

METASEZGİSEL ALGORİTMALARLA PORTFÖY OPTİMİZASYONU: BIST 30 UYGULAMASI¹

PORTFOLIO OPTIMIZATION WITH METAHEURISTIC ALGORITHMS: BIST 30 APPLICATION

Danyel BEKDAŞ * Hicabi ERSOY **

Arařtırma Makalesi / Geliř Tarihi: 07.03.2022
Kabul Tarihi: 31.03.2022

Öz

Hisse senetleri günlük fiyat deęişimlerine baęlı olarak yatırım kategorisinde riskli varlıklar olarak sınıflandırılmaktadır. Risk ile getiri pozitif yönlü iliřki olmakla beraber, risk ile getiri arasında denge kurmak portföy yönetimi açısından oldukça önemlidir. Yapay zeka algoritmaları ile hisse senetlerinin getirisi ve risk olarak ifade edilen standart sapmaları dikkate alınarak getiri ile risk arasında optimizasyon analizi yapılmaktadır. Çalışma kapsamında son dönemde geliştirilmiş yapay zeka algoritmalarından Jaya Algoritması, Öğrentme-Öğretme Tabanlı Algoritma ve Çiçek Tozlaşma Algoritmaları tanıtılmakta, bu algoritmalar kullanılarak BIST 30 hisse senetleri için optimizasyon yapılmakta ve bu üç algoritmadan elde edilen sonuçlar kıyaslanmaktadır.

Anahtar Kelimeler: *Portföy Getirisi, Portföy standart sapması, Risk, Optimizasyon, Yapay Zeka Algoritmaları, Finansal Analiz.*

JEL Sınıflaması: G11, G32, C61

Abstract

Stocks can be classified as risky financial instruments considering volatility in stock prices. Considering positive relationship between return and risk, its very important to balance the risk and return in portfolio management. Optimization analysis is made between return and risk with artificial intelligence algorithms by taking into account return and risk of stocks which expressed as the standard deviations. Within the scope of the study; Jaya Algorithm, Teaching-Learning Based Algorithm and Flower Pollination Algorithms, which are recently developed artificial intelligence algorithms, are introduced, optimization is made for BIST 30 stocks by using these algorithms and the results obtained from these three algorithms are compared.

Keywords: *Portfolio Return, Portfolio Standard Deviation, Risk, Optimization, Artificial Intelligence Algorithms, Financial Analysis.*

JEL Classification: G11, G32, C61

¹ **Bibliyografik Bilgi (APA):** FESA Dergisi, 2022; 7(1) , 164-176 / DOI: 10.29106/fesa.1084231

*Danyel Bekdaş, İstanbul Ticaret Üniversitesi, *İstanbul -Türkiye*, danyelbekdas@yahoo.com, ORCID: 0000-0002-3827-0431

**Doç. Dr. Hicabi Ersoy, İstanbul Ticaret Üniversitesi, *İstanbul - Türkiye*, hersoy@ticaret.edu.tr, ORCID: 0000-0002-3573-1976

1. Giriř

İnsanlar veya řirketler kazançlarının bir bölümünü ihtiyaları için harcarken, geriye kalan tasarrufları ile yatırım yaparlar. Yapılan bu yatırımlar bireysel olarak yapılabileceđi gibi bireysel tasarruflar toplanarak kurumsal tüzel kiřiler aracılıđıyla da yatırım yapmak mümkündür. Ancak; ister bireysel olsun ister kurumsal olsun yapılan tüm yatırımlarda temel ama tasarrufların korunması ve buna ek olarak mümkün olabilecek en fazla getiriyi elde etmektir.

Yatırımcıların, tüm tasarruflarını tek bir yatırım aracında deđerlendirmesi mümkün olduđu gibi, bu araların belirli bir yüzdesinden oluřan portföy oluřturmakta da mümkündür. Dolayısı ile portföy olarak adlandırılan ve yatırım ürünlerinden oluřan sepet, yatırımcının seçimine bađlı olarak farklı ürünler içermektedir. Bu nedenle portföy oluřturmada verilecek ilk karar yatırım aracını seçmek ve ikinci ařamada ise bu ürünlere hangi oranda yatırım yapmak olacaktır. Bu ařamada yatırımcıların risk algısı öne çıkmaktadır. Örneđin risk algısı düşük olan yatırımcıların sabit getirili mevduat veya tahvil ürünlerini seçerken, risk algısı yüksek yatırımcılar ise hisse senedi, emtia gibi riskli ürünleri seçmektedir.

Getiri ile risk arasında dođru orantılı iliřki bulunmaktadır. Risk arttıka getiri artarken, risk azaldıka getiri azalmaktadır. Mevduat ve tahvil gibi sabit getiriyi garanti eden yatırım aralarında risk düşük ancak bunun yanında getiri de sınırlıdır. Öte yandan hisse senedi gibi günlük fiyat dalgalanması yani riski yüksek ürünlere kazanç da yüksek olabilmektedir. Dolayısı ile yatırımcıların getiri ile risk arasındaki dengeye bađlı olarak karar vermeleri gerekmektedir. Bir yandan kazancı arttırmayı alıřan yatırımcılar, diđer yandan pozisyon nedeniyle alınan riskleri azaltmak ve kontrol edebilmek için farklı ürünlere yatırım yaparak portföy oluřturmaya alıřırlar. Diđer bir ifade ile farklı yatırım aralarına aynı anda yatırım yaparak riskleri azaltmak ve yönetmek mümkün olabilmektedir. Bu durumda portföy yatırımlarında bir yandan getiri maksimize edilmeye alıřılırken, diđer yandan riskin minimize edilmesi amalanır. Dolayısı ile portföy yatırımlarının maksimum getiri altında minimum riske dayalı bir optimizasyon sorunu olduđu görülmektedir.

Günümüzde geliřen teknoloji ile yatırımcılar finansal verilere olduka kolay ulařabilmekte ve buna bađlı olarak arařtırma ve analizler yapabilmektedir. Zaman içinde karmařık matematiksel iřlemlere dayalı optimum portföy oluřturmada yeni yöntem yöntemler geliřtirilmektedir. Bilgisayar teknolojisinin geliřmediđi dönemlerde basit yöntemlerle yapılan portföy optimizasyonu, geliřen teknoloji yardımıyla karmařık ve farklı algoritmalar kullanılarak yapılmakta ve daha bařarılı sonuçlar elde edilmektedir.

Bu alıřmanın ilerleyen bölümlerinde konuya iliřkin literatür arařtırmasına yer verildikten sonra sıklıkla kullanılmaları nedeniyle yapay zeka algoritmalarından Jaya Algoritması Öğrenme-Öğretme Tabanlı Algoritma ve iek Tozlaşma Algoritması olarak bilinen 3 metasezgisel algoritmanın optimizasyon BIST hisse senetleri üzerinde portföy optimizasyonları kıyaslanmaktadır.

2. Literatür Taraması

Son yıllarda bilgisayar ve yazılım teknolojisinin geliřmesiyle üretilen yazılımlar portföy optimizasyonunda kullanılmıřtır. Özellikle Yapay Zeka algoritmalarından metasezgisel algoritmaların zaman olarak hızlı uygulanması ve bařarılı sonuç vermesi bakımından optimum portföyü oluřturmada sıklıkla uygulanmaktadır. Çünkü gerek olayların birođu çok fazla faktör, parametreye ve kısıta bađlı olduđu için optimum özüm bulabilmek için temel bilimsel yöntemlerden farklı yaklařımlara da ihtiyaç duyulmaktadır (Koziel ve Yang, 2011). Öte yandan üzerinde alıřılan problemlerin karmařıklık düzeyinin yüksek olması potansiyel tüm özümlerin ve özüm kombinasyonlarının bulunmasını zorlařtırmakta ve insanları zaman kısıtı altında en bařarılı sonuca odaklanmaya yönlendirmektedir. Bu bakımdan iteratif arama prosedürüne bađlı olarak alıřan metasezgisel algoritmalar eřitli problemlere özüm olabilmektedir.

Metasezgisel algoritmaların ilk örneklerinden olan Genetik Algoritma, Holland (1975) tarafından optimizasyon problemlerine özüm bulmak amacıyla üretilmiřtir. En eski algoritma olmasına bađlı olarak olduka sıklıkla kullanılan Genetik Algoritma; güçlü olanın hayatta kalması ve güçlünün zayıfı yenmesine dayalı olan genetik bilimi esasına dayalıdır. Holland (1975) tarafından geliřtirilen Genetik Algoritma, kolay adapte olabilen ve gelecek nesillerde güçlü organizmalara dönüşen biyolojik yapılardan esinlenilmiř ve iftleşme, üreme, klonlama/ođalma, aprazlama ve mutasyona olarak adlandırılan biyolojik süreçlere göre düzenlenmiř bir algoritmadır (Murty, 2003). Genetik Algoritma (GA) sermaye piyasalarında optimum portföy oluřturmak amacıyla çok sayıda alıřmaya konu olmuřtur. Oh ve diđerleri (2005), endeks fonlarının performanslarını arttırmak amacıyla Genetik Algoritma kullanımını önermiřlerdir. KOSPI 200 endeks verilerinin analizinde GA kullanılarak yapılan portföy optimizasyon önerisinde, GA yönteminin endeks fon yönetiminde bařarılı sonuçlar verdiđi sonucuna varılmıřtır. Ek olarak GA ile elde edilen portföyün piyasanın yatay olması durumunda vasat diđer bir ifade ile ortalama performans gösterdiđi de tespit edilmiřtir. Chang ve diđerleri (2010), düzeltilmiř genetik algoritma kullanarak Taywan 50 endeksi için portföy optimizasyonu yapmıřlardır. Yapılan alıřma kapsamında; minimum risk ve maliyet, farklı

oranlarda dađılım ve sabit faiz oranına gre optimum performans konularında testler yapılmıřtır. Yapılan testler sonucunda dzeltiymiř genetik algoritmanın optimum portfy oluřturmada katkı sađladıđı grlmřtir. ankal (2015) ise yaptıđı alıřmada Genetik Algoritmayı kullanarak BIST 30 endeks hisseleri iin optimum portfy nerisinde bulunmuřtur. 2004-2013 arasındaki verilerin kullanıldıđı analizde, portfy seiminde yararlanılan parametrik katsayı olarak deđiřim katsayısını kullanmıřtır. Oluřturulan 8 portfy iinde minimum risk ve maksimum getiriye sahip portfyde genetik algoritma kullanıldıđı tespit edilmiřtir.

Genetik Algoritma dıřında dođadaki canlıların davranıřlarını baz alan ve sr zekası teknikleri iinde yer alan Paracık Sr Optimizasyonu (PSO) 1995 yılında Kennedy ve Eberhart tarafından geliřtirilmiř ve portfy optimizasyonunda sıklıkla kullanılan yntemlerden biri haline gelmiřtir. Tunchan (2009), Paracık Sr Optimizasyon tekniđini kullanarak portfy optimizasyonu alıřması yapmıřtır. Hang Seng, DAX 100, FTSE 100, S&P 100 endeksleri kullanılarak yapılan analiz sonuları; Genetik Algoritma, Tabu Arama ve Tavlama Benzetimi yntemleriyle karřılařtırılmıřtır. Yapılan alıřma sonrasında Paracık Sr Algoritmasının daha iyi sonular verdiđi grlmřtir. Benzer Őekilde Golmakani ve Fazel (2011), yaptıkları alıřmada kısıt altında PSO algoritması ile portfy optimizasyonu alıřmasını yrtmř ve geliřtirilen algoritmada kısıt olarak; sınır kořulları, hisse sayısı, minimum iřlem maliyet ve sektr kapitalizasyon kısıtları kullanılmıřtır. Analiz sonuları Genetik Algoritma ile kıyaslanmıř ve PSO sonularının daha etkin olduđu tespit edilmiřtir. Zhu ve diđerleri (2011) de, yaptıkları alıřmada Paracık Sr Optimizasyon yntemi kullanarak Shanghai Stock Exchange 50 Endeks hisselerinde oluřan 8 adet, 15 adet ve 49 adet hisseli portfy oluřturmuř ve elde edilen sonular Genetik Algoritma ile mukayese edilmiřtir. Sonu olarak, PSO'nun analiz edilen verilerde yksek hesaplama verimliđi gsterdiđi tespit edilmiřtir. elenli ve diđerleri (2015) ise, BIST 30 hisselerine Paracık Sr Optimizasyon modelini kullanarak optimum portfy oluřturmuřtur. Analiz kapsamında optimum portfydeki hisseler ve portfydeki hisse oranları tespit edilmiřtir. Elde edilen sonular Ortalama Varyans ve Garanti Yakınsamalı PSO modelleri ile karřılařtırılmıř ve sonu olarak PSO modelinin sonularının daha bařarılı olduđu grlmřtir.

Optimizasyonda sıklıkla kullanılan diđer bir yntem olan Yapay Arı Kolonisi Algoritması (YAK) arıların yiyecek kaynađı bulma davranıřlarına gre tasarlanmıř bir yntemdir (Karabođa, 2007). YAK algoritması, bařarılı sonulara bađlı olduđu yođun kullanılan algoritmalar arasında bulunmaktadır. Wang ve diđerleri (2012), hisse sayısı kısıtı altında Yapay Arı Kolonisi Algoritması ile portfy optimizasyonu nerisi yapmıřlardır. Hang Seng, DAX 100, FTSE 100, S&P 100 ve Nikkei 225 endeks hisselerinin veri olarak kullanıldıđı analiz sonuları; Genetik Algoritma, Tabu Arama, Tavlama Benzetimi, Paracık Sr Optimizasyon yntemleri ile karřılařtırılmıřtır. Sonu olarak Yapay Arı Kolonisi ynteminin daha kaliteli sonuları verdiđini belirtilmiřtir. elenli (2018) ise; Yapay Arı Kolonisi Algoritması ile elde edilen optimum portfyn Sharpe oranını, karesel programlama, genetik algoritma ve paracık sr optimizasyonu teknikleri ile kıyaslamıřtır. BIST 30 verilerinin kullanıldıđı analizde Yapay Arı Kolonisi algoritmasının maksimum getiri ve minimum varyansı ieren optimum portfy oluřturmada daha etkin sonu verdiđi belirtilmiřtir. Kartal (2015) da, 2005 yılında geliřtirilen meta sezgisel yntem olan Yapay Arı Kolonisi ile minimum risk ve maksimum getiriye baz alan portfy optimizasyonu uygulaması nermiřtir. BIST 30 verilerinin kullanıldıđı analiz kapsamında elde edilen sonular Tavlama Benzetimi (TA), Tabu Arama (TA), Genetik Algoritma (GA) ve Paracık Sr Optimizasyonu teknikleri ile kıyaslanmıř ve elde edilen sonulardan Yapay Arı Kolonisi tekniđinin daha bařarılı sonu verdiđi gzlemlenmiřtir. Benzer Őekilde Chen ve diđerleri (2012), Yapay Arı Kolonisi Algoritmasını kullanarak Hang Seng, DAX 100, S&P100 ve Nikkei 225 endekslerinden oluřan portfy optimizasyonu yapmıřtır ve sonular; Tabu Arama, Tavlama Benzetimi, Komřuluk Arařtırması Algoritmaları ile kıyaslanmıřtır. Elde edilen sonulara gre; Yapay Arı Kolonisi yatırımcılara daha etkili portfy oluřturmada yardımcı olduđu tespit edilmiřtir. te yandan bu alıřmaların dıřında Kumar ve Mishra (2017), kovaryans tabanlı Yapay Arı Kolonisi Algoritması kullanarak portfy optimizasyonu iin yeni bir algoritma geliřtirmiřtir. Yapılan testler sonrasında nerilen algoritmanın gerek portfylerde getiriye arttırırken riski azalttıđı tespit edilmiřtir.

Yapay zeka optimizasyonu algoritmaları srekli geliřim iinde olan ve zerine ok fazla alıřma yapılan alanlardan biri olması nedeniyle, zaman iinde yeni algoritmalar geliřtirilmektedir. alıřma kapsamında son dnemlerde geliřtirilen algoritmalar Jaya Algoritması, đrenme-đretme Tabanlı Algoritma ve iek Tozlařma Algoritmaları detaylı bir Őekilde Őematik olarak gsterilecek, bu yntemler kullanarak portfy optimizasyonu alıřması yapılacak olup, elde edilen sonular karřılařtırmalı olarak sunulacaktır.

3. Metodoloji

Sezgisel yntemler temel prensip olarak iterasyonla zm arama ilkesine dayanır. Mevcut sonua gre daha bařarılı bir sonu elde edebilmek iin, matematiksel fonksiyon baz alınarak ok sayıda iterasyon yapılır ve bu arama genellikle ilk optimum sonuta biter. Ancak, elde edilen zmlerden daha bařarılı sonu bulmak amacıyla kullanılan random fonksiyonu ve yeniden bařlatma prosedr ođu zaman beklenen etkiyi sađlayamamaktadır. Metasezgisel yntemler ise genellikle bazı sezgisel yaklařımları birleřtirerek yerel arama ile bulunanlardan daha iyi zmler elde edilmesini sađlar (Onwubolu ve Babu, 2004).

Metasezgisel algoritmaların ana bileşenleri analiz edildiğinde temel olarak yoğunlaşma ve çeşitlendirme olmak üzere iki kısımdan oluştuğu görülmektedir. Çeşitlendirme sürecinde arama alanının tamamında potansiyel çözümler araştırılır ve üretilir. Yoğunlaşma sürecinde ise herhangi bir çözüm bölgesine odaklanarak, ilgili alandaki en iyi sonucu seçmeyi ifade etmektedir. Çok sayıda çözüm üretiminde random (rastgele) fonksiyonu büyük fayda sağlarken, çok sayıda çözüm üretilmesi yerel optimum gibi hatalı çözüm bulunmasını engellemektedir. Çeşitlendirme aşaması ise, algoritmanın arama alanını daha verimli bir şekilde araştırmasını sağlar (Gandomi ve diğ., 2013).

Çalışma kapsamında, son dönemlerde geliştirilmiş Yeni Nesil Yapay Zeka (Metasezgisel) Optimizasyon Yöntemleri kullanılacaktır. Söz konusu algoritmalar;

- Jaya Algoritması
- Öğrenme-Öğretme Tabanlı Algoritma (Teaching-learning Based Algorithm)
- Çiçek Tozlaşma Algoritması (Flower Pollination Algorithm)

kullanılacak olup, bu algoritmaların güçlü ve zayıf yönleri ve BIST 30 endeksi üzerinde elde edilen sonuçlar mukayese edilerek en başarılı sonucu veren algoritmanın belirlenmesine çalışılacaktır.

3.1. Jaya Algoritması ve Akış Diyagramı

Rao (2016) tarafından geliştirilen Jaya Algoritması, temelde Öğrenme-Öğretme Tabanlı algoritmaya benzer bir çalışma prensibi bulunmaktadır. Bu kapsamda Jaya algoritması her zaman optimum sonuca ulaşma yaklaşımına bağlı olarak çalışmaktadır. Buna göre algoritma, en iyi çözüme ulaşmayı ve en kötü çözümden uzaklaşmayı aynı anda gerçekleştirmeyi hedeflemektedir. Diğer yandan algoritma adımı, çalışma ve sonuca ulaşma amaçları ile uyum içerisinde olan, Sanskritçe bir sözcük olan ve zafer anlamına gelen Jaya kelimesinden almıştır.

Jaya algoritmasını diğer yöntemlerle kıyasladığımız zaman daha az sayıda fonksiyonu baz alarak çalışmaktadır. Bunun sonucunda optimum sonuca ulaşmada daha az işlem gerekmektedir. Ek olarak, algoritma uygulamasının basit olması ve özel parametreler içermemesi Jaya Algoritmasını diğer yöntemlerden görece olarak üstün kılmaktadır (Rao, 2016). Öte yandan algoritma en iyi ve en kötü sonucu diğer yöntemlere kıyasla daha küçük bir arama alanı etrafında aradığı için, Jaya Algoritmasının yerel optimum noktalara takılma ihtimalini arttırırken, daha iyi sonuçların gözden kaçmasına neden olabilmektedir.

Jaya algoritmasını aşağıdaki formül ile ifade edebilmek mümkündür:

$$X_{i,yeni} = X_{i,j} + \text{rand}() (X_{i,geniyi} - |X_{i,j}|) - \text{rand}() (X_{i,genkötü} - |X_{i,j}|) \quad (1)$$

Bu formülde yer alan denklemin ana bileşenleri aşağıdaki şekilde tanımlanabilir.

$X_{i,yeni}$: i. değişkeni için belirlenecek yeni değer

$X_{i,j}$: i. değişkeninin j. aday çözümünün başlangıç matrisindeki değeri

$X_{i,geniyi}$: Amaç fonksiyonu açısından en iyi olan çözüme ait i. değişkeni değeri

$X_{i,genkötü}$: Amaç fonksiyonu açısından en kötü olan çözümün yer aldığı vektördeki i. değişkeni değeri

3.2. Öğrenme-Öğretme Tabanlı Algoritma ve Akış Diyagramı

Rao ve diğerleri (2011), bir öğretmenin sınıftaki öğrencileriyle arasında öğretim ve öğrenme ilişkisini kullanarak, öğretim-öğrenme tabanlı optimizasyon algoritması önermişlerdir. Bu algoritmadaki ana prensip, öğrenciler ve tüm sınıfın bilgi düzeyine arttırmaya yönelik yinelenmelere dayalı bir süreç bulunmakta ve öğretmenin öğrenciler üzerindeki ve öğrencilerin kendi aralarındaki etkileşimi dikkate alınmaktadır. Bu yöntemde göre, öğretmen sınıfın bilgi düzeyini yine sınıfın kapasitesine göre ancak belli bir düzeye kadar yükseltebilmektedir. Bu değişim süreci birçok faktöre bağlı olmakla beraber rassal bir süreçtir. Bu kapsamda bir sınıftaki bilgi düzeyi en yüksek kişi olan öğretmen, sınıftaki öğrencileri belirlenen bir hedefe ulaşma konusunda etkilemekte ve öğretmenin yeteneğine göre tüm sınıfın bilgi düzeyinin artması beklenmektedir (Toğan, 2012).

Öğretim-Öğrenme Tabanlı Optimizasyonunun temelinde öğretmen ve öğrenci olmak üzere iki temel bileşen bulunmasına bağlı olarak, optimizasyon süreci de iki aşamadan oluşmaktadır. İlk aşamada öğretmenin sahip olduğu bilgi düzeyi ve öğretim yeteneği (öğretim aşaması) bulunurken, ikinci aşamada ise öğrencilerin kişisel çabaları ile öğrenme süreci ve birbirleri ile etkileşimi bulunmaktadır (öğrenci aşaması). Bu kapsamda;

- $j=1, 2, \dots, sn$ olmak üzere sn adet aday çözüm içerisinde ilk olarak en yüksek nota sahip/bilgi düzeyi en yüksek olan öğrenci seçilerek, diğer aşama için öğretmen olarak atanır ve bu bireyin belirli bir oranda (öğretme faktörü: TF) bireyin bilgilerini diğer öğrencilere aktarması ile notların iyileştirilmesi sağlanır.

$$X_{i,yeni} = X_{i,j} + \text{rand}() \left(X_{i,geniyi} - (TF) X_{i,ortalama} \right) \quad (2)$$

$$TF = \text{round} \left(1 + \text{rand}() \right) \quad (3)$$

- Öğrenci aşamasındaysa notlarında güncellemeler olan öğrenciler arasından iki farklı çözüm (a ve b) rassal olarak seçilir ve amaç fonksiyonu bakımından daha iyi durumda olan çözüme (problemin minimizasyon olması durumunda) bağlı olarak notlar, Denklem (3.4)' de ifade edildiği şekilde yeniden güncellenir.

$$X_{i,yeni} = \begin{cases} AF_a < AF_b, X_{i,j} + \text{rand}() (X_{i,a} - X_{i,b}) \\ AF_a > AF_b, X_{i,j} + \text{rand}() (X_{i,b} - X_{i,a}) \end{cases} \quad (4)$$

Bu formülde yer alan denklemin ana bileşenleri aşağıdaki şekilde tanımlanabilir.

$X_{i,yeni}$: i. değişkeninin yeni değeri

$X_{i,j}$: i. değişkeninin j. aday çözümünün (öğrenci) başlangıç matrisi değeri

$X_{i,geniyi}$: Genel çözümler arasından amaç fonksiyonu açısından en iyi değere sahip çözüme (öğretmen) ait i. değişkeni değeri

$X_{i,ortalama}$: i. değişkenine ait tüm aday çözüm değerlerinin ortalaması

$X_{i,a}$: i. değişkeninin a. aday çözüm değeri

$X_{i,b}$: i. değişkeninin b. aday çözüm değeri

a : Başlangıç matrisinden rassal olarak seçilen a çözümü

b : Başlangıç matrisinden rassal olarak seçilen b çözümü

AF : Amaç fonksiyonu değerinin yer aldığı vektör

AF_a : a' nin amaç fonksiyonu değeri

AF_b : b' nin amaç fonksiyonu değeri

sn: Toplam öğrenci sayısı

min () : Belirli sayıda değer arasından en küçük olanı belirleyen fonksiyon

round () : İçerisinde yer alan ondalıklı sayıyı kendisine en yakın tam sayıya yuvarlayan fonksiyon

3.3. Çiçek Tozlaşma Algoritması ve Akış Diyagramı

Çiçekli bitkiler sahip olduğu özel renk, koku ve aromatik salgılar nedeniyle arı, sinek ve diğer böcekleri kendine çekebilme özelliğine sahiptir. Bu özelliklere ek olarak rüzgar, su gibi doğal kaynakların da yardımıyla böcekler çiçeklere yönelerek türlerinin devamı için tozlaşma sürecine katkı sağlamaktadır. Polen akışı adı verilen bu süreç; aynı türden ancak farklı bitkiler arasında olması durumunda çapraz tozlaşma, tek bir çiçek veya bir bitkinin farklı çiçekleri arasında gerçekleşmesi durumunda da kendi kendine tozlaşma adını almaktadır.

Yang ve diğerleri (2015), bu süreçten ilham alarak Çiçek Tozlaşma Algoritmasını geliştirmiş ve geliştirilen bu algoritma günümüzde sıklıkla kullanılan metasezgisel yöntemler arasında yerini almıştır. Çiçek Tozlaşma Algoritmasında, çiçekli bitkilerden esinlenerek formüle edilmiş ve tozlaşma işleminin özelliği, tozlaşma davranışı ve çiçek sabitliği ile ilgili olan dört farklı kural göz önünde bulundurulmaktadır (Yang ve diğ., 2013; Yang ve diğ., 2016):

- I. apraz tozlaşma, aynı türden farklı bitkilerin iekleri arasındaki polen aktarımı ile gerekleşir. Polen taşıyıcılarının uzak adımlarla zıplayarak veya uçarak Lévy dağılımı kurallarına uymaları gerekmektedir. Bu sürece küresel tozlaşma süreci denmektedir.
- II. Kendi kendine tozlaşma ise polenlerin, bir ieğin kendi içerisinde veya aynı bitkinin farklı iekleri arasındaki aktarımından dolayı oluşur. Bu tozlaşma türü ise yerel tozlaşmadır.
- III. iek sabitliği durumu ise polen taşıyıcıları ve iek türleri arasındaki işbirliğidir. Bu, iek tozlaşma sürecindeki bir gelişmedir.
- IV. Yerel ve küresel tozlaşma, anahtar olasılığı olarak adlandırılan ve 0 ile 1 arasında bir değere sahip olan olasılık değeri ile kontrol edilir.

Tüm bu kuralların uygulanmasıyla gerekleştirilecek optimizasyon sürecinde, optimum değerlerin elde edilmesi için iki farklı yol izlenmektedir. Bunun için iek tozlaşma algoritması parametrelerinden biri olan arama değışimi/anahtar olasılığı (sp) kontrol edilerek arama türünün belirlenmesi gerekmektedir.

- Eğer sp değeri rassal olarak üretilecek bir sayıdan daha büyükse, çözümlerin daha geniş bir alandan aranarak belirlendiği küresel (global) arama,
- Bu değeri üretilen rassal sayıdan daha küçükse, çözümlerin daha küçük bir alandan arandığı yerel (lokal) arama süreci olarak adlandırılmaktadır.
- iek tozlaşma algoritmasını aşağıdaki formül ile ifade edebilmek mümkündür:

$$X_{i,yeni} \begin{cases} sp > rand(), X_{i,j} + Lévy (X_{i,geniyi} - X_{i,j}) \\ sp < rand(), X_{i,j} + rand() (X_{i,m} - X_{i,k}) \end{cases} \quad (5)$$

$$k = ceil(rand() \cdot nf) \quad (6)$$

$$m = ceil(rand() \cdot nf) \quad (7)$$

$$Lévy = \left(\frac{1}{\sqrt{2\pi}} \right) (rand())^{-1.5} e^{\left(-\frac{1}{2 \cdot rand()} \right)} \quad (8)$$

Bu formülde yer alan denklemin ana bileşinleri aşağıdaki şekilde tanımlanabilir.

$X_{i,yeni}$: i. değışkeninin yeni değeri

$X_{i,j}$: i. değışkeninin j. aday çözümünün (öğrenci) başlangıç matrisi değeri

$X_{i,geniyi}$: Genel çözümler arasında amaç fonksiyonu açısından en iyi değere sahip çözüme (öğretmen) ait i. değışkeni değeri

$X_{i,k}$: i. değışkeninin k. aday çözüm değeri

$X_{i,m}$: i. değışkeninin m. aday çözüm değeri

AF : Amaç fonksiyonu değerinin yer aldığı vektör

K : Başlangıç matrisinden rastgele seçilen k. iek (aday vektör)

M : Başlangıç matrisinden rastgele seçilen m. iek (aday vektör)

4. BIST 30 Uygulaması ve Bulgular

alıřma kapsamında yapay zeka optimizasyon yöntemlerinden olan metasezgisel algoritmalarından; Jaya Algoritması (Rao, 2016), Öğrenme-Öğretme Tabanlı Algoritma (Rao vd., 2011) ve iek Tozlaşma Algoritması (Yang, 2011) kullanılarak hisse senedi portföy optimizasyonu yapılması amaçlanmaktadır. Ayrıca bu üç yöntem sonucu elde edilen sonuçlar kıyaslanarak elde edilen portföylerin performansları değerlendirilecektir.

Matlab Programı kullanılarak yukarı sıralanan üç yöntem için portföy optimizasyonu analiz edilecek ve portföylerin performansları getiri/risk oranı ile belirlenecektir. Bu kapsamda programdan elde edilen sonuçlar aşağıdaki şekilde özetlenebilir:

- Portföyün beklenen getirisi
- Portföyün riski
- Portföyün getiri/risk oranı
- Portföydeki hisse sayısı
- Portföydeki hisse senedi ağırlıkları

Analiz kapsamında tüm algoritmalarda ilk yapılacak işlem hisse senetlerinin getirisini hesaplamak olacaktır. Analizde günlük verilerin kullanılmasına bağılı olarak, hisse senedinin günlük getiri oranı

$$Getiri (R_i) = \frac{X_{t+1} - X_t}{X_t} * 100 \quad (9)$$

X_{t+1} ; hisse senedinin t+1 döneminde fiyatı

X_t ; hisse senedinin t döneminde fiyatı

R_i ; i hisse senedinin günlük getirisi

Hisse senetlerinin günlük getirisi (9) formülü kullanılarak hesaplandıktan sonra aynı hisse senetlerinin 253 günlük (1 yıllık) getirisi de aşağıdaki formül kullanılarak hesaplamak mümkündür:

$$\mu_i = \frac{\sum_{i=1}^N R_i}{N-1} \quad (10)$$

μ_i ; hisse senedinin beklenen getiri

N; yatırım yapılan gün sayısı

Portföy birden fazla sayıda hisse senedinden meydana geldiği dikkate alındığında, tüm portföyün beklenen getirisi de aşağıdaki şekilde hesaplanabilir:

$$E(R_p) = \sum_{i=1}^n P_i * \mu_i \quad (11)$$

$E(R_p)$; tüm portföyün beklenen getirisi

μ_i ; i. hisse senedinin beklenen getirisi

P_i ; i. hisse senedinin portföydeki ağırlığı

n; hisse senedi sayısı

Jaya Algoritma, Öğrenme-Öğretme Tabanlı Algoritma ve Çiçek Tozlaşma Algoritması dahil tüm yöntemlerde BIST 30 endeksinde yer alan hisse senetleri dikkate alınmış olup, ilgili hisse senetlerinin 01/2020-05/2021 arasındaki dönemde kapanış fiyatları kullanılarak analiz yapılacaktır.

Jaya Algoritma, Öğrenme-Öğretme Tabanlı Algoritma ve Çiçek Tozlaşma Algoritmasında amaç fonksiyonu getiri/risk oranı belirlenirken, fonksiyon kısıtlarında ise ortalama varyans modeli dikkate alınmıştır. Bu fonksiyonları aşağıdaki şekilde özetlemek mümkündür:

$$\text{Amaç fonksiyonu: } \text{Max} \frac{E(R_p)}{\sigma_p} = \frac{\sum_{i=1}^N P_i * \mu_i}{\sum_{i=1}^N \sum_{j=1}^N x_i * x_j * \sigma_{ij}} \quad (12)$$

$$\text{Kısıt Fonsiyonları: } \sum_{i=1}^N x_i = 1 \quad (13)$$

$$0 \leq x_i \leq 1 \quad i = 1, 2, 3, \dots, N$$

Tüm algoritmalarda aynı amaç ve kısıt fonksiyonları kullanılmış ve algoritmalarından elde edilen sonuçlar karşılaştırılmıştır.

- Analiz sonucunda BIST 30 hisseleri içerisinde çeşitli ağırlıkta EREGL, GUBRE ve SASA hisselerinden oluşan portföyün optimum olduğu görülmüştür. Söz konusu hisseler için algoritmaların bulunduğu optimum ağırlık değerleri
 - ✓ Jaya Algoritması (JA): %43.1 EREGL, %36.2 GUBRF, %20.7 SASA

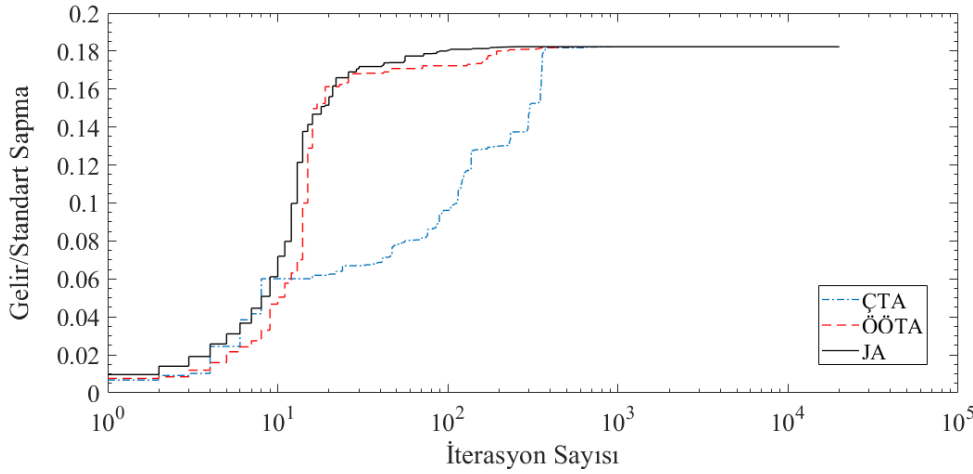
✓ Öğrenme-Öğretme Tabanlı Algoritma (ÖÖTA): %43.1 EREGL, %36.2 GUBRF, %20.7 SASA

✓ Çiçek Tozlaşma Algoritması (ÇTA): %43 EREGL, %36.3 GUBRF, %20.7 SASA

şeklinde olup kullanılan algoritmalar aynı hisselerde oldukça yakın sonuçlar verdiği anlaşılmaktadır.

- Optimizasyon metodolojisinde portföydeki hisse ağırlıkları belirlenirken, kullanıcı tarafından tanımlanan belli bir değerin altında ağırlık yüzdesi olması durumunda, optimum portföy içerisinde değeri sıfır atanarak ihmal edilir. Bundaki amaç portföy takibi açısından kolaylık sağlamaktır. Çalışmada bu değer %1 olarak tanımlanmıştır.
- Aşağıda detayları verilen sonuç tablolarında (Tablo1-3) da görüleceği üzere, algoritmaların sonuçları her bir popülasyonda (her bir algoritmada 20 adet olarak belirlendi) birbirine yakınsamaktadır. Bu gelişme elde edilen sonuçların doğruluğunu göstermesi bakımında ve güvenilirlik bakımından istenen bir durumdur.
- Tüm algoritmalarda günlük getiri %0.49 ve standart sapma %2.68 iken, getiri/standart sapma oranı da 0.1823 olarak gerçekleşmiştir.
- Algoritmalarda maksimum iterasyon sayısı 20 bin olarak belirlenmiş olup,
 - ✓ Jaya Algoritması 6.450
 - ✓ Öğrenme-Öğretme Tabanlı Algoritma 14.143
 - ✓ Çiçek Tozlaşma Algoritması 9.807 iterasyonla optimum sonuca ulaşmıştır.
 - ✓ Aşağıdaki grafikte algoritmaların performansı görülmektedir.

Şekil 4. Algoritma Performanslarının Kıyaslanması



5. Sonuç

Hisse senedi yatırımlarında menkul kıymet seçimi ve bu varlıkların portföydeki ağırlıkları olmak üzere iki temel karar bulunmaktadır. Bununla birlikte, hisse senedi portföy yatırımlarında getiri arttıkça risk de artmaktadır. Bu kapsamda getiri ile risk arasında denge kurulması için bilgisayar programlarının kullanımı yaygınlaşmakta ve sonucun optimize edilmesi için yapay zeka algoritmaları da sıklıkla uygulanmaktadır. Çalışma kapsamında yapay zeka optimizasyon algoritmalarından meta sezgisel algoritmalar kullanılarak BIST 30 hisse senetlerinden optimum portföy elde edilmesi için analiz yapılmıştır. Optimum sonuçlar bakımında algoritmaların performansı değerlendirilecek olursa, tüm algoritmaların optimum sonuca ulaştığı anlaşılmaktadır. Buna göre algoritmaların optimizasyon problemi için etkin algoritmalar olduğu söylenebilir. Yapay zeka optimizasyon algoritmalarının kıyaslandığı bir diğer parametre optimum sonuca ulaşmadaki hızlarıdır. Buna göre; Jaya Algoritmasının en hızlı optimum sonuca ulaşan algoritma olduğu görülmektedir. Sonuca ulaşma ve hız bakımından Jaya Algoritmasının diğer algoritmalarla kıyasla daha iyi olduğu anlaşılmaktadır.

Tablo 1. Jaya Algoritması Optimizasyon Sonuçları

AKBNK	ARCLK	ASELS	BIMAS	DOHOL	EKGYO	EREGL	GARAN	GUBRF	HALKB	ISCTR	KCHOL	KOZAA	KOZAL	KRDMD	OYAKC	PETKM	PGSUS	SAHOL	SASA	SISE	TAVHL	TCELL	THYAO	TKFEN	TSKB	TTKOM	TUPRS	VAKBN	YKBNK	Getiri	St. Sapma	Getiri/STD
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48984	2.68669	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48966	2.6857	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48986	2.6868	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48979	2.68643	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.4898	2.6865	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48976	2.68626	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48993	2.68719	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.49009	2.68806	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48998	2.6875	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48981	2.68652	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.4899	2.68702	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48974	2.68616	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48997	2.68743	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.4899	2.68705	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48984	2.68671	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48979	2.68641	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48966	2.68572	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48983	2.68664	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48975	2.68622	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48986	2.68683	0.18232

Tablo 2. Öğrenme-Öğretme Tabanlı Algoritma Optimizasyon Sonuçları

AKBNK	ARCLK	ASELS	BIMAS	DOHOL	EKGYO	EREGL	GARAN	GUBRF	HALKB	ISCTR	KCHOL	KOZAA	KOZAL	KRDMD	OYAKC	PETKM	PGSUS	SAHOL	SASA	SISE	TAVHL	TCELL	THYAO	TKFEN	TSKB	TTKOM	TUPRS	VAKBN	YKBNK	Getiri	St Sapma	Getiri/STD
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48998	2.68746	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48977	2.68633	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48968	2.68584	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.48996	2.68734	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.49001	2.68762	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.49008	2.68803	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.49006	2.68789	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48992	2.68716	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48979	2.68642	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.48998	2.68747	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48983	2.68665	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.4897	2.68595	0.18232
0	0	0	0	0	0	43.1%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48981	2.68653	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.48999	2.6875	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48986	2.68682	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48981	2.68654	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.48979	2.68642	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.49006	2.68792	0.18232
0	0	0	0	0	0	43.0%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.8%	0	0	0	0	0	0	0	0	0	0	0.4899	2.68703	0.18232
0	0	0	0	0	0	43.1%	0	36.2%	0	0	0	0	0	0	0	0	0	0	20.6%	0	0	0	0	0	0	0	0	0	0	0.48961	2.68546	0.18232

Tablo 3. Çiçek Tozlařma Algoritma Optimizasyon Sonuçları

AKBNK	ARCLK	ASELS	BIMAS	DOHOL	EKGYO	EREGL	GARAN	GUBRF	HALKB	ISCTR	KCHOL	KOZAA	KOZAL	KRDMD	OYAKC	PETKM	PGSUS	SAHOL	SASA	SISE	TAVHL	TCELL	THYAO	TKFEN	TSKB	TTKOM	TUPRS	VAKBN	YKBNK	Getiri	St Sapma	Getiri/STD
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232
0	0	0	0	0	0	43.0%	0	36.3%	0	0	0	0	0	0	0	0	0	0	20.7%	0	0	0	0	0	0	0	0	0	0	0.48994	2.68727	0.18232

Kaynakça

- Chang, Jui-Fang, Tien Chin Wang, and Yuan-Tzu Min. "Using Genetic Algorithms to construct a low-risk fund portfolio based on the Taiwan 50 Index." 2010 International Conference on Computational Aspects of Social Networks. IEEE, 2010.
- Cura, Tunchan. "Particle swarm optimization approach to portfolio optimization." *Nonlinear analysis: Real world applications* 10.4 (2009): 2396-2406.
- Çankal, Ahmet. "Genetik Algoritma Kullanarak Hisse Senedi Portföy Optimizasyonu: BİST-30'DA Bir Uygulama." (2015).
- Çelenli, Azize Zehra, Erol Eğriođlu, and Burçin Şeyda Çorba. "İMKB 30 indeksini oluřturan hisse senetleri için parçacık sürü optimizasyonu yöntemlerine dayalı portföy optimizasyonu." (2015).
- Çelenli A (2018). Yapay Arı Kolonisi Algoritması ile Sharpe Performans Oranına Dayalı Portföy Optimizasyonu: BIST 30 Uygulaması Doktora Tezi Doktora Tezi, Ondokuz Mayıs Üniversitesi.
- Chen, Angela HL, Yun-Chia Liang, and Chia-Chien Liu. "An artificial bee colony algorithm for the cardinality-constrained portfolio optimization problems." 2012 IEEE Congress on Evolutionary Computation. IEEE, 2012.
- Gandomi, A.H., Yang, X.S., Talatahari, S., Alavi, A.H. (Eds.), 2013, *Metaheuristic Algorithms in Modeling and Optimization, Metaheuristic Applications in Structures and Infrastructures*, Elsevier, ISBN: 9780123983640, 1-24.
- Golmakani, Hamid Reza, and Mehrshad Fazel. "Constrained portfolio selection using particle swarm optimization." *Expert Systems with Applications* 38.7 (2011): 8327-8335.
- Holland J H (1975). *Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence*. Ann Arbor, MI: University of Michigan Press
- Karabođa, D., Basturk, B., 2007, A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm, *Journal of Global Optimization*, 39(3), 459-471.
- Kartal B (2015). Yapay Arı Kolonisi Algoritması ile Finansal Portföy Optimizasyonu. Doktora Tezi Doktora Tezi, İstanbul Üniversitesi İşletme Anabilim Dalı, 143, Sosyal Bilimler Enstitüsü. (391453).
- Kennedy, James, and Russell Eberhart. "Particle swarm optimization." *Proceedings of ICNN'95-international conference on neural networks*. Vol. 4. IEEE, 1995.
- Koziel, S., Yang, X.S. (Eds.), 2011, *Computational Optimization, Methods and Algorithms* (vol. 356), Springer-Verlag, Heidelberg, Berlin, ISBN: 978-3-642-20858-4.
- Kumar, Divya, and K. K. Mishra. "Portfolio optimization using novel co-variance guided Artificial Bee Colony algorithm." *Swarm and Evolutionary Computation* 33 (2017): 119-130.
- Murty, K.G., 2003, *Optimization models for decision making: volume 1*, Ann Arbor, University Of Michigan, http://www_personal.umich.edu/~murty/books/opti_model/, [Ziyaret Tarihi: 2 Temmuz 2018].
- Oh, Kyong Joo, Tae Yoon Kim, and Sungky Min. "Using genetic algorithm to support portfolio optimization for index fund management." *Expert Systems with Applications* 28.2 (2005): 371-379.
- Onwubolu, G.C., Babu, B.V., 2004, *New Optimization Techniques in Engineering* (vol. 141), Springer-Verlag, Heidelberg, Berlin, ISBN: 978-3-540-39930-8.
- Rao, R., 2016, *Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems*, *International Journal of Industrial Engineering Computations*, 7(1), 19-34.
- Rao, R. Venkata, Vimal J. Savsani, and D. P. Vakharia. "Teaching-learning-based optimization: a novel method

for constrained mechanical design optimization problems." *Computer-Aided Design* 43.3 (2011): 303-315.

Tođan, V., 2012, Design of planar steel frames using teaching–learning based optimization, *Engineering Structures*, 34, 225-232.

Wang, Zhen, Sanyang Liu, and Xiangyu Kong. "Artificial bee colony algorithm for portfolio optimization problems." *International Journal of Advancements in Computing Technology* 4.4 (2012): 8-16.

Yang, X.S. (Eds.), Koziel, S., 2011, *Computational Optimization, Methods and Algorithms* (vol. 356), Springer-Verlag, Heidelberg, Berlin, ISBN: 978-3-642-20858-4.

Yang, X.S., Koziel, S., Leifsson, L., 2013, Computational optimization, modelling and simulation: recent trends and challenges, *International Conference on Computational Science ICCS 2013, 5-7 Haziran 2013 Barselona-İspanya, Procedia Computer Science*, 855-860.

Yang, X.S., Bekdař, G., Nigdeli, S.M. (Eds.), 2016, *Metaheuristics and Optimization in Civil Engineering*, Springer, Switzerland, ISBN: 9783319262451.

Zhu, Hanhong, et al. "Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem." *Expert Systems with Applications* 38.8 (2011): 10161-1016