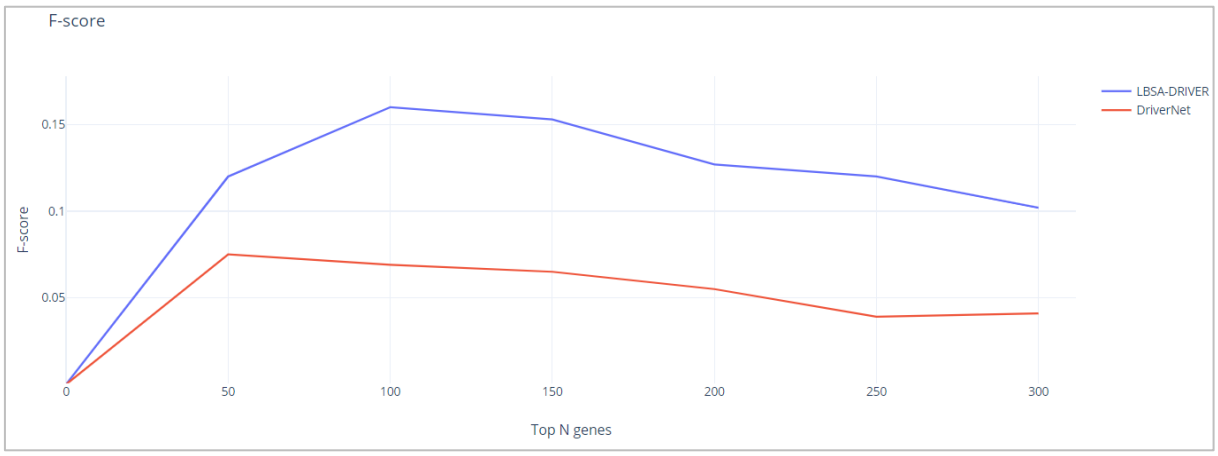


Figure 4.9. Curve comparing F-score values between LBSA-Driver and DriverNet for the GBM dataset

J - Prostate Adenocarcinoma (PRAD)

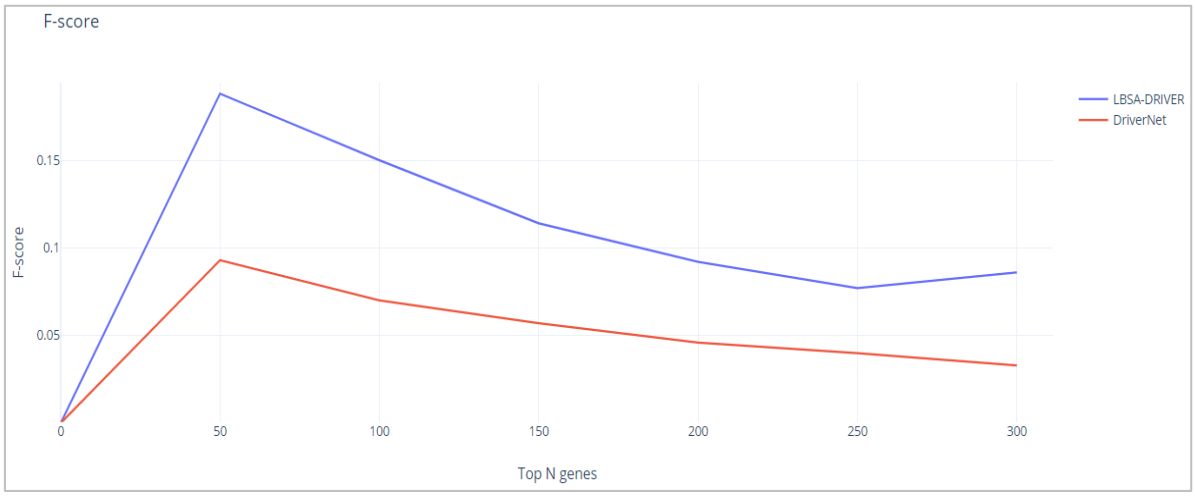


Figure 4.10. Curve comparing F-score values between LBSA-Driver and DriverNet for the PRAD dataset

K – Ovarian Cancer (OV)

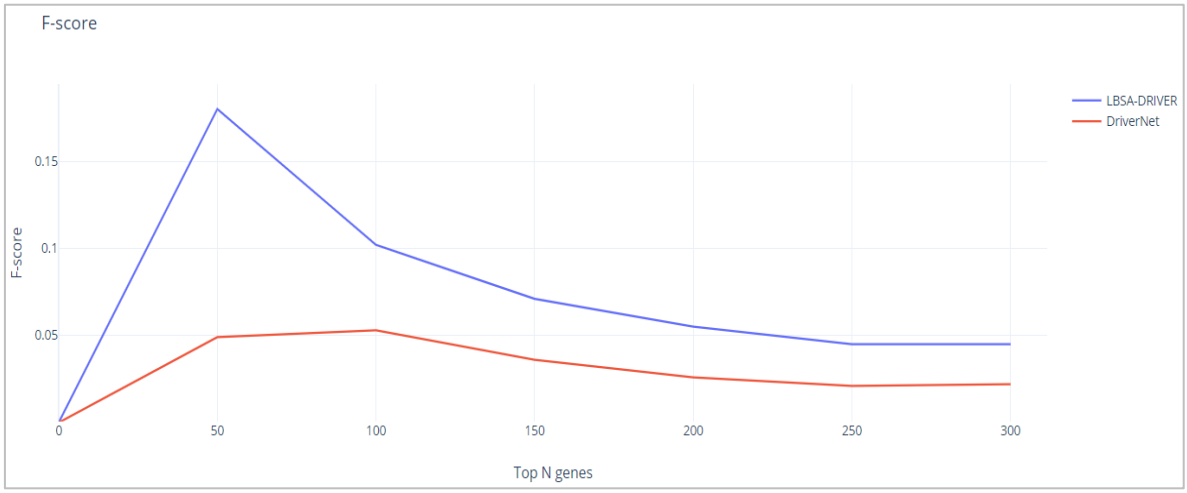


Figure 4.11. Curve comparing F-score values between LBSA-Driver and DriverNet for OV dataset

L – Breast Cancer (BRCA)

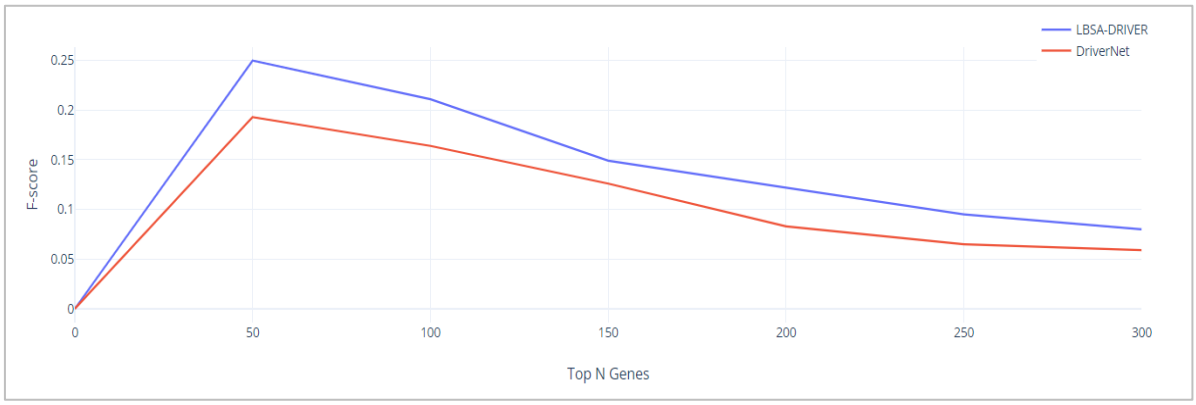


Figure 4.12. Curve comparing F-score values between LBSA-Driver and DriverNet for OV Dataset

4.1.1. Benchmarking analysis for BRCA

In the performance benchmarking analysis for the breast cancer dataset, the algorithm was able to identify up to 32 canonical driver genes in breast cancer, as acknowledged by the Network of Cancer Genes and Healthy Drivers (NGC7.0), with the TP53 being the top-ranked gene. After performing a comparison of the results of our algorithm with the results of DriverNet, DawnRank, IntDriver, and SubDyquency, the algorithm was able to identify up to five canonical driver genes, which were not identified by either of the above-mentioned

methods. These are AKT1, KMT2C, CBFB, ARID1A and CTCF. The method also discovered three other canonical genes, which were also identified by DriverNet but not by the other methods. Although they were identified by DriverNet, they were found to have a much lower rank in the DriverNet results. These genes are NOTCH1, which was ranked 7 by our algorithm ranked 1243 by the DriverNet algorithm, NF1 which was ranked 19 by our algorithm ranked 235 by the DriverNet algorithm and RB1, which was ranked 58 by DriverNet, 35 by DawnRank, 22 by Subdyquency and was not discovered by IntDriver. On the benchmarking analysis results shown in Table 4.4, the Is_Driver column shows if the indicated gene is a driver or not, where 1 means, it is a driver gene and 0 means it is not a driver gene.

4.1.2. Benchmarking analysis for Ovarian Cancer (OV) dataset

Out of the 11 canonical drivers listed by the NGC7.0, the proposed algorithm in this research was able to identify seven genes from the Ovarian Cancer Dataset. From these seven genes, six were ranked in the top 30 by this algorithm. These are: TP53, BRCA1, RB1, CTNNB1, FAT3 and KRAS. After comparing the results of this approach with the results of the DriverNet algorithm, the findings showed that this algorithm was able to discover the FAT3, which was not discovered by the DriverNet algorithm. The 1 gene discovered by this algorithm but not ranked in the top 30 is KREMEN1 ranked 571.

4.1.3. Benchmarking Analysis for Glioblastoma Multiforme (GBM) Dataset

In the benchmarking analysis of the results of this algorithm with the GBM dataset, the algorithm was able to discover 23 driver genes out of the 124 canonical and candidate drivers listed by the NGC version 7. Out of the 23 discovered, the algorithm was able to discover five genes that were not discovered by the DriverNet Algorithm. These five genes are PTPN11, STAG2, ATRX, COL1A2 and IGF1R. The corresponding ranks for these genes in our algorithm are 13, 16, 24, 27, and 29, respectively. Table 4.5 depicts the benchmarking analysis of both the OV dataset and the GBM dataset, which were compared with that of the DriverNet algorithm.

Table 4.4. Benchmarking analysis of the BRCA dataset in comparison with DriverNet, DawnRank, Int_Driver and Subdyquency.

Gene	Is driver	lbsa-driver	drivernet	DawnRank	IntDriver	Subdyquency
TP53	1	1	1	2	2	1
PIK3CA	1	2	2	1	1	2
EP300	1	3	5	175	-	5
AKT1	1	4	-	-	-	-
KMT2C	1	5	-	-	-	-
GATA3	1	6	8	4	4	8
NOTCH1	1	7	1243	-	-	-
NCOR2	0	8	-	-	-	-
PIK3R1	1	9	9	24	-	7
NCOR1	1	10	15	52	30	14
MAP3K1	1	11	20	3	7	11
CBFB	1	12	-	-	-	-
ARID1A	1	13	-	-	-	-
ERBB2	1	14	62	79	-	13
EGFR	1	15	6	-	-	12
BRCA1	1	16	7	23	-	14
CTCF	1	17	-	-	-	-
NCOA3	0	18	-	-	-	-
NF1	1	19	235	-	-	-
AKAP9	0	20	-	-	-	-
SMARCC2	0	21	-	-	-	-
MLL2	0	22	-	-	-	-
PTEN	1	23	118	262	41	20
FOXO3	1	24	312	-	-	-
SMARCC1	0	25	-	-	-	-
ERBB4	0-1	26	53	217	-	24
RB1	1	27	58	35	-	22
JAK1	0	28	997	-	-	-
KDM3A	0	29	-	-	-	-
PRKCZ	0	30	1018	-	-	-

Table 4.5. Benchmarking analysis of the OV and GBM datasets in comparison with DriverNet

		OV				GBM	
Gene	Is driver	LBSA-driver	DriverNet	Gene	Is driver	LBSA-Driver	DriverNet
TP53	1	1	1	TP53	1	1	1
EP300	0	2	141	EGFR	1	2	2
TTN	0	3	12	PTEN	1	3	6
EGFR	0	4	-	PIK3R1	1	4	5
CREBBP	0	5	62	PIK3CA	1	5	8
BRCA1	1	6	70	EP300	0	6	55
PRKACB	0	7	-	TTN	0	7	-
SRC	0	8	-	RB1	1	8	15
LRP2	0	9	278	PRKACA	0	9	-
FAT3	1	10	-	PLCG2	0	10	-
STAT3	0	11	262	UBC	0	11	-
CACNA1C	0	12	104	MUC16	0	12	-
SNRNP200	0	13	315	PTPN11	1	13	-
KRAS	1	14	2	LRP2	0	14	-
PIK3R2	0	15	-	APOB	0	15	-
COL6A3	0	16	299	STAG2	1	16	-
PCDH15	0	17	-	PIK3CB	0	17	-
MUC16	0	18	-	GNAT3	0	18	-
INSR	0	19	535	DYNC1H1	0	19	-
PRKACA	0	20	-	FLNA	0	20	175
DMD	0	21	174	KDR	0	21	357
CLTCL1	0	22	83	STAT1	0	22	201
POM121	0	23	-	FOXA1	0	23	-
HSPG2	0	24	-	ATRX	1	24	-
FAT1	0	25	585	MUC17	0	25	-
KIT	0	26	33	HUWE1	0	26	-
STAT6	0	27	-	COL1A2	1	27	-
NUP214	0	28	1341	NOS1	0	28	-
NUP205	0	29	107	IGF1R	1	29	-
HCK	0	30	1752	TPR	0	30	-

Table 4.6. Benchmarking analysis of the PRAD dataset in comparison with DriverNet, DawnRank, Int_Driver and Subdyquency

Gene	Is driver	LBSA-driver	Drivernet	DawnRank	IntDriver	Subdyquency
TP53	1	1	1	1	3	1
EP300	0	2	-	-	-	-
CTNNB1	1	3	2	21	-	7
FOXA1	1	4	5	2	6	4
PIK3CA	1	5	9	-	53	16
ATM	1	6	11	18	9	6
TTN	0	7	25	51	1	3
PTEN	1	8	147	-	38	18
HSPA8	0	9	14	-	-	27
PRKACA	0	10	-	-	-	-
FAT3	0	11	35	65	5	15
SP1	0	12	12	-	-	30
RELA	0	13	-	-	-	-
GRIA1	0	14	-	-	-	-
AKT1	1	15	-	-	-	-
PTPRC	0	16	-	-	-	-
SF3B1	0	17	-	-	-	-
RTP1	0	18	-	-	-	-
ACTB	0	19	-	-	-	-
VAV1	0	20	-	-	-	-
COL7A1	0	21	-	-	-	-
EPHB1	0	22	-	-	-	-
PRKCZ	0	23	-	-	-	-
HNRNPR	0	24	-	-	-	-
CTCF	0	25	-	-	-	-
PCDHB7	0	26	-	-	-	-
STAT1	0	27	-	-	-	-
ITGA2	0	28	-	-	-	-
PRPF8	0	29	-	-	-	-
KDR	0	30	-	-	-	-

4.1.4. Benchmarking Analysis for Prostate Cancer (PRAD) Dataset

After benchmarking analysis for the PRAD dataset, the results were compared with four other methods (DriverNet, DawnRank, Subdyquency, and IntDriver). The method proposed here was able to discover 13 genes listed as canonical driver genes by the NGC database. Of these 13 genes, TP53, CTNNB1, FOXA1, PIK3CA, ATM, PTEN, and AKT1 ranked 1, 3, 4, 5, 6, 8, and 15, respectively. The AKT1 gene was discovered by this algorithm and not discovered by the other algorithms. Benchmarking analysis proved that this algorithm was able to perform better than existing state-of-the-art methods. Table 4.6 depicts the benchmarking analysis in comparison with other state-of-the-art methods.

The top-ranked canonical driver genes which were identified by the approach proposed in this research and not by the compared approaches were as follows: For the Breast Cancer dataset, this approach discovered AKT1, KMT2C, CBFB, ARID1A and CTCF; for the Ovarian Cancer dataset, this approach discovered FAT3 and KREMEN1; for the Glioblastoma Multiforme dataset for brain cancer, this approach discovered PTPN11, STAG2, ATRX, COL1A2 and IGF1R; finally for the Prostate Adenocarcinoma dataset, this approach discovered the AKT1 driver gene. This is explicitly shown in Table 4.7.

Table 4.7. Top-ranked driver genes identified by the LBSA-DRIVER model and not discovered by DriverNet, DawnRank, IntDriver and Subdyquency.

Dataset	Canonical Driver Genes
BRCA	AKT1, KMT2C, CBFB, ARID1A, CTCF
OV	FAT3, KREMEN1
GBM	PTPN11, STAG2, ATRX, COL1A2, IGF1R
PRAD	AKT1

4.2. Results for Improved ARO

In this study, experiments were carried out on four benchmarking datasets from TCGA, as mentioned above in section 2. These were GBM, BRCA, OV, and PRAD datasets. The implementation was carried out in a Python environment, with the number of iterations being the input parameter. The fitness is calculated using the *Schwefel 2.22* function, which is an unconstraint test benchmark function (Trivedi, Pradeep, Narottam, Arvind, and Dilip, 2016).

The Top 30 ranked genes discovered by this algorithm for each dataset are represented in Table 4.8. Benchmarking performance analysis was then carried out on the results and compared with those of four state-of-the-art models: HotNet2, Dendrix, DriverNet, and QuaDMutNetEx. The analysis was conducted using the information of genes labeled as canonical cancer driver genes by the Network of Cancer Genes and Healthy Drivers (NCG7.0) data repository.

Table 4.8. Top 30 ranked genes discovered by ARO from experiments on GBM, BRCA, PRAD and OV.

Rank	GBM	BRCA	OV	PRAD
1	TP53	TP53	TP53	TP53
2	PTEN	PIK3CA	EP300	CTNNB1
3	PIK3R1	EP300	UBC	EP300
4	PIK3CA	AKT1	TTN	FOXA1
5	RB1	CDH1	DYNC1H1	PIK3CA
6	EB300	KMT2C	EGFR	ATM
7	NF1	GATA3	PIK3CA	PTEN
8	PLCG2	NCOR2	BRCA1	TTN
9	PRKACA	PIK3R1	RB1	STAT3
10	UBC	MAP3K1	PRKCB	CREBBP
11	PDGFRA	NOTCH1	PIK3CB	HSPA8
12	PTPN11	ERBB3	NCOA3	DYNC1H1
13	PIK3CB	NCOR1	DCTN1	SPOP
14	CREBBP	ERBB2	TAF1	SPTA1
15	SP1	NF1	LRRK2	HRAS
16	SPTA1	CBFB	SRC	PRKACA
17	LRP2	EGFR	PRKACB	KMT20
18	APOB	ARID1A	LRP2	FAT3
19	PIK3CG	BRCA1	STAT3	NCOR1
20	DYNC1I1	CTCF	SP1	EGFR
21	KDR	SMAD4	FAT3	PIK3CD
22	MUC16	NCOA3	HTT	RELA
23	ITGB2	JAK1	NCOR2	MUC16
24	JUN	RYNX1	PDGFRA	APC
25	SRC	ATR	ATM	POLR2B
26	PRKCB	AKAP9	POLR2A	HTT
27	STAG2	ERBB4	NFKB1	KMT2C
28	ARRB1	TAF1	ALMS1	PLCB4
29	ESR1	SMARCC2	GNAL	SMAD4
30	DYNC1H1	HERC2	PCDH15	ANK2

4.2.1. Performance benchmarking analysis

In the benchmarking performance analysis of the glioblastoma multiforme brain cancer dataset (GBM) with 68,802 genes, 15 canonical genes were identified out of the total 24 canonical genes presented by NCG7.0. Of these 15 genes, nine were ranked amongst the top 30 genes by the ARO algorithm. These genes include: TP53, PTEN, PIK3R1, PIK3CA, RB1, NF1, PDGFRA, PTPN11 and STAG2. For the triple-negative breast cancer dataset (BRCA), out of the total 110 canonical driver genes, 55 BRCA driver genes were identified in this study. From the 55 identified driver genes, 25 were ranked in the top 30 genes by ARO. For ovarian cancer (OV), out of the total of 11 canonical cancer genes, six were identified in this study, with 4 genes (TP53, BRCA1, RB1, FAT3) ranking in the top 30. Finally, for the prostate adenocarcinoma (PRAD) dataset, out of 43 canonical driver genes, 19 were identified in experiments with ARO, where 12 of these genes rank in the top 30 genes. Table 4.9 shows the cancer driver genes identified by ARO in this study.

Table 4.9. Driver genes discovered by ARO ranked amongst the top 30.

Dataset	Canonical Driver Genes
BRCA	TP53 PIK3CA EP300 AKT1 CDH1 KMT2C GATA3 PIK3R1 MAP3K1 NOTCH1 ERBB3 NCOR1 ERBB2 NF1 CBFB EGFR ARID1A BRCA1 CTCF SMAD4 ATR RB1 RUNX1 PTEN
OV	TP53 BRCA1 RB1 FAT3
GBM	TP53 PTEN PIK3R1 PIK3CA RB1 NF1 PDGFRA PTPN11 STAG2
PRAD	TP53 CTNNB1 FOXA1 PIK3CA ATM PTEN SPOP HRAS KMT2D NCOR1 APC KMT2C

4.2.2. Performance evaluation

The algorithm was evaluated by calculating the precision, recall, f-score, and accuracy of the results. Here, true positives are represented as TP, true-negative as TN, false-positive represented as FP, and false-negative represented as FN. The value for TP was obtained by calculating the total number of genes discovered by the algorithm that have also been labeled

as canonical driver genes by the NCG7.0. The value for TN was taken as the number of genes that have not been labeled by NCG7.0 as canonical driver genes and also have not been ranked amongst the top N genes by the algorithm (N representing different population sizes). The value for FP was obtained by calculating the total number of genes identified by our algorithm but have not been labeled by the NCG7.0 as canonical driver genes. Lastly, the value of FN was obtained by calculating the total number of genes that have not been discovered by the algorithm but have been labeled as driver genes by the NCG7.0 database. Table 4.10 below shows the number of genes labeled as canonical genes by NCG7.0 for the different datasets used in this research.

Table 4.10. Canonical driver gene count in the NCG7.0 for datasets

Dataset	Number of Canonical Genes
GBM	24
BRCA	110
OV	11
PRAD	43

Precision, recall, and the f-score are calculated as shown in Equations 4.1, 4.2, and 4.3, respectively. Table 4.11 shows the comparison of the results of ARO with HotNet2, Dendrix, DriverNet, and QuaDMutNetEx. The table depicts the genes found in the solution for each algorithm, the canonical genes identified, and the True Positives and False positives for each dataset.

4.3. Enrichment Analysis

When trends have to be identified in large biological datasets, functional enrichment analysis tends to be the most famous technique used, as this can be applied medically to discover drug mechanisms and diseases (Wijesooriya, Jadaam, Perera, Kaur, and Ziemann, 2022). Because this technique is often applied to Gene sets, it is also known as Gene Set Enrichment Analysis (GSEA). The identification process is done using statistical methods and approaches to discover either a depleted or an enriched group of genes. There are several procedures involved in the underlying structure of GSEA. These include gene-level

Table 4. 11. Comparison of canonical genes with HotNet2, DriverNet, Dendrix and QuaDMutNetEx

Dataset	Algorithm	Genes in Solution	Canonical Genes	TP	FP	p-score
OV	ARO	11	6	6	5	0.545
	HotNet2	11	4	4	7	0.363
	DriverNet	-	-	-	-	-
	Dendrix	11	3	3	8	0.273
	QuaDMutNetEx	11	7	7	5	0.636
GBM	ARO	24	11	11	13	0.458
	HotNet2	24	11	11	26	0.458
	DriverNet	24	9	9	8	0.375
	Dendrix	24	4	4	20	0.167
	QuaDMutNetEx	24	6	6	18	0.250
PRAD	ARO	43	19	19	24	0.442
	HotNet2	43	9	9	34	0.209
	DriverNet	43	14	14	32	0.326
	Dendrix	43	8	8	35	0.186
	QuaDMutNetEx	43	13	13	30	0.302
BRCA	ARO	110	55	55	55	0.500
	HotNet2	110	18	18	92	0.164
	DriverNet	110	33	33	77	0.300
	Dendrix	110	16	16	94	0.145
	QuaDMutNetEx	110	25	25	85	0.227

.

statistics, global tests, transformation, gene set statistics, null hypotheses, and significance assessment. The gene-level statistics analyze parameters, such as fold change, signal-to-noise ratio, t-statistic, regression coefficient, log-likelihood ratio, and correlation coefficient. For the global tests, the global Ancova and Hotelling's T^2 tests are used. The transformation could be binary, rank, p-value, local FDR, absolute value, square-based transformation, or, in some cases, no transformation applied. Gene set statistics consist of several tests, including the Wilcoxon rank test, the MaxMean statistic, the mean, sum, and median, the enrichment score, the sign test, and the conditional FDR. The significance assessment tests include sample label permutation, gene sampling, and re-standardization. There exist different tools for performing enrichment analysis. Some of these tools include but are not limited to NASQAR (Yousif, Drou, Rowe, Khalfan, and Gunsalus, 2020), Enrichr

(Laboratory Ma'ayan, 2014), AmiGO 2 (Consortium Gene Ontology) and Blast2GO (Conesa et al., 2005). In this research, Gene Ontology (GO) analysis is performed using the PANTHER Overrepresentation test in the Gene Ontology Resource (GOR) (The Gene Ontology Resource, 2022). This is the latest release of the GOR consisting of 43,648 GO terms with 7,510,543 annotations, 1,473,200 gene products, and 5,183 species.

As shown in figure 4.13 below, the GO analysis was carried out using the following parameters from the PANTHER analysis platform:

- Analysis Type: Panther Overrepresentation Test
- Reference list: Homosepiens
- Annotation Dataset: GO Bio-process Complete
- Test Type: Fischer's Test
- Correction Test: False Discovery Rate (FDR)

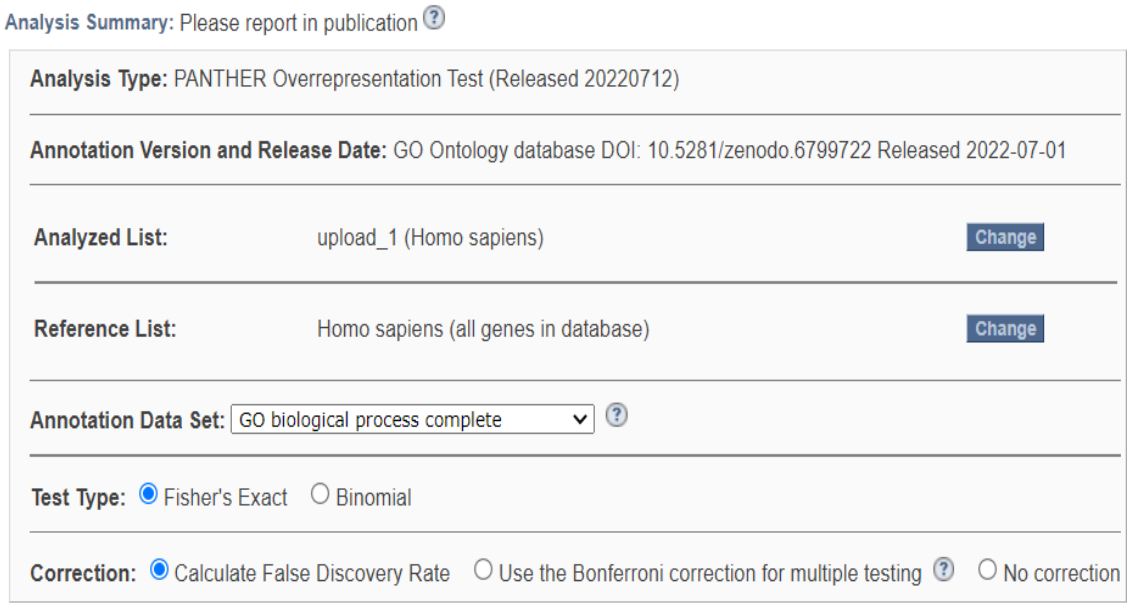


Figure 4. 13. Settings for performing GO using PANTHER Overrepresentation Test

5. DISCUSSION

In this research, two methods have been proposed. The first method is called LBSA-DRIVER, which is a network-based model used to identify cancer driver genes by making use of a transcriptional network. There have been similar studies done in this area to identify subnetworks rather than single stand-alone genes (Vandin, Upfal, and Raphael, 2011; Bashashati et al., 2012; Bokhari, Alhareeri, and Arodz, 2020; Xi, Wang, and Li, 2018; Sarkar A. et al., 2020). In this work, a gene-gene interaction network is used to verify the impact of the corresponding genes. The proposed model here makes use of a bipartite graph similar to that used in the DriverNet algorithm. This bipartite graph is generated such that the bipartite graph's left-side nodes are gene mutations while the right-side nodes are outlier expression mutations. After generating the bipartite graph, an optimization problem is created by trying to map the genes found on the left side of the graph to those found on the right side based on their effect on a given biological network (A gene-gene interaction network, in this case). To solve this optimization problem, a novel method proposed by (Zhan, Lin, Zhang, and Zhong, 2016) called List-based Simulated Annealing (LBSA) is used. This method makes use of a specially generated temperature list to perform simulated annealing. Experimental analysis is then performed on four benchmarking data sets of TCGA. These datasets include the TCGA Glioblastoma Multiforme (GBM) dataset, which is related to brain cancer, the TCGA Breast Cancer (BRCA) dataset for breast cancer, the TCGA Prostate Adenocarcinoma (PRAD) dataset for prostate cancer, and the TCGA Ovarian Serous Cystadenocarcinoma (OV) dataset for ovarian cancer. Benchmarking analysis enrichment analysis was performed on the results from the experiments and the results showed that our algorithm was able to effectively identify canonical cancer genes as represented by the NCG7.0 database.

In the analysis of the GBM dataset, we were able to identify 23 driver genes. Out of these, this approach was able to identify PTPN11, STAG2, ATRX, COL1A2, and IGF1R, which were not identified by the DriverNet algorithm. In the same way, during the benchmarking analysis of the OV dataset, the algorithm was able to identify TP53, BRCA1, RB1, CTNNB1, FAT3, and KRAS, of which FAT3 was not discovered by the other state-of-the-art methods comparisons were carried out with. The results here on the PRAD dataset were similar to those of the OV dataset, as the method was able to discover one extra gene, AKT1, which was not discovered by the other algorithms. Finally, the performance on the BRCA

dataset identified AKT1, KMT2C, CBFB, ARID1A, and CTCF, which were not identified by the other state-of-the-art methods. An evaluation of the results was then performed by calculating the precision, f-score, recall, and accuracy and comparing the obtained results with the results of state-of-the-art methods, including DriverNet, SubDyquency, DawnRank, and IntDriver. This algorithm has one setback, which is the processing time. Because the simulated algorithm is iterative, the model developed here compromises running time for accuracy. The larger the dataset, the more time it takes for the algorithm to produce the results. Due to this, we work to tackle the issue of processing time by implementing a modern-day metaheuristic algorithm called the Artificial Rabbits Optimization Algorithm, which works faster while maintaining the accuracy of the approach. This allows the model to be used effectively on PanCancer datasets, which are much larger. Another example of an algorithm that will work effectively to solve the optimization problem in this research is the Grey Wolf Optimization algorithm (2022) (Besharatnia, Talebpour, and Aliakbary, 2022).

The second method in this research is an improved version of the bio-inspired Artificial Rabbit's Optimization algorithm proposed by Wang L et al. in 2022. This algorithm was improved by adding a mutation phase before implementing the detour foraging and random hiding phases of the ARO. The model in this study is as follows. First, using the cancer data for the respective cancer type obtained from the TCGA data repository and a biological interaction network, a bipartite graph was generated as that presented in DriverNet and QuaDMutNetEx. ARO was then applied to this bipartite graph to identify cancer driver genes. This improved ARO algorithm is implemented thus: A random solution was initialized in the neighborhood space. Mutation was then applied to this solution to generate a more diverse solution that covers the search space. Next, the energy factor, A, was calculated to determine whether detour foraging or random hiding would be implemented on the current rabbit solution. This is the energy-shrinking mechanism that is used by rabbits for survival. If the energy factor is greater than 1, then detour foraging (exploration) is implemented to generate the new position of the rabbit. If the energy factor is less than or equal to 1, then random hiding is implemented to generate the new position of the current rabbit. At every iteration, the fitness of the current solution was calculated using the *Schwefel 2.22* function and compared with that of the candidate solution to select the new current solution. The best solution was then obtained at the end of the iterations. Performance analysis was then applied to evaluate the results of this model. Performance analysis was performed on the four benchmarking datasets: GBM brain cancer dataset, PRAD prostate

cancer dataset, BRCA dataset, and OV dataset. A comparison of results was performed with four other state-of-the-art models used for identifying cancer driver mutations: Dendrix, DriverNet, HotNet2, and QuaDMutNetEx. ARO proposed in this study proved to be quite efficient in identifying cancer driver mutations compared to these methods. ARO was able to identify the following driver genes labeled as canonical genes by the NCG7.0. For GBM: TP53, PTEN, PIK3R1, PIK3CA, RB1, NF1, PDGFRA, PTPN11 and STAG2 were identified among the top 30 ranked; for OV: TP53, BRCA1, RB1 and FAT3 were identified among the top 30 ranked; for PRAD: TP53, CTNNB1, FOXA1, PIK3CA, ATM, PTEN, SPOP, HRAS, KMT2D, NCOR1, APC and KMT2C were identified among the top 30 ranked; for BRCA: TP53, PIK3CA, EP300, AKT1, CDH1, KMT2C, GATA3, PIK3R1, MAP3K1, NOTCH1, ERBB3, NCOR1, ERBB2, NF1, CBFB, EGFR, ARID1A, BRCA1, CTCF, SMAD4, ATR, RB1, RUNX1 and PTEN were identified among the top 30 ranked.

There are several reasons to why ARO serves as an appropriate algorithm for this study. In ARO, the detour foraging helps in accomplishing global search, while random hiding helps in accomplishing local search. The parameter, R, presented in Equation 3.6, could be adaptatively adjusted as the number of iterations increases to foster the gradual alternation from exploration to exploitation. Approximately half of the iterations are assigned to exploitation ($A \leq 1$), and the other half is assigned to exploration ($A > 1$) as an outcome of the energy factor, A. This energy factor is a time-dependent factor that effectively switches between exploitation and exploration and enhances these phases. Furthermore, ARO has a good ability to bear fault diagnosis obtained from its ability to balance exploitative and explorative searches. The computation complexity of ARO is linear and given by: $O(ARO) = O(1 + n + Tn + 0.5*Tnd + 0.5\ Tnd) = O(Tnd + Tn + n)$. Despite having a superior performance, ARO also has a couple of shortcomings. ARO lacks multiple search mechanisms during the optimization of problems that have variable types of certainty. Also, ARO faces shortcomings in handling unimodal problems that have multiple extrema. Hence, ARO is less efficient in solving NP-Hard problems. These problems could be solved by implementing multi-objective and binary versions of the algorithm. Finally, GO analysis was performed on the top 30 ranked genes for each dataset to identify their effect on biological processes.

6. CONCLUSION

The experts and researchers in bioinformatics have developed methods to identify gene driver mutations from passenger mutations. The problem of identifying driver mutations is directly related to finding out what mutations qualify as passenger mutations and are not directly involved in the development of the cancer disease. Cancer driver mutations are mutations that directly drive cancer initiation and progression, while passenger mutations do not. Excessive work has been carried out in trying to identify cancer driver genes from passenger genes, as passenger genes consist of quite a large percentage of mutations in cancer tumors.

The problem of finding a good technique to identify cancer driver genes has existed in the bioinformatics and computational biology society for decades and has seen many methods and attempts by scientists to solve the problem at hand. This problem has also gradually moved from identifying only single genes or mutations to identifying pathways that consist of a group of genes sharing similar functional and biological properties. Originally, methods to do these were all frequency-based, whereby a gene was considered a driver mutation based solely on its frequency. That is, highly frequent genes were considered driver genes and less frequent genes were labelled as passenger gene mutations. However, this had a significant setback, given that most driver genes were also low frequency but held a vital role in cancer development. Thus, network-based methods were then developed, which were not totally based on frequency but rather on the effect of related genes on a biological network.

The methods proposed in this research are a network-based method, which uses a novel list-based simulated annealing approach to solve the optimization problem and an improved bio-inspired algorithm known as ARO to be used for identifying cancer driver genes. The algorithm is chosen because of the advantages it provides as compared to the previously proposed methods, most notably with its ability to exhaust the search space while being efficient. Through experimental analysis of benchmarking datasets, this method proves to be effective in identifying cancer driver genes. The approaches are compared with DriverNet, DawnRank, IntDriver, and Subdyquency, which are state-of-the-art methods developed, over time. These approaches here evidently perform better than these methods in identifying canonical driver genes. Functional enrichment analysis on the top identified genes also shows the importance of the found genes in biological processes in humans. This research is important as it aims to bring forward an accurate method for detecting these

driver mutations, which will help prevent the disease. This research is also crucial as it gives other researchers a lifting point for their research and serves as a knowledge platform for information about cancer research.

REFERENCES

Futreal, P., Coin, L., Marshall, M., Down, T., Hubbard , T., Wooster, R., Stratton, M. (2004). A census of human cancer genes. *Nature Reviews Cancer*, (4),177-183.

Internet: (May 2022). cBioPortal for Cancer Genomics. Web: http://www.cbioportal.org/ Last accessed date: 05.05.2022

Akhavansafar, M., and Teimourpour, B. (2020). A network Based method to predict cancer causal genes in GR Network. *bioRxiv*.(3),24-27.

Alipanahi, B., Delong, A., Weirauch, M., and Frey, B. (2015). Predicting the sequence specifcities of DNA and RNA-binding proteins by deep learning. *Nature Biotechnology*., 33(8),831.

Angermueller, C., Lee, H, Reik, W, and Stegle, O. (2017). DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. *Genome Biology*, 18(1),1–13.

Angermueller, C., Pärnamaa, T., Parts, L., and Stegle, O. (2016). Deep learning for computational biology. *Molecular Systems Biology*, 12(7),878.

Annelies, B., Daria,F.,An, S., Tessa, V., Hans, D., and Kris, G. (2022). Limited evidence for protein products of non-coding transcripts in the HEK293T cellular cytosol. *Molecular and Cellular Proteomics*,(12), 44-49.

Aparisi, F., Amado-Labrador, H., Calabuig-Fariñas, S., Torres, S., Herreros-Pomares, A., Jantus-Lewintre, E., . . . Camps, C. (2019). Passenger mutations in cancer evolution. *Cancer Republic*, (9), 11-15.

Bailey, M., Tokheim, C., Porta-Pardo, E., Sengupta, S., Bertrand, D., Weerasinghe, A., and Schein, J. (2018). Comprehensive characterization of cancer driver genes and mutations. *Cell*, 173(2), 371-385.

Bashashati, A., Haffari, G., Ding, J., Ha, G., Lui, K., Rosner, J., and Shah, P. (2012). DriverNet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer. *Genome Biology*, 13(12), 1-14.

Benjamin, L., and Keshet, E. (1997). Conditional switching of vascular endothelial growth factor (VEGF) expression in tumors: induction of endothelial c ell shedding and regression of hemangioblastoma-like vessels by VEGF withdrawal. *Proceedings of the National Academy of Sciences of the USA*, 94, 8761-8766.

Besharatnia, F., Talebpour, A., and Aliakbary, S. (2022). An Improved Grey Wolves Optimization Algorithm for Dynamic Community Detection and Data Clustering. *Applied Artificial Intelligence*. 1-22.

Bokhari, Y., Alhareeri, A., and Arodz, T. (2020). QuaDMutNetEx: a method for detecting cancer driver genes with low mutation frequency. *BMC Bioinformatics*, 21(1), 1-12.

Internet: (November 2019). Cancer Biomarkers: Improving Detection and Treatment. Retrieved from Crown Bioscience. Web: https://blog.crownbio.com/cancer-biomarkers-improving-detection-and-treatment . Last accessed date: 30.09.2022

Camp, M., Rachlow, J., Shipley, L., Johnson, T., and Bockting, K. (2014). Grazing in sagebrush rangelands in western north america: implications for habitat quality for a sagebrush specialist, the pygmy rabbit. *Rangel*, 36 (2), 151–159.

Chaudhary, M., Pham, V., Le, T., and Robinson, P. (2021). NIBNA: a network-based node İmportance approach for identifying breast cancer drivers. *Bioinformatics*, 37(17), 2521-2528.

Chen, Z., Lu, Y., Cao, B., Zhang, W., Edwards, A., and Zhang, K. (2022). Driver Gene Detection through Bayesian Network Integration of Mutation and Expression Profiles. *Bioinformatics*. 30(1), 2331-2348

Cheng, F., Zhao, J., and Zhao, Z. (2015). Advances in computational approaches for prioritizing driver mutations and significantly mutated genes in cancer genomes. *Brief Bioinformatics*., 17(4),642–56.

Cho, A., Shim, J. E., Kim, E., Supek, F., Lehner, B., and Lee, I. (2016). MUFFINN: cancer gene discovery via network analysis of somatic mutation data. *Genome Biology*, 17, 129.

Conesa, A., Götz, S., García-Gómez, J., Terol, J., Talón, M., and Robles, M. (2005). Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics*, 21 (18), 3674–3676.

Dees, N., Zhang, Q., Kandoth, C., Wendl, M., S chierding, W., and Koboldt, D. (2012). MuSiC: identifying mutational significance in cancer genomes. *Genome Research*, 22(8),1589–98.

Erten, C., Houdjedj, A., and Kazan,H. (2021).Ranking cancer drivers via betweenness-based outlier detection and random walks. *BMC Bioinformatics*, 22(1), 1-16.

Guo, W., Zhang, S., Zeng, T., Li, Y., Gao, J., and Chen, L. (2019). A novel network control model for identifying personalized driver genes in cancer. *PLoS Computational Biology*, 15(11).

Hai, Y., Qiang, W., Xue, Z., Hushan, Y., and Bingshan, L. (2017). Cancer driver gene discovery through an integrative genomics approach in a non-parametric Bayesian framework. *Bioinformatics*, 33, (4) 483–490.

Han, Y., Yang, J., Qian, X., C heng, C., Liu, S., Hua, X., and Lu, Y. (2019). DriverML: a machine learning algorithm for identifying driver genes in cancer sequencing studies. *Nucleic Acids Research*, 47(8), 45-46.

Hsiehchen, D., and Hsieh, A. (2018). Nearing saturation of cancer driver gene discovery. *Journal of Human Genetics*, 63, 941-943.

Irving, J. (2017). Chapter 4 - *The Biology, Prognostic Relevance, and Targeted Treatment of Ras Pathway–Positive Childhood Acute Lymphoblastic Leukemia*. In J. Irving, Conquering RAS From Biology to Cancer Therapy, 61-78.

Juan, Q. (2017). Rabbits do not eat grass around the nest. *Knowledge Window*, 13, 39.

Kumar, S., Warrell, J., Li, S., McGillivray, P., Meyerson, W., Salichos, L., and Gerstein, M. (2020). Passenger mutations in more than 2,500 cancer genomes: overall molecular functional impact and consequences. *Cell*, 180(5), 915-927.

Internet: Laboratory Ma'ayan. (October 24). Computational Systems Biology. Icahn School of Medicine at Mount Sinai. Web: labs.icahn.mssm.edu. Last accessed date: 20.12.2022

Lawrence, M., Stojanov, P., Polak, P.,Kryukov,G.,Cibulskis, K., and Sivachenko, A. (2013). *Nature*, 499(7457),214.

Lee, D. S.,Vassiliadis,V. S., and Park, J. M. (2002). List-based threshold accepting algorithm for zero-wait scheduling of multiproduct batch plants. *Industrial and Engineering Chemistry Research*, 41 (25) 6579–6588.

Li , Q., Ishikawa, T.,Oshima, M., and Taketo, M. (2005). The threshold level of adenomatous polyposis coli protein for mouse intestinal tumorigenesis. *Cancer Research*, 65, 8622-8627.

Lijin, W., Rongying, C., Min, L., and Yiwen, Z. (2019). Enhanced List-based Simulated Annealing Algorithm for Large-Scale Traveling Salesman Problem. *IEEE Access*, (7). 144366 - 144380.

Liu, C., Dai, Y., Yu , K., and Zhang, Z.-K. (2022). Enhancing Cancer Driver Gene Prediction by Protein-Protein Interaction Network. *IEEE*, 2231 - 2240.

Malebary, S., and Khan, Y. (2021). Evaluating machine learning methodologies for identification of cancer driver genes. *Scientific Reports*, 11(1), 1-13.

Mamoshina, P., Vieira, A., Putin, E., and Zhavoronkov, A. (2016). Applications of deep learning in biomedicine. *Molecular Pharmarcy*., 13(5),1445–54.

Mao, C., Yao, L.,Pan,Y.,Luo, Y., and Zeng,Z. (2018). Deep generative classifers for thoracic disease diagnosis with chest x-ray images. *IEEE*, 1209–1214.

Metropolis, N., Rosenbluth, W. A., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *Calculations by Fast Computing Machines*, (21), 6.

Michael, S. L., Stojanov, P., Mermel, C. H., Garraway, L. A., Golub, T. R., Meyerson, M., Getz, G. (2014). Discovery and saturation analysis of cancer genes across 21 tumor types. *PubMed Central*, 505(7484), 495–501.

Mirjalili,S., and Lewis, A. (2016). The whale optimization algorithm. *Advanced Engineering Software*, 95, 51–67.

Internet: NCG. (May 2022). NCG7.0. Retrieved from The Network of Cancer Genes and healthy. Web: drivers: http://ncg.kcl.ac.uk/old_versions.php Last accessed date: 21.07.2022

Network, C. G. (2008). Comprehensive genomic characterization defines human glioblastoma genes and core pathways. In C. G. Network, Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*. (455),1061-1068.

Internet: NIH. (2022, May 21). About Cancer. Retrieved from U.S. National Institute of Health. National Cancer Institute. Web: https://www.cancer.gov/about-cancer Last accessed date: 21.05.2022

Nikolakopoulos, A., and Sarimveis, H. (2007). A threshold accepting heuristic with intense local search for the solution of special instances of the traveling salesman problem. *European Journal of Operational Research*, 177, (3) 1911–1929.

Oliveira Jr, H., and Petraglia, A. (2014). Establishing nash equilibria of strategic games: a multistart fuzzy adaptive simulated annealing approach. *Applied Soft Computing Journal*, (19) 188–197.

Oliveira, H. E., and Petraglia, A. (2013). Solving nonlinear systems of functional equations with fuzzy adaptive simulated annealing. *Applied Soft Computing Journal* 13, (11) 4349– 4357.

Quang, D., and Xie, X. (2016). DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Research*., 44(11):e107–e107.

Saed, A., Rebennack, S., and Pardalos, P. M. (2008). Improving the neighborhood selection strategy in simulated annealing using the optimal stopping problem. I-*Tech Education and Publishing*, 420.

Shui, L., Xiaoxiao, Y., Dan, Z., Jiyao, S., Xin, W., Qingyu, W., . . . Xuewen , Z. (2018). Analysis of Transcription Factor-Related Regulatory Networks Based on Bioinformatics Analysis and Validation in Hepatocellular Carcinoma. *BioMed Research International*. (23) 128–137.

Simone de Jong, M. P. (2012). A Gene Co-Expression Network in Whole Blood of Schizophrenia Patients Is Independent of Antipsychotic-Use and Enriched for Brain-Expressed Genes. *PLOS ONE*, 7(6): 39498.

Song, J., Peng, W., and Wang, F. (2019b). A random walk-based method to identify driver Genes by integrating the subcellular localization and variation frequency into bipartite graph. *BMC Bioinformatics*, 20(1), 1-17.

Song, J., Peng, W., and Wang, F. (2019c). An Entropy-based method for identifying mutual exclusivedriver genes in cancer. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 17(3), 758-768.

Song, J., Peng, W.,Wang, F., and Wang, J. (2019a). Identifying driver genes involving gene dysregulated expression, tissue-specific expression and gene-gene network. *BMC Medical Genomics*, 12(7), 1-12.

Talbi, E. G., and Muntean, T. (1993). Hill-climbing, simulated annealing and genetic algorithms: a comparative study and application to the mapping problem. *Proceedings of the Twenty-sixth Hawaii International Conference on System Sciences*., (2), 565-573.

Tang, Y., Wei, P., Zhao, J., Xia, J., Cao, R., and Zheng, C. (2021). Identification of driver genes based on gene mutational effects and network centrality. *BMC Bioinformatics*, 22(3), 1-16.

Internet: The Gene Ontology Resource. (May 2022). GOR. Web: http://geneontology.org/ Last accessed date: 16.05.2022

Thomas, R., Gerrit, A. M., David, J. H., Godfrey , G., Sebastian, F.-E., Romina, B., . . . Jordi, C. (2019). The landscape of genomic copy number alterations in colorectal cancer and their consequences on gene expression levels and disease outcome. *Molecular Aspects of Medicine*, 48-61.

Tian, R.,Basu,M., and Capriotti, E. (2014). ContrastRank: a new method for ranking putative cancer driver genes and classification of tumor samples. *Bioinformatics*, 30(17),i572–8.

Trivedi, I., Pradeep, J., Narottam, J., Arvind, K., and Dilip, L. (2016). Novel adaptive whale optimization algorithm for global optimization. *Indian Journal of Science and Technology*., 9(38),319-26.

Tůmová, E., Martinec, M., and Chodová, D. (2011). Analysis of Czech rabbit genetic resources. *Science Agriculture Bohem*, 42,113-8.

Tynes, V. V. (2010). *Behavior of Exotic Pets*. (New Jersey, USA) John Wiley and Sons. 13

Vandin, F., Upfal, E., and Raphael, B. (2011). Algorithms for detecting significantly mutated pathways in cancer. *Journal of Computational Biology*, 18(3),507–22.

Wang, C. Y., Lin, M., Zhong, Y. W., and Zhang, H. (2015). Solving travelling salesman problem using multiagent simulated annealing algorithm with instance-based sampling. *International Journal of Computing Science and Mathematics*. 6, (4)336–353.

Wang, L., Cao, Q., Zhang, Z., Mirjalili, S., and Zhao, W. (2022). Artificial rabbit's optimization: A new bio-inspired meta-heuristic algorithm for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, 114,105082.

Wangxiao, X., Hui, J., Huifang, G., Yaowen, L., and Xingchun, G. (2022). Integrated gene co-expression network analysis reveals unique developmental processes of Aurelia aurita. *Gene*, 2(31), 125

Wijesooriya, K., Jadaam, S. A., Perera, K.L.,Kaur, T., and Ziemann, M. (2022). Urgent need for consistent standards in functional enrichment analysis. *PLoS Computational Biology*, 18(3), e1009935.

Wodarz, D., Newell, A., and Komarova, N. (2018). Passenger mutations can accelerate tumour suppressor gene inactivation in cancer evolution. *Journal of The Royal Society Interface*, 15(143), 967.

Xi, J., Wang, M., and Li, A. (2017). Discovering potential driver genes through an integrated model of somatic mutation profiles and gene functional information. *Molecular BioSystems*, 13(10),2135–44.

Xi, J., Wang, M., and Li, A. (2018). Discovering mutated driver genes through a robust and sparse co-regularized matrix factorization framework with prior information from mRNA expression patterns and interaction network. *BMC bioinformatics*, 19(1), 1-14.

Yamamoto, K. (2022, 10 23). *The Elements of Precision Medicine*. (San Fransisco, USA) University of California San Fransisco. 27-31.

Yang, H., Zhang, L., Zhou, D., Li, D., Zhang, J., and Wang, Z. (2022). Trans-Driver: a deep learning approach for cancer driver gene discovery with multi-omics data. *Cold Spring Harbour Laboratory bioRXIV*. 2 (3), 26

Yousif, A., Drou, N.,Rowe,J.,Khalfan, M., and Gunsalus, K. (2020). NASQAR: a web-based platform for high-throughput sequencing data analysis and visualization. *BMC Bioinformatics*., 21 (1), 267.

Zeng, Z., C, M., V., Xin., Jo, N., Sa, K., and Se, C. (2021). Deep learning for cancer type classification and driver gene identification. *BMC Bioinformatics*, 22(4),1-3.

Zhan, S., Lin, j., Zhang, Z., and Zhong, Y. W. (2016). List-based simulated annealing algorithm for traveling salesman problem. *Computational Intelligence and Neuroscience*. 3, 12-23

Zhang, T., and Zhang, D. (2017). Integrating omics data and protein interaction networks to prioritize driver genes in cancer. *PubMedCentral*,. 8(35), 58050–58060.

Zhang, T., Zhang, S., and Li, Y. (2021). Identifying driver genes for individual patients through inductive matrix completion. *Bioinformatics*, 37(23), 4477-4484.

Zhang, W., and Wang, S. (2020). A novel method for identifying the potential cancer driver genes based on molecular data integration. *Biochemical Genetics*, 58(1), 16-39.

Zhou, J., and Troyanskaya, O. (2015). Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods*, 12(10),931



Gazili olmak ayrıcalıktır