

Uçuş Aşamalarının Bölümlendirilmesi: Havacılık Verilerinde Gözetimsiz Öğrenme Uygulaması

Oğuz BEKTAŞ

İstanbul Medeniyet Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Makine Mühendisliği Bölümü, İstanbul.

PhD: University of Warwick, MSc: Queen Mary University of London

e-posta: oguz.bektas@medeniyet.edu.tr ORCID ID: https://orcid.org/0000-0003-3687-3703

Geliş Tarihi: 04.03.2020

Kabul Tarihi: 02.12.2020

Öz

Anahtar kelimeler
Havacılık; Hava Araçları; Uçuş Verileri ve Rejimleri; Gözetimsiz Öğrenme; Özdüzenleyici Haritalar Yöntemi; Kümeleme

Hava araçları, işlevleri ve yapıları gereği, buldukları operasyon boyunca birden fazla değişken rejim içinde bulunmaktadır. Kalkış, iniş, tırmanış, düz uçuş, yatay düzlemde dönüş vb durumlarda hava aracından elde edilen veriler farklılık göstermekte ve sinyal değerleri içinde buldukları çevre ve uygulamalara göre değişmektedir. Bu makalede, aynı grupta bulunan uçuş veri kümelerinin diğer veri kümelerine göre daha ayırt edici olan karakteristik özelliklerine göre sınıflandırılması ve uçuş rejimlerinin bu kümeler amacıyla saptanması amaçlanmıştır. Bu amaçla, yapay sinir ağlarının gözetimsiz bir eğitim biçimini kullanan "Özdüzenleyici Haritalar" yöntemi kullanılmış ve bölümlendirme işlemi rekabetçi öğrenme ve haritalama sistemiyle tamamlanmıştır. Böylece, birden fazla boyutta bulunan farklı verilerin işaretlenerek, rejim bazında çalışılmasına olanak sağlanmıştır.

Segmentation of Flight Stages: An Unsupervised Learning Model for Aviation Data

Keywords
Aerospace; Aircrafts; Flight Data and Regimes; Unsupervised Learning; Self-Organizing Maps; Clustering

Abstract

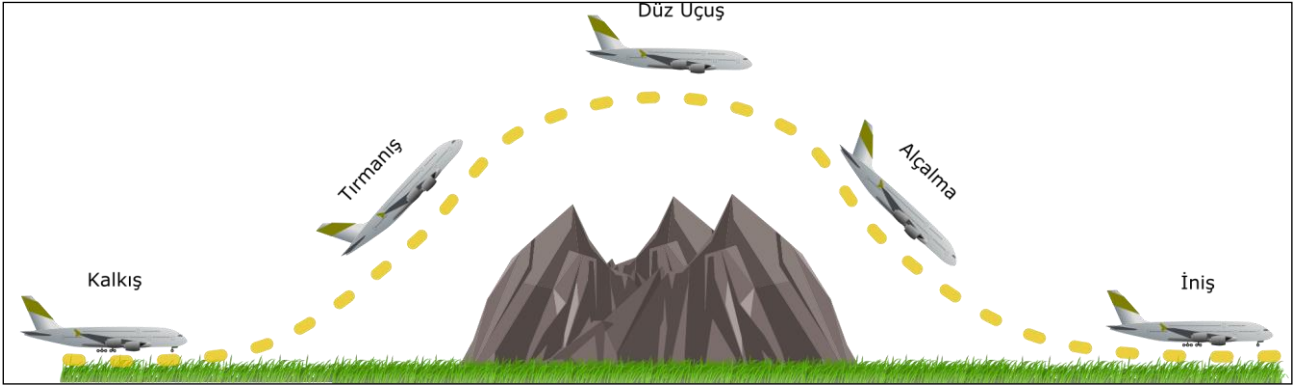
Due to their functions and characteristics, the aircrafts function in multiple variable regimes throughout their operations. In the cases of take-off, landing, climbing, cruise flight, rolling etc., the condition monitoring data from an aircraft differ and their values vary according to the environment and applications in which they are in. In this article, it is aimed to classify the flight data clusters in the same group as more distinctive than other clusters and to determine flight regimes for these data. For this purpose, the "Self-Organizing Maps" method, which uses a type of unsupervised neural network training, is used and the clustering is completed with a competitive learning and mapping system. Thus, it is possible to work on regime basis by marking different data in multiple regime dimensions.

© Afyon Kocatepe Üniversitesi

1. Giriş

İçerdiği geniş kapsamlı aşamalar nedeniyle, uçuşu basit bir tanımla sınırlandırmak kolay değildir- hava taşıtları, taşıma kapasitelerine ve her nevi insanlı ya da insansız uçan çok geniş bir yelpazedeki araçların çeşitliliğine göre farklı uçuş rejimlerine sahip olabilmektedir. Genel olarak, kimi hava taşıtlarına ait verilerin toplanması ve işlenmesi göreceli olarak daha az önem teşkil etse de genişleyen ticari havacılık sektörünün ve yüksek güvenilirlik gerektiren yolcu uçaklarının ürettiği "durum izleme verileri" giderek artan bir öneme sahiptir.

"Nesnelerin İnterneti" ve "Büyük Veri Analitiği" ile ilgili uygulamalar arttıkça ve yeni güncellemeler geldikçe bu verilerin daha detaylı olarak uygulanabilme imkânı ortaya çıkmıştır (Bektaş 2020a). Her ne kadar, koruyucu bakım uygulamaları kompleks sistemlerden yüksek verim almak ve üretim kayıplarını en aza indirmek için, operasyon süresince ölçüm cihazları ile izleme olarak sonuçları analiz etse de; havacılık sistemlerinde bulunan farklı uçuş aşamalarının analiz edilmesi bu verilerin istenilen seviyede dönüştürülmesi, daha iyi anlaşılması ve ön işleme tabi tutulabilmesi için gereklidir.



Şekil 1. Uçuş operasyon aşamaları

Özellikle, küme analizi (veya yaygın bilinen ismi ile kümeleme) bu ön işlemenin başlıca aşamalarından biri olup sinyal işleme problemlerini çözmek için kullanılan dönüştürme işlemlerinden biridir. Farklı boyutlarda bulunan sinyal değerlerinin aynı sınıflandırmada yer alması için diğer grup elementlerinden farklı olarak herhangi bir şekilde birbirine paralel olup olmadıklarına karar verilebilir. Böylece, verilerin tek düzeye indirgenebilmeleri ve/veya istenilen düzeyde etiketlenilerek buldukları boyutta işlenebilmelerine olanak sağlanabilmektedir. Kümeleme yöntemleri, temelde bir sınıflandırmayı nelerin oluşturduğu ve bunların nasıl etkili bir şekilde tespit edileceği bakımından farklılık gösterir.

Havacılık sistemlerinde operasyon durumunun izlenmesinde başlıca sorun, verilerin çok boyutlu ve gürültülü olmalarından dolayı sahip oldukları karakteristikleridir. Veri analizi doğru bir şekilde ele alınsa bile, veri işleme birden fazla operasyonel rejim yapısının neden olacağı kısıtlamaların üstesinden gelebilecek etkili tekniklere dayanmalıdır. Ayrıca, bu verilerin anlamlı ve kullanışlı bilgilere dönüştürülmesine ihtiyaç vardır. Literatürde, Fourier Dönüşümü (Bracewell 1986, Ozaktas ve Kutay 2001, Lindon and Ferrige 1980), Dalgacık Dönüşümü (Aballe et al. 1999, Garguet-Duport et al. 1996, Koçyiğit ve Korürek 2010), Kalman filtresi (Chui and Chen 2017) gibi birçok veri işleme stratejisi olmuştur. Bunlar gibi sinyal işleme yaklaşımları çok çeşitli problemlerle başa çıkmak ve farklı olayları anlamak için vazgeçilmez bir rol oynarken, küme analizi her zaman önemli ve birincil

bir görev olmuştur. Kümeleme algoritmalarının temel amacı, büyük yapılandırılmış veya yapılandırılmamış veri kümelerini anlamlandırmak ve bunlardan daha kullanışlı olacak değerler çıkarmaktır. Büyük boyutlu yapılandırılmamış verilerle çalışıldığında, verileri analiz etmeye başlamadan önce istikrarlı bir tür gruplamaya ayırmaya çalışmak mantıklıdır. Kümeleme ve verileri sınıflandırma, toplu olarak kapsamlı bir bakış açısına sahip olmaya ve ardından yapılacak olan analize daha derinlemesine girmeden önce bazı mantıksal yapılar oluşturulmasına olanak tanır. Kümeleme yöntemleri, bir kümedeki iki veri noktasının birbirine benzeyeceği şekilde belirli sayıda grup veya kategoriye verileri tahsis ederken, farklı kümelere gelen veri noktalarını ayrı olarak sınıflar.

Havacılık sistemleri ve benzeri karmaşık sistemlerde, verilerin sınıflandırılması ve kümelmesi amacıyla Gauss karışım modeli (McLachlan and Peel 2004), en yakın komşu algoritması (Ramasso 2014), K-ortalama (Lam 2014), ve bulanık eseme c-ortalama (Sugeno 1993) gibi modeller kullanılmıştır. Kullanılan bu uygulamalardan gözetimli öğrenme algoritmaları farklı kümelere keskin bir sınıflandırma sağladığından ötürü; uçuş boyunca her bir rejimde bulunan verilerin centroidlerinin değişmesine karşı elverişsiz olabilmektedirler. Bu sorun k-ortalama gibi gözetimsiz modeller ile çözülebilsede, tam olarak bir öğrenme olmadığı için görece daha basit bu gözetimsiz modeller karmaşık verileri kümelemede kötü bir performans çıkarır. Bununla birlikte, günümüzde giderek kullanım alanları artan ve adını duyurmaya başlayan Big Data (Büyük Veri)

konusunun kümeleme ile birlikte ele alınması gerekmektedir. Her ne kadar bu olgu teknolojinin ilerlemesi ile önem kazanmış olarak görülse de yıllardır havacılık alanında durum izleme verilerinin elde edilmesi gelişimini tamamlamıştır. Fakat, bu boyutta veri içeriklerinin görece daha gelişmiş modellemeler ile analiz edilmesi gerekmektedir. Uçuş rejimlerinin karmaşıklığından (bknz. **Şekil 1**. Uçuş operasyon aşamaları) ve hedef tanımlamasının etkilemesinden kaçınmak için, verilerin sebep-sonuç biçiminde etiketlenmesinden farklı olarak birbirleri arasındaki ilişkiler genel olarak giriş verisi ile öğrenilebilir. Böylece, öğrenme için çıkış verisi sağlanmadan örneklerin birbirine olan uzaklıklarını, komşuluk ilişkilerini ve yoğunluklarını kullanarak sınıflandırma ile ilgili çıkarımlar yapılabilir (Bektaş 2020b). Tam bu noktada, özdüzenleyici haritalar yönteminin kullanılması, karmaşık uçuş rejimlerine sahip büyük verilerin yüksek performans ile kümelenebilmesi için bir avantaj sağlar. Burada modelin sunduğu bir diğer fayda ise; rejimlerin kolay bir biçimde ve iki boyutlu olacak şekilde görselleştirilmeleri; böylelikle bölümlenmek için kullanılmalarıdır. Ayrıca; özdüzenleyici haritalar diğer yöntemlere kıyasla hesaplama gereksinimlerini de azaltır.

Bu çalışma beş bölümden oluşmaktadır. İkinci bölüm, alan ve modelle ile ilgili geçmişte bulunan çalışmalar ve literatürü içermektedir. Sonrasındaki bölüm ise uygulama seti ve özdüzenleyici haritalar yöntemini incelemektedir. Dördüncü bölüm, test uygulaması ve ortaya çıkan sonuçları analiz etmektedir. Son bölümde ise çalışmanın çözümlemesi, kısıtları ve gelecekte olabilecek potansiyel çalışma önerileri yer almaktadır.

2. Metot ve Materyal

Özdüzenleyici haritalar yönteminde öğrenmenin temel amacı, ağ bölümlerinin belirli girdi örüntülerine karşı benzer şekilde karşılık vermesini sağlamaktır. Ağ bölümleri aynı girişleri kendilerinin dahili parametreleriyle karşılaştırır. Sonrasında, kazanan (winner) olarak da adlandırılan en iyi eşleşmeye sahip bölüm (best match) daha sonra bu girdiye göre ayarlanır. Bu tür gerçek bir rekabetçi

(gözetimsiz) öğrenme, birincil verilerin kümelenebilmesini ve nihayetinde hiyerarşik olarak düzenlenmiş bir şekilde tespit edilmelerini sağlar (Özçalıcı 2017).

Bu çalışmada sunulan öğrenme algoritması, "X" girişinin bir nöral ağ yapısındaki "M" nöronları kümesine eşlenmesini sağlayan doğrusal olmayan bir topoloji oluşturur ve aşağıda belirtilen şekilde formülize edilir (Kohonen 1990, Wittek 2013, Wittek et al. 2017).

$$X = \{x(t) \mid t \in \{t_0, \dots, t_f\}\} \quad (1)$$

$$M = \{n_1, \dots, n_k\} \quad (2)$$

Burada, sırasıyla t_0 ve t_f eğitimin başlangıç ve bitiş noktalarını temsil etmektedir. n_1, \dots, n_k nöronları, aşağıdaki ağırlık vektörleriyle bir ızgaraya yerleştirilir.

$$W = \{w_1(t), \dots, w_k(t)\} \quad (3)$$

Daha sonra, her bir veri noktası için "M" nöronları kümesindeki en iyi eşleşen "BM" birim hesaplanır.

$$BM(x(t)) = n_b \in M \quad (4)$$

Böylece, "d" veri kümesindeki mesafe fonksiyonu özellik alanında aşağıdaki gibi düzenlenir:

$$d(x(t), w_b(t)) \leq d(x(t), w_j(t)) \quad (5)$$

$$\forall w_j(t) \in W, \quad (6)$$

Daha sonra dizi, yeni $w_j(t + 1)$ 'in eski $w_j(t)$ ve yeni $x(t)$ 'den yinelemeli olarak hesaplandığı düzgünleştirme tipi bir işlem olarak formüle edilir.

$$w_j(t + 1) = w_j(t) + \alpha h_{b_j}(t) (x(t) - w_j(t)) \quad (7)$$

Burada " α ", düzeltme boyutunu tanımlayan skaler bir öğrenme faktörüdür; değer " t " endeksi ile azalır. " $h_{b_j}(t)$ " konumu koşullara bağlı olan komşuluk işlevidir. Bu işlev genellikle Gaussian olarak alınır (Wittek et al. 2017).

$$h_{b_j} = \exp\left(\frac{-\|r_b - r_j\|}{\delta(t)}\right) \quad (8)$$

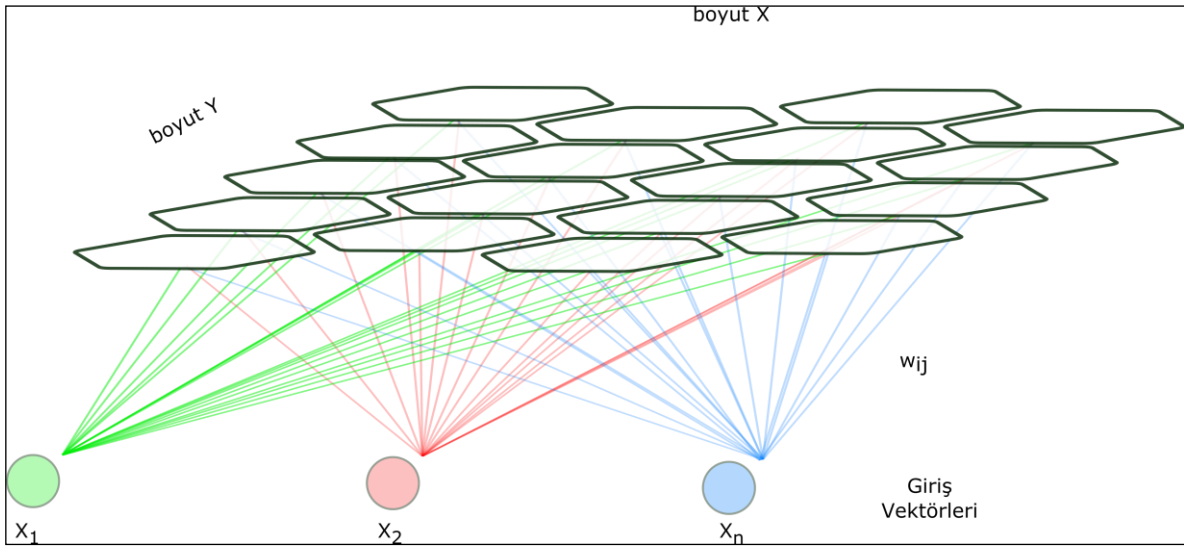
Burada " r_b " ve " r_j " düğümlerin koordinatlarıdır. " $\delta(t)$ " genişliği, etki alanını daraltmak için monoton

olarak azalır. Bundan sonra, ağırlık güncellemesinin toplu formülasyonu şu şekilde hesaplanır:

$$w_j(t_f) = \frac{\sum_{x'=t_0}^{t_f} h_{b_j}(t') x(t')}{\sum_{x'=t_0}^{t_f} h_{b_j}(t')} \quad (9)$$

Burada belirtilen özdüzenleyici haritalar yönteminin görseli **Şekil 2**'de grafik ile açıklanmıştır. Görseldeki metot, x_1, \dots, x_n , boyunca yayılan giriş veri uzayından iki boyutlu bir düğüm dizisine bir eşleme

tanımlar. Sinir ağlarında bulunan diğer eğitim algoritmalarının metodolojilerinden farklı olarak, özdüzenleyici haritalar yöntemi bir çıkış vektörü gerektirmez ve x_1, \dots, x_n 'yi herhangi bir dış denetim olmadan sınıflandırabilir. **Şekil 2** görselinden de anlaşılacağı üzere özdüzenleyici haritaların yapısı nispeten daha kolay anlaşılabilir grafiklere olanak sağlamaktadır. Fakat, yapılarının detaylı analizleri diğer yapay sinir ağlarına paralel şekilde oldukça zordur (Özçalıcı 2017).



Şekil 2. Özdüzenleyici Haritalar Yöntemi

3. Veri Seti ve Çalışmanın Vaka Analizi

Çalışmanın vaka analizi için kullanılan senaryo, operasyonel döngü geçmişleri boyunca izlenmiş set verileri yardımıyla yapılmaktadır. Bu amacı gerçekleştirmek için bir uçuştan diğerine farklı uçuş koşullarında çalışan bir uçak motor ünitesinin modellemesinden alınan bilgi kullanılmaktadır (Saxena et al. 2008). Bu nedenle veri seti, çeşitli faktörlere bağlı olarak, miktar ve oran bakımından her uçuş aşaması için farklı olacaktır. Belirli bir uçuş sırasında değişen rejimler, yalnızca uçuş süresi ve uçuş koşullarına bağlı olarak doğrudan ölçülemeyeceği ve dolayısıyla her uçuş sırasında toplanan sensor verilerinden çıkarılan bilgilere dayanması gerekmektedir. Modellenen bu senaryoda, motor performansı üzerinde önemli bir

etkisi olan üç çalışma ayarı vardır. Bu ayarlar, 0-42 bin ft. arasında değişen rakım, 0-0.84 arası Mach sayısı ile 60 veya 100 değerinde throttle resolver açısı (TRA) olarak dahil edilmiştir (bknz **Çizelge 1.** Operasyon Set Verileri). Veriler ayrıca sensor gürültüsü ile kirletilmiştir. Çalışmada, bu veri setinden bir uçuş yörüngesi (trajectory) alınarak sunulan metot uygulanmıştır. Burada kullanılan yörünge, aynı uçak motorundan uzun bir zamanda toplanan verileri içeren ve normal çalışma koşullarından bir performans bozukluğuna kadar olan operasyonel süreleri, run-to-failure, ihtiva eden bir operasyonlar bütünüdür. Dolayısıyla, bu yörünge uçak motorunun kullanışlı operasyon süresince birçok uçuşundan alınmış enstantane (snapshot) verilerden oluşmaktadır. Bu özelliğinden ötürü, veri seti görece yoğundur. Enstantane veriler rastgele

seçildiği için rejimler arası geçişlerde bir sıra bulunmamakta ve bu da; yapılacak olan analizi daha da zorlayarak kullanılan model performansının daha iyi bir biçimde ölçülmesine olanak sağlamaktadır.

Çizelge 1. Operasyon Set Verileri

Operasyonel döngüde süre	Rakım (0-42 bin ft.)	Mach sayısı (0-0.84)	Throttle resolver açısı (TRA) (20-100)
0	42.0049	0.84	100
1	20.002	0.7002	100
2	42.0038	0.8409	100
3	42	0.84	100
4	25.0063	0.6207	60
5	34.9996	0.84	100
6	0.0019	0.0001	100
7	41.9981	0.84	100
8	42.0016	0.84	100
9	25.0019	0.6217	60
10	20.0016	0.7	100
11	34.9993	0.84	100
12	24.9986	0.62	60
13	20.0056	0.7008	100
14	25	0.62	60

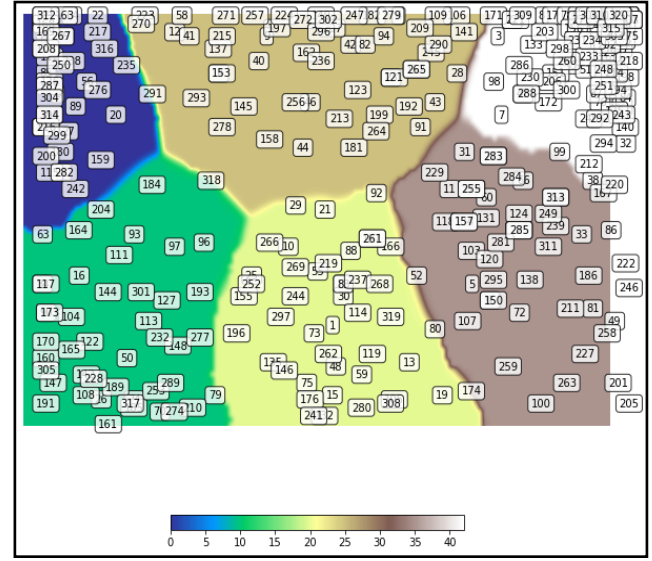
306	42.0038	0.84	100
307	42.0074	0.8412	100
308	20.0038	0.7015	100
309	42.0014	0.84	100
310	42.0044	0.8401	100
311	35.0045	0.8401	100
312	0.0001	0	100
313	35.0069	0.84	100
314	0.0019	0	100
315	42.005	0.8402	100
316	0.0014	0.0014	100
317	10.0046	0.25	100
318	10.0009	0.252	100
319	20.006	0.7002	100
320	42.0058	0.84	100

Her ne kadar Özdüzenleyici haritalar modellenmeden önce vaka analizinde altı farklı uçuş aşamasının verildiği bilinse de, modellenen sistem büyük bir harita boyutu ile (100x100 nöron bileşeni) nitelendirilmiştir. Çalışmada analizi gerçekleştirmek için Python 3 kullanılmış ve özdüzenleyici haritalar, Wittek et al. (2017) tarafından geliştirilen metot ile modellenmiştir.

4. Bulgular ve Analiz

Bu bölümde eğitilen algoritmanın kümeleme ve uçuş aşamalarını tanımlanmasıyla ortaya çıkan sonuçlar, çizelgeler ve grafikler ile açıklanacaktır. Eğitim süresi boyunca özdüzenleyici haritalarda verilen nöronlar birbirleriyle -bir bakıma- çekişme içinde bulunurlar. Bu çekişme sonrasında galip gelen

nöron ve çevresinde bulunan ağırlık vektörleri, verilen gözlemlere göre yenilenmektedir (Kohonen 2001).



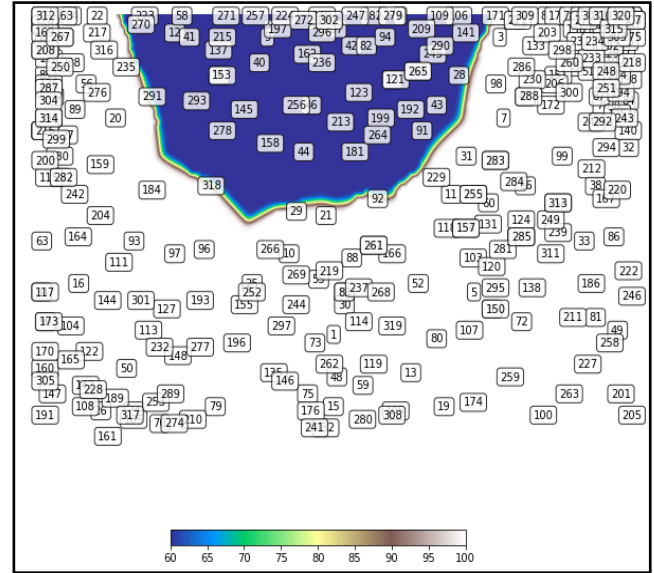
Şekil 3. Rakım (0-42 bin ft.) set verisine göre rejimlerin kümelenmesi

Bu eğitim sonucunda elde edilen verilerin benzer örüntüleri aynı rejim katmanı ve renginde konumlandırılmıştır (bknz Şekil 3-4-5'de bulunan çıktı katmanlarında farklı renkler ile gösterilen rejim kümeleri). Bu şekillerde bulunan rejim kümeleri, komşu özdüzenleyici harita ağırlıkları (matris U - Kohonen katmanı) ile her bir nöronda hangi aracın olduğunu gösteren başarı oranı tablosu ile temsil edilmektedir (Özçalıcı 2016). Bu rejim çıktı haritaları nöronların sahip olduğu izlenimleri ve birbirine olan uzaklıkları göstermektedir.

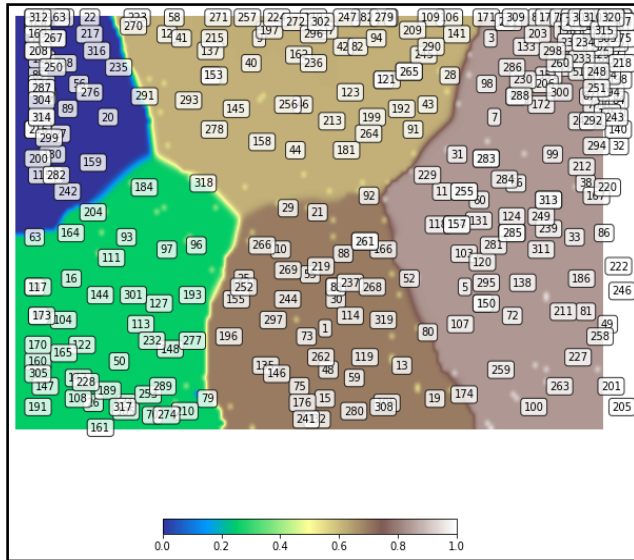
Bu şekillerde bulunan katmanlardaki renk değişimleri ve renkler arası geçiş, komşu rejimlerin birbirlerine olan farklılıkları gösterir. Şekillerde görüleceği üzere bulunan rejim kümeleri oldukça keskindir ve renk geçişleri sadece katman sınırları arasında çok kısıtlı olarak çıkmıştır. Bu durumdan ötürü, farklı renk kümeleri arasındaki sınırlar açık bir şekilde görülebilmektedir. Ayrıca, küme katmanları tek renk titreminde sınıflanmıştır ve bir ton farklılığı gözlemlenmemektedir. Bu ise, girdi olarak verilen motor performans verilerinin benzer merkezler

üzerinde birbirlerine benzer olduğunu belirtmektedir.

Şekil 3'de rakım set veri girdisi ile sınıflandırma yapılmıştır. Diğer iki operasyon set verisinin aksine altı farklı rejimin hepsi de kümelenmiştir. Koyu mavi düşük rakım verisini içermektedir ve bu nedenle yer anındaki (deniz seviyesi) aşamalarını temsil etmektedir. Deniz seviyesi çalışma alanı dışındaki bölgelerde bulunan veriler, rakım seviyesinden öngörüleceği üzere uçuşun farklı koşullarına işaret edecektir. Bu durumda, eğitilen algoritma doğal olarak farklı koşulları kümelemiştir. Örneğin, maksimum rakım seviyesi ~42 bin ft beyaz renk tabakası ile belirtilirken, değişkenlik belirten bölümler diğer renklerle gösterilmiştir.



Şekil 5. Throttle resolver açısı (TRA) (20-100) set verisine göre rejimlerin kümelenmesi



Şekil 4. Mach sayısı (0-0.84) set verisine göre rejimlerin kümelenmesi

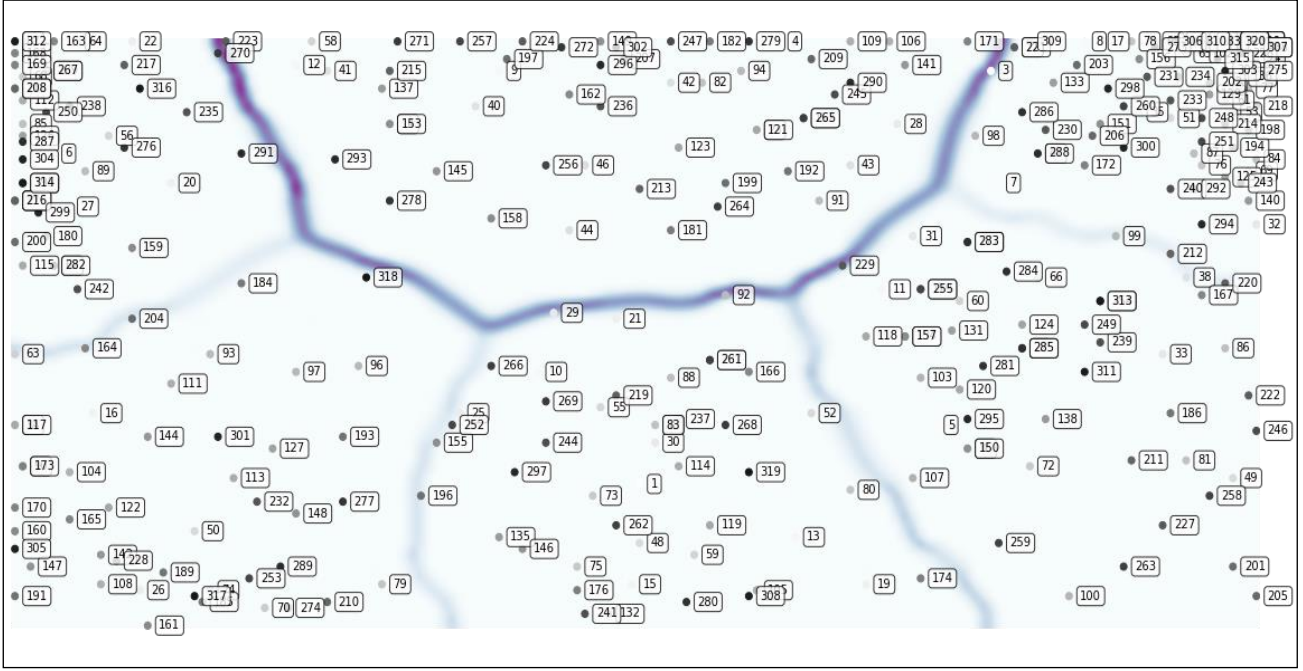
Rakım analizi sonucundaki görseller farklı rejim renkleriyle sunulduktan sonra, sıra Mach sayısı ile küme tabakalarının belirlenmesine gelmektedir. Rakımın ~35 ve ~42 bin ft olduğu bölümlerde Mach sayısı ~0.84 civarındadır ve aynı tabaka rengi ile gösterimi yapılmış ve bir önceki grafiğe göre beş farklı grupta yapılmıştır. Ayrıca, deniz seviyesi gruplamasında beklenildiği üzere hız seviyesi oldukça düşüktür ve ara rejimlerde rakıma bağlantılı olarak değişen Mach sayıları mevcuttur.

Analiz sonucunda ortaya çıkan önceki görseller kümelerin yerlerinin belirlenmesinde etkili olarak kullanılmasına rağmen; **Şekil 5'**de throttle resolver açısı verisi kısıtlı olarak sadece ikili bir grupta yapılmıştır.

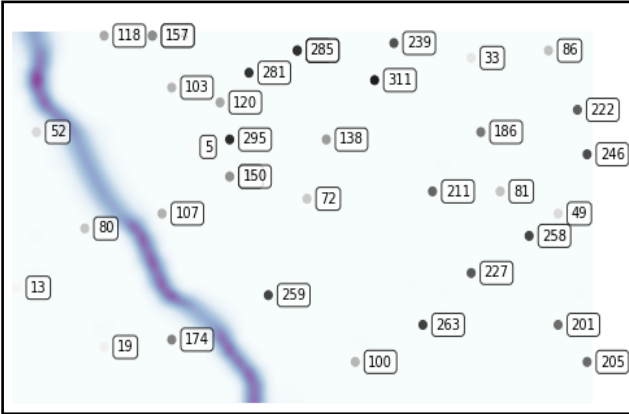
Çizelge 2. Altı uçuş rejimi: küme merkezleri ve yarıçapları

Rakım	Küme Merkezleri		Küme Yarıçapı
	Mach	TRA	
42	0.84	100	0.006
35	0.84	100	0.006
0	0	100	0.003
20	0.7	100	0.006
25	0.62	60	0.006
10	0.25	100	0.006

Küme tabakaları tanımlandıktan sonra, rejim sınıflandırmasına ilişkin tanımlayıcı istatistiklerin **Çizelge 2'**de incelenmektedir. Burada küme merkezleri ve verilerin bu merkezlere olan uzaklık seviyeleri (küme yarıçapı) verilmektedir. Her rejim sahip olduğu belirgin özelliklere göre önceki görsellerdeki tabakalara ayrıldıktan sonra, **Şekil 6'**da küme sınırları belirtilmiştir.



Şekil 6. Küme Sınırları



Şekil 7. Küme Sınırları yoğun gösterim

Komşu tabakalara olan uzaklık arttıkça sınırlarının koyuluğu artmıştır. Ayrıca, daha yoğun bir analiz için belirli bir alan (bknz **Şekil 7** sağ alt köşe) yakınlştırabilir ve verilerin birbirlerine olan uzaklıkları daha yakın olarak incelenebilir. Bu görselde U-Matrix ile her veri noktası için en iyi eşleşen birimler çizilmiştir.

Genel olarak, sunulan yöntem veri setinde belirtildiği üzere uçuş verilerinde altı farklı rejimi işaretleyerek istenilen sağlam sonuçları elde edebilmiştir. Veri seti, çalışmada kümele amacıyla

kullanılan operasyon set verilerinden (rakım, Mach sayısı ve throttle resolver açısı) başka sensor verilerini içermektedir. Çalışmamızda bulunan kümelemeye dayanarak bu sensor verilerinin içinde bulunduğu her bir rejim kendi içlerinde standart skorları alınıp normalize edildiğinde ve rejimler tekrar aynı zaman serisinde birleştirildiğinde anlamlı bir bütün teşkil etmektedir. Özellikle bu normalize işleminden sonra sensor verilerinin eksponansiyel bir artış veya azalış göstermesi ve veri setinde bahsedildiği üzere normal çalışma koşullarından bir performans bozukluğuna kadar olan süreç ile (run-to-failure) paralellik göstermesi çalışmanın sonucunun doğruluğunu ispat etmekte ve modelin veri analizinde kullanışlı olduğunu göstermektedir.

5. Tartışma ve Sonuç

Bu çalışmada özdüzenleyici haritalar yöntemi, farklı uçuş rejimlerini temsil eden uçuş verileri ile eğitilmiştir. Verilen parametrelerle uçuş altı farklı rejime bölünmüş ve bir renk spektrumu ile tabakalar gösterilmiştir. Çalışmada, ticari havacılık sektörünün gün geçtikçe artan çok rejimli büyük veri analizi

ihtiyacına, durum izleme verilerini başarılı bir şekilde kümeleyerek karşılık verilmiştir. Bundan ötürü, uygulama önemli bir potansiyele sahiptir. Fakat uçuş rejimleri genellikle bu makalede kullanılan vaka analizinden daha karmaşık bir yapıdadır ve bu durum gelecek projeler için önemli bir çalışma alanına işaret etmektedir.

6. Kaynaklar

- Aballe, A., Bethencourt, M., Botana, F., and Marcos, M. 1999. Using wavelets transform in the analysis of electrochemical noise data. *Electrochimica Acta*, **44(26)**, 4805-4816.
- Bektaş, O. 2020a. Kestirimci Bakım İçin Döner Mekanizma Bozulma Eğrisinin Tanımlanması. *Avrupa Bilim ve Teknoloji Dergisi*, **(19)**, 420-428.
DOI: 10.31590/ejosat.708257
- Bektaş, O. 2020b. Uçak İtki Takatinin Segmentasyonu. 8. *Ulusal Havacılık Ve Uzay Konferansı (UHUK'2020)*, Ankara, Turkey.
- Bracewell, R. N. 1986. *The Fourier transform and its applications*. New York: McGraw-Hill.
- Chui, C. K., and Chen, G. 2017. Kalman filtering. Springer International Publishing, 19-26.
- Garguet-Duport, B., Girel, J., Chassery, J. M., and Patou, G. 1996. The use of multiresolution analysis and wavelets transform for merging SPOT panchromatic and multispectral image data. *Photogrammetric Engineering and remote sensing*, **62(9)**, 1057-1066.
- Jack Lam, S. S. 2014. Enhanced trajectory based similarity prediction with uncertainty quantification. *Annual Conference of the Prognostics and Health Management Society*, 623-634.
- Koçyiğit, Y., and Korürek, M. 2010. EMG işaretlerini dalgacık dönüşümü ve bulanık mantık sınıflayıcı kullanarak sınıflama. *İTÜDERGİSİ*, **4(3)**.
- Kohonen, T. 1990. The self-organizing map. *Proceedings of the IEEE*, **78(9)**: 1464-1480.
- Kohonen, T. 2001. Self-Organizing Maps. In *Springer Series in Information Sciences*, **30**, 501.
- Lindon, J. C., and Ferrige, A. G. 1980. Digitisation and data processing in Fourier transform NMR. *Progress in Nuclear Magnetic Resonance Spectroscopy*, **14(1)**, 27-66.
- McLachlan, G., and Peel, D. 2004. *Finite mixture models*. John Wiley & Sons.
- Ozaktas, H., and Kutay, M. 2001. The fractional Fourier transform. *European Control Conference (ECC) IEEE*, 1477-1483.
- Özçalıcı, M. 2017. Özdüzenleyici Haritalar Yardımıyla Piyasa Bölümlendirmesi: Türkiye İkinci El Otomobil Piyasası Örneği. *Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi*, **12(2)**, 23-36.
- Özçalıcı, M. 2016. Hisse Senetlerinin Özdüzenleyici Haritalarla Kümelendirilmesi: BIST50 Endeksinde Yer Alan Hisseler Üzerine Bir Uygulama. *Istanbul University Journal of the School of Business*, **45(1)**, 22-33.
- Peter Wittek, S. C. 2017. Somoclu: An Efficient Parallel Library for Self-Organizing Maps. *Journal of Statistical Software*, **78(9)**, 1-21.
- Ramasso, E. 2014. Investigating computational geometry for failure prognostics. *International Journal of Prognostics and Health Management*, **5(1)**, 1-18.
- Saxena, A., Goebel, K., Simon, D., & Eklund, N. 2008. Damage propagation modeling for aircraft engine run-to-failure simulation. *International Conference on Prognostics and Health Management*. Denver: IEEE.
- Sugeno, M. 1993. Fuzzy hierarchical control of an unmanned helicopter. *Fifth IFSA World Congress*, Seoul, 179-182.
- Wittek, P., 2013. Somoclu: An efficient distributed library for self-organizing maps. *arXiv*, **1305.1422**, 1-7.

Witteck, P., Gao, S. C., Lim, I. S., & Zhao, L. 2017. Somoclu:
An Efficient Parallel Library for Self-Organizing Maps.
Journal of Statistical Software, **78(9)**, 1-21