



T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ
BİYOİSTATİSTİK ANABİLİM
DALI



**KAYIP LANDMARK TAHMİNİNDE KULLANILABİLECEK
YÖNTEMLERİN PERFORMANSININ KARŞILAŞTIRILMASI VE
F YAKLAŞIMININ ÖNERİLMESİ**

FATMA EZGİ CAN

(DOKTORA TEZİ)

BURSA-2020

Fatma Ezgi CAN

BİYOİSTATİSTİK ANABİLİM DALI DOKTORA TEZİ

2020





T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ
BİYOİSTATİSTİK ANABİLİM DALI



**KAYIP LANDMARK TAHMİNİNDE KULLANILABİLECEK
YÖNTEMLERİN PERFORMANSININ KARŞILAŞTIRILMASI VE
F YAKLAŞIMININ ÖNERİLMESİ**

Fatma Ezgi CAN

(DOKTORA TEZİ)

**DANIŞMAN:
Prof.Dr. İlker ERCAN**

BURSA-2020

**T.C.
ULUDAĞ ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ**

ETİK BEYANI

Doktora tezi olarak sunduğum

“Kayıp Landmark Tahmininde Kullanılabilecek Yöntemlerin Performansının Karşılaştırılması ve F Yaklaşımının Önerilmesi” adlı çalışmanın, proje safhasından sonuçlanmasına kadar geçen bütün süreçlerde bilimsel etik kurallarına uygun bir şekilde hazırlandığını ve yararlandığım eserlerin kaynaklar bölümünde gösterilenlerden oluştuğunu belirtir ve beyan ederim.

Fatma Ezgi CAN
28/01/2020



SAĞLIK BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ'NE

Biyostatistik Anabilim Dalı ~~Yüksek Lisans/Doktora~~ öğrencisi Fatma Ezgi CAN tarafından hazırlanan "Kayıp Landmark Tahmininde Kullanılabilecek Yöntemlerin Performansının Karşılaştırılması ve F Yaklaşımının Önerilmesi" konulu Yüksek Lisans/Doktora tezi 28/01/2020 günü, 11:00-12:00 saatleri arasında yapılan tez savunma sınavında jüri tarafından oy birliği/oy çokluğu ile kabul edilmiştir.

Adı-Soyadı

Tez Danışmanı

Prof.Dr.İlker ERCAN

Üye

Prof.Dr.Berna YAZICI

Üye

Doç.Dr. Güven ÖZKAYA

Üye

Doç.Dr. Mehmet KOÇAK

Üye

Dr.Öğr.Üyesi Mehmet Onur KAYA

İmza



Bu tez Enstitü Yönetim Kurulu'nun tarih ve sayılı toplantısında alınan numaralı kararı ile kabul edilmiştir.

Prof.Dr. Gülşah ÇEÇENER
Enstitü Müdürü

TEZ KONTROL ve BEYAN FORMU

28/01/2020

Adı Soyadı: Fatma Ezgi CAN

Anabilim Dalı: Biyoistatistik

Tez Konusu: Kayıp Landmark Tahmininde Kullanılabilecek Yöntemlerin Performansının Karşılaştırılması ve F Yaklaşımının Önerilmesi

<u>ÖZELLİKLER</u>	<u>UYGUNDUR</u>	<u>UYGUN DEĞİLDİR</u>	<u>ACIKLAMA</u>
Tezin Boyutları	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Dış Kapak Sayfası	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
İç Kapak Sayfası	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Kabul Onay Sayfası	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Sayfa Düzeni	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
İçindekiler Sayfası	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Yazı Karakteri	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Satır Aralıkları	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Başlıklar	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Sayfa Numaraları	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Eklerin Yerleştirilmesi	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Tabloların Yerleştirilmesi	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Kaynaklar	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

DANIŞMAN ONAYI

Unvanı Adı Soyadı: Prof.Dr.İlker ERCAN

İmza:

İÇİNDEKİLER

Dış Kapak

İç Kapak

ETİK BEYAN.....	II
KABUL ONAY.....	III
TEZ KONTROL BEYAN FORMU	IV
İÇİNDEKİLER	V
TÜRKÇE ÖZET	VII
İNGİLİZCE ÖZET	VIII
1. GİRİŞ	1
2. GENEL BİLGİLER.....	4
2.1. İstatistiksel Şekil Analizi.....	4
2.2. Morfometri.....	5
2.2.1. Geleneksel Morfometri	7
2.2.2. Geometrik Morfometri	8
2.3. Landmark	9
2.3.1. Landmark Güvenilirliği	12
2.3.2. Landmark Seçim Kriterleri	14
2.4. Geometrik Morfometride Landmark Tabanlı Yaklaşımlar	17
2.4.1. Procrustes Analizi	17
2.4.2. Öklid Uzaklık Matrisi Analizi (EDMA).....	19
2.5. Kayıp Veri Tahmin Yöntemleri	20
2.5.1. EM (Expectation Maximization) Algoritması	23
2.5.2. Çoklu Regresyon Atama Yöntemi	27
2.5.3. Temel Bileşenler Analizi (PCA).....	30
2.5.3.1. Olasılıksal Temel Bileşenler Analizi (Probabilistic PCA, PPCA	34
2.5.3.2. Bayes Yaklaşımlı Temel Bileşenler Analizi (Bayesian PCA, BPCA)	36
2.5.3.3. Ters Doğrusal Olmayan Temel Bileşenler Analizi (Inverse Non-Linear PCA, NLPCA)	38
2.5.3.4. İteratif Kısmi En Küçük Kareler İle Doğrusal Olmayan Tahmine Dayalı Temel Bileşenler Analizi (Non-Linear Estimation by Iterative Partial Least Squares PCA, INIPALS)	39

2.5.3.5. Temel Bileşenler Analizi ile Kayıp Veri Tahmini	41
2.6. F Yaklaşımı	42
3. GEREÇ VE YÖNTEM	46
3.1. Simülasyon Senaryosu	47
4. BULGULAR	61
4.1. Gerçek Veriye Bağlı Simülasyon	61
4.2. Simülasyon Çalışmasından Elde Edilen Sonuçlar	63
4.3. Gerçek Veri Uygulaması.....	78
5. TARTIŞMA VE SONUÇ.....	81
6. KAYNAKLAR	85
7. SİMGELER VE KISALTMALAR	91
8. EKLER.....	92
9. TEŞEKKÜR	135
10. ÖZGEÇMİŞ.....	136

TÜRKÇE ÖZET

İstatistiksel şekil analizi; nesnelere elde edilen geometrik bilginin kullanıldığı yöntemleri içermektedir. İstatistiksel şekil analizinde geometrik bilginin kullanılmasında en önemli girdi landmarklardır. Şekil analizinde kayıp veriler landmark koordinatlarına ait bilgi kaybı olduğu durumda ortaya çıkmaktadır. Landmarkların kartezyen koordinatlarında veri kaybı meydana gelmesi o landmarkı kullanılamaz hale getirerek ilgili birimin araştırmadan çıkmasına yol açmaktadır. Tez çalışmasında kayıp landmarkların tahminine yönelik kullanılan EM algoritması, çoklu regresyon atama, Bayes yaklaşımı temel bileşenler analizi, olasılıksal temel bileşenler analizi, ters doğrusal olmayan temel bileşenler analizi ve iteratif kısmi en küçük kareler ile doğrusal olmayan tahmine dayalı temel bileşenler analizi yöntemleri ve amaca yönelik olarak tez çalışması kapsamında önerilen min(F) ve max(F) yaklaşımının performansı değerlendirilmiştir. Simülasyon senaryosunda landmark sayıları 3, 6, 9 ve 12 ve örneklem büyüklükleri 30, 50 ve 100 olarak alınmıştır. Çok değişkenli normal dağılımdan isotropik ve isotropik olmayan modellere dayalı olarak veri türetilmiş ve 10 farklı simülasyon senaryosu dikkate alınmıştır. Küçük, orta ve büyük örneklem büyüklükleri dikkate alındığında, performans değerlendirmesinde en iyi ve en farklı sonucu, tez çalışmasında önerilen F-yaklaşımı algoritması Min(F) ölçütü vermiştir.

Anahtar kelimeler: Landmark, Şekil Analizi, Kayıp Veri

İNGİLİZCE ÖZET

F APPROACH PROPOSAL AND COMPARISONS OF METHODS USED IN MISSING LANDMARK ESTIMATION

Statistical shape analysis involves methods that use geometric information obtained from objects. The most important input to the use of geometric information in statistical shape analysis is landmarks. Missing data in shape analysis occurs when there is a loss of information about landmark coordinates. The loss of data in the cartesian coordinates of landmarks makes that landmark unusable and causes the related unit to be dropped out of the survey. Performances are evaluated for the following methods used in the estimation of missing data; EM algorithm, multiple regression imputation, Bayesian principal component analysis, probabilistic principal component analysis, Inverse non-linear principal component analysis, non-linear estimation by iterative partial least squares principal component analysis and proposed Min(F) and Max(F) approaches in the thesis. Landmark counts were taken as 3, 6, 9, 12 and sample sizes were taken as 30, 50, 100 in the simulation study. The data are generated based on multivariate normal distribution from isotropic and non-isotropic models and 10 different simulation scenarios are considered. The best and the most different result in the performance evaluation according to small, medium and large sample sizes is the Min(F) criterion of the F-approximation algorithm proposed in the thesis study.

Key words: Landmark, Shape analysis, Missing data

1. GİRİŞ

Anatomik şekiller ve şekil çeşitliliği sağlık alanında yapılan araştırmalarda önemli bir konu olarak yer almaktadır (Anwary, 2012). Sağlık alanında yapılan çalışmalarda, organ veya organizmaların geometrik özellikleri araştırılmaktadır. Bu çalışmalarda veri setleri kantitatif veya kalitatif ölçüm değerlerinden oluşurken, görüntüleme tekniklerindeki gelişmeyle bir organın veya organizmanın görüntüsü veya şekli de veri setlerini oluşturabilmektedir (Ercan ve ark., 2012, 2015). Şekil ile ilgili sağlık alanında yapılan çalışmalara, şekillerin karşılaştırılması, şekli kullanarak nasıl sınıflandırma yapılabileceği ve şekil değişkenliğinin nasıl tanımlanabileceği örnek olarak verilebilir (Dryden ve Mardia, 1998; Ercan ve ark., 2012).

İstatistiksel şekil analizi, anatomik varyasyonların araştırılmasında ve olasılıksal sonlu eleman modellerinin oluşturulmasında yaygın olarak kullanılmıştır (Yates and Untaroiu, 2018). Şekil analizi, hem tıbbi teşhisin doğruluğunu hem de büyüme ve hastalıkların arkasındaki süreçleri anlama becerisini geliştirme potansiyeline sahip olduğu için tıbbi görüntü işlemenin önemli bir alanı olarak ortaya çıkmaktadır (Joshi ve ark., 2002)

Şekil analizi, nesnelere elde edilen geometrik bilginin kullanıldığı yöntemleri içermektedir. İstatistiksel şekil analizinde geometrik bilginin kullanılmasında en önemli girdi ise landmarklardır. Her bir landmark iki boyutlu düzlemde sıralı ikili ya da üç boyutlu uzayda sıralı üçlü biçimindeki kartezyen koordinatlara sahiptir. Şekil analizinde kullanılan en önemli girdi olan landmarkların belirlenmesiyle nesnelere ait iki boyutlu ve üç boyutlu uzaydaki kartezyen koordinatlar elde edilebilmektedir.

Birçok arařtırmada kayıp veri sorunu yařandığı gibi, Őekil analizinde de kayıp landmark sorunu yařanabilmektedir. Veri setlerinde meydana gelen kayıplar, kayıp veri yoęunluęuna da baęlı olarak istatistiksel alıřmalarda önemli problemler yaratabilmektedir. Arařtırmalarda kayıp veri olması parametre tahminlerinde sapmalara, bilgi kaybına istatistiksel ıkarırsamalarda gücün düşmesine ve standart hatanın artmasına neden olabilmektedir (Dong and Peng, 2013). alıřmalarda veri setinde kayıp deęerler olması ilgilenilecek ilk konu olmamasına raęmen ciddi problemlere yol aabilmektedir.

Veri setindeki kayıp gözlemlerin giderilememesi, veri setinin istatistiksel yöntemler için uygunsuz olmasına ve istatistiksel analiz varsayımlarının sağlanamamasına neden olabilmektedir (Dong and Peng, 2013). Özellikle çok deęişkenli analizlerin uygulanacağı durumda kayıp veri sorununun çok daha önem taşıdığı görülmektedir.

İstatistiksel Őekil analizinde de incelenen nesnelere ait landmarklarda kayıpların olması durumunda ilgili birim tamamen alıřmadan ıkmaktadır. Dolayısıyla Őekil analizinde kayıp landmark olması, incelenen Őeklin bütünlüęünü bozması nedeniyle ve arařtırmada birim kaybı olmasından dolayı daha da önem taşımaktadır.

Őekil analizinde kayıp veriler landmark koordinatlarına ait bilgi kaybı olduęu durumda ortaya ıkmaktadır. Saęlık ve antropoloji alanındaki alıřmalarda landmark koordinatlarında meydana gelen veri kayıpları, incelenen kemik yapısındaki kırılmalardan veya görüntü kalitesindeki bozulmadan kaynaklanabilmektedir. Landmarkların kartezyen koordinatlarında veri kaybı meydana gelmesi o landmarkı kullanılamaz hale getirmekte ve ilgilenilen birimin arařtırmadan ıkmasına yol amaktadır.

Bazı iki boyutlu Őekillerde kayıp landmarklarla karşılaşılabilmektedir. Özellikle adli tıp, paleontoloji ve arkeoloji gibi bilim dallarında kayıp landmark sorunu, alıřmada yer alan örneklem sayısı az olması durumunda, daha da önem kazanmaktadır. Literatür incelendiğinde Őekil analizinde kayıp landmarkı tahmin etmek için EM (Expectation Maximization) algoritması, oklu regresyon atama

yöntemi ve temel bileşenler analizi (PCA) yöntemlerinin yaygın olarak kullanıldığı görülmektedir (Couette and White, 2010).

Kayıp veri tahmin yöntemlerinden en çok kullanılan yöntemlerden biri EM algoritması olmakla birlikte, son yıllarda en çok olabilirlik ve çoklu atama gibi dağılımsal modellere dayalı yöntemlerin de kullanıldığı görülmektedir (Pigott, 2001). Tez çalışmasında kayıp landmarkların tahminine yönelik kullanılan yöntemler ve amaca yönelik olarak tez çalışması kapsamında önerilen Min(F) ve Max(F) yaklaşımının performansı değerlendirilmiştir.



2. GENEL BİLGİLER

2.1. İstatistiksel Şekil Analizi

Biyolojik yapıların en temel özelliklerinden birisi olan şekil, yapıların genel görünümlerin benzersiz olmasını sağlamaktadır (Cho ve ark., 2019). Başka bir tanımla şekil, görünümleri analizde büyük rol oynayan nesnelerin fiziksel özelliğidir (Xu and Hong, 2017). Şekil terimi genellikle bir nesnenin görünümünü ifade etmek için de kullanılmaktadır (Anwary, 2012). Kendall tarafından şekil, bir nesneye ait konum, ölçek ve döndürme etkileri arındırıldığında geriye kalan geometrik bilgi olarak tanımlanmıştır (Dryden and Mardia, 1998; Kendall, 1977).

1960 ve 1970' lerde araştırmacılar şekil değişkenliğini tanımlamak için şeklin sayısal ifadesini istatistiksel şekil analizi ile birleştirilmiştir. İstatistiksel şekil analizi teorisi Kendall ve Bookstein' in bağımsız çalışmaları ile başlamıştır (Bookstein, 1984, 1986, 1992; Kendall, 1977). Dryden ve Mardia (1998) istatistiksel şekil analizi teorisinin geliştirilmesinde önemli rol oynamıştır. Şekil analizi yöntemleri ile şeklin uygun matematiksel gösterimleri ve şekil farklılıklarını belirlemede kullanılacak istatistiksel yöntemler geliştirilmiştir. İstatistiksel şekil analizi, şekli rastgele bir nesne gibi ele alarak şeklin betimleyici istatistiklerini hesaplamak için yöntemler sunmaktadır (Cho ve ark., 2019).

İstatistiksel şekil analizi, benzer şekillerin ya da benzer şekillerden oluşan farklı grupların özelliklerini tanımlamak amacıyla ölçülen istatistiklere ait şekiller kümesinin geometrik analizi olarak tanımlanabilir (Ercan ve ark., 2012). Özet bir tanımla, şeklin matematiksel olarak ifade edilmesidir (Brombin ve Salmaso, 2013).

İstatistiksel şekil analizinin, gerek farklı disiplinlerde yapılan ve gerek ise tıbbi görüntü analizlerindeki uygulamalarda şekilleri verimli bir şekilde temsil etmede güçlü bir analiz yöntemi olduğu gösterilmiştir (Joshi ve ark., 2002). Özellikle, istatistiksel şekil analizinin tıpta sınıflandırma ve tanı amaçlı ölçümler oluşturmak için kullanılabilir, şekle dayalı özellikler sağlayabildiği gösterilmiştir. İstatistiksel şekil analizinde, belirli uygulamaların spesifik özelliklerine, kısıtlamalarına ve hedeflerine bağlı olarak geliştirilen ve kullanılan farklı teknikler vardır. Ayrıca, bugüne kadar yapılan istatistiksel şekil analizi çalışmaları, şekiller arasında ilişki oluşturmak için gerekli ön işleme yöntemleriyle birlikte verilen bir kümedeki şekillerin matematiksel gösterimini geliştirmeye odaklanmıştır (Cho ve ark., 2019; Wu ve ark., 2014). İstatistiksel şekil analizinde amaç, landmark özelliklerine dayalı şekil metriği geliştirmektir (Krim and Yezzi, 2006).

İstatistiksel şekil analizi; nesnelerin öteleme, ölçekleme ve döndürme yöntemlerini kullanarak şekli incelemeye yönelik çalışmaları içermektedir. Herhangi bir nesneye ait şekil, süperimpozisyon uygulandığında değişmeden kalacaktır.

Anatomik şekil ve çeşitliliği, her zaman biyoloji ve sağlık alanındaki araştırmaların önemli bir ilgi konusu olmuştur. Şekil analizinde analiz yaklaşımlarından bazıları, şekiller arasındaki farklılıkları belirlemek, ortalama şekli ve şekil değişkenliğini tahmin etmektir (Anwary, 2012; Dryden and Mardia, 1998).

2.2. Morfometri

Morfometri, nesnelerin şekli ve büyüklüğünü araştırmak için matematik, geometri, biyometri, biyoloji, bilgisayar bilimi ve modern mühendislik dallarıyla istatistiği bir araya getiren ve umut vadeden bir bilim dalıdır (Brombin ve Salmaso, 2013). Literatür incelendiğinde şekil değişkenliği ve şeklin diğer değişkenlerle olan ilişkisini araştıran ve şekil karşılaştırmalarında ele alınan yöntemleri içerdiği görülmektedir (Adams, 1999; Brombin ve Salmaso, 2013; Webster ve Sheets, 2010; Zelditch ve ark., 2004). Morfometrik verilerden bilgi elde edilme yöntemleri, biyolojik sezgilere veya klasik morfolojiye dayanan kavramlardan daha çok, matematiksel

işlemleri içermesi nedeniyle morfometri, matematiksel şekil analizinin de dalı olarak görülmektedir (Zelditch ve ark., 2004). Ayrıca, türler arasındaki farklılıkların konumlandırılması ve karakterizasyonu ile ilgili olduğu için ontojenik ve organizma şekillerinde meydana gelen değişiklikleri inceleyen biyolojik çalışmalarda önemli yere sahiptir (Bookstein, 1986, 1992; Brombin ve Salmaso, 2013).

Bookstein tarafından şekil değişkenliğinin, içsel ve dışsal faktörler arasındaki ilişkisinin araştırılması olarak tanımlanan morfometri, organizma formlarının kantitatif çalışmasıdır (Bookstein, 1984, 1986, 1992). Boyut ve şekil ile ilgili bilgilerin sayısal terimlerle yeterince nitelendirilmesine, kesin ve tam olarak ölçülmesine yardımcı olur. Morfometrik çalışmalar, bireyler ve organizma grupları arasındaki varyasyonların anlaşılması amacıyla yapılan çalışmaların başlangıcını oluşturmaktadır (Reitner ve ark., 2011).

Geçmiş yıllarda türlerin filogenetik ağaçlarını ortaya çıkarmak için kullanılan morfometri, biyolojik şekillerin nicel analizi olarak da tanımlanabilmektedir (Brombin ve Salmaso, 2009; Ercan ve ark., 2012; Henderson, 2006). Şekillerin kullanımıyla, filogenileri nicel morfolojik verilerden istatistiksel olarak çıkartabilmek için tasarlanan çeşitli yöntemler geliştirilmiştir (Krim and Yezzi, 2006)

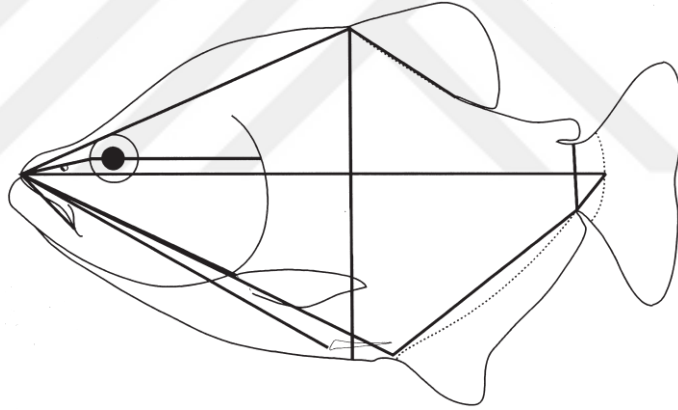
Morfometri, geleneksel ve geometrik morfometri olmak üzere iki farklı dalda incelenmektedir. Geleneksel morfometri, şekiller üzerinde landmarkları belirlemek amacıyla kullanılabilir. Geometrik morfometri ise landmark tabanlı istatistiksel şekil analizi yaklaşımlarını içermektedir (Ercan ve ark., 2012). Geleneksel morfometride landmarklar arasındaki uzaklıklar ölçülürken geometrik morfometride ilgilenilen değişkenler arasındaki geometrik ilişkiler de dikkate alınmaktadır (Boxler ve Schultheiß, 2010; Sheets ve ark., 2006; Zelditch ve ark., 2004).

2.2.1. Geleneksel Morfometri

Geleneksel morfometrik yöntemler, biyolojik ve anatomik anlamları olan landmarkları kullanarak şekli tanımlamak için uzunluk ve açı ölçümleri gibi geleneksel ölçümlere çok değişkenli analizlerin uygulanmasını içermektedir (Ocakoglu ve Ercan, 2013).

Karl Pearson çok değişkenli morfometri yöntemlerini ilk kullanan araştırmacılardandır. Karl Pearson, landmarklar arasında uzunluklara bağlı olarak kraniumlar arasında benzerlik ölçütü olarak tanımlanan “ırksal benzerlik katsayı” ile ilgili çalışmalar yapmıştır (Dryden ve Mardia, 1998; Pearson, 1926).

Geleneksel olarak, morfometrik veriler Şekil 1’ de olduğu gibi uzunluk, derinlik ve genişlik ölçümlerinden oluşmaktadır.



Şekil 1. Bir teleostun dış vücut formunun geleneksel morfometrik ölçümleri (Lagler ve ark., 1963; Zelditch ve ark., 2004).

Şekil 1’ deki ölçümlerle elde edilecek veri seti, şekil hakkında az ve yetersiz bilgi verecektir. Geleneksel morfometride, orijinal formun şeklini ve uzunluk ölçümlerini kullanarak şeklin genel ifadesini elde etmek mümkün olmamaktadır. Çok değişkenli yaklaşımda uygulamalar, landmarklar arasındaki uzunlukların açıları ve oranlarını kullanarak biyolojik organizmaların cinsiyetini belirlemeyi veya türlerin sınıflandırılmasını içermektedir. Geleneksel morfometride en çok kullanılan yöntemler PCA, temel koordinat analizi, faktör analizi, diskriminant analizi ve çok değişkenli varyans analizidir. PCA’ da her bir “bileşen” şeklin ya da büyüklüğün bazı

açıların ölçümü olarak ele alınmaktadır. Genellikle ilk temel bileşen her değişkende eşit yüklere sahip olmakta ve şeklin genel büyüklüğü olarak yorumlanmaktadır (Dryden and Mardia, 1998).

Çok değişkenli yaklaşımda kullanılan diğer bir yaklaşım da allometri çalışmalarıdır. Huxley tarafından ortaya atılan allometri kavramı, şekil ve büyüklük arasındaki ilişki olarak tanımlanabilir (Dryden ve Mardia, 1998; Ercan ve ark., 2012; Huxley, 1924, 1932). İstatistiksel şekil analizinde, büyüklük ölçümü, doğrusal mesafeleri veya ölçümleri dikkate almak yerine, bir nesnenin veya organizmanın geometrik bilgisini kullanarak elde edilir (Sigirli and Ercan, 2013). Mosimann ise pozitif uzunluk ölçümlerinin şekli ve büyüklüğü ile ilgili çalışmalar yapmıştır. Mosimann çeşitli dağılımların karakterizasyonunu da içeren, şekil ve örneklem büyüklüğünün bağımsızlığı hakkında teoremler önermiştir (Mosimann, 1970).

Geleneksel morfometride uzaklıklar, açılar, oranlar vb. leri kullanılarak çok sayıda çalışma yapılmıştır. Günümüzde de biyolojik çalışmalarda hala sıklıkla kullanılmaktadır. Bazı durumlarda açı ve uzunluk oranlarının önemli lineer kombinasyonlarının yorumlanması zor olabilmektedir. Bu gibi durumlarda çok değişkenli uzaydan elde edilen verilerle çalışmak yerine, orijinal uzaydan elde edilen şekillerle çalışarak yorumlamak her zaman daha kolaydır (Dryden and Mardia, 1998).

2.2.2. Geometrik Morfometri

Geometrik morfometri istatistiksel analizler yardımıyla nesnelerin geometrik bilgilerini analiz ederek, organizma noktalarının veya bölümlerinin birbirlerine göre konumlandırılmasına odaklanan morfometri dalıdır (Brombin ve Salmaso, 2013). Geometrik morfometride temel fikir organizmalardan elde edilen değerler yerine geometrik nesnenin tamamıyla çalışmaktır (Dryden and Mardia, 1998). Son yıllarda geometrik morfometrik yöntemler şekil bilgilerini yakalama ve görselleştirme konusundaki etkinlikleri nedeniyle morfolojik çalışmalarda önemli yere sahiptir (Bookstein, 1992; Mitteroecker ve Gunz, 2009). Genellikle geometrik veriler

ilgilenilen organizmalardan elde edilmiş iki ya da üç boyutlu kartezyen koordinatlara sahip noktalardan oluşmaktadır (Watanabe, 2018).

Şekil analizi araçlarının organizma şekillerine çevrilmesi ve bu araçlar yardımıyla şekil farklılıklarının matematiksel olarak analiz edilmesi ve açıklanması, morfometrinin hem morfoloji hem de istatistik ile ilişkili olduğunu göstermektedir. Geometrik morfometri, şekilde meydana gelen farklılıklar ve değişimler ile ilgili kesin ve doğru sonuçlar sunarak karmaşık şekiller arasındaki farkların görselleştirilmesine olanak sağlamaktadır (Zelditch ve ark., 2004).

Şekil analizi, biyolojik organizmalarda meydana gelen varyasyon nedenlerinin ve morfolojik dönüşümlerin anlaşılmasında kullanılan bir yaklaşımdır. Morfolojik dönüşümlerin resimlerini çizebilmek geometrik morfometrinin önemli avantajlarından biridir (Zelditch ve ark., 2004). Geometrik morfometri ile birlikte biyolojik yapılara ait şekil değişkenliği çalışılmaya başlanmıştır (Rohlf, 1999, 2000). Biyolojik şekiller arasındaki fark belirlenirken çok boyutlu şekil uzayının özelliklerinin açıklanması, geometrik yaklaşımların geleneksel yaklaşımlara göre temel avantajlarından biridir (Brombin ve Salmaso, 2013; Rohlf, 1999, 2000). Geometrik morfometrinin geleneksel morfometrik yöntemlere göre en önemli avantajı ise güçlü istatistiksel yöntemleri kullanmasıdır.

2.3. Landmark

Landmark, iki ya da üç boyutlu uzayda, ilgilenilen nesne üzerindeki belli bir özelliğinin pozisyonuna karşılık gelen noktadır (Lele and Richtsmeier, 2001). Farklı bir ifadeyle şekil analizinde kullanılan landmarklar, anakütle içinde veya anakütleler arasında eşleşen her nesneye karşılık gelen noktalar olarak tanımlanmaktadır (Dryden and Mardia, 1998). Her bir landmark, iki boyutlu düzlemde sıralı ikili ya da üç boyutlu uzayda sıralı üçlü biçimindeki kartezyen koordinatlara sahiptir.

Literatürde farklı landmark sınıflamaları bulunmaktadır. Dryden ve Mardia (1998), landmarkları anatomik landmarklar, matematiksel landmarklar, pseudo landmarklar olarak üç grupta sınıflandırmışlardır.

- **Anatomik landmarklar:** Organizmalar arasında biyolojik olarak anlamlı olacak şekilde uyum gösteren ve uzmanlar tarafından oluşturulan noktalardır (Şekil 2) (Dryden and Mardia, 1998). Anatomik landmarklar biyolojik köken bakımından uyuşan organizma bölümlerini tanımlamaktadırlar ve bu bölümler homolog olarak adlandırılmaktadır.
- **Matematiksel landmarklar:** Şeklin matematiksel ya da geometrik özelliklerine göre konumlandırılan landmarklardır (Şekil 3).
- **Pseudo landmarklar:** Bir organizmanın üzerinde, dış hat çizgisinde veya matematiksel ya da anatomik landmarklar arasında konumlandırılan landmarklardır.

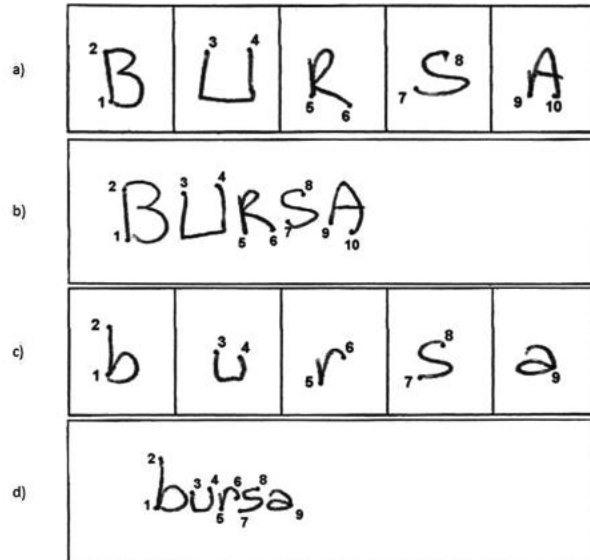
Bookstein (1992) ise landmarkları tip I, tip II ve tip III olmak üzere üç gruba ayırmıştır. Tip I landmarklar, dokuların ya da kemiklerin birleşme noktalarında konumlanmaktadır. Tip II landmarklar, maksimum eğim gibi yerel özellikler ile tanımlanmaktadır. Tip III landmarklar ise maksimum çap ve ağırlık merkezi gibi noktalarda yer alan landmarklardır (Bookstein, 1992).

Lele ve Richestmeier (2001) ise landmarkları geleneksel, fuzzy ve yapılandırılmış landmarklar olmak üzere üç farklı gruba ayırmıştır.

- **Geleneksel landmarklar:** Bazı biyolojik anlamlılıklara göre belirlenen özelliklerin konumlarına karşılık gelen noktalardır. Geleneksel landmarklar, tanımı bir koordinat sistemine bağlı olanlar ve bağlı olmayanlar olarak iki gruba ayrılmaktadır.
- **Fuzzy landmarklar:** Gözlemcinin referans sisteminden daha geniş alanı kapsayan ve tam olarak tanımlanmamış biyolojik yapıya karşılık gelen noktalardır.
- **Yapılandırılmış landmarklar:** Geometrik bilgi ve geleneksel landmarkların kombinasyonu kullanılarak tanımlanan konumlara karşılık gelen noktalardır.



Şekil 2. Kraniumda landmarkların gösterimi (Ozdemir ve ark., 2010). 6-21 numaralı landmarklar anatomik landmarklar, 1-5 numaralı landmarklar ise fuzzy landmarklar



Şekil 3. El yazısıyla yazılmış Bursa kelimesinde landmarkların belirlenmesi (Ozkaya ve ark., 2012)

2.3.1. Landmark Güvenilirliği

Herhangi bir istatistiksel analiz uygulanmadan önce veri elde etme yöntemlerinin güvenilirliği araştırılmalıdır. Tekrarlı ölçümlerden elde edilen sonuçların tutarlılığı güvenilirlik ile belirlenmektedir (Carmines ve Zeller, 1979; Ercan ve ark., 2008; Lele ve Richtsmeier, 2001). İstatistiksel şekil analizi çalışmalarında nesnelere üzerindeki landmarkların belirlenmesi, çalışmanın güvenilirliği açısından çok önemlidir. Landmarklar konumlandırılırken ölçümde meydana gelecek hata kaynakları göz önünde bulunmalıdır. Kesin olmayan ölçümler, aynı örnekten tekrarlanan ölçümler arasında değişkenliğe neden olmaktadır. Herhangi bir veri toplama yönteminde, aynı örneğin tekrarlanan ölçümleri arasındaki değişkenlik artışı, güvenilirliğinde azalmasına neden olacaktır (Ercan ve ark., 2008; Kohn ve Cheverud, 1992; Lele ve Richtsmeier, 2001).

Birbirini izleyen ölçümler arasındaki değişkenlik;

- a) Landmarkları konumlandırırken gözlemci hatası
- b) Landmark koordinatlarını belirlerken ölçme aracı hatası

olmak üzere iki farklı hatadan kaynaklanmaktadır (Ercan ve ark., 2012; Etoz ve Ercan, 2012; Lele ve Richtsmeier, 2001).

Değerlendirici deneyimi çalışmalarda güvenilir landmark elde etmek için önemli bir faktördür. Gözlemciler arası değişkenliğin etkisini ortadan kaldırmak için landmark koordinatlarını tek bir gözlemcinin belirlemesi önerilmektedir (Ercan ve ark., 2008; Lele ve Richtsmeier, 2001).

Bazı durumlarda, çalışmada birden fazla değerlendirici olması kaçınılmaz olabilir. Bu gibi durumlarda küçük bir örneklem belirlenerek landmark koordinatlarının güvenilirliği ve gözlemciler arası varyasyon değerlendirilmelidir. Eğer landmark koordinatlarının güvenilirliği yeterliyse veri toplama aşamasına başlanmalıdır. Eğer güvenilirlik yeterli değilse gözlemcilerin belirlediği koordinatlar arasında uyumsuzluk olduğu anlaşılır (Ercan ve ark., 2008; Lele ve Richtsmeier, 2001). Çalışmalarda gözlemcilerin belirlediği koordinatlar arasında farklılık varsa

bunun farklı nedenleri olabilir. Değerlendirici veya değerlendiricilerin ölçüm aracı ya da landmarkların konumlandırıldığı nesne hakkında fazla bilgisi olmayabilir. Bununla birlikte eğer landmark koordinatları bir görüntüden belirleniyorsa, görüntünün çözünürlüğü de landmark konumlandırılmasını etkileyebileceği için kontrol edilmelidir (Ercan ve ark., 2008; Lim ve Foong, 1997; McWilliam ve Welander, 1978).

Tek bir değerlendirici olması durumunda, tüm deneklere ya da deneklerin bir kısmına ait landmark koordinatları farklı bir değerlendirici tarafından da belirlenmelidir. Bu işlem çalışmada landmark koordinatlarını belirleyen değerlendirici ile diğer değerlendiriciler arasındaki uyum düzeyi hakkında bilgi verecektir. Landmark koordinatlarını belirlemek için kullanılan bir diğer yöntem ise aynı gözlemcinin landmark koordinatlarını tekrar belirlemesidir. Landmark koordinatlarının tekrar belirlenmesi değerlendiricinin belirlediği koordinatlar arasındaki tutarlılığı hakkında bilgi verecektir (Ercan ve ark., 2008).

Landmark türü, landmark güvenilirliğini etkileyebilecek faktörlerden biridir. Bazı landmarkların konumları ve belirleme kriterleri diğer landmarklara göre daha belirsizdir. Bu tip landmarkların güvenilirliği de daha düşük olacaktır (Dryden and Mardia, 1998; Lele and Richtsmeier, 2001; Valeri ve ark., 1998; Williams and Richtsmeier, 2003).

Literatür incelendiğinde landmark güvenilirliğini belirlemede bağımsız örneklem t testi, varyans analizi ve sınıf içi korelasyon katsayısının kullanıldığı görülmektedir (Aldridge ve ark., 2005; Jamison ve Ward, 1993; Williams ve Richtsmeier, 2003). Ercan ve ark. (2008), iki veya üç boyutlu landmark koordinatlarını birim, değerlendirici, landmark ve etkileşimlerini eşzamanlı olarak dikkate alarak güvenilirlik düzeyi hakkında bilgi elde etmek amacıyla, genelleştirilmiş güvenilirlik teorisi yaklaşımından hesaplanan, G katsayısını uyarlamışlardır.

2.3.2. Landmark Seçim Kriterleri

Landmarklar, organizmalardaki aynı bölge veya nokta olarak tanımlanabilecek anatomik bölgelerdir. Landmarklar, biyolojik veya anatomik anlamlılığa sahip özelliklerin keşfedilmesi için yeterince kapsamlı morfoloji örneği sağlamalıdır. Biyolojik veya anatomik anlamlılığa sahip özelliklerin keşfi, biyolojik veya anatomik olarak anlamlı olmayacağı düşünülen değişkenlerin çalışma dışında bırakılması anlamına gelmez. Çalışma başında seçilen landmarklar, çalışma sonucunda neyi keşfedebileceğinizi belirlediği için landmark seçimi çok önemlidir (Zelditch ve ark., 2004).

Landmark koordinatlarının önemli filogenetik, gelişimsel ve işlevsel bilgileri taşıyabileceği göz önüne alındığında, herhangi bir morfometri çalışmasına başlamadan önce hangi landmarkların dikkate alınacağı ve analiz edileceği dikkatlice düşünülmelidir (Reitner ve ark., 2011).

Seçilen landmarkların morfolojik değişkenliği belirlemede yeterli olması gerekmektedir. Yetersiz sayıda landmark seçimi nesnelere arasında genel şekil farklılıklarını belirleyen yerel şekil farklılıklarının belirlenmesini zorlaştıracaktır. Doğru ve yeterli landmark seçimi, nesnelere meydana gelen deformasyonlardan dolayı arkeoloji ve paleontoloji dallarında daha da önem kazanmaktadır. Bununla birlikte, çok fazla landmark seçimi de veri toplama aşamasının uzamasına ve karmaşıklaşmasına neden olacaktır. Fazla landmark seçimi şeklin görselleştirilmesi için yararlı olmasına rağmen, landmark sayısı örneklem büyüklüğünden fazla olacağı için istatistiksel yöntemlerin uygulanmasında sorun olacaktır (Watanabe, 2018).

İdeal landmark seçiminde dikkat edilmesi gereken bazı özellikler vardır (Webster ve Sheets, 2010; Zelditch ve ark., 2004). Bunlar;

- Homolog anatomik noktalar olması
- Diğer landmarklarla ilişkili olarak topolojik konumlarının değişmemesi
- Yeterli morfolojik kapsama sahip olması
- Tekrarlanabilir ve güvenilir olması
- Aynı düzlemde yer alması

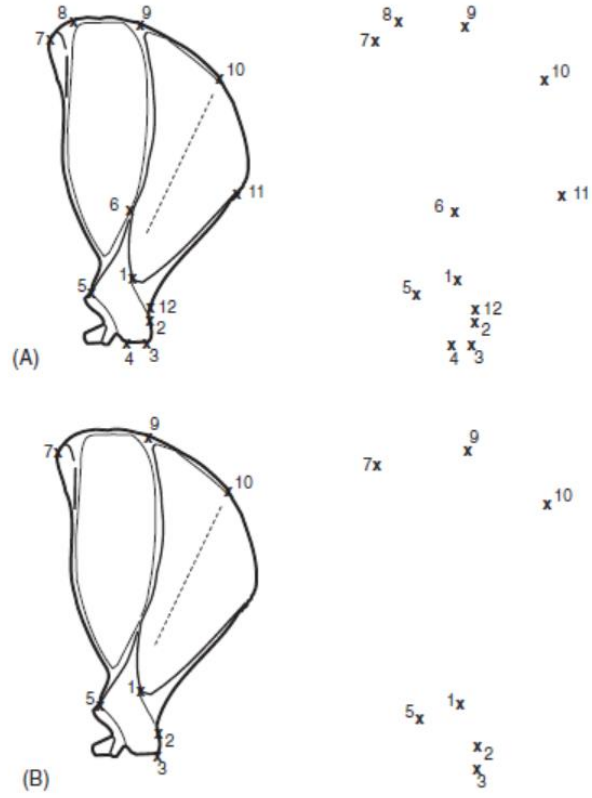
Homoloji kavramı, landmark tabanlı morfometride önemli bir yere sahiptir. Birçok geleneksel morfometrik çalışma homoloji ile ilgili olsa da, ölçümleri seçerken homoloji temel sorun olarak ele alınmamıştır. Eğer homoloji temel sorun olarak ele alınsaydı en büyük kranium genişliği gibi bazı standart değişkenler ölçülemezdi. En büyük kranium genişliği farklı organizmalarda farklı noktalar yardımıyla ölçülebileceğinden, bu gibi değişkenlerin homolog olması gerekmemektedir. Sonuç olarak, kraniumun genişlemesinin kranium şeklindeki değişikliklerden nasıl etkilendiği belirlenemez.

Geometrik morfometride ise homoloji hem matematiksel hem de biyolojik nedenlerden dolayı landmarkların seçiminde en önemli kriterlerden biridir. Homolojinin hem matematiksel hem de biyolojik önemini anlamak önemlidir. Bazı durumlarda homolojileri şüpheli olsa da landmarkları çalışmaya dahil etmek için nedenler vardır.

Morfometride biyolojik homolojinin önemi, morfometri literatüründe yer alan homoloji tanımında gizlidir. Bazı morfometrisyenler ile biyologlar homolojiyi farklı tanımlarlar. Biyologlar genellikle homolojiyi organizmadaki kısımlar veya karakterler açısından ele alırken, matematikçiler homolojiyi bu kısımlar üzerindeki bireysel noktalar olarak ele alırlar.

İlgilenilen şekiller çok farklı olduğunda morfometrik analizler doğru bir şekilde uygulanamaz. Örneğin, zamanla incelenen kemiklerin yapısı tamamen değişirse, şekiller geometrik olarak analiz etmek için çok farklı hale gelebilir. Ayrıca bazı durumlarda landmarklar tamamen deforme olabilir. Bu gibi değişikliklere uğramış şekil, morfometrik analiz için uygun olmayacaktır.

Landmark seçiminde önemli kriterlerden bir diğeri de şeklin yeterince kapsanmasıdır. Şekil analizinde temel veriler landmarklar olduğu için şekli en iyi şekilde ifade edecek landmarkların belirlenmesi gereklidir. Ayrıca eğer landmarklar tanımlanmazsa spesifik bölgelerdeki değişikliklerin belirlenmesi mümkün olmayacaktır. Şeklin yeterli kapsanıp kapsanmadığı seçilen landmarklar yardımıyla anlaşılmaktadır (Şekil 4).



Şekil 4. Farklı kapsama derecelerine göre sincapın kürek kemiğindeki landmarklar:
A) ayrıntılı kapsama B) sınırlı kapsama (Zelditch ve ark., 2004)

Şekil 4-A) da görüldüğü gibi doğru ve yeterli landmarkların seçimi ile şekil kapsamlı olarak ifade edilmiştir.

Landmark seçiminde önemli kriterlerden biri, landmarkların hatasız olarak tekrar belirlenebilmesidir (Shimoda and Yamazaki, 2015). Eğer bir landmarkı belirlemek zorsa, tekrarlanan ölçümlerde belirli bir yönde yanlı işaretlemeler meydana gelecektir.

Landmarkların seçiminde son kriter, üç boyutlu organizmaların iki boyutlu analiz edilmesiyle ilişkilidir. Üç boyutlu analizlerin uygulama zorluğu, kullanılan programların maliyeti gibi sebepler araştırmacıları iki boyutlu analizleri uygulamaya yönlendirmektedir. Bu durumda üç boyutlu uzayda şekiller daha yeterli kapsanacakken çalışmaların iki boyutlu düzlemde yapılmasına ve veri kaybına neden olmaktadır.

2.4. Geometrik Morfometride Landmark Tabanlı Yaklaşımlar

Landmark tabanlı geometrik morfometri, şeklin biyolojik yapısını, şekil değişkenliğini ve şeklin diğer faktörlerle olan ilişkisini açıklayan güçlü bir yaklaşımdır. Analizler sonucunda şekil farklılıklarından ortaya çıkan grafiksel gösterimler görsel olarak çekici ve sezgiseldir. Geleneksel morfometri, morfolojiyi uzunluk ölçümleri, oranlar veya açılar açısından özetlemeyi içerirken landmark tabanlı geometrik morfometride ise landmark konfigürasyonu açısından iki ya da üç boyutlu kartezyen koordinatları kullanılarak şekil özetlenmektedir. Geometrik morfometri, organizmalardan elde edilen landmark verileri arasındaki spatial ilişkiyi belirlemesi nedeniyle güçlü ve popülerdir (Webster and Sheets, 2010).

Geometrik morfometride kullanılan landmark tabanlı yöntemlerinden bazıları ince levha eğri analizi (Thin plate spline analysis), sonlu element morfometrisi, Procrustes analizi ve Öklid Uzaklık Matrisi analizidir (EDMA) olarak belirtilebilir. Bu yöntemler arasında Procrustes analizi ve EDMA sıklıkla kullanılmaktadır (Ercan ve ark., 2012).

2.4.1. Procrustes Analizi

Procrustes analizi, geometrik morfometride en çok kullanılan süperimpozisyon yöntemidir. Procrustes analizinde şekil konumundan kaynaklanan farklılıklar, konfigürasyonların merkezileştirilmesiyle sağlanır. Her bir konfigürasyonun yeniden ölçeklendirilmesiyle konfigürasyonlar arası şekil farklılıkları ortadan kaldırılır. İki konfigürasyon arasındaki farklılıkların ortadan kaldırılması, bir konfigürasyonun (hedef form) merkez çevresinde döndürülmesiyle gerçekleştirilir. Hedef ve referans landmark konumları arasındaki uyumsuzluk, ilgili landmarklar arasındaki uzaklığın karesi alınarak hesaplanır. Hesaplanan uzaklığın karesi, ilgili konfigürasyonlar arasındaki Procrustes uzaklığı olarak adlandırılmaktadır. Procrustes analizinde, bu uzaklıkların toplamını minimuma indirmek amaçlanmaktadır. Tüm landmark konfigürasyonlarının genel bir noktaya ötelenmesi, yeniden ölçeklendirilmesi ve

döndürülmesi Genelleştirilmiş Procrustes analizi olarak adlandırılmaktadır (Dryden and Mardia, 1998). Procrustes analizi ile öteleme, ölçekleme ve döndürme işlemleri kullanılarak landmark konfigürasyonları arasındaki farklılıklar ortadan kaldırıldığı için bu konfigürasyonlar arasındaki herhangi bir farklılık, şekil farklılıklarının sonucu olmalıdır (Webster and Sheets, 2010).

$v = (v_1, v, \dots, v_k)^T$ ve $w = (w_1, w_2, \dots, w_k)^T$ aynı düzlemde merkezileştirilmiş konfigürasyonlar olmak üzere v^* , v' nin kompleks eşleniği ve $v^*1_k = 0 = w^*1_k$ olarak ele alınsın. Şekil konfigürasyonlarının karşılaştırılması için iki şekil arasındaki uzaklığın belirlenmesi gerekmektedir. Şekiller arasındaki uzaklığın belirlenmesi için uygun olan yöntem, benzerlik dönüşümleri kullanılarak w ve v arasındaki şekil farklılığının büyüklüğünü gösteren, gözlenen ve tahmin edilen v arasındaki farkların kullanılmasıdır. Bu amaçla kullanılan kompleks regresyon denklemi eşitlik-1' de verilmiştir (Dryden and Mardia, 1998):

$$\begin{aligned} v &= (a + ib)1_k + \beta e^{i\theta} w + \varepsilon \\ &= [1_k, w]A + \varepsilon \\ &= X_D A + \varepsilon \end{aligned} \tag{1}$$

Eşitlik-1' de $A = (A_1, A_2)^T = (a + ib, \beta e^{i\theta})^T$, öteleme $(a + ib)$, ölçekleme $(\beta > 0)$ ve döndürme $(0 \leq \theta < 2\pi)$ ile 2×1 ' lik kompleks parametrelerdir. Ayrıca ε , $k \times 1$ ' lik hata vektörü ve $X_D = [1_k, w]$ $k \times 2$ ' lik tasarım matrisidir.

Süperimpozisyonu uygulamak için, en küçük kareler fonksiyonu minimuma indirgenerek A tahmin edilmelidir. Hata kareleri toplamı eşitlik-2' de belirtilmiştir.

$$D^2(y, w) = \varepsilon^* \varepsilon = (v - X_D A) * (v - X_D A) \tag{2}$$

\hat{A} ' nın tahmin edilmesiyle w ' nun y üzerine tam Procrustes süperimpozisyonu elde edilebilir.

$$\hat{A} = (\hat{a} + i\hat{b}, \hat{\beta} e^{i\hat{\theta}})^T = \operatorname{arginf} \varepsilon^* \varepsilon = \operatorname{arginf} (v - X_D A) * (v - X_D A) \tag{3}$$

2.4.2. Öklid Uzaklık Matrisi Analizi (EDMA)

İstatistiksel şekil analizinde bir nesnenin formu, öteleme, ölçekleme ve döndürme işlemleri sonucunda değişmeden kalan karakteristikleri olarak tanımlanmaktadır. EDMA, Lele ve Richtsmeier (2001) tarafından Öklid uzaklık matris gösterimi kullanılarak ortalama form ve form değişkenliğini tahmin etmek için önerilmiştir. EDMA, eşdeğer konfigürasyonlardaki landmark oranlarının karşılaştırılmasıyla form değişkenliğinin ve büyüme farklılıklarının incelenmesine olanak sağlamaktadır. Bu yöntem ile organizmaların formu iki ya da üç boyutlu koordinatlar kullanılarak karşılaştırılabilmektedir (Anwary, 2012).

X konfigürasyonundaki tüm landmarklar arasındaki uzaklıkları gösteren $k \times k$ boyutlu form matrisi $FM(X)$ ile gösterilsin. X_1, X_2, \dots, X_n X anakütlesinden seçilen n birimlik örneklemin landmark koordinat matrisleri olsun. $e_{lm,i}$ i' ninci birey için l ve m landmarkları arasındaki karesi alınmış Öklid uzaklığı ve $lm=1, 2, \dots, k$ olmak üzere X anakütlesinden $FM(X)$ ' i tahmin etmek için izleyen denklemler uygulanır (Brombin ve Salmaso, 2013):

$$\bar{e}_{lm} = n^{-1} \sum_{i=1}^n e_{lm,i}$$

$$s^2 = n^{-1} \sum_{i=1}^n (e_{lm,i} - \bar{e}_{lm})^2$$

$$\hat{\delta}_{lm} = (\bar{e}_{lm}^2 - s^2(e_{lm}))^{0,5}$$

$$FM(X) = (\hat{\delta}_{lm}^{0,5})$$

(4)

Benzer şekilde, Y_1, Y_2, \dots, Y_n Y anakütlesinden seçilen n birimlik örneklemin landmark koordinat matrisleri olmak üzere, $FM(Y)$ hesaplandıktan sonra $FM(X)$ ve $FM(Y)$ matrisleri kullanılarak EDMA-I test istatistiği hesaplanır.

$$T = \frac{\max(FDM(X, Y))}{\min(FDM(X, Y))}$$

(5)

Eşitlik-5' te $FDM(X,Y)_{ij} = FM(X)_{ij}/FM(Y)_{ij}$ olarak hesaplanmaktadır. EDMA-II test istatistiği ise eşitlik-6' da gösterildiği gibi hesaplanmaktadır.

$$Z = \max|S_X - S_Y|$$

(6)

Eşitlik-6' da S_X ve S_Y ortalama form matrislerini göstermektedir. EDMA-I' de varyans-kovaryans matrislerinin eşitliği varsayımı varken EDMA-II de varyans-kovaryans matrislerinin eşit olması gerekmemektedir. EDMA-I ve EDMA-II yöntemlerinin özellikleri Tablo 1' de verilmiştir (Lele ve Richstmeier, 2001).

Tablo 1. EDMA yöntemlerinin genel özellikleri ve karşılaştırılması

EDMA-I	EDMA-II
Formlar koordinat sisteminden bağımsız olarak tanımlanır ve karşılaştırılır.	Formlar koordinat sisteminden bağımsız olarak tanımlanır ve karşılaştırılır.
Yokluk hipotezi şekillerin benzer olduğunu belirtir.	Yokluk hipotezi şekillerin benzer olduğunu belirtir.
Sadece ölçek ya da form farklılıklarından kaynaklanan farklılıkların belirlenmesini sağlar.	İstatistiksel olarak anlamlı farklılıkların, şekildeki ve ölçeklendirme faktöründeki farklılıklara ayrışmasını sağlar.
Ortalama form matrisi ölçeklendirme faktörünün seçiminden etkilenmez.	Ortalama form matrisi ölçeklendirme faktörünün seçimine bağlıdır.
Örneklemlerin varyans-kovaryans matrisi eşit olmalıdır.	Örneklemlerin varyans-kovaryans matrisinin eşit olması gerekmemektedir.
Test istatistiği konumdan kaynaklanan farklılıkları belirleyememektedir. Bu amaçla form farklılıkları matris elementlerine ait güven aralıkları geliştirilmiştir.	Test istatistiği konumdan kaynaklanan farklılıkları belirleyememektedir. Bu amaçla şekil farklılıkları matris elementlerine ait güven aralıkları geliştirilmiştir.

2.5. Kayıp Veri Tahmin Yöntemleri

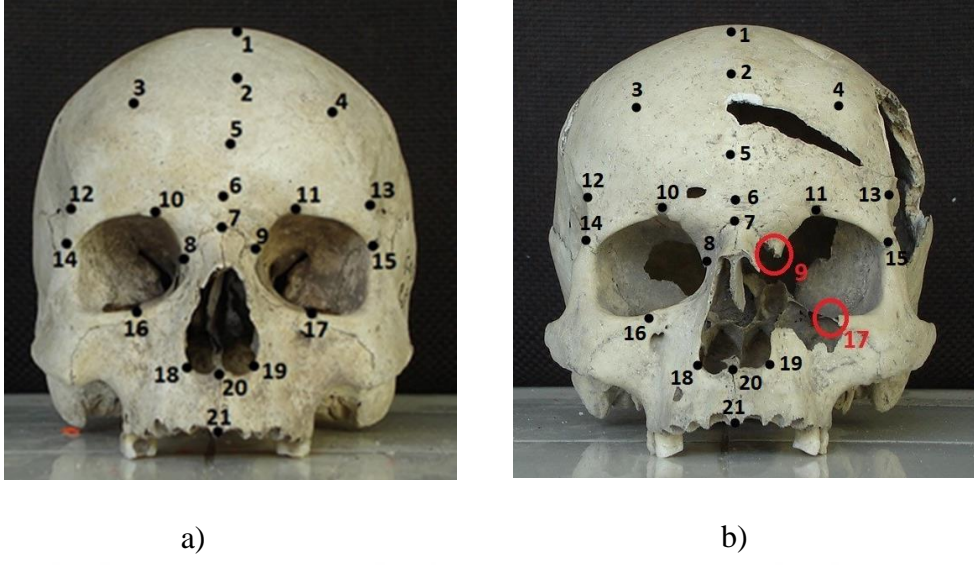
Kayıp veriler, birçok araştırmada önemli bir sorun olarak ortaya çıkmaktadır. İstatistiksel analizlerde, araştırmacıların ilgilendikleri durum hakkında çıkarım yapmak için veri matrisi kullanmaları gerekmektedir. İlgilenilen veri matrisinde gözlenemeyen durumlar olduğunda kayıp veri ortaya çıkmaktadır.

Kayıp veri kavramı, Little ve Rubin (1987) tarafından tamamen rastgele (missing completely at random (MCAR)), rastgele (missing at random (MAR)) ve rastgele olmayan (not missing at random (MNAR)) kayıp veri olmak üzere üç grupta

sınıflandırılmıştır (Rubin, 1976; Schmitt ve ark., 2015). MCAR, kayıp değer meydana gelme olasılığının veri karakteristiklerinden bağımsız olduğu durumlarda ortaya çıkmaktadır. MAR ise sadece gözlenen değişkenlere bağlı olarak kayıp veri meydana gelme olasılığıdır. Kayıp veri meydana gelme olasılığı, kayıp veri içeren değişkenlere bağlı olduğunda ise MNAR olarak adlandırılmaktadır (He, 2010).

Literatür incelendiğinde anket veya ölçek çalışmalarında kayıp veri meydana gelme sebeplerinin madde yanıtlanmama (item nonresponse) ve birim yanıtlanmama (unit nonresponse) olmak üzere iki gruba ayrıldığı görülmektedir (Graham, 2012). Madde yanıtlanmama durumunda bazı değişkenlerde kayıp veri meydana gelirken; birim yanıtlanmama durumunda ise tüm değişkenlerde kayıp veri meydana gelmektedir. Ölçekte madde yanıtlanmama ve cevapsız bırakılan anket soruları madde yanıtlanmamaya örnek verilebilir (Schmitt ve ark., 2015). Birim yanıtlanmama ise deneklere ulaşılamadığında, tedaviyi reddetme vb. sebeplerle ortaya çıkmaktadır.

Morfolojik ve özellikle paleontolojik çalışmalarda fosilleşme ve zamana bağlı aşınmalar nedeniyle landmark koordinatlarında kayıplar sıklıkla meydana gelmektedir. Geometrik morfometrik yöntemlerde tüm ölçümlerin homolog ve aynı landmark sayısına sahip olması gerektiği için kayıp veri içeren birimler ya analizden çıkartılmalıdır ya da eksik verilerin tahmin edilmesi gerekmektedir (Mitteroecker and Gunz, 2009). Kayıp verilerin çalışmadan çıkarılması, özellikle fosilleşme nedeniyle denek sayısının az olduğu paleontolojik çalışmalarda çok daha az arzu edilen bir durumdur. Paleontolojik verilerde sıklıkla kayıp landmarklarla karşılaşmaktadır. Bu nedenle paleontolojistlerin yaklaşımları mevcut olan landmarklarla çalışmak, kayıp landmark içermeyen örnekleri seçmek, ya da kayıp landmark içeren örnekleri çalışma dışı bırakmaktır. Bu yaklaşımlar sonucunda paleontologlar ya morfolojik bilgi kaybı ya da örneklem büyüklüğünü azaltmak seçenekleriyle karşı karşıya kalmaktadır. Bu durumlar büyük örneklerle çalışıldığında sorun olmazken, küçük örneklerde bu yaklaşımlar istatistiksel analizlerin uygulanmasında sorun yaratabilmektedir (Şekil 5) (Couette and White, 2010).



Şekil 5. Kranium görünümü. a) Kayıp landmark içermeyen kranium, b) Deformasyona uğrayan kranium

Kayıp verilerin göz ardı edilmesi, örneklemin rastgeleliğini bozarak sonuçların genelleştirme ihtimalini ortadan kaldırmaktadır (Little ve Rubin, 1987; Rubin, 1976). Kayıp verinin yoğunluğu, istatistiksel çıkarımlarda gücün düşmesine ve parametre tahminlerinde sapmalara neden olabilmektedir.

Kayıp veri tahmini için literatürde birçok yöntem yer almaktadır. Bu yöntemler genellikle kayıp veri atama yöntemleri olarak adlandırılmaktadır (Schmitt ve ark., 2015). Kayıp veri ile ortalamanın yer değiştirmesi, EM algoritması ve çoklu regresyon atama yöntemleri en çok kullanılan kayıp veri atama yöntemleri arasında yer almaktadır (Adams ve Rohlf, 2004). Bu yöntemlerin dışında PCA' nın modifiye edilmesiyle kayıp veri tahmini yapan yaklaşımlar da önerilmiştir (Nounou ve ark., 2002; Scholz ve ark., 2005; Stacklies ve ark., 2007).

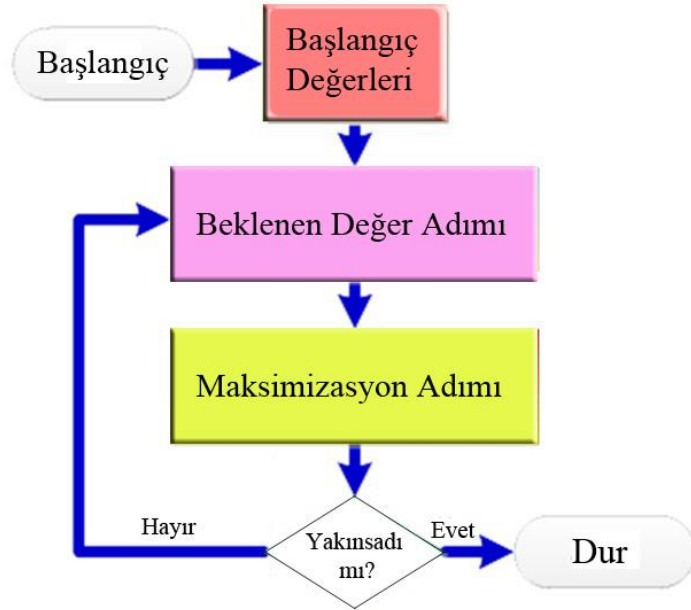
İstatistiksel şekil analizinde kayıp veri sorunu ilgilenilen landmarklara ait kartezyen koordinatların belirlenemediği durumlarda ortaya çıkmaktadır. Landmark koordinatlarının belirlenememesi, ilgili landmarkın çalışma dışı kalmasına neden olmaktadır. Kayıp veri tahmininde sıklıkla kullanılan EM algoritması, çoklu regresyon atama, Bayes yaklaşımı PCA (BPCA), olasılıksal PCA (PPCA), ters doğrusal olmayan PCA (NLPCA) ve iteratif kısmi en küçük kareler ile doğrusal olmayan

tahmine dayalı PCA (INIPALS) yöntemleri ve tez çalışmasında önerilen F yaklaşımı algoritmasının kayıp landmark tahminindeki performansları karşılaştırılmıştır.

2.5.1. EM (Expectation Maximization) Algoritması

EM algoritması kayıp verilerin tahmininde maksimum benzerlik tahminlerini kullanmaktadır (Couette and White, 2010; Hunt and Jorgensen, 2003). EM algoritması, veri seti tamamen gözlenemediğinde parametrik modeller için en çok olabilirlik tahminlerini kullanan genel bir yöntemdir (Schafer, 1997).

EM algoritması kayıp gözlem içermeyen veri setinde log olabilirlik fonksiyonunu maksimize ederek parametreleri tahmin etmektedir (Dempster ve ark., 1977; Dong and Peng, 2013). EM algoritması iteratif ve iki aşamalı bir yöntemdir (Ng ve ark., 2002). E aşamasında eksik veri için en iyi tahminler, M aşamasında ise eksik veri atandığında ortalama, varyans, kovaryans vb. için tahminler elde edilir (Dempster ve ark., 1977; Honaker ve ark., 2016; Ng ve ark., 2002). EM algoritmasının akış şeması Şekil 6' da verilmiştir.



Şekil 6. EM algoritmasının akış şeması

Kayıp veri içermeyen veri setleri ile çalışırken betimleyici istatistikler ve betimleyici istatistiklere ait en çok olabilirlik tahminleri kolaylıkla hesaplanabilmektedir. Kayıp veri içeren veri setlerinde de yapılan tahminler en çok olabilirlik tahminlerine dayanmaktadır. Bununla birlikte kayıp veri olduğunda olabilirlik tahminlerinin hesaplanması kayıp veri olmayan durumda olduğu kadar kolay olmamaktadır. Bu nedenle, Dempster ve ark. (1977) EM algoritmasını önermişlerdir.

EM algoritması iki aşamalı bir süreçtir. E aşamasında kayıp veriler için “beklenen değerler” gözlenen değerleri kullanarak regresyon denklemleri ile hesaplanarak regresyon denklemlerinden elde edilen sonuçlara göre kayıp gözlem koşullu ortalama ile yer değiştirilir (Dempster ve ark., 1977; Rubin ve ark., 2007). M aşamasında ise E aşamasında elde edilen istatistiklere dayalı olarak log olabilirlik fonksiyonunu maksimize ederek tahminler güncellenir. Bu iki aşamalı süreç araştırmacının belirlediği iterasyon katsayısı kadar, tahminlerde yakınsama hedeflenerek, devam eder (Rubin ve ark., 2007).

$f(x|\Phi)$ regular üstel aile fonksiyonlarını göstermek üzere

$$f(x|\Phi) = b(x) \exp(\Phi t(x)^T) / a(\Phi)$$

(7)

şeklinde tanımlanmaktadır. Eşitlik-7’ de Φ $1 \times r$ vektör parametrelerini, $t(x)$ kayıp veri olmayan veri setine ait istatistiklerin $1 \times r$ vektörünü göstermektedir. Regular terimi, Φ vektörünün r -boyutlu konveks Ω seti ile sınırlandırıldığını ifade etmektedir. $t(x)$ seçimine karşılık olduğu gibi Φ parametrelendirmesi tekil olmayan $r \times r$ lineer dönüşümüne bağlı olarak benzersizdir. Bu gibi parametreler genellikle doğal parametreler olarak adlandırılmaktadır.

Eşitlik-7’ de tanımlanan fonksiyonun varsayımları yerine getirildiğinde EM algoritması aşağıdaki gibi tanımlanmaktadır:

Φ^p , nin algoritma p kez uygulandıktan sonra Φ ’ nin değerini gösterdiğini varsayalım. Bir sonraki adım izleyen şekilde devam eder:

E adımı: t^p fonksiyonu bulunarak $t(x)$ ’ e istatistikler tahmin edilir.

$$t^p = E(t(x)|y, \Phi^p) \quad (8)$$

M adımı: Denklemlerin çözümü olarak Φ^{p+1} bulunur.

$$E(t(x)|\Phi) = t^p \quad (9)$$

Eşitlik-9' da yer alan denklemler üstel fonksiyona uygunluk gösteren verilerin en çok olabilirlik tahmini için kullanılan olabilirlik denklemlerine benzer formdadır. Dolayısıyla t^p ' nin eşitlik-7' den elde edilen gözlenen x ' ten hesaplanan istatistikleri gösterdiğini varsayarsak, eşitlik-9'daki denklem Φ tahminine ait en çok olabilirlik tahmincilerini ifade etmektedir (Dempster ve ark., 1977).

EM algoritmasının kayıp veri tahmin aşamalarında daha ayrıntılı gösterimi Dong ve ark. (2013) tarafından ifade edilmiştir. E adımında veri setinde yer alan parametrelerin log olabilirlik fonksiyonunun beklenen değeri hesaplanmaktadır. Veri setinin (Y) gözlenen kısım Y_{obs} ve kayıp gözlem içeren kısım Y_{mis} olmak üzere iki bölümden oluştuğunu varsayalım. Y değişkeninin dağılımı, bilinmeyen θ parametresine bağlı olarak eşitlik-10' daki gibidir.

$$P(Y|\theta) = P(Y_{obs}, Y_{mis}|\theta) = P(Y_{obs}|\theta)P(Y_{mis}|Y_{obs}, \theta) \quad (10)$$

Eşitlik-10 olabilirlik fonksiyonu olarak eşitlik-11' deki gibi yazılabilir.

$$L(\theta|Y) = L(\theta|Y_{obs}, Y_{mis}) = cL(\theta|Y_{obs})P(Y_{mis}|Y_{obs}, \theta) \quad (11)$$

Eşitlik-11' de c, model parametreleri ve kayıp parametrelerin bağımsızlığı varsayımı altında, Θ ' dan bağımsız, görmezden gelinebilen sabit bir katsayıdır (Dong and Peng, 2013; Schafer, 1997). Eşitlik-11' deki eşitliğin iki tarafının da logaritması alındığında, log olabilirlik fonksiyonu, eşitlik-12' deki şekilde elde edilir.

$$l(\theta|Y) = l(\theta|Y_{obs}) + \log P(Y_{mis}|Y_{obs}, \theta) + \log c \quad (12)$$

Eşitlik-12' de $l(\theta|Y) = \log P(Y|\theta)$ kayıp gözlem içermeyen veriye ait log olabilirlik fonksiyonunu, $l(\theta|Y_{obs})$ gözlenen veri setine ait log olabilirlik fonksiyonunu, $P(Y_{mis}|Y_{obs}, \theta)$ θ verildiğinde kayıp gözlemin tahmini dağılımını ve $\log c$ ise sabit bir değeri göstermektedir (Dong and Peng, 2013; Schafer, 1997). $\log c$, θ parametresinin tahminini etkilemediği için izleyen hesaplamalarda bu terim göz ardı edilmiştir.

Y_{mis} bilinmediği için kayıp gözlem içermeyen veri setine ait log olabilirlik fonksiyonu doğrudan belirlenememektedir. Eğer θ parametresinin geçici ya da $\theta^{(t)}$ ile gösterilen ilk tahmini yapılmışsa kayıp gözleme ait dağılımın $P(Y_{mis}|Y_{obs}, \theta^{(t)})$ olduğu varsayıldığında $l(\theta|Y)$ ' nin beklenen değeri eşitlik-13' te gösterildiği gibi hesaplanabilmektedir.

$$\begin{aligned}
Q(\theta|\theta^{(t)}) &= E[l(\theta|Y)|Y_{obs}, \theta^{(t)}] \\
&= \int l(\theta|Y)P((Y_{mis}|Y_{obs}, \theta^{(t)})dY_{mis} \\
&= l(\theta|Y_{obs}) + \int \log P(Y_{mis}|Y_{obs}, \theta)P(Y_{mis}|Y_{obs}, \theta^{(t)})dY_{mis}
\end{aligned}
\tag{13}$$

Eşitlik-13' te belirtilen $Q(\theta|\theta^{(t)})$ hesaplanması EM algoritmasının E adımıdır.

M adımında ise E adımında elde edilen değer maksimize edilerek θ parametresinin bir sonraki tahmini elde edilir:

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} Q(\theta|\theta^{(t)})
\tag{14}$$

EM algoritması gözlenen veri setine bağlı olan θ^0 tahmini ile başlayıp E ve M adımları hesaplanılarak devam etmektedir. θ parametresine ait tahminler neredeyse aynı olduğunda algoritma sonlandırılır (Dempster ve ark., 1977; Dong and Peng, 2013).

Araştırmalarda kullanılan farklı EM algoritmaları literatürde yer almaktadır. Collins ve ark. (1992), gizil modellerde kullanılmak üzere EM algoritmasını

uyarlamışlardır. Rubin ve ark. (1982), faktör analizinde kullanılmak üzere EM algoritmasını önermişlerdir. EM algoritmasının önerilen her versiyonunda temel amaç ilgilenilen parametre için en çok olabilirlik tahminlerini elde etmektir (Graham, 2012).

2.5.2. Çoklu Regresyon Atama Yöntemi

Çoklu regresyon atama yöntemi, kayıp veri tahmininde sıklıkla kullanılan yöntemlerden biridir. Her bir kayıp değere tek tek değer atamak yerine çoklu değer atama, her bir kayıp değeri olası değerler seti ile yer değiştirir. Daha sonra çoklu değer atama ile tahmin edilmiş veri setleri standart yöntemler kullanılarak analiz edilerek elde edilen sonuçlar birleştirilir. Standart yöntemlerden hangisinin kullanıldığına bakılmaksızın farklı veri setlerinden elde edilen sonuçları birleştirme aşaması temelde aynıdır.

Çoklu değer atama, türetilmiş verilerle her bir değeri tahmin etmeye çalışmak yerine kayıp değerlerin rasgele örneklemine temsil eder. Bu süreç, kayıp veriler nedeniyle belirsizliği düzgün şekilde yansıtan geçerli istatistiksel çıkarımlarla sonuçlanır.

Çoklu değer atama 3 farklı aşamadan oluşmaktadır:

Adım 1: Kayıp veri m adet veri seti üretmek için m kez tahmin edilir.

Adım 2: m adet tahmin edilmiş veri seti standart yöntemlerle analiz edilir.

Adım 3: m adet tahmin edilmiş veri setinin analizinden elde edilen sonuçlar birleştirilir.

Değer atama adımı, çoklu atama yöntemine ait adımlardan en karmaşık olanıdır. Değer atama adımının amacı, gözlenen veri setine ait bilgiyi birden fazla kullanarak kayıp değeri tahmin etmektir. Bu aşamada regresyon analizi ve Markov Chain Monte Carlo yöntemi gibi birçok farklı istatistiksel yöntem kullanılmaktadır (Dong and Peng, 2013).

Eksik veri içeren her bir değişkenin diğer tüm değişkenlerle regresyon analizi yapılmaktadır (Couette and White, 2010; Dong and Peng, 2013; Little and Rubin, 1987; Zhou ve ark., 2001). Veri setinde Y_1, Y_2, \dots, Y_p olmak üzere p adet değişken olduğu ve $1 < j \leq p$ olmak üzere Y_j ile Y_p arasında kayıp gözlemler olduğu varsayalım. j ' ninci değişkene ait kayıp gözleme değer atamak için Y_1 ' den Y_{j-1} ' e kadar olan gözlenen değerler kullanılarak, kayıp değer içeren Y_j değişkeni için regresyon modeli uygulanır (Dong and Peng, 2013).

$$Y_j = \beta_0 + \beta_1 Y_1 + \beta_2 Y_2 + \dots + \beta_{(j-1)} Y_{(j-1)} \quad (15)$$

Tahmin edilen modelde $(\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_{(j-1)})$ parametre tahminleri yer almaktadır (Yuan, 2010).

Her bir atamada, yeni parametreler $(\hat{\beta}_{*0}, \hat{\beta}_{*1}, \dots, \hat{\beta}_{*(j-1)})$ ve σ_j^2 , kayıp veri için bir sonraki tahmin edilen dağılımdan elde edilir. Tahmin edilen kayıp veriler eşitlik-16' dan elde edilen değerler ile yer değiştirilir.

$$\hat{\beta}_{*0} + \hat{\beta}_{*1} y_1 + \hat{\beta}_{*2} y_2 + \dots + \hat{\beta}_{(j-1)} y_{(j-1)} + z_i \sigma_{*j}^2 \quad (16)$$

Y_j ' ye ait kayıp gözleme değer atandıktan sonra, kayıp gözlem içermeyen veri seti elde edilinceye kadar Y_{j+1}, \dots, Y_p değişkenleri için de regresyon analizi uygulanır (Dong ve Peng, 2013; Rubin, 1987).

İkinci adım olan istatistiksel analiz adımında m adet veri seti birbirinden bağımsız şekilde analiz edilir. Bu adımın sonunda m adet veri setinin analizi sonucunda m adet parametre tahmini elde edilir.

Çoklu değer atamanın üçüncü adımında m adet tahmin birleştirilmektedir. Little ve ark. (1987) m nokta tahminini ve standart hatalarını birleştiren bir formül önermiştir (Dong ve Peng, 2013). \hat{Q}_i ' nin i ' inci veri setindeki Q parametresine (örneğin regresyon katsayısı) ait parametre tahminini ve \hat{U}_l varyans tahminini

gösterdiği varsayılsın. Birleştirilmiş nokta tahmini Q , eşitlik-17' deki gibi hesaplanmaktadır.

$$\bar{Q} = \frac{1}{m} \sum_{i=1}^m \hat{Q}_i \quad (17)$$

\bar{Q} parametresine ait varyans, grup içi değer atamasına ait varyans değeri (\bar{U}) ve gruplar arası değer atamasına ait varyans değerinin (B) ağırlıklandırılmış toplamıdır.

$$\bar{U} = \frac{1}{m} \sum_{i=1}^m \hat{U}_i \quad (18)$$

$$B = \frac{1}{m-1} \sum_{i=1}^m (\hat{Q}_i - \bar{Q})^2 \quad (19)$$

$$T = \bar{U} + \left(1 + \frac{1}{m}\right)B \quad (20)$$

Eşitlik-20' de $\frac{1}{m}$ sonlu sayıda değer atama ile ilişkili rastgelelik düzeltmesidir.

$(Q - \bar{Q})/\sqrt{T}$, t dağılımına uygunluk göstermektedir. Serbestlik dereceleri (v_m ya da v_m^*) eşitlik 21-23' teki denklemlerle hesaplanmaktadır.

$$r = \frac{\left(1 + \frac{1}{m}\right)B}{\bar{U}} \quad (21)$$

$$v_m = (m-1) \left[1 + \frac{1}{r}\right]^2 \quad (22)$$

$$v_m^* = \left[\frac{1}{v_m} + \frac{1}{\frac{(1 - \text{gamma})v_0(v_0 + 1)}{v_0 + 3}} \right]^{-1} \quad (23)$$

Eşitlik-21' de r, grup içi değer atamasına ait varyans ile standardize edilmiş gruplar arası değer atamasına ait varyans değeri olarak tanımlanmaktadır. r, varyansta kayıp gözleme bağlı olarak meydana gelen artışı göstermektedir. Eşitlik-23' te $\text{gamma} = \left(1 + \frac{1}{m}\right)B/T$ ve v_0 ise veri setinde kayıp gözlem olmadığı durumda kullanılan serbestlik derecesidir. v_m^* , v_0 küçük olduğunda, v_m için kullanılan düzeltmedir.

Çoklu değer atama yönteminde, veri setinde yer alan her bir kayıp gözlem için olası değerler tahmin edilir. Standart istatistiksel analizler kullanılarak her bir veri seti analiz edilir (Pigott, 2001).

Kayıp gözlemler için önerilen olası değerler seti veri setinin dağılımına bağlıdır. Bu durumda çok değişkenli normal dağılıma uygunluk göstermelidir. En çok olabilirlik tahmininden farklı olarak çoklu değer atama yönteminin amacı uygun istatistiklerin beklenen değerini elde etmek yerine kayıp gözlemlerin tahminini elde etmektir (Pigott, 2001).

2.5.3 Temel Bileşenler Analizi (PCA)

PCA, temel olarak çok değişkenli analizlerde boyut indirgeme ve değişkenler arasındaki bağımlılık yapısının ortadan kaldırılması için kullanılmaktadır (Arbour and Brown, 2014; Sokal and Rohlf, 1995; Stacklies and Redestig, 2016). PCA, başlı başına bir analiz olduğu gibi farklı analizler için veri hazırlama tekniği olarak da kullanılmaktadır (Tatlıdil, 2002).

PCA uygulamalarında Hotelling tarafından önerilen yöntemde $X_{p \times n}$ ham veri matrisi veya $Z_{p \times n}$ biçiminde ifade edilen standartlaştırılmış değerler matrisi de

kullanılmaktadır. Ham veri matrisinin kullanılması durumunda, temel bileşenlerin bulunmasında varyans-kovaryans matrisinden, standartlaştırılmış veri matrisinin kullanılması durumunda ise korelasyon matrisinden yararlanılmaktadır.

Temel bileşenleri aşağıdaki şekilde elde edilmektedir:

$Z_{p \times n}$ standartlaştırılmış değerler matrisi ve $T_{p \times p}$ dönüşüm matrisi olsun.

$$Y_{p \times n} = T'_{p \times p} Z_{p \times n} \quad (24)$$

Eşitlik-24 yardımıyla birbirleriyle ilişkili z_{ij} değerlerinden dönüştürme sonunda, birbirinden bağımsız ortagonal vektörleri oluşturan y_{ij} değerlerine ulaşılmaktadır. Dönüşüm sonunda elde edilen Y matrisinin ortalama vektörü ve kovaryans matrisi eşitlik-25 ve eşitlik-26' da verilmiştir.

$$E(Y) = E(T'Z) = T'E(Z) = 0 \quad (25)$$

$$Var(Y) = T'E(ZZ')T = T'RT \quad (26)$$

Eşitlik-26' da R, $p \times p$ boyutlu korelasyon matrisidir.

Dönüştürülmüş Y matrisine ait vektörlerin birbirlerinden bağımsız olabilmeleri için $Var(Y)$ matrisinin köşegen matris olması gerekmektedir. $Var(Y)$ matrisinin köşegenleştirilmesinde çok sayıda T dönüşüm matrisi kullanılabilir. Birbirlerinden farklı bu dönüşüm matrislerinden amaca en uygun olanının seçilebilmesi için uyulması gereken bazı koşullar mevcuttur. Bu koşullar aşağıda belirtilmiştir:

- Y vektörlerinin ilki y_1 'in varyansı en büyük olmalıdır.

$$Var(y_1) = \max \frac{1}{n-1} \sum_{i=1}^n (y_{1i})^2 \quad (27)$$

- y_1 vektörünün bulunmasında kullanılan t_1 vektörünün elemanlarının kareleri toplamı 1 olmalıdır.

$$t_1' t_1 = 1 \quad (28)$$

Bu kısıtlayıcılar yardımıyla z_i vektöründen, dönüşüm sonucu elde edilen y_1 vektörünün i ' inci elemanı eşitlik-29' daki gibi elde edilir.

$$y_{1i} = t_1' z_i \quad (29)$$

İlk koşul nedeniyle y_1 vektörünün varyansı ise eşitlik-30' da belirtildiği gibi elde edilmektedir.

$$\begin{aligned} \frac{1}{n-1} \sum_{i=1}^n (y_{1i})^2 &= \frac{1}{n-1} \sum_{i=1}^n (t_1' z_i)^2 = \frac{1}{n-1} \sum_{i=1}^n t_1' z_i z_i' t_1 \\ \text{Var}(y_1) &= t_1' \frac{1}{n-1} Z Z' t_1 = t_1' R t_1 \end{aligned} \quad (30)$$

y_1 vektörünün varyans değeri olan $t_1' R t_1$ ' in ikinci koşula uyularak maksimum değere ulaşması gerekmektedir. Bu amaçla eşitlik-31' deki fonksiyon çözülmelidir.

$$\varphi_1 = t_1' R t_1 - \lambda_1 (t_1' t_1 - 1) \quad (31)$$

Eşitlik-31' deki fonksiyonun t_1 ' e göre türevi alınıp sıfıra eşitlendiğinde eşitlik-32 elde edilir.

$$\frac{\partial \varphi_1}{\partial t_1} = 2R t_1 - 2\lambda_1 t_1 = 0 \Rightarrow (R - \lambda_1 I) t_1 = 0 \quad (32)$$

Eşitlik-32' de λ_1 değeri R matrisinin özdeğeri, t_1 vektörü de R matrisinin özvektörü olarak adlandırılır. Özdeğerleri elde etmek için eşitlik-33' teki denklemin açılımından elde edilen p' inci dereceden polinom denklemden p tane λ değeri bulunur.

$$|R - \lambda I| = 0 \quad (33)$$

R matrisi pozitif tanımlı ve simetrik olduğu için elde edilecek değerlerin tümü gerçek değerler olacaktır. Eşitlik-33' ten elde edilen p tane özdeğer kullanılarak her birine karşılık gelen p tane özvektör elde edilir.

Eşitlik-33' teki denklemin kullanımı ile elde edilen λ_j ' lerden biri λ_1 ve ilgili vektörü de t_1 olsun. Bu denklem soldan t_1' ile çarpıldığında eşitlik-34 elde edilir.

$$t_1' R t_1 - \lambda_1 t_1' t_1 = 0 \quad (34)$$

Eşitlik-28 dikkate alındığında da ($t_1' t_1 = 1$), $t_1' R t_1 = \lambda_1$ elde edilir. Sonuç olarak y_1 değişkeninin varyansı λ_1 olarak bulunur.

$$Var(y_1) = Var(\sqrt{\lambda_1 t_1}) = E(\sqrt{\lambda_1 t_1})(\sqrt{\lambda_1 t_1})' = \lambda_1 t_1' t_1 = \lambda_1 \quad (35)$$

PCA' da y_1 ' in varyansının en büyük olması istendiğinden, λ_1 değeri λ_j değerleri arasında en büyük değerli olarak seçilir. Seçilen λ_1 değerinin kullanımı ile elde edilen t_1 vektörü birinci özvektör, birinci özdeğer λ_1 ve birinci özvektör t_1 olmak üzere; elde edilen birinci temel bileşen eşitlik-36' da verilmiştir.

$$y_1 = t_1' Z \quad (36)$$

İkinci temel bileşen y_2 bulunurken, y_1 vektörünün bulunmasında kullanılan iki koşulla birlikte üçüncü bir koşul da göz önüne alınır:

- y_2 vektörünün varyansı y_1 ' den sonra en büyük olmalıdır.

- t_2 vektörü birim normal vektör olmalıdır ($t_2' t_2 = 1$),
- y_1 ve y_2 vektörleri birbirine dik olmalıdır ($t_2' t_1 = 0$)

Bu koşullar altında benzer işlemler uygulandığında eşitlik-37 elde edilir. Eşitlik-37' de λ_2 , R matrisinin ikinci özdeğeri ve t_2 , R matrisinin ikinci özvektörüdür. Elde edilen y_2 vektörüne ikinci temel bileşen, $\sqrt{\lambda_2} t_2$ değerlerine ise ikinci asıl temel bileşen katsayıları adı verilir. İkinci temel bileşen, varyansı, y_1 ' in varyansından sonra en büyük ve y_1 vektörüne diktir.

$$R t_2 - \lambda_2 t_2 = 0 \text{ ve } (R - \lambda_2 I) t_2 = 0 \quad (37)$$

Temel bileşenlerin elde edilmesi için benzer işlemlere devam edildiğinde, $j=1, \dots, p$ için tüm λ_j, t_j ve y_j değerleri elde edilir. Bu durumda λ_p en küçük değere sahip özdeğer ve y_p ise en küçük varyanslı temel bileşendir.

PCA, genellikle analizlerde ilk aşamada kullanılır. Kovaryans matrisi özdeğer ayrışmasına bağlı olduğu için standart analiz yaklaşımı, kayıp veri tahmini için uygun değildir (Stacklies ve ark., 2007; Stacklies and Redestig, 2016). Bu nedenle kayıp veri tahmininde kullanılmak amacıyla PCA modifiye edilmiştir. Tez çalışmasında kullanılan modifiye edilmiş yöntemler, BPCA, PPCA, NLPCA ve INIPALS' dir.

2.5.3.1. Olasılıksal Temel Bileşenler Analizi (Probabilistic PCA, PPCA)

EM algoritmasını olasıksal bir model ile birleştiren bir yaklaşım olan PPCA, Tipping ve ark. (1999) tarafından önerilmiştir. EM algoritması, gürültü (noise) gibi gizil değişkenlerin normal dağılıma uygun olduğu varsayımına dayalıdır. Standart PCA' da veriler eğitim setine uzak fakat modeli iyi tahmin eden temel bileşenlerle tanımlanmış alt uzaya yakındır. PPCA, veriler için olabilirlik modeli gibi bir olasılık modeli tanımlamaktadır (Stacklies ve ark., 2007; Stacklies and Redestig, 2016).

q temel eksenler, $j \in \{1, \dots, q\}$ olmak üzere w_j ortonormal eksenleri göstermektedir. $W = (w_1, \dots, w_q)$, $C = WW^T + \sigma^2 I$, I birim matris, t_n d boyutlu veri vektörünü, $n \in \{1, \dots, N\}$, x_n kayıp veriyi ve “tam veri” kavramı gizil değişkenleri de içeren tüm veri olmak üzere tam veri setinin log olabilirliği eşitlik-38 deki gibidir.

$$\mathcal{L}_c = \sum_{n=1}^N \ln\{p(t_n, x_n)\} \quad (38)$$

PPCA için olasılık modeli ise eşitlik-39’ da belirtilmiştir.

$$p(t_n, x_n) = (2\pi\sigma^2)^{-d/2} \exp\left(-\frac{\|t_n - Wx_n - \mu\|^2}{2\sigma^2}\right) (2\pi)^{-q/2} \exp\left(-\frac{\|x_n\|^2}{2}\right) \quad (39)$$

M $q \times q$ boyutlu varyans-kovaryans matrisi olmak üzere E adımında eşitlik-40 ve eşitlik-41’ deki model parametrelerinden bağımsız olarak \mathcal{L}_c ’ nin beklenen değeri eşitlik-42’ deki gibi hesaplanır.

$$\langle x_n \rangle = M^{-1} W^T (t_n - \mu) \quad (40)$$

$$\langle x_n x_n^T \rangle = \sigma^2 M^{-1} + \langle x_n \rangle \langle x_n \rangle^T \quad (41)$$

$$\begin{aligned} \langle L_c \rangle = - \sum_{n=1}^N \left\{ \frac{d}{2} \ln(\sigma^2) + \frac{1}{2} \text{tr}(\langle x_n x_n^T \rangle) + \frac{1}{2\sigma^2} (t_n - \mu)^T (t_n - \mu) - \frac{1}{\sigma^2} \langle x_n \rangle^T W^T (t_n \right. \\ \left. - \mu) + \frac{1}{2\sigma^2} \text{tr}(W^T W \langle x_n x_n^T \rangle) \right\} \quad (42) \end{aligned}$$

M adımında ise \mathcal{L}_c maksimize edilir.

$$\bar{W} = \left\{ \sum_{n=1}^N (t_n - \mu) \langle x_n \rangle^T \right\} \left(\sum_{n=1}^N \langle x_n x_n^T \rangle \right)^{-1} \quad (43)$$

$$\tilde{\sigma}^2 = \frac{1}{Nd} \sum_{n=1}^N \left\{ \|t_n - \mu\|^2 - 2 \langle x_n \rangle^T \bar{W}^T (t_n - \mu) + \text{tr}(\langle x_n x_n^T \rangle \bar{W}^T \bar{W}) \right\} \quad (44)$$

Olabilirliği maksimize etmek için eşitlik-41 ve eşitlik-44' ten parametre tahminleri elde edildikten sonra koşullu dağılımlara ait istatistikler eşitlik-40 ve eşitlik-41' den elde edilir. Eşitlik-(40-44) algoritma yakınsayıncaya kadar tekrarlanır (Tipping and Bishop, 1999).

2.5.3.2 Bayes Yaklaşımlı Temel Bileşenler Analizi (Bayesian PCA, BPCA)

BPCA, tahmin edilen değerın olabilirliğini hesaplamak için Bayes tahminini EM algoritması ile birleştirir. BPCA, özellikle kayıp değer tahmini için geliştirilmiştir ve değişken Bayes yapısına dayalıdır. BPCA yöntemi, PCA ile kıyaslandığında özdeğerler, skorlar ve temel bileşenler farklı ölçeklenmektedir.

Bir bileşene ait Öklid normu bu bileşenden gözlenen gürültünün varyansına göre küçük olduğunda, bileşen sıfıra doğru yaklaşacaktır. Veri setindeki eksiklik ve gürültü nedeniyle gerçek bileşenleri doğru belirlemek için gerekli olan bilgi eksikliği nedeniyle küçük örneklem büyüklüğüne sahip veri setlerinde gerçek ve tahmin edilen özdeğerler arasındaki fark daha fazla olabilir. BPCA' nın PCA' dan farklı başka bir yönü de değişken Bayes yapısında temel bileşenler arasında ortogonalite koşulu olmamasıdır.

BPCA, temel bileşenler regresyonu, Bayes tahmini ve EM algoritması olmak üzere üç farklı yöntemin bir araya gelmesinden oluşmaktadır.

Temel bileşenler regresyonunda y vektörünün kayıp verileri içeren kısmı y^{miss} PCA' yı kullanılarak, kayıp değer içermeyen kısmı y^{obs} , den tahmin edilir. Her bir temel eksen w_1 olmak üzere w_1^{obs} gözlenen değerleri, w_1^{miss} ise kayıp değerleri gösterebilir. Benzer şekilde W^{obs} ve W^{miss} sütun vektörleri $w_1^{obs}, \dots, w_K^{obs}$ ve $w_1^{miss}, \dots, w_K^{miss}$ olmak üzere $W = (W^{obs}, W^{miss})$ olarak ifade edilmektedir.

y vektörü için faktör skorları $x = (x_1, \dots, x_K)$ eşitlik-45' te verilen hata değeri minimize edilerek elde edilir.

$$err = \|y^{obs} - W^{obs}x\|^2 \quad (45)$$

En küçük kareler yöntemi kullanılarak elde edilen sonuç eşitlik-46' te verilmiştir.

$$x = (W^{obsT}W^{obs})^{-1}W^{obsT}y^{obs} \quad (46)$$

Eşitlik-47 kullanılarak y^{miss} tahmin edilir.

$$y^{miss} = W^{miss}x \quad (47)$$

Bayes tahmini adımımda Tipping (1999) tarafından önerilen PPCA kullanılarak Bayesian tahmin yöntemi önerilmiştir (Oba ve ark., 2003). Bayesian tahmini, X ve θ ' nin Bayes teoremine göre sonsal dağılımını içermektedir.

$$p(\theta, X/Y) \propto p(Y, X/\theta) p(\theta) \quad (48)$$

Eşitlik-48' de θ tahmini için önsel tahmini gösteren $p(\theta)$, önsel dağılım olarak adlandırılmaktadır. Önsel dağılım tahminden önce belirlenmelidir. Eğer θ_{true} parametresi biliniyorsa temel bileşenler regresyonu ile tahmin edilen kayıp değerlere ait önsel değerler eşitlik-49' da verilmiştir.

$$q(Y^{miss}) = p(Y^{miss}/Y^{obs}, \theta_{true}) \quad (49)$$

Eşitlik-49' da $p(Y^{miss}/Y^{obs}, \theta_{true})$ gözlenen değerlere göre olabilirlik olasılığını marjinalleştirerek elde edilir. Eğer gerçek parametre yerine sonsal parametre $q(\theta)$ mevcutsa, kayıp değerlerin sonsal değerleri eşitlik-50' deki gibi elde edilir.

$$q(Y^{miss}) = \int d\theta q(\theta) p(Y^{miss}/Y^{obs}, \theta) \quad (50)$$

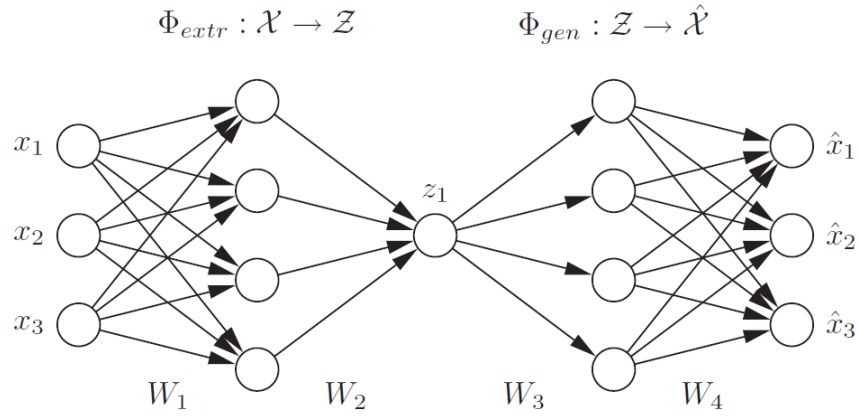
2.5.3.3. Ters Doğrusal Olmayan Temel Bileşenler Analizi (Inverse Non-Linear PCA, NLPCA)

NLPCA, özellikle bağımlı değişkenin doğrusal olmadığı deneylerdeki veri setleri için uygundur (Stacklies ve ark., 2007; Stacklies and Redestig, 2016). NLPCA, standart doğrusal PCA' nın doğrusal olmayan genelleştirmesi olarak ele alınmaktadır (Scholz ve ark., 2005).

NLPCA ilk olarak öğrenmeye dayalı topoloji (auto-associative topology) ile çok katmanlı algılayıcıya (multi-layer perceptron) dayalı olarak Kramer tarafından önerilmiştir (Kramer, 1991).

NLPCA, öğrenmeye dayalı sinir ağı (auto-associative neural network) ve “dar-boğaz (bottle-neck)” gibi davranan katman bileşimlerine dayalıdır. Katmanlardan biri gizli doğrusal olmayan katman diğeri ise yeniden oluşturulmuş veriye dayalı olan sonuç katmanıdır.

Bu ağda yükler “gizli” olarak ele alınabilir. Eğitim veri setindeki kayıp değerler hata olarak hesaplanırken geriye yayılım boyunca göz ardı edilir. Dolayısıyla NLPCA geleneksel PCA gibi kayıp veri tahmininde kullanılabilir. NLPCA' daki tek fark yüklerin bir sinir ağı ile ifade edilmesidir (Şekil 7).



Şekil 7. Standart öğrenmeye dayalı sinir ağı (Scholz ve ark., 2005)

Öğrenmeye dayalı ağ, özdeşlik dönüşümü (identity mapping) yapmaktadır. Hataların karesini $\|x - \hat{x}\|^2$ minimize ederek x ' in tahmin değeri \hat{x} ' ya eşit olması gerekmektedir.

Bu yöntemde birleştirici ağ iki kısma ayrılmaktadır. $\Phi_{extr}: Z \rightarrow \hat{\chi}$ olmak üzere, ilk kısım çıkarım (extraction) fonksiyonudur $\Phi_{extr}: \chi \rightarrow Z$. İkinci kısım ise ters fonksiyonu göstermektedir.

2.5.3.4 İteratif Kısmi En Küçük Kareler İle Doğrusal Olmayan Tahmine Dayalı Temel Bileşenler Analizi (Non-Linear Estimation by Iterative Partial Least Squares PCA, INIPALS)

INIPALS, Wold ve ark. (1966) tarafından önerilen ve basitçe kayıp değerleri dikkate almadan PCA uygulayan kısmi en küçük kareler regresyonuna dayalı bir algoritmadır. INIPALS yöntemi ile hem PCA uygulanabilirken hem de kayıp veri tahmini yapılabilmektedir (Preda ve ark., 2010).

Bu yöntem %5' ten daha az kayıp veri içeren veri setlerinde daha iyi sonuçlar vermektedir (Stacklies and Redestig, 2016).

$p \geq 1$ ve $\forall i \in 1, \dots, p$ olmak üzere $X = (X_1, X_2, \dots, X_p)'$ p boyutlu rasgele vektör ve $E(X_i) = 0$ olsun. X vektörüne PCA uygulandığında elde edilen sonuç eşitlik-51' de belirtilmiştir.

$$X = \sum_{h=1}^q \xi_h u_h \quad (51)$$

Eğer eşitlik-51' de sadece ilk r bileşen ($r < q$) kullanılırsa $i \in 1, \dots, p$ olmak üzere eşitlik-52 elde edilir.

$$\hat{X}_i^{(r)} = \sum_{h=1}^r \xi_h u_h(i) \quad (52)$$

INIPALS algoritmasında, $h = 1, \dots, q$ olmak üzere $u_h(i)$ ξ_h bileşenine ait X_i değişkeninin lineer regresyonundaki eğim katsayısını göstermektedir. n örneklem büyüklüğü, $i = 1, \dots, n$, $j = 1, \dots, p$ ve X $n \times p$ boyutlu rasgele vektör olmak üzere INIPALS algoritmasının adımları aşağıda belirtilmiştir (Preda ve ark., 2010):

Adım-1. $X_0 = X$

Adım-2. $h = 1, 2, \dots, q$ için

Adım-2.1 $\xi_h = X_{h-1}$ (X_{h-1} vektörünün ilk sütunu)

Adım-2.2 u_h yakınsayınca kadar devam edilir.

Adım-2.2.1 $i = 1, 2, \dots, p$ için

$$u_h(i) = \frac{\sum_{j:x(j,i)\xi_h(j)} x_{h-1}(j,i)\xi_h(j)}{\sum_{j:x(j,i)\xi_h(j)} \xi_h^2(j)} \quad (53)$$

Adım-2.2.2 u_h normalleştirilir.

Adım-2.2.3 $i = 1, 2, \dots, n$ için

$$\xi_h(i) = \frac{\sum_{j:x(i,j)} x_{h-1}(i,j)u_h(j)}{\sum_{j:x(i,j)} u_h^2(j)} \quad (54)$$

Adım-2.3 $X_h = X_{h-1} - \xi_h u_h'$

Kayıp veri olmaması durumunda INIPALS algoritması Jacobi algoritmasına eşdeğerdir. Kayıp veri olması durumunda, $\{\hat{\xi}_h\}_{h=1,\dots,q}$ ve $\{\hat{u}_h\}_{h=1,\dots,q}$ Nipals algoritması ile elde edilen tahminler olmak üzere $x(i,j)$ eşitlik-55' te olduğu gibi tahmin edilmektedir.

$$\hat{x}(i,j) = \sum_{h=1}^q \hat{\xi}_h(i) \hat{u}_h(j) \quad (55)$$

Eşitlik-54' de kayıp veri olması durumunda INIPALS algoritması tanımlanmaktadır.

2.5.3.5 Temel Bileşenler Analizi ile Kayıp Veri Tahmini

Farklı temel bileşenler analiz yöntemleri uygulanarak kayıp verinin tahmin edilmesinde sadece anlamlı bileşenleri ele alındığında, bileşenler eşitlik-56' da olduğu gibi ifade edilir.

$$X = 1 \times \bar{x}^T + TP^T + V \quad (56)$$

Eşitlik-56' da X gözlemleri, $T=t_1, t_2, \dots, t_k$ gizil değişkenleri ya da skorları, $P=p_1, p_2, \dots, p_k$ kovaryans matrisinin anlamlı özvektörlerini içeren dönüşüm matrisini ve V artıkları göstermektedir.

Bu yöntemde kayıp değerler, skorların orijinal uzaydaki değerlerinin tahmin edilmesi ile bulunmaktadır.

$$\hat{X} = 1 \times \bar{x}^T + TP^T + V \quad (57)$$

Elde edilen tahmin, önemli bilginin ilk k bileşenle açıklanabildiğini belirten, artıkların (V) küçük olduğu durumda daha iyi sonuç vermektedir (Stacklies ve ark., 2007). Ayrıca kayıp değerlerin tahmininde, veri setindeki korelasyon yapısı da dikkate alınmaktadır (Arbour and Brown, 2014).

2.6. F Yaklaşımı

Tez çalışmasında yaygın olarak kullanılan kayıp veri tahmin yöntemlerine alternatif olarak Bookstein koordinatları, çember denklemi ve F istatistiği kullanılarak kayıp landmarkların tahmin edilmesine yönelik F yaklaşımı önerilmiştir. F yaklaşımı önerisinde, Min(F) ve Max(F) ölçütleri değerlendirilmiştir.

(x_j, y_j) landmark koordinatları, $j = 1, 2, \dots, k$ ve k landmark sayısı ($k \geq 3$) olmak üzere Bookstein koordinatları aşağıdaki şekilde hesaplanmaktadır.

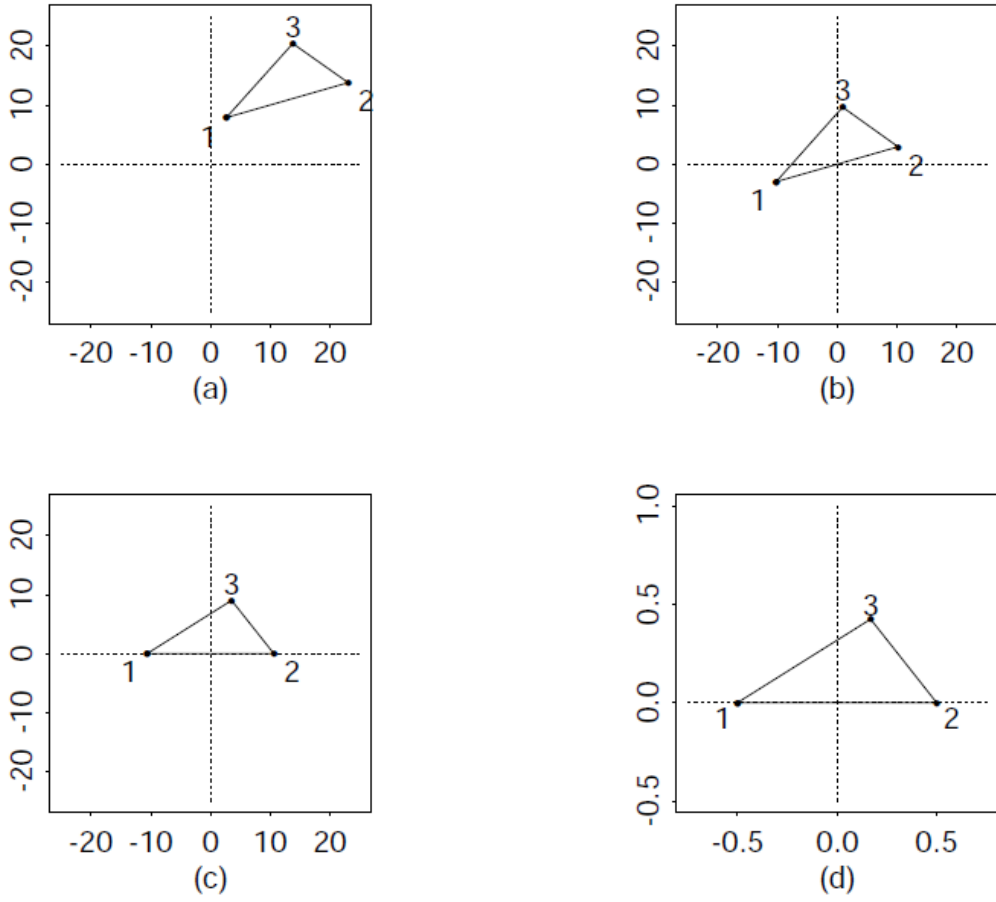
$j = 3, \dots, k$ olmak üzere bir nesneye ait koordinatlar öteleme, ölçekleme ve döndürme sonrasında referans alınan iki landmarkın koordinatları $(\frac{1}{2}, 0)$ ve $(-\frac{1}{2}, 0)$ konumlarına taşınarak elde edilen Bookstein koordinatları eşitlik-58' de belirtilmiştir.

$$u_j^B = \frac{\{(x_2 - x_1)(x_j - x_1) + (y_2 - y_1)(y_j - y_1)\}}{D_{12}^2} - \frac{1}{2}$$

$$v_j^B = \frac{\{(x_2 - x_1)(x_j - x_1) + (y_2 - y_1)(y_j - y_1)\}}{D_{12}^2}$$

(58)

Eşitlik-58' de $D_{12}^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2 > 0$ ve $-\infty < u_j^B, v_j^B < \infty$ şeklindedir (Dryden ve Mardia, 1998; Ercan ve ark., 2015). Bookstein koordinatlarının geometrik gösterimi Şekil 8' de verilmiştir.



Şekil 8. Bookstein koordinatlarının geometrik gösterimi. a) Ham veri b) Öteleme c) Döndürme d) Ölçeklendirme (Dryden and Mardia, 1998)

Tez çalışmasında kayıp landmark tahmini için önerilen F yaklaşımı algoritması aşağıda verilmiştir.

Adım-1. n birimlik k landmarklı veri setinde, ilgili birimdeki kayıp olan m' inci landmark belirlenir.

Adım-2. İlgili birimdeki kayıp olan m' inci landmarkı tahmin etmede kullanılacak olan iki referans landmark olarak i' inci ve j' inci landmarklar belirlenir.

Adım-3. i' inci ve j' inci landmarklar referans alınarak, veri seti Bookstein koordinatlarına dönüştürülür.

Adım-4. m' inci landmarkın kayıp olmadığı birimlerde, Öklid uzaklıkları kullanarak i' inci ve j' inci landmarklar ile m' inci landmark arası uzaklıklar hesaplanır.

Adım-5. Her bir birim için hesaplanan landmark arası uzaklıklara göre i-m (d_2 uzaklığı) ve j-m (d_3 uzaklığı) uzaklıklarının ortalama ve standart hatası hesaplanır.

Adım-6. d_2 ve d_3 için güven aralıkları hesaplanır ($d_2 \pm 1.96SH$ ve $d_3 \pm 1.96SH$)

Adım-7. Tahmin için her iki güven aralığının alt sınır değerleri iterasyon başlangıcı ve üst sınır değeri de iterasyon sonu olarak kabul edilir.

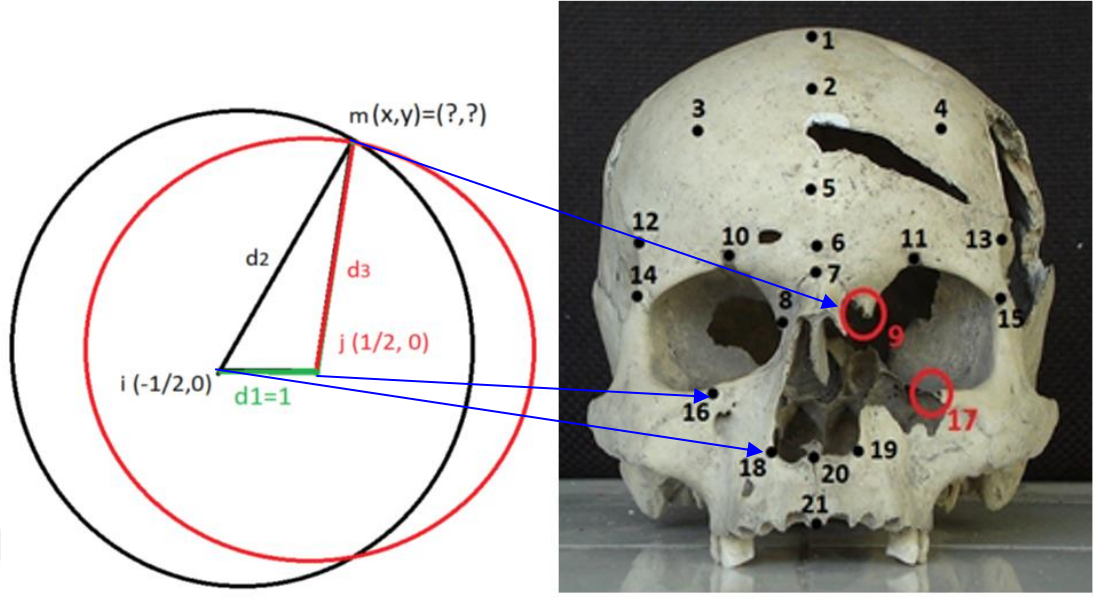
Adım-8. İterasyon katsayısı belirlenir.

Adım-9. Her bir iterasyonda ilgili birimde kayıp olan m' inci landmark için eşitlik-59'daki iki noktası bilinen çember denklemleri kullanılarak koordinatlar tahmin edilir (Şekil 9).

$$\left(x - \left(-\frac{1}{2}\right)\right)^2 + (y - 0)^2 = d_2^2$$

$$\left(x - \frac{1}{2}\right)^2 + (y - 0)^2 = d_3^2$$

(59)



Şekil 9. İki noktası bilinen çember denklemini kullanılarak koordinatlar tahmin edilmesi

Adım-10. İterasyonda, tahmin edilen m ' inci landmarkına ait koordinatlar olan x_m ve y_m için s_b^2 ve s_w^2 istatistikleri ve F istatistiği hesaplanır.

Adım-11. İterasyon d_2 ve d_3 uzaklıkları üst sınır değerine ulaşıncaya kadar tekrarlanır.

Adım-12. Tüm iterasyonlar dikkate alınarak Min(F) ve Max(F) istatistiği bulunur.

Adım-13. Min(F) ve Max(F) istatistiklerine göre ilgili x ve y koordinatları kayıp landmark koordinatları olarak kabul edilir.

Adım-14. Tüm landmarkların koordinatları Bookstein koordinatlarından orijinal koordinatlara geri dönüştürülür.

3. GEREÇ VE YÖNTEM

Tez çalışmasında EM algoritması, çoklu regresyon atama yöntemi, PCA uyarlanarak elde edilen BPCA, PPCA, NLPCA, INIPALS ve önerilen F yaklaşımı algoritmasının kayıp landmark tahminindeki performansları karşılaştırılmıştır. Simülasyon çalışması HP Z8000 Windows 7, 64-bit işletim sistemine sahip 16 gb RAM kapasiteli Workstation' da R 3.4.0 programında yapılmıştır (R Core Team, 2019). Simülasyonların çalışma süresi 1 hafta ile 1 ay arasında değişmektedir.

Veri türetiminde ve düzenlenmesinde mvtnorm, shapes, readxl, dplyr ve Matrix R paketleri kullanılmıştır (Bates ve Maechler, 2017; Dryden, 2017; Genz ve ark., 2017; Wickham ve Bryan, 2017; Wickham ve ark., 2019).

EM algoritmasını kullanarak kayıp veri tahmin etmek amacıyla "Amelia" R paketi kullanılmıştır. "Amelia" paketi ile bootstrap yöntemi ve EM algoritması kullanılarak kayıp değerler tahmin edilmektedir (Honaker ve ark., 2011).

Çoklu regresyon atama yöntemi ile kayıp değer tahmininde eksik veriler için Gibbs örneklemesini kullanarak çoklu değer ataması yapan "mice" R paketi kullanılmıştır. "mice" fonksiyonu ile regresyon ataması yapmak için Bayesian lineer regresyon analizi uygulama olanağı sağlayan "norm" seçeneği kullanılmıştır. Uygulanan algorithmada, veri setindeki diğer veriler için olası yapay değerler türeterek eksik veri içeren hedef sütuna değer atanarak kayıp veriler tahmin edilmektedir. (Buuren ve Groothuis-Oudshoorn, 2011).

PCA modifiye edilerek kayıp veri tahmini yapılmasını sağlayan yöntemler için "pcaMethods" R paketi kullanılmıştır (Stacklies ve Redestig, 2007).

3.1. Simülasyon Senaryosu

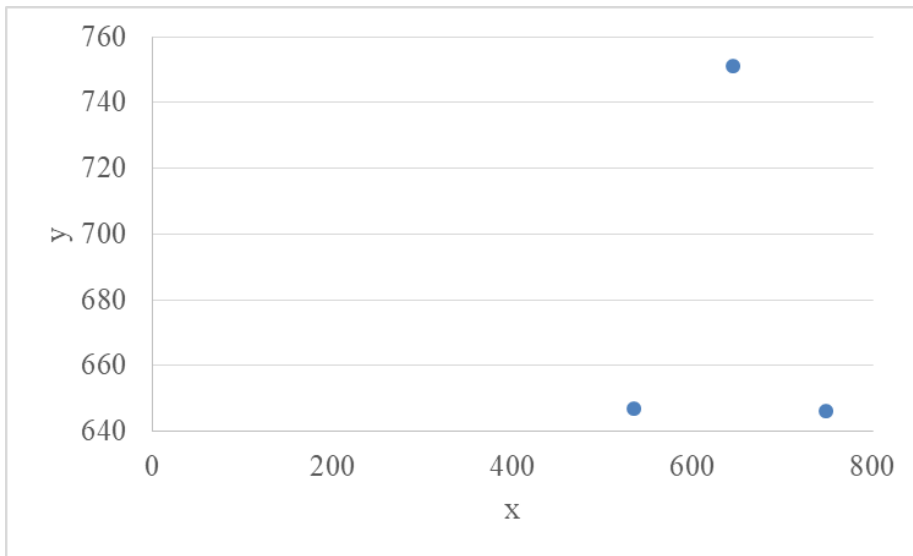
Simülasyon çalışmasında Ozdemir ve ark. (2010) yaptığı çalışmadaki landmarklar referans alınmıştır. Simülasyon senaryosunda landmark sayıları 3, 6, 9 ve 12 olarak alınmıştır. Örneklem büyüklükleri ise sırasıyla 30, 50 ve 100 olarak alınmıştır. Çok değişkenli normal dağılımdan isotropik ve isotropik olmayan modellere dayalı olarak veri türetilerek 10 farklı simülasyon senaryosu dikkate alınmıştır.

Veri türetiminde Ozdemir ve ark. (2010) yaptığı çalışmadaki landmarklara ait ortalama vektörleri kullanılmıştır. Üç landmarklı durumda kraniumda 1' inci, 3' üncü, 4' üncü landmarklar, 6 landmarklı durumda 1' inci, 3' üncü, 4' üncü 6' ıncı, 12' inci, 13' üncü landmarklar, 9 landmarklı durumda 1' inci, 3' üncü, 4' üncü 6' ıncı, 12' inci, 13' üncü 16' ıncı, 17' inci, 20' inci landmarklar, 12 landmarklı durumda ise 1' inci, 3' üncü, 4' üncü 6' ıncı, 8' inci, 9' uncu, 12' inci, 13' üncü 16' ıncı, 17' inci, 20' inci ve 21' inci landmarklar dikkate alınmıştır (Şekil 2).

- Landmark sayısı 3 olması durumunda kullanılan ortalama vektörleri:

$$\bar{X} = [644.18 \quad 534.73 \quad 747.82]$$

$$\bar{Y} = [751 \quad 646.82 \quad 646.18]$$

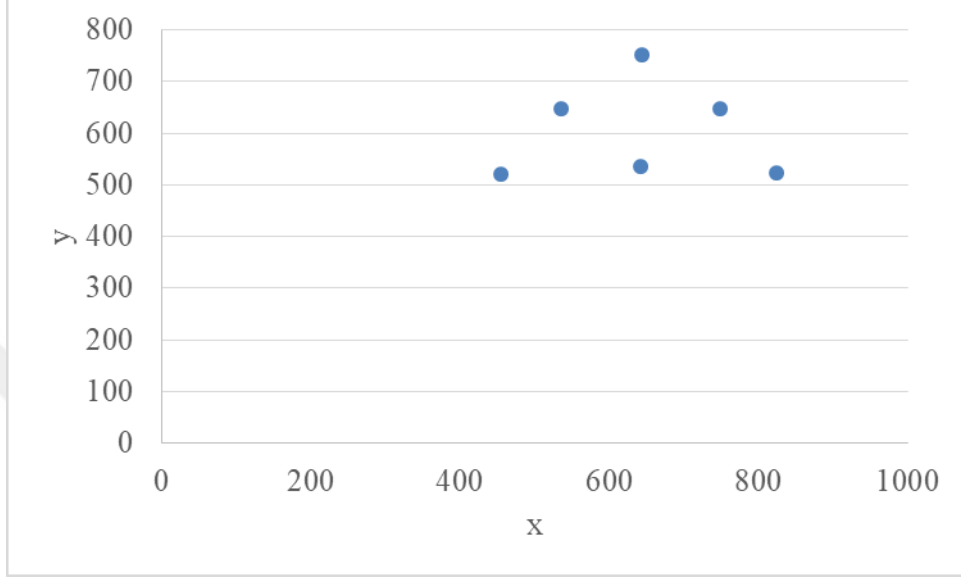


Grafik 1. 3 landmarklı durumda landmark koordinatlarının gösterimi

- Landmark sayısı 6 olması durumunda kullanılan ortalama vektörleri:

$$\bar{X} = [644.18 \quad 534.72 \quad 747.82 \quad 454.54 \quad 641.90 \quad 824.45]$$

$$\bar{Y} = [751 \quad 646.82 \quad 646.18 \quad 521.72 \quad 534.72 \quad 523.90]$$

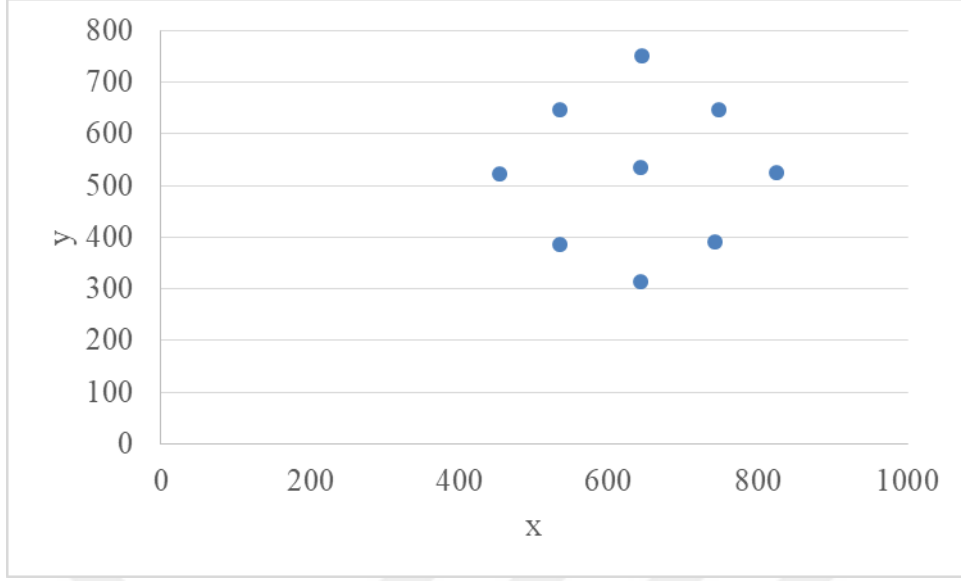


Grafik 2. 6 landmarklı durumda landmark koordinatlarının gösterimi

- Landmark sayısı 9 (Şekil 2 de landmark numaraları:) olması durumunda kullanılan ortalama vektörleri:

$$\bar{X} = [644.18 \quad 534.72 \quad 747.82 \quad 454.54 \quad 641.90 \quad 824.45 \quad 533.90 \quad 743 \quad 643.18]$$

$$\bar{Y} = [751 \quad 646.82 \quad 646.18 \quad 521.72 \quad 534.72 \quad 523.90 \quad 385.81 \quad 390.54 \quad 312.90]$$

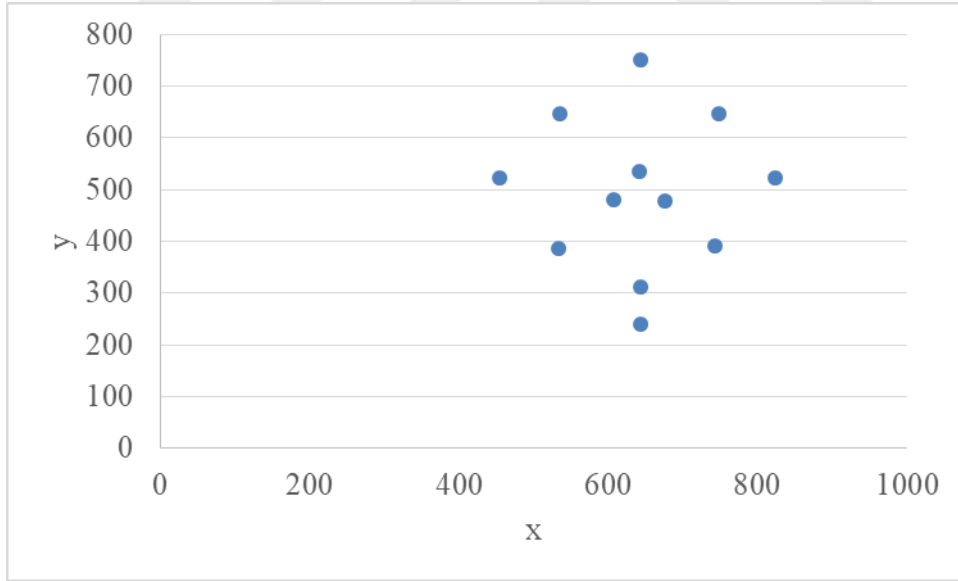


Grafik 3. 9 landmarklı durumda landmark koordinatlarının gösterimi

Landmark sayısı 12 olması durumunda:

$$\bar{X} = [644.18 \ 534.72 \ 747.82 \ 454.54 \ 641.90 \ 824.45 \ 533.90 \ 743 \ 643.18 \ 607.81 \ 675.18 \ 643.45]$$

$$\bar{Y} = [751 \ 646.82 \ 646.18 \ 521.72 \ 534.72 \ 523.90 \ 385.81 \ 390.54 \ 312.90 \ 480 \ 478.81 \ 238.54]$$



Grafik 4. 12 landmarklı durumda landmark koordinatlarının gösterimi

Tez çalışmasında dikkate alınan simülasyon senaryoları aşağıda belirtilmiştir.

- Senaryo 1' de Ozdemir ve ark. (2010) tarafından yapılan çalışmadan alınan varyans-kovaryans matrisi gözden geçirilerek kullanılmıştır.

- Senaryo 2’ de isotropik model kullanılmış ve varyans-kovaryans matrisi birim matris (I) olarak alınmıştır.
- Senaryo 3’ te isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 5$, kovaryans değerleri 0 olarak alınmıştır.
- Senaryo 4’ te isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 0.5$, kovaryans değerleri 0 olarak alınmıştır.
- Senaryo 5’ te isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 1$, kovaryans değerleri 2 olarak alınmıştır.
- Senaryo 6’ da tarafımızdan oluşturulan varyans-kovaryans matrisi kullanılmıştır.
- Senaryo 7’ de isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 0.1$, kovaryans değerleri 0 olarak alınmıştır.
- Senaryo 8’ de isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 0.05$, kovaryans değerleri 0 olarak alınmıştır.
- Senaryo 9’ da ise isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 0.01$, kovaryans değerleri 0 olarak alınmıştır.
- Senaryo 10’ da ise isotropik model kullanılmış ve varyans değerleri $\sigma^2 = 0.001$, kovaryans değerleri 0 olarak alınmıştır.

Tüm simülasyon senaryolarında varyans-kovaryans matrisi pozitif tanımlı olacak şekilde veri türetilmiştir ve aşağıda belirtilmiştir.

- Senaryo 1-a: Senaryo 1 ve landmark sayısı 3 için kullanılan varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 6.51 & 610.27 & 896.93 & 26.15 & 26.16 & 26.16 \\ 610.27 & 5.41 & 687.11 & 21.81 & 13.95 & 14.31 \\ 896.93 & 687.11 & 7.88 & 30.08 & 33.49 & 33.78 \\ 26.15 & 21.81 & 30.08 & 0.51 & 6.55 & 6.74 \\ 26.16 & 13.95 & 33.49 & 6.55 & 0.56 & 11.68 \\ 26.16 & 14.31 & 33.78 & 6.74 & 11.68 & 0.60 \end{bmatrix}$$

- Senaryo 1-b: Senaryo 1 ve landmark sayısı 6 için kullanılan varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 6.51 & 610.27 & 896.93 & 787.10 & 706.48 & 681.94 & 26.14 & 26.15 & 26.15 & 27.96 & 30 & 29.49 \\ 610.27 & 5.41 & 687.11 & 700.68 & 587.11 & 537.56 & 21.81 & 13.95 & 14.31 & 26.11 & 24.46 & 21.10 \\ 896.93 & 687.11 & 7.88 & 916.24 & 853.78 & 845.88 & 30.08 & 40.40 & 33.77 & 29.67 & 34.50 & 36.28 \\ 787.10 & 700.68 & 916.24 & 7.12 & 752.36 & 70.29 & 0.50 & 26.55 & 6.73 & 31.44 & 32.12 & 28.98 \\ 706.48 & 587.11 & 853.78 & 752.36 & 6.21 & 649.50 & 6.55 & 30.76 & 11.68 & 27.37 & 29.56 & 29.07 \\ 681.94 & 537.56 & 845.88 & 702.93 & 649.50 & 6.20 & 6.73 & 32.24 & 0.59 & 24.12 & 27.85 & 28.82 \\ 26.14 & 21.81 & 30.08 & 0.50 & 6.55 & 6.73 & 97.18 & 600.52 & 30.52 & 625.10 & 770.58 & 671.85 \\ 26.15 & 13.95 & 40.40 & 26.55 & 30.76 & 32.24 & 600.52 & 0.59 & 144.75 & 8.28 & 11.13 & 10.97 \\ 26.15 & 14.31 & 33.77 & 6.73 & 11.68 & 0.59 & 30.52 & 144.75 & 35.56 & 243.24 & 277.94 & 229.36 \\ 27.96 & 26.11 & 29.67 & 31.44 & 27.37 & 24.12 & 625.10 & 8.28 & 243.24 & 0.73 & 16.71 & 13.13 \\ 30 & 24.46 & 34.50 & 32.12 & 29.56 & 27.85 & 770.58 & 11.13 & 277.94 & 16.71 & 0.86 & 14.89 \\ 29.49 & 21.10 & 36.28 & 28.98 & 29.07 & 28.82 & 671.85 & 10.97 & 229.36 & 13.13 & 14.89 & 0.74 \end{bmatrix}$$

- Senaryo 1-c: Senaryo 1 ve landmark sayısı 9 için kullanılan varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 6.51 & 610.27 & 896.93 & 787.10 & 706.48 & 681.94 & 735.54 & 585.82 & 705.69 & 26.14 & 26.15 & 26.15 & 27.96 & 30 & 29.49 & 45.42 & 39.39 & 81.21 \\ 610.27 & 5.41 & 687.11 & 700.68 & 587.11 & 537.56 & 592.92 & 499.10 & 581.47 & 21.81 & 13.95 & 14.30 & 26.11 & 24.46 & 21.10 & 39.08 & 30.41 & 80.37 \\ 896.93 & 687.11 & 7.88 & 916.24 & 853.78 & 845.88 & 919.54 & 712.72 & 868.95 & 30.08 & 40.40 & 33.77 & 29.67 & 34.50 & 36.28 & 51.11 & 46.96 & 85.74 \\ 787.10 & 700.68 & 916.24 & 7.12 & 752.36 & 702.93 & 778.02 & 630.44 & 749.03 & 0.50 & 26.55 & 6.73 & 31.44 & 32.12 & 28.98 & 51.16 & 42.04 & 96.72 \\ 706.48 & 587.11 & 853.78 & 752.36 & 6.21 & 649.50 & 705.94 & 564.81 & 678.28 & 6.55 & 30.76 & 11.68 & 27.37 & 29.56 & 29.07 & 44.97 & 39.05 & 79.30 \\ 681.94 & 537.56 & 845.88 & 702.93 & 649.50 & 6.20 & 693.72 & 548.35 & 662.89 & 6.73 & 32.24 & 0.59 & 24.12 & 27.85 & 28.82 & 40.13 & 36.23 & 66.80 \\ 735.54 & 592.92 & 919.54 & 778.02 & 705.94 & 693.72 & 6.72 & 600.48 & 727.64 & 97.18 & 24.83 & 30.52 & 21.83 & 25.07 & 25.49 & 38.66 & 34.95 & 69.05 \\ 585.82 & 499.10 & 712.72 & 630.44 & 564.81 & 548.35 & 600.48 & 5.35 & 577.97 & 600.52 & 21.95 & 144.75 & 19.79 & 21.85 & 22.17 & 30.14 & 26.91 & 59.96 \\ 705.69 & 581.47 & 868.95 & 749.03 & 678.28 & 662.89 & 727.64 & 577.97 & 6.33 & 30.52 & 28.27 & 35.56 & 23.54 & 26.55 & 27.08 & 39.32 & 35.53 & 70.97 \\ 26.14 & 21.81 & 30.08 & 0.50 & 6.55 & 6.73 & 97.18 & 600.52 & 30.52 & 625.10 & 770.58 & 243.24 & 625.10 & 770.58 & 671.85 & 1214.73 & 935.97 & 1225.74 \\ 26.15 & 13.95 & 40.40 & 26.55 & 30.76 & 32.24 & 24.83 & 21.95 & 28.27 & 600.52 & 0.59 & 144.75 & 8.28 & 11.13 & 10.97 & 15.12 & 13.34 & 16.66 \\ 26.15 & 14.30 & 33.77 & 6.73 & 11.68 & 0.59 & 30.52 & 144.75 & 35.56 & 243.24 & 144.75 & 35.56 & 243.24 & 277.94 & 229.36 & 405.41 & 284.49 & 429 \\ 27.96 & 26.11 & 29.67 & 31.44 & 27.37 & 24.12 & 21.83 & 19.79 & 23.54 & 625.10 & 8.28 & 243.24 & 0.73 & 16.71 & 13.13 & 25.13 & 16.62 & 22.64 \\ 30 & 24.46 & 34.50 & 32.12 & 29.56 & 27.85 & 25.07 & 21.85 & 26.55 & 770.58 & 11.13 & 277.94 & 16.71 & 0.86 & 14.89 & 26.41 & 18.50 & 23.32 \\ 29.49 & 21.10 & 36.28 & 28.98 & 29.07 & 28.82 & 25.49 & 22.17 & 27.08 & 671.85 & 10.97 & 229.36 & 13.13 & 14.89 & 0.74 & 21.10 & 15.57 & 18.72 \\ 45.42 & 39.08 & 51.11 & 51.16 & 44.97 & 40.13 & 38.66 & 30.14 & 39.32 & 1214.73 & 15.12 & 405.41 & 25.13 & 26.41 & 21.10 & 1.32 & 32.47 & 39.67 \\ 39.39 & 30.41 & 46.96 & 42.04 & 39.05 & 36.23 & 34.95 & 26.91 & 35.53 & 935.97 & 13.34 & 284.49 & 16.62 & 18.50 & 15.57 & 32.47 & 0.98 & 27.84 \\ 81.21 & 80.37 & 85.74 & 96.72 & 79.30 & 66.80 & 69.05 & 59.96 & 70.97 & 1225.74 & 13.66 & 429 & 22.64 & 23.32 & 18.72 & 39.67 & 27.84 & 1.37 \end{bmatrix}$$

- Senaryo 1-d: Senaryo 1 ve landmark sayısı 12 için kullanılan varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 6.51 & 610.27 & 896.93 & 787.10 & 706.48 & 681.94 & 735.54 & 585.82 & 705.69 & 718.60 & 689.43 & 706.66 & 26.14 & 26.15 & 26.15 & 27.96 & 30 & 29.49 & 45.42 & 39.39 & 81.21 & 28.14 & 27.47 & 72.26 \\ 610.27 & 5.41 & 687.11 & 700.68 & 587.11 & 537.56 & 592.92 & 499.10 & 581.47 & 589.74 & 588.63 & 576.67 & 21.81 & 13.95 & 14.30 & 26.11 & 24.46 & 21.10 & 39.08 & 30.41 & 80.37 & 23.98 & 21.84 & 64.68 \\ 896.93 & 687.11 & 7.88 & 916.24 & 853.78 & 845.88 & 919.54 & 712.72 & 868.95 & 878.61 & 820.38 & 878.43 & 30.08 & 40.40 & 33.77 & 29.67 & 34.50 & 36.28 & 51.11 & 46.96 & 85.74 & 32.25 & 32.47 & 81.77 \\ 787.10 & 700.68 & 916.24 & 7.12 & 752.36 & 702.93 & 778.02 & 630.44 & 749.03 & 762.76 & 741.57 & 747.33 & 0.50 & 26.55 & 6.73 & 31.44 & 32.12 & 28.98 & 51.16 & 42.04 & 96.72 & 30.26 & 28.39 & 82.30 \\ 706.48 & 587.11 & 853.78 & 752.36 & 6.21 & 649.50 & 705.94 & 564.81 & 678.28 & 687.08 & 659.69 & 680 & 6.55 & 30.76 & 11.68 & 27.37 & 29.59 & 29.07 & 44.97 & 39.05 & 79.30 & 27.81 & 27.32 & 70.56 \\ 681.94 & 537.56 & 845.88 & 702.93 & 649.50 & 6.20 & 693.72 & 548.35 & 662.89 & 666.55 & 633.25 & 667.11 & 6.73 & 32.24 & 0.59 & 24.12 & 27.85 & 28.82 & 40.13 & 36.23 & 66.80 & 26.03 & 26.10 & 61.99 \\ 735.54 & 592.92 & 919.54 & 778.02 & 705.94 & 693.72 & 6.72 & 600.48 & 727.64 & 729.01 & 682.26 & 737.22 & 97.18 & 24.83 & 30.52 & 21.83 & 25.07 & 25.49 & 38.66 & 34.95 & 69.05 & 24.02 & 23.36 & 64.24 \\ 585.82 & 499.10 & 712.72 & 630.44 & 564.81 & 548.35 & 600.48 & 5.35 & 577.97 & 577.59 & 556.45 & 581.17 & 600.52 & 21.95 & 144.75 & 19.79 & 21.85 & 22.17 & 30.14 & 26.91 & 59.96 & 20.24 & 20.20 & 53.01 \\ 705.69 & 581.47 & 868.95 & 749.03 & 678.28 & 662.89 & 727.64 & 577.97 & 6.33 & 696.16 & 662.82 & 700.26 & 30.52 & 28.27 & 35.56 & 23.54 & 26.55 & 27.08 & 39.32 & 35.53 & 70.97 & 24.87 & 24.81 & 64.89 \\ 718.60 & 589.74 & 878.61 & 762.76 & 687.08 & 666.55 & 729.01 & 577.59 & 696.16 & 6.36 & 669.54 & 699.70 & 625.10 & 600.52 & 243.24 & 25.69 & 28.52 & 28.50 & 42.73 & 37.93 & 75.93 & 26.56 & 26.24 & 69.17 \\ 689.43 & 588.63 & 820.38 & 741.57 & 659.69 & 633.25 & 682.26 & 556.45 & 662.82 & 669.54 & 6.13 & 661.49 & 770.58 & 0.59 & 144.75 & 28.24 & 30.24 & 29.52 & 45.38 & 39.01 & 80.40 & 28.28 & 27.97 & 70.30 \\ 706.66 & 576.67 & 878.43 & 747.33 & 680 & 667.11 & 737.22 & 581.17 & 700.26 & 699.70 & 661.49 & 6.36 & 53.87 & 26.41 & 19.57 & 21.95 & 25.01 & 25.84 & 37.13 & 33.93 & 67.96 & 23.67 & 23.57 & 62.54 \\ 26.14 & 21.81 & 30.08 & 0.50 & 6.55 & 6.73 & 97.18 & 600.52 & 30.52 & 625.10 & 770.58 & 53.87 & 97.18 & 600.52 & 30.58 & 625.10 & 770.58 & 671.85 & 12.14 & 935.97 & 12.25 & 723 & 685.18 & 10.27 \\ 26.15 & 13.95 & 40.40 & 26.55 & 30.76 & 32.24 & 24.83 & 21.95 & 28.27 & 600.52 & 0.59 & 26.41 & 600.52 & 0.59 & 144.75 & 8.28 & 11.13 & 10.97 & 15.12 & 13.34 & 13.66 & 9.17 & 10.05 & 13.06 \\ 26.15 & 14.30 & 33.77 & 6.73 & 11.68 & 0.59 & 30.52 & 144.75 & 35.56 & 243.24 & 144.75 & 19.57 & 30.58 & 144.75 & 35.56 & 243.24 & 277.94 & 229.36 & 405.41 & 284.49 & 429 & 282.24 & 245.72 & 323.15 \\ 27.96 & 26.11 & 29.67 & 31.44 & 27.37 & 24.12 & 21.83 & 19.79 & 23.54 & 25.69 & 28.24 & 21.95 & 625.10 & 8.28 & 243.24 & 0.73 & 16.71 & 13.13 & 25.13 & 16.62 & 22.64 & 18.56 & 15.30 & 16.41 \\ 30 & 24.46 & 34.50 & 32.12 & 29.56 & 27.85 & 25.07 & 21.85 & 26.55 & 28.52 & 30.24 & 25.01 & 770.58 & 11.13 & 277.94 & 16.71 & 0.86 & 14.89 & 26.41 & 18.50 & 23.32 & 18.96 & 16.40 & 17.92 \\ 29.49 & 21.10 & 36.28 & 28.98 & 29.07 & 28.82 & 25.49 & 22.17 & 27.08 & 28.50 & 29.52 & 25.84 & 671.85 & 10.97 & 229.36 & 13.13 & 14.89 & 0.74 & 21.10 & 15.57 & 18.72 & 14.90 & 13.49 & 15.12 \\ 45.42 & 39.08 & 51.11 & 51.16 & 44.97 & 40.13 & 38.66 & 30.14 & 39.32 & 42.73 & 45.38 & 37.13 & 12.14 & 15.12 & 405.41 & 25.13 & 26.41 & 21.10 & 1.32 & 32.47 & 39.67 & 29.17 & 25.16 & 29.82 \\ 39.39 & 30.41 & 46.96 & 42.04 & 39.05 & 36.23 & 34.95 & 26.91 & 35.53 & 37.93 & 39.01 & 33.93 & 935.97 & 13.34 & 284.49 & 16.62 & 18.50 & 15.57 & 32.47 & 0.98 & 27.84 & 19.30 & 17.61 & 22.22 \\ 81.21 & 80.37 & 85.74 & 96.72 & 79.30 & 66.80 & 69.05 & 59.96 & 70.97 & 75.93 & 80.40 & 67.96 & 12.25 & 13.66 & 429 & 22.64 & 23.32 & 18.72 & 39.67 & 27.84 & 1.37 & 25.10 & 21.83 & 29.89 \\ 28.14 & 23.98 & 32.25 & 30.26 & 27.81 & 26.03 & 24.02 & 20.24 & 24.87 & 26.56 & 28.28 & 23.67 & 723.59 & 9.17 & 282.84 & 18.56 & 18.96 & 14.90 & 29.17 & 19.30 & 25.10 & 0.84 & 17.54 & 18.13 \\ 27.47 & 21.84 & 32.47 & 28.39 & 27.32 & 26.10 & 23.36 & 20.20 & 24.81 & 26.24 & 27.94 & 23.57 & 685.18 & 10.05 & 245.72 & 15.30 & 16.40 & 13.49 & 25.16 & 17.61 & 21.83 & 17.54 & 0.77 & 16.59 \\ 72.26 & 64.68 & 81.77 & 82.30 & 70.56 & 61.99 & 64.24 & 53.01 & 64.89 & 69.17 & 70.30 & 62.54 & 10.27 & 13.06 & 323.15 & 16.41 & 17.92 & 15.12 & 29.82 & 22.22 & 29.89 & 18.13 & 16.59 & 1.11 \end{bmatrix}$$

- Senaryo 2-a: Senaryo 2 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6$
- Senaryo 2-b: Senaryo 2 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12}$
- Senaryo 2-c: Senaryo 2 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18}$
- Senaryo 2-d: Senaryo 2 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24}$
- Senaryo 3-a: Senaryo 3 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 5$
- Senaryo 3-b: Senaryo 3 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 5$
- Senaryo 3-c: Senaryo 3 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 5$
- Senaryo 3-d: Senaryo 3 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 5$
- Senaryo 4-a: Senaryo 4 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 0.5$
- Senaryo 4-b: Senaryo 4 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 0.5$
- Senaryo 4-c: Senaryo 4 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 0.5$
- Senaryo 4-d: Senaryo 4 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 0.5$
- Senaryo 5-a: Senaryo 5 ve 3 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & 2 & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

- Senaryo 5-b: Senaryo 5 ve 6 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} \left(\begin{array}{cc} 1 & 2 \\ & \ddots \\ 2 & 1 \end{array} \right) \end{bmatrix}_{12 \times 12}$$

- Senaryo 5-c: Senaryo 5 ve 9 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} \left(\begin{array}{cc} 1 & 2 \\ & \ddots \\ 2 & 1 \end{array} \right) \end{bmatrix}_{18 \times 18}$$

- Senaryo 5-d: Senaryo 5 ve 12 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} \left(\begin{array}{cc} 1 & 2 \\ & \ddots \\ 2 & 1 \end{array} \right) \end{bmatrix}_{24 \times 24}$$

- Senaryo 6-a: Senaryo 6 ve 3 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 1 & 2 & 2.1 & 2.1 & 2.3 & 2.4 \\ 2 & 1.1 & 2.3 & 2.4 & 2.5 & 2.3 \\ 2.1 & 2.3 & 1.4 & 2.4 & 2.3 & 2.3 \\ 2.1 & 2.4 & 2.4 & 1 & 2.2 & 2.3 \\ 2.3 & 2.5 & 2.3 & 2.2 & 1.1 & 2 \\ 2.4 & 2.3 & 2.3 & 2.3 & 2 & 1.4 \end{bmatrix}$$

- Senaryo 6-b: Senaryo 6 ve 6 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2.1 & 2.3 & 2.4 & 2.1 & 2 & 2.1 \\ 2 & 1.1 & 2.3 & 2.1 & 2.4 & 2.5 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 & 2.5 \\ 2.1 & 2.3 & 1.4 & 2.1 & 2.3 & 2.3 & 2.4 & 2.3 & 2.3 & 2.2 & 2.3 & 2.3 \\ 2.3 & 2.1 & 2.1 & 1.3 & 2.2 & 2.3 & 2.2 & 2.4 & 2.2 & 2.2 & 2.2 & 2.1 \\ 2.4 & 2.4 & 2.3 & 2.2 & 1.5 & 2 & 2.3 & 2.4 & 2.1 & 2 & 2.1 & 2.3 \\ 2.1 & 2.5 & 2.3 & 2.3 & 2 & 1.2 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 & 2.5 \\ 2.1 & 2.4 & 2.4 & 2.2 & 2.3 & 2.4 & 1 & 2.2 & 2.3 & 2.2 & 2.1 & 2.2 \\ 2.3 & 2.5 & 2.3 & 2.4 & 2.4 & 2.5 & 2.2 & 1.1 & 2 & 2.1 & 2.3 & 2.4 \\ 2.4 & 2.3 & 2.3 & 2.2 & 2.1 & 2.3 & 2.3 & 2 & 1.4 & 2 & 2.1 & 2.3 \\ 2.1 & 2.1 & 2.2 & 2.2 & 2 & 2.1 & 2.2 & 2.1 & 2 & 1.3 & 2.3 & 2.1 \\ 2 & 2.4 & 2.3 & 2.2 & 2.1 & 2.4 & 2.1 & 2.3 & 2.1 & 2.3 & 1.5 & 2.1 \\ 2.1 & 2.5 & 2.3 & 2.1 & 2.3 & 2.5 & 2.2 & 2.4 & 2.3 & 2.1 & 2.1 & 1.2 \end{bmatrix}$$

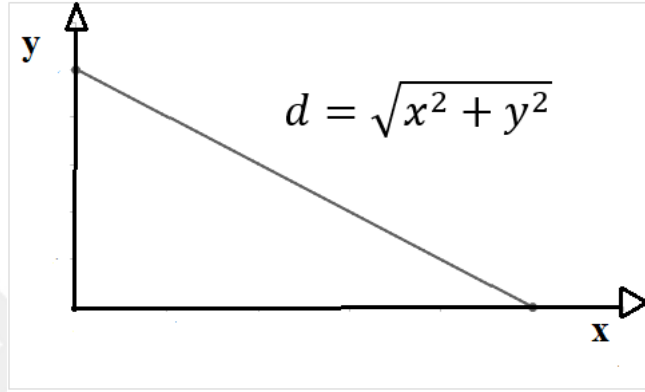
- Senaryo 6-c: Senaryo 6 ve 9 landmarklı durumda varyans-kovaryans matrisi

$$\Sigma = \begin{bmatrix} 1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2 & 2.1 & 2.3 & 2.1 & 2.3 & 2.4 & 2.1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 \\ 2 & 1.1 & 2.3 & 2.1 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 \\ 2.1 & 2.3 & 1.4 & 2.1 & 2.3 & 2.3 & 2.1 & 2.3 & 2.3 & 2.4 & 2.3 & 2.3 & 2.2 & 2.3 & 2.3 & 2.3 & 2.3 & 2.3 \\ 2.3 & 2.1 & 2.1 & 1.3 & 2.2 & 2.3 & 2.2 & 2.1 & 2.2 & 2.2 & 2.4 & 2.2 & 2.2 & 2.2 & 2.1 & 2.5 & 2.1 & 2.2 \\ 2.4 & 2.4 & 2.3 & 2.2 & 1.5 & 2 & 2.1 & 2.3 & 2.4 & 2.3 & 2.4 & 2.1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2.2 \\ 2.1 & 2.5 & 2.3 & 2.3 & 2 & 1.2 & 2.3 & 2.1 & 2.4 & 2.4 & 2.5 & 2.3 & 2.1 & 2.4 & 2.5 & 2.3 & 2.1 & 2 \\ 2 & 2.3 & 2.1 & 2.2 & 2.1 & 2.3 & 1 & 2.1 & 2.3 & 2.3 & 2.5 & 2.3 & 2.3 & 2.4 & 2.3 & 2.3 & 2.2 & 2.3 \\ 2.1 & 2.1 & 2.3 & 2.1 & 2.3 & 2.1 & 2.1 & 1.1 & 2.2 & 2.2 & 2.4 & 2.2 & 2.5 & 2.2 & 2.4 & 2.2 & 2.2 & 2.2 \\ 2.3 & 2.4 & 2.3 & 2.2 & 2.4 & 2.4 & 2.3 & 2.2 & 1.4 & 2.4 & 2.1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2 & 2.1 \\ 2.1 & 2.4 & 2.4 & 2.2 & 2.3 & 2.4 & 2.3 & 2.2 & 2.4 & 1 & 2.2 & 2.3 & 2.2 & 2.1 & 2.2 & 2.4 & 2.2 & 2.5 \\ 2.3 & 2.5 & 2.3 & 2.4 & 2.4 & 2.5 & 2.5 & 2.4 & 2.1 & 2.2 & 1.1 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2 & 2.1 \\ 2.4 & 2.3 & 2.3 & 2.2 & 2.1 & 2.3 & 2.3 & 2.2 & 2 & 2.3 & 2 & 1.4 & 2 & 2.1 & 2.3 & 2.4 & 2.1 & 2 \\ 2.1 & 2.1 & 2.2 & 2.2 & 2 & 2.1 & 2.3 & 2.5 & 2.1 & 2.2 & 2.1 & 2 & 1.3 & 2.3 & 2.1 & 2.4 & 2.5 & 2.3 \\ 2 & 2.4 & 2.3 & 2.2 & 2.1 & 2.4 & 2.4 & 2.2 & 2.3 & 2.1 & 2.3 & 2.1 & 2.3 & 1.5 & 2.1 & 2.3 & 2.3 & 2.1 \\ 2.1 & 2.5 & 2.3 & 2.1 & 2.3 & 2.5 & 2.3 & 2.4 & 2.2 & 2.4 & 2.3 & 2.1 & 2.1 & 1.2 & 2.2 & 2.3 & 2.3 & 2.2 \\ 2.3 & 2.3 & 2.3 & 2.5 & 2.4 & 2.3 & 2.3 & 2.2 & 2.1 & 2.4 & 2.1 & 2.4 & 2.4 & 2.3 & 2.2 & 1 & 2 & 2.1 \\ 2.4 & 2.1 & 2.3 & 2.5 & 2.4 & 2.3 & 2.3 & 2.2 & 2.1 & 2.4 & 2.1 & 2.4 & 2.4 & 2.3 & 2.2 & 1 & 2 & 2.1 \\ 2.1 & 2.4 & 2.3 & 2.2 & 2 & 2.4 & 2.3 & 2.2 & 2.1 & 2.5 & 2.1 & 2 & 2.3 & 2.1 & 2.2 & 2.1 & 2 & 1.4 \end{bmatrix}$$

- Senaryo 7-a: Senaryo 7 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 0.1$
- Senaryo 7-b: Senaryo 7 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 0.1$
- Senaryo 7-c: Senaryo 7 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 0.1$
- Senaryo 7-d: Senaryo 7 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 0.1$
- Senaryo 8-a: Senaryo 8 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 0.05$
- Senaryo 8-b: Senaryo 8 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 0.05$
- Senaryo 8-c: Senaryo 8 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 0.05$
- Senaryo 8-d: Senaryo 8 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 0.1$
- Senaryo 9-a: Senaryo 9 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 0.01$
- Senaryo 9-b: Senaryo 9 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 0.01$
- Senaryo 9-c: Senaryo 9 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 0.01$
- Senaryo 9-d: Senaryo 9 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 0.01$
- Senaryo 10-a: Senaryo 10 ve 3 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_6 \times 0.001$
- Senaryo 10-b: Senaryo 10 ve 6 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{12} \times 0.001$
- Senaryo 10-c: Senaryo 10 ve 9 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{18} \times 0.001$
- Senaryo 10-d: Senaryo 10 ve 12 landmarklı durumda varyans-kovaryans matrisi: $\Sigma = I_{24} \times 0.001$

Simülasyon çalışması 1000 tekrarla yürütülmüştür ve iki boyutlu olarak türetilen landmark veri setinden i' inci landmarka ait kartezyen koordinatlar kayıp veri tahmin performansını test etmek için çıkartılmıştır.

Yöntemlerin kayıp landmarkı tahmin etmedeki performanslarını karşılaştırmak için x ve y noktaları arasındaki uzaklık (d: hipotenüs) dikkate alınarak hata kareler ortalamasının karekökü (RMSE) ölçütü kullanılmıştır (Şekil 10).



Şekil 10. RMSE ölçütünün hesaplanmasında kullanılan x ve y noktaları arasındaki d uzaklığı

$$RMSE = \sqrt{\frac{\sum_{i=1}^r (d_i^{tahmin\ edilen} - d_i^{gözlener})^2}{r}}, \quad i=1,2,3, \dots, r.$$

(60)

Eşitlik-60' da r tekrar sayısını göstermektedir.

4. BULGULAR

4.1. Gerçek Veriye Baęlı Simülasyon

Ozdemir ve ark. (2010) alıřmasında yer alan gerek kranium landmark koordinatlarının betimleyici deęerleri kullanılarak oluřturulan Senaryo 1' den elde edilen sonular Tablo 2' de verilmiřtir. Senaryo 1, 3 landmarklı ve $n=50$ durumunda tez alıřmasında nerilmiř olan F yaklařımına ynelik iterasyonda hesaplanan F istatistik deęerlerinin daęılım grafięi Grafik 5' te verilmiřtir.

Tablo 2. Senaryo 1 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

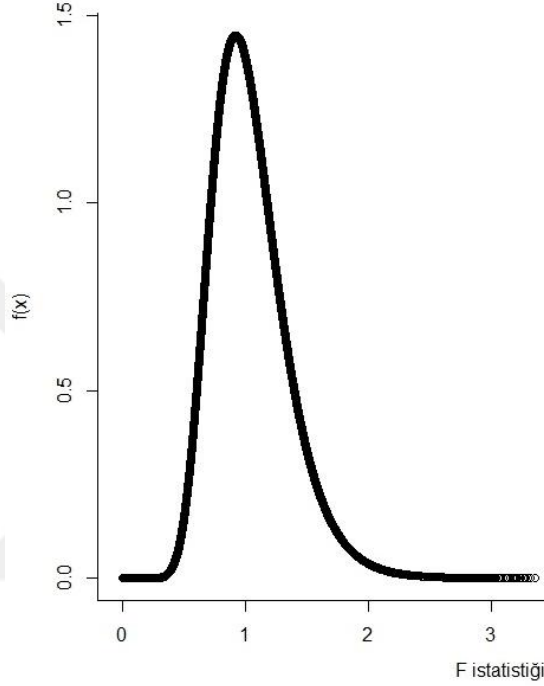
Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	1089.711	<u>1507.406</u>	<u>2279.734</u>
Önerilen algoritma (MaxF)		3153.889	2262.167	3081.448
Regresyon		<u>515.733</u>	1995.172	3323.355
Önerilen algoritma (MinF)	6	1961.090	8131.829	9772.201
Önerilen algoritma (MaxF)		3438.837	11426.520	11514.820
Regresyon		3938.094	7767.117	13513.070
BPCA		<u>1527.543</u>	5018.074	10073.970
INIPALS		1784.895	3982.793	11361.390
NLPCA		5072.692	26661.56	<u>5621.938</u>
PPCA		2328.066	7409.171	8836.567
EM		1732.616	<u>3460.328</u>	12795.690
Önerilen algoritma (MinF)		9	3966.026	2074.778
Önerilen algoritma (MaxF)	<u>618.969</u>		1368.918	1606.864
Regresyon	5043.554		946.610	1343.914
BPCA	2470.691		515.439	9032.632
INIPALS	935.411		<u>508.683</u>	895.751
NLPCA	8254.187		715.875	4480.708
PPCA	3728.743		515.480	8277.032
EM	3813.620		608.465	<u>761.700</u>
Önerilen algoritma (MinF)	12	799.118	5499.235	2261.604
Önerilen algoritma (MaxF)		<u>637.459</u>	3226.846	762.285
Regresyon		729.539	4824.130	1213.680
BPCA		8320.772	1087.778	390.345
INIPALS		6720.646	<u>982.646</u>	<u>338.597</u>
NLPCA		1272.289	2126.765	527.953
PPCA		9540.851	1117.035	358.511
EM		-**	2553.860	5224.780

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 1' de 3 landmarklı durumda n=30 alındığında regresyon atama yöntemi iyi sonuç verirken n=50 ve 100 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30 alındığında BPCA,

n=50 alındığında EM algoritması, n=100 alındığında ise NLPCA yöntemi iyi sonuçlar vermiştir. 9 landmark dikkate alındığında n=30 alındığında Max(F) yaklaşımı, n=50 alındığında INIPALS yöntemi, n=100 alındığında ise EM algoritmasının daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında n=30 alındığında Max(F) yaklaşımı yöntemi iyi sonuç verirken n=50 ve n=100 alındığında INIPALS yöntemi daha iyi sonuç vermiştir (Tablo 2).



Grafik 5. Senaryo 1, 3 landmarklı ve n=50 durumunda tahmin edilen F istatistik değerleri

4.2. Simülasyon Çalışmasından Elde Edilen Sonuçlar

Oluşturulan simülasyon senaryolarına göre Senaryo 2-10' dan elde edilen sonuçlar sırasıyla Tablo 3-11' de verilmiştir. Simülasyon sonuçlarının genel değerlendirilmesi Tablo 12' de verilmiştir.

Tablo 3. Senaryo 2 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	177.966	<u>74.941</u>	301.128
Önerilen algoritma (MaxF)		<u>56.131</u>	573.342	<u>17.524</u>
Regresyon		220.315	261.166	217.142
Önerilen algoritma (MinF)	6	235.479	641.505	457.469
Önerilen algoritma (MaxF)		150.349	32.105	2362.921
Regresyon		78.467	1907.181	1991.355
BPCA		18.481	72.227	216.414
INIPALS		16.9137	60.334	222.229
NLPCA		31.519	204.129	445.717
PPCA		19.03337	75.938	<u>214.446</u>
EM		<u>14.586</u>	<u>22.380</u>	239.021
Önerilen algoritma (MinF)		9	<u>81.105</u>	576.318
Önerilen algoritma (MaxF)	174.169		4833.972	495.501
Regresyon	152.825		1261.711	601.870
BPCA	1270.039		573.269	415.394
INIPALS	215.652		<u>573.246</u>	24.296
NLPCA	2505.003		573.459	2190.256
PPCA	1502.868		573.268	136.025
EM	35320.180		1087.312	1130.672
Önerilen algoritma (MinF)	12		<u>35.939</u>	342.974
Önerilen algoritma (MaxF)		46.322	<u>23.229</u>	<u>19.574</u>
Regresyon		36.877	567.373	58.723
BPCA		167.497	709.556	466.303
INIPALS		157.298	585.279	43.572
NLPCA		329.280	1654.940	4413.822
PPCA		171.446	686.577	449.878
EM		-**	71.924	231.933

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 2' de 3 landmarklı durumda n=30 ve n=100 alındığında Max(F) yaklaşımı iyi sonuç verirken n=50 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30 ve n=50 alındığında EM algoritması, n=100 alındığında ise PPCA yöntemi iyi sonuçlar vermiştir. 9 landmark

dikkate alındığında n=30 ve n=100 alındığında Min(F), n=50 alındığında ise INIPALS yönteminin daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında n=30 alındığında Min(F) yaklaşımı iyi sonuç verirken n=50 ve n=100 alındığında Max(F) yaklaşımı daha iyi sonuç vermiştir (Tablo 3).

Tablo 4. Senaryo 3 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	418.117	834.993	762.743
Önerilen algoritma (MaxF)		<u>133.281</u>	276.627	<u>295.578</u>
Regresyon		588.477	<u>231.013</u>	2162.870
Önerilen algoritma (MinF)	6	528.412	840.502	1174.188
Önerilen algoritma (MaxF)		339.011	112.559	2709.025
Regresyon		178.803	<u>2734.097</u>	2367.703
BPCA		37.119	174.794	160.812
INIPALS		45.933	237.572	394.099
NLPCA		77.278	253.658	1077.016
PPCA		54.379	165.671	<u>159.550</u>
EM		<u>32.569</u>	<u>156.932</u>	492.629
Önerilen algoritma (MinF)	9	<u>187.985</u>	<u>38.188</u>	464.639
Önerilen algoritma (MaxF)		402.669	450.889	753.644
Regresyon		353.332	156.481	770.229
BPCA		1261.984	248.145	770.229
INIPALS		235.295	248.147	<u>435.870</u>
NLPCA		2548.733	248.200	1901.599
PPCA		1551.787	248.154	578.259
EM		48.630	114.138	522.588
Önerilen algoritma (MinF)	12	<u>82.235</u>	398.285	18862.870
Önerilen algoritma (MaxF)		114.936	60.568	15036.350
Regresyon		85.305	<u>21.942</u>	12877.220
BPCA		193.060	718.043	5430.659
INIPALS		183.221	586.351	<u>776.101</u>
NLPCA		364.576	1013.874	47449.830
PPCA		198.239	698.184	6811.833
EM		-**	22.320	50925.030

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 3' de 3 landmarklı durumda $n=30$ ve $n=100$ alındığında $\text{Max}(F)$ yaklaşımı iyi sonuç verirken $n=50$ alındığında regresyon atama yönteminin daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında $n=30$ ve $n=50$ alındığında EM algoritması, $n=100$ alındığında ise PPCA yöntemi iyi sonuçlar vermiştir. 9 landmark dikkate alındığında $n=30$ ve $n=50$ alındığında $\text{Min}(F)$ yaklaşımı, $n=100$ alındığında ise INIPALS yönteminin daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında $n=30$ alındığında $\text{Min}(F)$ yaklaşımı, $n=50$ alındığında regresyon atama yöntemi ve $n=100$ alındığında INIPALS yöntemi daha iyi sonuç vermiştir (Tablo 4).



Tablo 5. Senaryo 4 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	127.177	<u>26.901</u>	26.097
Önerilen algoritma (MaxF)		<u>39.887</u>	488.237	28.3674
Regresyon		140.979	168.675	<u>18.061</u>
Önerilen algoritma (MinF)	6	159.094	864.705	552.453
Önerilen algoritma (MaxF)		101.137	46.172	2287.706
Regresyon		52.460	2544.643	1943.674
BPCA		13.055	71.553	271.057
INIPALS		14.372	59.297	175.982
NLPCA		26.210	340.703	494.404
PPCA		16.159	75.563	269.914
EM		<u>10.037</u>	<u>17.196</u>	<u>167.230</u>
Önerilen algoritma (MinF)		9	<u>55.436</u>	307.456
Önerilen algoritma (MaxF)	119.872		1774.176	311.199
Regresyon	105.142		450.889	753.644
BPCA	1274.708		<u>236.716</u>	369.483
INIPALS	211.907		237.084	152.546
NLPCA	2452.317		239.083	2333.573
PPCA	1478.323		237.118	439.969
EM	28506.560		328.739	1024.552
Önerilen algoritma (MinF)	12	<u>26.394</u>	321.617	321.617
Önerilen algoritma (MaxF)		33.618	<u>19.435</u>	<u>19.435</u>
Regresyon		27.104	1943.674	1943.674
BPCA		162.829	517.005	517.005
INIPALS		152.931	383.425	383.425
NLPCA		326.5101	814.314	814.3148
PPCA		166.683	479.165	479.165
EM		-**	94.445	94.445

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.
** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 4' te 3 landmarklı durumda n=30 alındığında Max(F) yaklaşımının, n=50 alındığında Min(F) yaklaşımının ve n=100 alındığında ise regresyon atama yönteminin daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında

n=30, n=50 ve n=100 alındığında EM algoritması iyi sonuçlar vermiştir. 9 landmark dikkate alındığında n=30 ve n=100 alındığında Min(F) yaklaşımının, n=50 alındığında ise BPCA yönteminin daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında n=30 alındığında Min(F) yaklaşımı iyi sonuç verirken n=50 ve n=100 alındığında Max(F) yaklaşımı daha iyi sonuç vermiştir (Tablo 5).



Tablo 6. Senaryo 5 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	36.760	878.573	<u>23.904</u>
Önerilen algoritma (MaxF)		<u>24.987</u>	<u>26.359</u>	1737.706
Regresyon		26.949	35.719	24.070
Önerilen algoritma (MinF)	6	404.795	2221.138	1563.661
Önerilen algoritma (MaxF)		259.394	<u>131.904</u>	408.363
Regresyon		214.955	6212.969	3045.866
BPCA		106.418	243.098	2387.016
INIPALS		144.268	180.219	94.992
NLPCA		209.668	335.959	2800.942
PPCA		234.010	243.197	2391.650
EM		<u>46.991</u>	443.652	<u>27.470</u>
Önerilen algoritma (MinF)		9	<u>87.694</u>	<u>1325.220</u>
Önerilen algoritma (MaxF)	193.087		2712.694	58.479
Regresyon	187.661		30848.560	95.584
BPCA	1154.764		6479.549	158.694
INIPALS	568.407		6479.503	145.854
NLPCA	1945.908		6480.792	158.749
PPCA	423.932		6479.542	162.960
EM	883.799		1710.746	102.896
Önerilen algoritma (MinF)	12		72.683	<u>1004.977</u>
Önerilen algoritma (MaxF)		215.858	1514.610	63.810
Regresyon		<u>33.596</u>	12365.493	177.428
BPCA		568.068	3361.323	122.918
INIPALS		463.884	3160.148	40.417
NLPCA		1065.992	3356.589	512.134
PPCA		597.359	3286.985	114.320
EM		-**	1012.658	967.972

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 5' te 3 landmarklı durumda n=30 ve n=50 alındığında Max(F) yaklaşımının, n=100 alındığında ise Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30 ve n=100 alındığında EM algoritması iyi sonuçlar verirken n=50 alındığında Max(F) yaklaşımının daha iyi sonuç

verdiği görülmektedir. 9 landmark dikkate alındığında n=30, n=50 ve n=100 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında n=30 alındığında regresyon atama yöntemi iyi sonuç verirken n=50 ve n=100 alındığında Min(F) yaklaşımı daha iyi sonuç vermiştir (Tablo 6).

Tablo 7. Senaryo 6 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	63.761	1077.795	82.224
Önerilen algoritma (MaxF)		<u>46.557</u>	<u>29.794</u>	411.453
Regresyon		94.529	145.555	<u>54.400</u>
Önerilen algoritma (MinF)	6	118.870	<u>25.125</u>	<u>20.648</u>
Önerilen algoritma (MaxF)		87.279	367.370	352.584
Regresyon		<u>17.719</u>	82.673	80.149
BPCA		65.336	114.577	104.632
INIPALS		90.949	50.278	48.258
NLPCA		53.451	412.905	409.592
PPCA		60.239	132.697	125.753
EM		29.838	148.254	146.634
Önerilen algoritma (MinF)	9	<u>76.843</u>	<u>83.362</u>	39.455
Önerilen algoritma (MaxF)		231.750	580.059	133.896
Regresyon		252.828	653.750	116.761
BPCA		572.052	113.807	35.163
INIPALS		434.743	113.720	21.333
NLPCA		1722.553	114.012	784.462
PPCA		481.873	113.810	34.811
EM		1823.677	658.180	<u>16.808</u>
Önerilen algoritma (MinF)	12	45.122	922.990	<u>60.633</u>
Önerilen algoritma (MaxF)		67.845	<u>62.448</u>	96.080
Regresyon		<u>44.057</u>	2399.408	136.115
BPCA		271.670	376.220	145.520
INIPALS		260.166	144.293	68.774
NLPCA		1208.678	746.408	706.603
PPCA		278.899	385.887	163.034
EM		..**	589.365	1008.654

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 6' da 3 landmarklı durumda $n=30$ ve $n=50$ alındığında $\text{Max}(F)$ yaklaşımının, $n=100$ alındığında ise regresyon atama yönteminin daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında $n=30$ alındığında regresyon atama yönteminin, $n=50$ ve $n=100$ alındığında $\text{Min}(F)$ yaklaşımının daha iyi sonuç verdiği görülmektedir. 9 landmark dikkate alındığında $n=30$, $n=50$ alındığında $\text{Min}(F)$ yaklaşımı iyi sonuç verirken, $n=100$ alındığında EM algoritmasının daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında $n=30$ alındığında regresyon atama yöntemi, $n=50$ alındığında $\text{Max}(F)$ yaklaşımı ve $n=100$ alındığında $\text{Min}(F)$ yaklaşımı daha iyi sonuç vermiştir (Tablo 7).



Tablo 8. Senaryo 7 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	52.558	23.107	145.002
Önerilen algoritma (MaxF)		16.215	27.640	12.309
Regresyon		36.975	42.897	32.4687
Önerilen algoritma (MinF)	6	73.241	1095.754	613.839
Önerilen algoritma (MaxF)		45.997	60.296	80.831
Regresyon		23.557	855.401	250.203
BPCA		6.0177	70.743	58.268
INIPALS		8.644	44.939	26.438
NLPCA		11.747	120.946	63.481
PPCA		10.079	74.423	57.678
EM		4.476	6.286	3.082
Önerilen algoritma (MinF)	9	23.737	13.557	34.8346
Önerilen algoritma (MaxF)		52.034	24.414	273.922
Regresyon		45.519	73.8211	72.648
BPCA		1289.328	902.4544	62.5413
INIPALS		210.963	903.4501	37.01711
NLPCA		1695.408	894.6384	1298.046
PPCA		948.726	903.619	58.42228
EM		16483.040	278.1173	829.4518
Önerilen algoritma (MinF)	12	11.884	158.1613	9.100
Önerilen algoritma (MaxF)		15.639	771.8525	6.962
Regresyon		12.247	360.9658	4.246
BPCA		156.641	4195.298	454.952
INIPALS		146.419	367.7007	35.126
NLPCA		317.106	7445.648	1541.222
PPCA		161.427	3461.769	435.341
EM		-**	553.9291	153.260

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 7' de 3 landmarklı durumda n=30 ve n=100 alındığında Max(F) yaklaşımının, n=50 alındığında ise Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30, n=50 ve n=100 alındığında EM algoritması iyi sonuçlar vermiştir. 9 landmark dikkate alındığında n=30, n=50 ve n=100 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 12

landmark dikkate alındığında $n=30$ ve $n=50$ alındığında Min(F) yaklaşımı iyi sonuç verirken $n=100$ alındığında regresyon atama yöntemi daha iyi sonuç vermiştir (Tablo 8).



Tablo 9. Senaryo 8 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	37.428	46.156	18.596
Önerilen algoritma (MaxF)		<u>11.518</u>	67.817	25.062
Regresyon		23.723	<u>23.723</u>	<u>0.430</u>
Önerilen algoritma (MinF)	6	46.057	505.117	11.883
Önerilen algoritma (MaxF)		28.856	2212.690	21.056
Regresyon		14.845	142.572	12.651
BPCA		4.055	336.577	72.779
INIPALS		6.067	166.669	34.115
NLPCA		8.087	388.799	87.654
PPCA		7.224	346.082	63.330
EM		<u>2.999</u>	<u>9.011</u>	<u>7.484</u>
Önerilen algoritma (MinF)		9	<u>10.325</u>	<u>9.232</u>
Önerilen algoritma (MaxF)	38.085		16.088	6.217
Regresyon	31.678		16.598	11.805
BPCA	1693.828		284.079	449.125
INIPALS	521.288		284.037	34.047
NLPCA	1188.922		284.304	428.992
PPCA	900.452		284.082	424.296
EM	6157.813		44.419	104.871
Önerilen algoritma (MinF)	12		<u>8.609</u>	<u>92.648</u>
Önerilen algoritma (MaxF)		11.494	132.612	<u>4.950</u>
Regresyon		8.887	285.399	11.805
BPCA		159.815	535.136	449.125
INIPALS		150.135	113.758	34.047
NLPCA		309.512	809.780	4278.992
PPCA		164.478	512.525	424.296
EM		-**	146.628	104.871

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 8' de 3 landmarklı durumda n=30 alındığında Max(F) yaklaşımının, n=50 ve n=100 alındığında ise regresyon atama yönteminin daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30, n=50 ve n=100 alındığında EM algoritması iyi sonuçlar vermiştir. 9 landmark dikkate alındığında n=30, n=50 ve

n=100 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında n=30 ve n=50 alındığında Min(F) yaklaşımı iyi sonuç verirken n=100 alındığında Max(F) yaklaşımı daha iyi sonuç vermiştir (Tablo 9).

Tablo 10. Senaryo 9 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	<u>15.895</u>	<u>7.289</u>	<u>3.973</u>
Önerilen algoritma (MaxF)		4.982	9.511	9.07
Regresyon		15.971	10.192	10.193
Önerilen algoritma (MinF)	6	19.549	66.057	<u>3.567</u>
Önerilen algoritma (MaxF)		12.251	81.947	7.357
Regresyon		6.348	203.448	4.689
BPCA		1.767	319.285	184.988
INIPALS		2.895	154.408	96.843
NLPCA		3.479	355.035	184.537
PPCA		3.468	327.603	184.040
EM		<u>1.277</u>	<u>3.477</u>	4.479
Önerilen algoritma (MinF)	9	<u>7.024</u>	<u>18.698</u>	<u>5.137</u>
Önerilen algoritma (MaxF)		15.283	23.176	7.889
Regresyon		13.371	362.459	8.657
BPCA		1287.701	1896.967	1577.92
INIPALS		213.928	1896.958	213.7781
NLPCA		2103.321	1896.984	3538.665
PPCA		916.259	1896.975	791.9374
EM		10582.830	95.63426	4185.635
Önerilen algoritma (MinF)	12	<u>2.324</u>	<u>6.475</u>	<u>7.077</u>
Önerilen algoritma (MaxF)		4.563	12.686	8.200
Regresyon		2.556	7.648	9.646
BPCA		115.743	528.8316	354.619
INIPALS		100.387	114.2012	37.890
NLPCA		290.380	860.0972	693.913
PPCA		118.837	504.2585	342.870
EM		-**	172.9367	702.651

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 9' da 3 landmarklı durumda $n=30$, $n=50$ ve $n=100$ alındığında $\text{Min}(F)$ yaklaşımının iyi sonuçlar verdiği görülmektedir. 6 landmark dikkate alındığında $n=30$ ve $n=50$ alındığında EM algoritması iyi sonuçlar verirken $n=100$ alındığında $\text{Min}(F)$ yaklaşımı daha iyi sonuç vermiştir. 9 landmark dikkate alındığında $n=30$, $n=50$ ve $n=100$ alındığında $\text{Min}(F)$ yaklaşımının daha iyi sonuç verdiği görülmektedir. 12 landmark dikkate alındığında $n=30$, $n=50$ ve $n=100$ alındığında $\text{Min}(F)$ yaklaşımının iyi sonuçlar verdiği görülmektedir (Tablo 10).



Tablo 11. Senaryo 10 ve 3, 6, 9 ve 12 landmark durumlarında elde edilen sonuçlar

Metot	k	n=30	n=50	n=100
Önerilen algoritma (MinF)	3*	5.524	<u>3.904</u>	<u>1.628</u>
Önerilen algoritma (MaxF)		<u>1.813</u>	5.825	2.033
Regresyon		6.990	7.967	3.846
Önerilen algoritma (MinF)	6	<u>1.425</u>	<u>2.874</u>	<u>0.811</u>
Önerilen algoritma (MaxF)		3.343	4.851	1.070
Regresyon		1.837	17.865	6.584
BPCA		0.540	36.577	72.779
INIPALS		0.899	66.669	34.115
NLPCA		1.077	88.799	87.654
PPCA		1.079	46.082	63.330
EM		0.389	9.011	7.484
Önerilen algoritma (MinF)	9	<u>2.003</u>	<u>3.984</u>	<u>0.111</u>
Önerilen algoritma (MaxF)		9.155	12.685	4.035
Regresyon		7.156	22.673	5.348
BPCA		333.802	370.054	227.504
INIPALS		176.872	370.152	24.074
NLPCA		1460.914	370.051	138.711
PPCA		380.388	370.172	218.057
EM		12.851	26.629	7.350
Önerilen algoritma (MinF)	12	<u>1.049</u>	<u>3.216</u>	<u>2.282</u>
Önerilen algoritma (MaxF)		1.466	4.897	3.059
Regresyon		1.088	11.458	5.234
BPCA		157.911	215.985	460.321
INIPALS		150.666	208.743	33.630
NLPCA		280.095	211.684	589.487
PPCA		161.023	216.257	424.237
EM		._**	32.435	20.013

*3 landmarklı durumda landmark sayısının az olmasından dolayı EM algoritması ve PCA tabanlı yöntemler kullanılamamaktadır.

** Gerekli tahminleri yapabilmek için alınan örneklem büyüklüğü çok azdır.

Senaryo 10' da 3 landmarklı durumda n=30 alındığında Max(F) yaklaşımının, n=50 ve n=100 alındığında ise Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 6 landmark dikkate alındığında n=30, n=50 ve n=100 alındığında Min(F) yaklaşımı iyi sonuçlar vermiştir. 9 landmark dikkate alındığında da n=30, n=50 ve n=100 alındığında Min(F) yaklaşımının daha iyi sonuç verdiği görülmektedir. 12

landmark dikkate alındığında ise $n=30$, $n=50$ ve $n=100$ alındığında Min(F) yaklaşımının iyi sonuçlar verdiği görülmektedir (Tablo 11).

Tablo 12. Farklı örneklem büyüklüklerine göre en iyi sonucu veren yöntemlerin frekansları

	n=30					n=50					n=100				
	3	6	9	12	Toplam	3	6	9	12	Toplam	3	6	9	12	Toplam
MinF	1	1	9	7	18	6	2	7	5	20	4	3	7	4	18
MaxF	8		1		9	2	1		3	6	3			3	6
Regresyon	1	1		3	5	2			1	3	3			1	4
BPCA		1			1			1		1					
NİPALS								2	1	3			1	2	3
NLPCA														1	1
PPCA														2	2
EM		7			7		7			7		4	2		6

4.3. Gerçek Veri Uygulaması

Kayıp landmark tahmini için tez çalışmasında önerilmiş olan F yaklaşımının adımlarını sayısal örnekle açıklamak amacıyla Senaryo 1 ve 3 landmarklı durum ve $n=5$ alınmıştır (Tablo 13).

Tablo 13. Kayıp landmark içermeyen veri seti

	x1	x2	x3	y1	y2	y3
1	636.130	1066.270	739.854	1282.627	646.954	646.333
2	623.368	719.535	726.976	935.716	645.830	645.185
3	643.320	905.782	747.023	1121.972	646.897	646.351
4	625.732	2139.795	729.541	2355.921	647.164	646.573
5	620.284	425.833	723.876	642.201	643.994	643.492

Birinci birimde 3. Landmark kayıp landmark (m) olarak belirlendi (Tablo 14).

Tablo 14. Kayıp landmark içeren veri seti

	x1	x2	x3	y1	y2	y3
1	636.130	1066.270	?	1282.627	646.954	?
2	623.368	719.535	726.976	935.716	645.830	645.185
3	643.320	905.782	747.023	1121.972	646.897	646.351
4	625.732	2139.795	729.541	2355.921	647.164	646.573
5	620.284	425.833	723.876	642.201	643.994	643.492

Veri seti Bookstein koordinatlarına dönüştürülür. Birinci birime ait Bookstein koordinatları:

[,1] [,2]

[1,] -0.5 0

[2,] 0.5 0

[3,] ? ?

İkinci birime ait Bookstein koordinatları:

[,1] [,2]

[1,] -0.500000 0.000000e+00

[2,] 0.500000 6.695574e-17

[3,] 2.193868 -2.925077e+00

Üçüncü birime ait Bookstein koordinatları:

[,1] [,2]

[1,] -0.50000000 0.00000000

[2,] 0.50000000 0.00000000

[3,] -0.03471818 -0.04327598

Dördüncü birime ait Bookstein koordinatları:

[,1] [,2]

[1,] -0.5000000 0.0000000000

[2,] 0.5000000 0.0000000000

[3,] 0.4972266 -0.0001573331

Beşinci birime ait Bookstein koordinatları:

[,1] [,2]

[1,] -0.5000000 0.0000000

[2,] 0.5000000 0.0000000

[3,] -0.1103701 -1.259385

Her bir birim için hesaplanan landmark arası uzaklıklara göre i-m (d_2 uzaklığı) ve j-m (d_3 uzaklığı) uzaklıklarının ortalama ve standart hatası hesaplanır. Elde edilen d_2 uzaklıkları sırasıyla 3.976, 0.467, 0.997, 1.318 şeklindedir. d_3 uzaklıkları ise sırasıyla 3.380, 0.536, 0.002, 1.399 şeklindedir. d_2 uzaklığının ortalaması 1.689, d_3 uzaklığının ortalaması 1.329 olarak elde edilmiştir. d_2 uzaklığının standart hatası, d_3 uzaklığının standart hatası 0.782 olarak elde edilmiştir.

d_2 uzaklığının % 95' lik güven aralığı (0.156; 3.222) ve d_3 uzaklığının % 95' lik güven aralığı (-0.123; 2.783) olarak elde edilmiştir. İterasyon katsayısı 1 alınarak F istatistik değerleri elde edilmiştir.

Kriter olarak seçilen Min(F) ve Max(F) istatistik değerine karşılık gelen x ve y koordinatları tahmin edilen koordinatlar olarak ele alınır. Min(F) kriterine göre (x,y) (0.157, 1.88), F istatistiği ise 1.26' dır. Max(F) kriterine göre (x,y) (2.16, -0.124), F istatistiği ise 5.62' dır. Tahmin edilen koordinatlar orijinal koordinat değerlerine geri dönüştürülür. Min(F) ölçütü kullanılarak tahmin edilen koordinatlar (438.166, 598.209), Max(F) ölçütü kullanılarak tahmin edilen koordinatlar (407.991, 587.313) olarak bulunmuştur.

5. TARTIŞMA VE SONUÇ

Araştırma süreci ne kadar titiz ve dikkatli yürütülürse yürütülsün, elde olmayan nedenlerle bazı birimlerden verilerin tamamı veya bir kısmı elde edilememektedir. Kayıp veri sorunu bazen birimlere ulaşamamaktan, bazen teknik nedenlerden ve bazen de birimlerin doğası gereği ortaya çıkmaktadır. Kayıp veri sorunu nedeniyle, model kuramama, bazı istatistiksel analizleri yapamama ve kestirimlerde yanlışlıklar görülebilmektedir. Kayıp veri sorunundaki bu durum, araştırmacıların kayıp veri tahmin yöntemlerine önem vermelerine neden olmuştur.

Kayıp veri tahmininde kullanılan başlıca yöntemler değer atama yöntemleri, kayıp veri ile ortalamanın yer değiştirmesi, EM algoritması, PCA modifiye edilerek elde edilmiş BPCA, INIPALS, NLPCA, PPCA ve regresyon atama olarak sayılabilirler (Arbour ve Brown, 2014; Couette ve White, 2010; Neeser ve ark., 2009; Rubin ve ark., 2007). Kayıp veri tahmini için geliştirilen bu yöntemlerden çoklu atama yöntemleri, EM algoritması ve PCA tabanlı yöntemler MAR kayıp veri yapısını kullanmaktadır (Dong and Peng, 2013). EM algoritması ve çoklu atama yöntemleri model tabanlı yöntemler olarak ele alınırken BPCA, INIPALS, NLPCA, PPCA yöntemleri PCA'ya dayanan yöntemlerdir (Pigott, 2001).

Kayıp veri tahmin yöntemleri geliştirilmesindeki genel ve temel düşünce veri setlerindeki kayıpları tahmin etmektir. Oysa günümüzde veri analizleri artık şekil ve görüntü üzerinden de yapılmaktadır. Teknolojik gelişimle birlikte istatistiksel şekil analizi yaklaşımlarındaki gelişmeyle, şekil ve görüntülerde de veri kayıplarına rastlanılmaya başlanmıştır. Şekil analizinde kayıp veri sorunu landmark kayıpları dolayısıyla kartezyen koordinatları şeklinde olmaktadır. Literatür incelendiğinde şekilsel temelli kayıp landmarklara yönelik geliştirilen özel bir yöntem görülememiştir. Tez çalışmasındaki amaç da standart kayıp verilerde kullanılan yöntemlerin kayıp landmark tahminindeki performanslarının değerlendirilmesi ve

kayıp landmark tahminine yönelik, tez çalışması kapsamında şekil temel olarak geliştirilen “F-yaklaşımı” algoritmasının Min(F) ve Max(F) ölçütlerine göre performansını görmek ve diğer standart kayıp veri yöntemleriyle karşılaştırılmasıdır. Tez çalışmasında literatür incelenerek genel kabul görmüş olan yöntemlerden EM algoritması, BPCA, INIPALS, NLPCA, PPCA ve regresyon atama yöntemleri inceleme kapsamına alınmıştır.

Şekil analizine yönelik kayıp landmark tahmini için yapılan uygulamalar literatürde araştırıldığında az sayıda çalışmaya rastlanılmıştır. Literatürde şekil analizindeki kayıp landmark sorununa yönelik uygulamalar incelendiğinde Bookstein ve ark. (1999) kranium profilleri ile ilgili çalışmalarında kayıp landmark tahmini için ince levha eğri relaksasyon yöntemini (thin-plate spline relaxation method) kullandığı görülmüştür. Literatürde bazı araştırmacılar ise çalışmalarında kayıp landmarkları dikkate almadan o birimleri çıkartarak analizleri uygulamışlardır (Beumer ve ark. (2006).

Şekle yönelik istatistiksel analizlerin uygulandığı morfometri çalışmalarında genellikle veri setleri iskeletlerden, kemiklerden vb. ler gibi anatomik yapılardan oluşmaktadır. Bu tarz anatomik yapılarda zamana bağlı olarak meydana gelen kırılmalar nedeniyle landmark koordinatlarında kayıplar meydana gelmektedir. Landmark koordinatlarının bilinmemesi, o birimin çalışma dışında kalmasına neden olmaktadır. Genellikle küçük örneklem büyüklüğüne sahip paleontoloji ve arkeoloji çalışmalarında her bir landmark çok daha değerlidir. Bu nedenlerden dolayı araştırmalarda Landmarkların konumlandırılması ve kayıp landmark koordinatlarının tahmin edilmesi önem taşımaktadır. Landmarkları konumlandırmada gözlemci hatası rastgele hata kapsamında değerlendirilirken, ölçme aracı hatası da sistematik hata olarak değerlendirilebilir. Küçük örneklem çalışmalarında, kayıp landmarkı olan birim yerine yeni birim konulması mümkün olmadığı durumlarda da kayıp landmark tahmini bir zorunluluk haline gelmektedir.

Morfometrik çalışmalarda kayıp veri tahmini amacıyla yapılan çalışmalar incelendiğinde Couette ve White (2010) üç boyutlu morfometrik verilerde kayıp verinin önemini araştırdığı çalışmasında EM algoritması ve çoklu regresyon atama yöntemlerini karşılaştırmışlardır. Karşılaştırma sonucunda EM algoritması ve çoklu

regresyon atama yöntemlerinin benzer sonuçlar verdiğini göstermişlerdir. Strauss ve ark. (2003) ise morfometrik verilerde kayıp veri tahmininde kullanılmak üzere EM algoritması ve temel bileşenler tahminini karşılaştırarak, yöntemlerin benzer sonuçlar verdiğini bulmuşlardır. Tez çalışmasında ise EM algoritması kayıp landmark tahmininde bu tez çalışmasında önerilmiş olan F-yaklaşımı algoritmasında Max(F) ölçütüne yakın, Min(F) ölçütüne göre ise oldukça düşük performans vermiştir.

Arbour ve Brown (2014), geometrik morfometride kayıp veri tahmininde kullanılan BPCA, en küçük kareler regresyonu, ince levha eğri analizini ve kayıp veri ile ortalamanın yer değiştirmesi yöntemlerinin performanslarını karşılaştırmışlardır. Karşılaştırma sonucunda BPCA ve en küçük kareler regresyonunun güvenilir yöntemler olduğunu belirtmişlerdir. Tez çalışmasında simülasyon sonuçlarına göre BPCA ve regresyon atama yöntemleri bu tez çalışmasında önerilmiş olan Min(F) ve Max(F) ölçütlerine göre oldukça düşük performans göstermiştir.

Brown ve ark. (2012), morfometrik verilerde kayıp veri tahmin yöntemlerinin performansını araştırdığı çalışmasında kayıp veri ile ortalamanın yer değiştirmesi, regresyon atama ve BPCA yöntemlerini karşılaştırarak kayıp veri oranı düşük olduğunda en iyi performansı BPCA yönteminin, en kötü performansı ise ortalamanın yer değiştirmesi yönteminin verdiğini belirtmişlerdir. Neeser ve ark. (2009), fosil kraniumlarının yeniden yapılandırılması amacıyla kayıp landmark tahmini için kayıp veri ile ortalamanın yer değiştirmesi, ince levha eğri analizini ve çoklu regresyon atama yöntemlerini karşılaştırarak regresyon atama yönteminin en iyi sonuçları verdiğini belirtmişlerdir.

Simülasyon çalışmasında örneklem büyüklükleri 30, 50 ve 100 olarak dikkate alındığında, performans değerlendirmesinde en iyi ve en farklı sonucu, tez çalışmasında önerdiğimiz F-yaklaşımı algoritmasına ait Min(F) ölçütü vermiştir. Performans sıralamasına göre diğer yöntemler değerlendirildiğinde yine bu tez çalışmasında önerilmiş olan Max(F) ölçütlü F-yaklaşımı ve EM algoritmaları benzer sonuçlar vermişlerdir. Kayıp landmark tahminine yönelik performans başarısında regresyon atama yaklaşımı sonraki yöntem olarak değerlendirilebilir. PCA temelli BPCA, INIPALS, NLPCA, PPCA yöntemlerinin tez çalışmasında incelenen yöntemler gibi başarılı olamadığı ifade edilebilir. Landmark tabanlı kayıp veri

tahminine yönelik diğerk arařtırmalarda kayıp veri oranının az olduđu durumlarda BPCA yönteminin iyi performans gösterdiğini belirtmişlerdir (Arbour ve Brown, 2014; Brown ve ark., 2012).

Kayıp landmark tahmininde kullanılan literatürde sıklıkla karşılaşılan yöntemlerin ve tez çalışmasında incelenen yöntemlerin birçođu istatistik teorisi yoğun ve dolayısıyla bazı istatistiksel varsayımların sağlanamaması durumunda performans düşüklüğü gösterebilecek yöntemlerdir. Bu yöntemlerde landmark sayısı da oldukça önem taşımaktadır. Regresyon atama yöntemi dışındaki, kayıp landmark tahmini için tez çalışmasında incelenen yöntemlerde, landmark sayısı 3 olması durumunda sorun yaşanmaktadır. Tez çalışmasında önerdiğimiz F-yaklaşımı algoritması şekli oluşturan 3 landmarkın olması durumunda kayıp landmark tahmini için kullanılabilir. Üç landmark olması durumuna göre regresyon atama yöntemi ile önerdiğimiz F-yaklaşımı algoritmasına yönelik simülasyon sonuçları değerlendirildiğinde $n=30$ olduğunda $Max(F)$ ölçütünün iyi sonuçlar verdiği, $n=50$ olduğunda ise $Min(F)$ ölçütünün iyi sonuçlar verdiği ve $n=100$ ise regresyon atama yöntemi, $Max(F)$ ve $Min(F)$ yakın sonuçlar verdiği görülmüştür. Landmark sayısının 3'den fazla olması da diğerk yöntemler açısından bir avantaj haline dönüşmemektedir.

6. KAYNAKLAR

- Adams DC (1999) Methods for shape analysis of landmark data from articulated structures. *Evolutionary Ecology Research* 1: 959-970.
- Adams DC, Rohlf FJ (2004) Geometric morphometrics ten years of progress following the “revolution”. *Italian Journal of Zoology* 71: 5-16.
- Aldridge K, Boyadjiev SA, Capone GT et al (2005) Precision and error of three-dimensional phenotypic measures acquired from 3dMD photogrammetric images. *American Journal of Medical Genetics* 138A: 247-253.
- Anwary AR (2012) Statistical Shape Analysis for the Human Back Statistical Shape Analysis for the Human Back. <https://core.ac.uk/download/pdf/9559663.pdf>
- Arbour JH, Brown CM (2014) Incomplete specimens in geometric morphometric analyses. *Methods in Ecology and Evolution* 5: 16-26.
- Bates D, Maechler M (2017) Matrix: Sparse and dense matrix classes and methods. R package version 1.2-11. <https://CRAN.R-project.org/package=Matrix>
- Bookstein FL (1984) A statistical method for biological shape comparisons. *Journal of Theoretical Biology* 107: 475-520.
- Bookstein FL (1986) Size and shape spaces for landmark data in two dimensions. *Statistical Science* 1: 181-242.
- Bookstein FL (1992) *Morphometric Tools for Landmark Data: Geometry and Biology*. Cambridge University Press, Cambridge.
- Bocxlaer VB, Schultheiß R (2010) Comparison of morphometric techniques for shapes with few homologous landmarks based on machine-learning approaches to biological discrimination. *Paleobiology* 36: 497-515.
- Brombin C, Salmaso L (2009) Multi-aspect permutation tests in shape analysis with small sample size. *Computational Statistics and Data Analysis* 53: 3921-3931.
- Brombin C, Salmaso L (2013) *Permutation Tests in Shape Analysis*. Springer, New York.
- Brown CM, Arbour JH, Jackson DA (2012) Testing of the effect of missing data estimation and distribution in morphometric multivariate data analyses. *Systematic Biology* 61: 941-954.
- Buuren SV, Groothuis-Oudshoorn K (2011) mice: multivariate imputation by chained equations in R. Versiyon: 2.30. *Journal of Statistical Software* 45(3), 1-67. URL <http://www.jstatsoft.org/v45/i03/>. (Erişim tarihi: 1/12/2017)

- Carmines EG, Zeller RA (1979) Reliability and Validity Assessment. Sage Publications: London.
- Cho MH, Asiaee A, Kurtek S (2019) Elastic Statistical Shape Analysis of Biological Structures with Case Studies: A Tutorial. *Bulletin of Mathematical Biology* 81: 2052-2073.
- Collins LM, Wugalter SE (2010) Latent class models for stage-sequential dynamic latent variables. *Multivariate Behavioral Research* 27: 131-157.
- Couette S, White J (2010) 3D geometric morphometrics and missing-data. Can extant taxa give clues for the analysis of fossil primates? *Comptes Rendus Palevol* 9: 423-433.
- Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B* 39: 1-38.
- Dong Y, Peng CY (2013) Principled missing data methods for researchers. *SpringerPlus* 2: 1-17.
- Dryden IL, Mardia KV (1998) *Statistical Shape Analysis*. John Wiley and Sons, Chichester.
- Dryden I (2017) shapes: Statistical shape analysis. R package version 1.2.0. <https://CRAN.R-project.org/package=shapes>
- Ercan I, Ocakoglu G, Guney I et al (2008) Adaptation of generalizability theory for inter-rater reliability for landmark localization. *International Journal of Tomography & Simulation* 9: 51-58.
- Ercan I, Ocakoglu G, Sigirli D et al (2012) Statistical shape analysis and usage in medical sciences: review. *Turkiye Klinikleri Journal of Biostatistics* 4: 27-35.
- Ercan I, Sigirli D, Ozkaya G (2015) Examining the variations in the results of the Hotelling T^2 test in case of changing baseline landmarks in the Bookstein coordinates. *Interdisciplinary Sciences: Computational Life Sciences* 7: 186-193.
- Etoz A, Ercan I (2012) Anthropometric Analysis of the Nose. Editor: PREEDY VR, *Handbook of Anthropometry: Physical Measures of Human Form in Health and Disease*. Springer, New York, pp. 919-926
- Genz A, F Bretz, T Miwa ve ark. (2017) mvtnorm: Multivariate normal and t distributions. R package version 1.0-6. URL <http://CRAN.R-project.org/package=mvtnorm>
- Graham JW (2012) *Missing Data Analysis and Design*. Springer, New York.
- He Y (2010) Missing data analysis using multiple imputation: getting to the heart of the matter. *Circulation: Cardiovascular Quality and Outcomes* 3:98.
- Henderson A (2006) Traditional morphometrics in plant systematics and its role in palm systematics. *Botanical Journal of the Linnean Society* 151: 103-111.
- Wickham H, François R, Henry L et al (2019) dplyr: A Grammar of data

manipulation. R package version 0.8.0.1. <https://CRAN.R-project.org/package=dplyr>

- Honaker J, King G, Blackwell M (2011) Amelia II: A Program for Missing Data. *Journal of Statistical Software*, 45(7), 1-47. URL <http://www.jstatsoft.org/v45/i07/>. (Erişim tarihi: 1/12/2017)
- Hunt L, Jorgensen M (2003) Mixture model clustering for mixed data with missing information. *Computational Statistics & Data Analysis* 41: 429-440.
- Huxley JS (1924) Constant differential growth ratios and their significance. *Nature* 114: 895-896.
- Huxley JS (1932) *Problems of Relative Growth*. Methuen, London.
- Jamison PL, Ward RE (1993) Brief communication: measurement size, precision, and reliability in craniofacial anthropometry: bigger is better. *American Journal of Physical Anthropology* 90: 495-500.
- Joshi S, Pizer S, Fletcher PT et al (2002) Multiscale deformable model segmentation and statistical shape analysis using medial descriptions. *IEEE Transactions on Medical Imaging* 21: 538-550
- Kendall DG (1977) The diffusion of shape. *Advances in Applied Probability* 9: 428-430.
- Kohn L, Cheverud J (1992) Calibration, validation and evaluation of scanning systems: anthropometric imaging system repeatability, electronic imaging of the human body. *Proceedings of a Working Group: CSERIAC, Dayton* pp. 114-123.
- Kramer MA (1991) Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal* 37: 233-243.
- Krim H, Yezzi AJ (2006) *Statistics and analysis of shapes. Modeling and Simulation in Science, Engineering and Technology*, Birkhauser, Boston.
- Lagler KF, Bardach JE, Miller RR (1963) *Ichthyology*. John Wiley & Sons, Michigan.
- Lele SR, Richtsmeier JT (2001) *An Invariant Approach to Statistical Analysis of Shapes*. Chapman&Hall, Florida.
- Lim KF, Foong KW (1997) Phosphor-stimulated computed cephalometry: reliability of landmark identification. *British Journal of Orthodontics* 24: 301-308.
- Little RJA, Rubin DB (1987) *Statistical Analysis with Missing Data*. John Wiley & Sons, New York.
- McWilliam JS, Welander U (1978) The effect of image quality on the identification of cephalometric landmarks. *Angle Orthodontist* 48: 49-56.
- Mitteroecker P, Gunz P (2009) Advances in geometric morphometrics. *Evolutionary Biology* 36: 235-247.
- Mosimann JE (1970) Size allometry: size and shape variables with characterizations of the lognormal and generalized gamma distributions. *Journal of the American*

Statistical Association 65: 930-945.

- Neuser R, Ackermann RR, Gain J (2009) Comparing the accuracy and precision of three techniques used for estimating missing landmarks when reconstructing fossil hominin crania. *American Journal of Physical Anthropology* 140:1–18.
- Ng HKT, Chan PS, Balakrishnan N (2002) Estimation of parameters from progressively censored data using EM algorithm. *Computational Statistics & Data Analysis* 39: 371-386.
- Nounou MN, Bakshi BR, Goel PK et al (2002) Bayesian principal component analysis. *Journal of Chemometrics* 16: 576-595.
- Oba S, Sato M, Takemasa I et al (2003) A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics* 19: 2088-2096.
- Ocakoglu G, Ercan I (2013) Traditional and modern morphometrics: review. *Turkiye Klinikleri Journal of Biostatistics* 5: 37-41.
- Ozdemir ST, Ercan I, Ozkaya G et al (2010) Geometric morphometric study and cluster analysis of late Byzantine and modern human crania. *Collegium antropologicum* 34: 353-383.
- Ozkaya G, Taskapilioglu O, Ercan I (2012) Statistical shape analysis of handwriting of patients with multiple sclerosis. *Turkiye Klinikleri Journal of Medical Sciences* 32: 1702-1709.
- Pearson K (1926) On the coefficient of racial likeness. *Biometrika* 18: 105-117.
- Pigott TD (2001) A review of methods for missing data. *Educational Research and Evaluation* 7: 353-383.
- Preda C, Saporta G, Mbarek MHBH (2010) The NIPALS algorithm for missing functional data. *Revue roumaine de mathématiques pures et appliquées* 55: 315-326.
- R Core Team (2013) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.
- Reitner J, Trauth MH, Stüwe K et al (2011) *Morphology for nonmorphometricians*. Springer, New York.
- Rohlf FJ (1999) Shape statistics: procrustes superimpositions and tangent spaces. *Journal of Classification* 16: 197-223.
- Rohlf FJ (2000) On the use of shape spaces to compare morphometric methods. *Hystrix* 11: 9-25.
- Rubin DB (1976) Inference and missing data. *Biometrika* 63: 581-592.
- Rubin DB (1987) *Multiple Imputation for Nonresponse in Surveys*. John Wiley & Sons, New York.
- Rubin DB, Thayer DT (1982) EM algorithms for ml factor analysis. *Psychometrika*

47: 69-76.

- Rubin LH, Witkiewitz K, Andre JS et al (2007) Methods for handling missing data in the behavioral neurosciences: don't throw the baby rat out with the bath water. *Journal of Undergraduate Neuroscience Education* 5: A71-A77.
- Schafer JL (1997) *Analysis of Incomplete Multivariate Data*. Chapman&Hall, New York.
- Schmitt P, Mandel J, Guedj M (2015) A comparison of six methods for missing data imputation. *Journal of Biometrics & Biostatistics* 6: 1-6.
- Scholz M, Kaplan F, Guy CL et al (2005) Non-linear PCA: a missing data approach. *Bioinformatics* 21: 3887-3895.
- Sheets HD, Covino KM, Panasiewicz JM et al (2006) Comparison of geometric morphometric outline methods in the discrimination of age-related differences in feather shape. *Frontiers in Zoology* 3: 1-12.
- Shimoda M, Yamazaki K (2015) Quality visual landmark selection based on distinctiveness and repeatability. *ROBOMECH Journal* 2: 1-10.
- Sigirli D, Ercan I (2013) Growth and allometry in modern morphometrics: review. *Turkiye Klinikleri Journal of Biostatistics* 5: 42-48.
- Sokal RR, Rohlf FJ (1995) *Biometry: The Principles and Practice of Statistics in Biological Research*. 3rd edition, W. H. Freeman and Company, New York.
- Stacklies W, Redestig H, Scholz M et al (2000) *pcaMethods* -- a Bioconductor package providing PCA methods for incomplete data. *Bioinformatics* 23: 1164-1167 (Erişim tarihi: 1/12/2017)
- Stacklies W, Redestig H, Scholz M et al (2007) *pcaMethods* - A bioconductor package providing PCA methods for incomplete data. *Bioinformatics* 23: 1164-1167.
- Tatlıdil H (2002) *Uygulamalı Çok Değişkenli İstatistiksel Analiz*. Ziraat Matbaacılık: Ankara.
- Tipping ME, Bishop CM (1999) Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 61: 611-622.
- Valeri CJ, Cole TM, Lele S et al (1998) Capturing data from threedimensional surfaces using fuzzy landmarks. *American Journal of Physical Anthropology* 107: 113-124.
- Watanabe A (2018) How many landmarks are enough to characterize shape and size variation? *PLoS ONE* 13: e0198341.
- Webster M, Sheets HD (2010) A practical introduction to landmark-based geometric morphometrics. *The Paleontological Society Papers* 16: 163-188.
- Williams FL, Richtsmeier JT (2003) Comparison of mandibular landmarks from computed tomography and 3d digitizer data. *Clinical Anatomy* 16: 163-188.
- Wu J, Brigham KG, Simon MA et al (2014) An implementation of independent

component analysis for 3d statistical shape analysis. *Biomedical Signal Processing and Control* 13: 345-356.

Xu L, Hong Y (2017) Functional and shape data analysis. *Journal of Quality Technology* 49: 419-420.

Yates KM, Untaroiu CD (2018) Finite element modeling of the human kidney for probabilistic occupant models: Statistical shape analysis and mesh morphing. *Journal of Biomechanics* 74: 50-56.

Yuan Y (2010) Multiple imputation for missing data: concepts and new development (Version 9.0). SAS Institute Inc 1-13.

Zelditch ML, Swiderski DL, Sheets HD et al (2004) *Geometric Morphometrics for Biologists*. Elsevier Academic Press, New York.

Zhou XH, Eckert GJ, Tierney WM (2001) Multiple imputation in public health research. *Statistics in Medicine* 20: 1541-1549.

7. SİMGELER VE KISALTMALAR

Temel Bileşenler Analizi: PCA

Öklid Uzaklık Matrisi Analizi: EDMA

Tamamen rastgele veri: MCAR

Rastgele veri: MAR

Rastgele olmayan kayıp veri: MNAR

Bayes yaklaşımı PCA: BPCA

Olasılıksal PCA: PPCA

Ters doğrusal olmayan PCA: NLPCA

İteratif kısmi en küçük kareler ile doğrusal olmayan tahmine dayalı PCA: INIPALS

Hata kareler ortalamasının karekökü: RMSE

8. EKLER

EK1

Simülasyon çalışmasında 3 landmarklı durumda kullanılan R kodları aşağıda belirtilmiştir.

```
#####Kullanılan paketler
```

```
library(mvtnorm)
```

```
library(shapes)
```

```
library(readxl)
```

```
library(mice)
```

```
library(pcaMethods)
```

```
library(Amelia)
```

```
library(Matrix)
```

```
library(dplyr)
```

```
##RMSE değerlerinin hesaplanmasında kullanılan fonksiyon
```

```
rmse <- function(error)
```

```
{
```

```
  sqrt(mean(error^2))
```

```
}
```

```
set.seed(100)
```

```
t=1000 #####tekrar sayısı
```

```
n=30 #####örneklem büyüklüğü
```

```
la=3 #####landmark sayısı
```

```
sayac2<-0
```

```
sayac_mr<-0
```

```
for (l in 1:t) {
```

```
  sayac_mr<-sayac_mr+1
```

```
  sayac_em<-sayac_em+1
```

```
  sayac_ppca<-sayac_ppca+1
```

```
  sayac_bpca<-sayac_bpca+1
```

```
  sayac_nlpca<-sayac_nlpca+1
```

```

sayac_nipals<-sayac_nipals+1
#####Çok deęişkenli normal daęılımdan veri türetilmesi
KOVARYANS <- read_excel("KOVARYANS.xlsx", col_names = FALSE)
KOVARYANS<-as.matrix(KOVARYANS)
KOVARYANS<-nearPD(KOVARYANS)
cov<-KOVARYANS$mat@x
dim(cov)<-c(la*2,la*2)
data<-rmvnorm(n, mean=c(644.18, 534.72, 747.81, 454.54, 641.90, 824.45, 751,
646.81, 646.18, 521.72, 534.72, 523.90), sigma=cov)
my.dat<-array(data, dim=c(la,2,n))
#####Kayıp landmark tahmininde kullanılacak landmarkların belirlenmesi
xorj<-my.dat [3,1,1]
x1<- my.dat [1,1,1]
x2<-my.dat [2,1,1]
yorj<-my.dat [3,2,1]
y1<-my.dat [1,2,1]
y2<-my.dat [2,2,1]
##### Veri setinin Bookstein koordinatlarına dönüştürülmesi
my.d<-bookstein2d(my.dat)
my.d.cor<-bookstein2d(my.dat)$bshpv
#####Kayıp landmarkın oluřtutulması
my.dat [3,1,1]<-NA
my.dat [3,2,1]<-NA
#####Kayıp landmark içeren veri seti üzerinden Bookstein koordinatlarının
oluřturulması
my.dat.book<-bookstein2d(my.dat)
my.dat.book.cor<-my.dat.book$bshpv
veri<-my.dat.book.cor
#####Veri setinin F istatistięini uygulamak için düzenlenmesi
i=1
yeni<-data.frame(matrix(nrow = n, ncol = 2))

```

```

colnames(yeni)<-c("x", "y")
for (i in 1:n) {
yeni[i,1]<-veri[3,1,i]
yeni[i,2]<-veri[3,2,i]
}
#####F yaklaşımı algoritmasının uygulanması:
i=2
d31<-matrix(nrow = n, ncol = 1)
for (i in 2:n) {
deneme2=my.dat.book.cor[:,i]
xk1<-deneme2[1,1]
yk1<-deneme2[1,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d31[i,1]=sqrt((xk3-xk1)^2+(yk3-yk1)^2)#####Öklid uzaklığının hesaplanması
}
d31<-d31[-1,]
d31ort=mean(d31)
i=2
d32<-data.frame(matrix(nrow = n, ncol = 1))
for (i in 2:n) {
deneme2=my.dat.book.cor[:,i]
deneme2<-deneme2[1:3,]
xk2<-deneme2[2,1]
yk2<-deneme2[2,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d32[i,1]=sqrt((xk3-xk2)^2+(yk3-yk2)^2) #####Öklid uzaklığının hesaplanması
}
d32<-d32[-1,]
d32ort=mean(d32)

```

```

d32ort
c=(-1/2)*(d31ort^2-d32ort^2)
d=sqrt(d31ort^2-(c+0.5)^2)
y4<-yeni
y4<-as.matrix(y4)
y4[1,1]<-c
y4[1,2]<-d
####Güven aralıklarının hesaplanması:
d2<-d31
d2as<-d31ort-1.96*(sd(d2)/sqrt(length(d2))) #### d2 alt sınır
d2üs<- d31ort +1.96*(sd(d2)/sqrt(length(d2))) #### d2 üst sınır
d3<-d32
d3as<- d32ort -1.96*(sd(d3)/sqrt(length(d3))) #### d3 alt sınır
d3üs<- d32ort+1.96*(sd(d3)/sqrt(length(d3))) #### d3 üst sınır
closeAllConnections()
sayac<-0
fstatt<-matrix(byrow=TRUE)
ls<-matrix(byrow=TRUE)
sayacs<-matrix(byrow=TRUE)
cs<-matrix(byrow=TRUE)
ds<-matrix(byrow=TRUE)
closeAllConnections()
####F istatistiğinin hesaplanması
for ( c in seq(d2as, d2üs, 0.2) ) {
for ( d in seq(d3as, d3üs, 0.2) ) {
sayac<-sayac+1
y4[1,1]<-c
y4[1,2]<-d
gakt=n*(mean(y4[,1])-mean(y4))^2+n*(mean(y4[,2])-mean(y4))^2
cat(gakt, sep="\n", file="gakt.txt", append=TRUE)
sink("gkt.txt")

```



```

k=2
for( i in 1:k){
for( j in 1:n) {
t=((y4[j,i]-mean(y4))^2)
cat(t, sep="\n", file="gkt.txt", append=TRUE)
j=j+1
}
i=i+1
}
gkt <- read.table("gkt.txt")
gktson=sum(gkt)
gikt=gktson-gakt
fstat=((gakt/(k-1))/(gikt/(2*n-k)))
fstatt[sayac]<-fstat
closeAllConnections()
ls[sayac]<-l
sayacs[sayac]<-sayac
cs[sayac]<-c
ds[sayac]<-d
}
closeAllConnections()
}
#####Çoklu regresyon atama yöntemi ile kayıp değer tahmini
mr_veri<-yeni
xt_mrs<-matrix(byrow=TRUE)
yt_mrs<-matrix(byrow=TRUE)
colnames(mr_veri)<-c("v1", "v2")
imputed_Data <- mice(mr_veri, method="norm")
xt_mr=imputed_Data$imp$v1[1,1]#####Tahmin edilen x koordinatı
yt_mr=imputed_Data$imp$v2[1,1]#####Tahmin edilen y koordinatı
xt_mrs[sayac_mr]=xt_mr

```

```

yt_mrs[sayac_mr]=yt_mr
#####Min(F) ölçütü için sonuçların düzenlenmesi
ts<-rbind(ls, sayacs, cs, ds, fstatt)
ts<-t(ts)
ts<-as.data.frame(ts)
colnames(ts)<-c("L", "SAYAC", "X", "Y", "F")
minn<-ts %>%
group_by(L) %>%
slice(which.min(F))
maxx<-ts %>%
group_by(L) %>%
slice(which.max(F))
#####Min(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
D=(x2-x1)^2+(y2-y1)^2
A=x2-x1
B=y2-y1
ub<- minn$X
vb<- minn$Y
C= (ub+0.5)
yj2={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinatı
yj<- yorj
xj2={C*D+A*x1-B*yj+B*y1}/A ##### Tahmin edilen x koordinatı
xj2<-as.numeric(unlist(xj2))
yj2<-as.numeric(unlist(yj2))
zorj=sqrt(xorj^2+yorj^2)#####Orijinal hipotenüs değeri
ztah_min=sqrt(xj2^2+yj2^2)#####MinF ölçütüne göre hipotenüs değeri
error_min=ztah_min-zorj
error_min<-as.numeric(unlist(error_min))
cat(error_min, sep="\n", file="error min.txt", append=TRUE)
#####Max(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri dönme

```

```

ubm<- maxx$X
vbm<- maxx$Y
C= (ubm+0.5)
ym={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinati
yj<- yorj
xm={C*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xm<-as.numeric(unlist(xm))
ym<-as.numeric(unlist(ym))
ztah_max=sqrt(xm^2+ym^2)#####Max(F) ölçütü ile elde edilen hipotenüs değeri
error_max=ztah_max-zorj
error_max<-as.numeric(unlist(error_max))
cat(error_max, sep="\n", file="error max.txt", append=TRUE)
closeAllConnections()
#####Çoklu regresyon atama yöntemi için Bookstein koordinatlarından orijinal
koordinatlara geri dönme
ub_mr<- xt_mr
vb_mr<- yt_mr
C_mr= (ub_mr+0.5)
yj2_mr={(A^2+B^2)*y1+B*C_mr*D+vb*A*D}/(A^2+B^2)#####Tahmin edilen
ykoordinati
yj<- yorj
xj2_mr={C_mr*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xj2_mr<-as.numeric(unlist(xj2_mr))
yj2_mr<-as.numeric(unlist(yj2_mr))
ztah_mr=sqrt(xj2_mr^2+yj2_mr^2)#####Çoklu regresyon atama yöntemi ile elde
edilen hipotenüs değeri
error_mr=ztah_mr-zorj
error_mr<-as.numeric(unlist(error_mr))
cat(error_mr, sep="\n", file="error mr.txt", append=TRUE)
}
error_min<- read.table("error min.txt")
error_min<-as.numeric(unlist(error_min))

```

```
rmse_f_min<-rmse(error_min)#####Min(F) ölçütü için RMSE değeri
error_max<- read.table("error max.txt")
error_max<-as.numeric(unlist(error_max))
rmse_f_max<-rmse(error_max) #####Max(F) ölçütü için RMSE değeri
error_mr<- read.table("error mr.txt")
error_mr<-as.numeric(unlist(error_mr))
rmse_mr<-rmse(error_mr) #####Çoklu regresyon atama yöntemi için RMSE değeri
```



Simülasyon çalışmasında 6 landmarklı durumda kullanılan R kodları aşağıda belirtilmiştir.

```
#####Kullanılan paketler
library(mvtnorm)
library(shapes)
library(readxl)
library(mice)
library(pcaMethods)
library(Amelia)
library(Matrix)
library(dplyr)
##RMSE değerlerinin hesaplanmasında kullanılan fonksiyon
rmse <- function(error)
{
  sqrt(mean(error^2))
}
set.seed(100)
t=1000 #####tekrar sayısı
n=30#####örneklem büyüklüğü
n1=n
n2=n1+1
n3=2*n1
n4=n3+1
n5=3*n1
n6=n5+1
n7=4*n1
la=6#####landmark sayısı
l=1
sayac2<-0
sayac_mr<-0
```

```

sayac_em<-0
sayac_ppca<-0
sayac_bpca<-0
sayac_nlpca<-0
sayac_nipals<-0
for (l in 1:t) {
sayac_mr<-sayac_mr+1
sayac_em<-sayac_em+1
sayac_ppca<-sayac_ppca+1
sayac_bpca<-sayac_bpca+1
sayac_nlpca<-sayac_nlpca+1
sayac_nipals<-sayac_nipals+1 #####Çok deęişkenli normal daęılımdan veri türetilmesi
KOVARYANS <- read_excel("KOVARYANS.xlsx", col_names = FALSE)
KOVARYANS<-as.matrix(KOVARYANS)
KOVARYANS<-nearPD(KOVARYANS)
cov<-KOVARYANS$mat@x
dim(cov)<-c(la*2,la*2)

data<-rmvnorm(n, mean=c(644.18, 534.72, 747.81, 454.54, 641.90, 824.45, 751,
646.81, 646.18, 521.72, 534.72, 523.90), sigma=cov)
my.dat<-array(data, dim=c(la,2,n))

#####Kayıp landmark tahmininde kullanılacak landmarkların belirlenmesi
xorj<-my.dat [3,1,1]
x1<- my.dat [1,1,1]
x2<-my.dat [2,1,1]
yorj<-my.dat [3,2,1]
y1<-my.dat [1,2,1]
y2<-my.dat [2,2,1]

##### Veri setinin Bookstein koordinatlarına dönüştürülmesi
my.d<-bookstein2d(my.dat)
my.d.cor<-bookstein2d(my.dat)$bshpv

#####Kayıp landmarkın oluřtutulması

```

```

my.dat [3,1,1]<-NA
my.dat [3,2,1]<-NA
#####Kayıp landmark içeren veri seti üzerinden Bookstein koordinatlarının
oluşturulması
my.dat.book<-bookstein2d(my.dat)
my.dat.book.cor<-my.dat.book$bshpv
veri<-my.dat.book.cor
#####Veri setinin F istatistiğini uygulamak için düzenlenmesi
i=1
yeni<-data.frame(matrix(nrow = n, ncol = 2))
colnames(yeni)<-c("x", "y")
for (i in 1:n) {
yeni[i,1]<-veri[3,1,i]
yeni[i,2]<-veri[3,2,i]
}
#####F yaklaşımı algoritmasının uygulanması:
i=2
d31<-matrix(nrow = n, ncol = 1)
for (i in 2:n) {
deneme2=my.dat.book.cor[,i]
xk1<-deneme2[1,1]
yk1<-deneme2[1,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d31[i,1]=sqrt((xk3-xk1)^2+(yk3-yk1)^2)#####Öklid uzaklığının hesaplanması
}
d31<-d31[-1,]
d31ort=mean(d31)
i=2
d32<-data.frame(matrix(nrow = n, ncol = 1))
for (i in 2:n) {

```

```

deneme2=my.dat.book.cor[:,i]
deneme2<-deneme2[1:3,]
xk2<-deneme2[2,1]
yk2<-deneme2[2,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d32[i,1]=sqrt((xk3-xk2)^2+(yk3-yk2)^2) #####Öklid uzaklığının hesaplanması
}
d32<-d32[-1,]
d32ort=mean(d32)
d32ort
c=(-1/2)*(d31ort^2-d32ort^2)
d=sqrt(d31ort^2-(c+0.5)^2)
y4<-yeni
y4<-as.matrix(y4)
y4[1,1]<-c
y4[1,2]<-d
#####Güven aralıklarının hesaplanması:
d2<-d31
d2as<-d31ort-1.96*(sd(d2)/sqrt(length(d2))) ##### d2 alt sınır
d2üs<- d31ort +1.96*(sd(d2)/sqrt(length(d2))) ##### d2 üst sınır
d3<-d32
d3as<- d32ort -1.96*(sd(d3)/sqrt(length(d3))) ##### d3 alt sınır
d3üs<- d32ort+1.96*(sd(d3)/sqrt(length(d3))) ##### d3 alt sınır
closeAllConnections()
sayac<-0
fstatt<-matrix(byrow=TRUE)
ls<-matrix(byrow=TRUE)
sayacs<-matrix(byrow=TRUE)
cs<-matrix(byrow=TRUE)
ds<-matrix(byrow=TRUE)

```



```

closeAllConnections()
#####F istatistiğinin hesaplanması
for ( c in seq(d2as, d2üs, 0.2) ) {
for ( d in seq(d3as, d3üs, 0.2) ) {
sayac<-sayac+1
y4[1,1]<-c
y4[1,2]<-d
gakt=n*(mean(y4[,1])-mean(y4))^2+n*(mean(y4[,2])-mean(y4))^2
cat(gakt, sep="\n", file="gakt.txt", append=TRUE)
sink("gkt.txt")
k=2
for( i in 1:k){
for( j in 1:n) {
t=((y4[j,i]-mean(y4))^2)
cat(t, sep="\n", file="gkt.txt", append=TRUE)
j=j+1
}
i=i+1
}
gkt <- read.table("gkt.txt")
gktson=sum(gkt)
gikt=gktson-gakt
fstat=((gakt/(k-1))/(gikt/(2*n-k)))
fstatt[sayac]<-fstat
closeAllConnections()
ls[sayac]<-l
sayacs[sayac]<-sayac
cs[sayac]<-c
ds[sayac]<-d
}
closeAllConnections()

```

```

}
#####Çoklu regresyon atama yöntemi ile kayıp değer tahmini
mr_veri<-yeni
xt_mrs<-matrix(byrow=TRUE)
yt_mrs<-matrix(byrow=TRUE)
colnames(mr_veri)<-c("v1", "v2")
imputed_Data <- mice(mr_veri, method="norm")
xt_mr=imputed_Data$simp$v1[1,1]
yt_mr=imputed_Data$simp$v2[1,1]
xt_mrs[sayac_mr]=xt_mr
yt_mrs[sayac_mr]=yt_mr
#####EM algoritması ile kayıp değer tahmini
k=6
j=1
sink("yenix.xls")
while (j <=n) {
i=3
while (i <= k) {
aa<-veri[i,1,j]
cat(aa, sep="\n")
i=i+1
}
j=j+1
}
j=1
sink("yeniy.xls")
while (j <=n) {
i=3
while (i <= k) {
bb<-veri[i,2,j]
cat(bb, sep="\n")

```

```

i=i+1
}
j=j+1
}
yenix<-read.table("yenix.xls")
yeniy<-read.table ("yeniy.xls")
yenix<-as.matrix(yenix)
yeniy<-as.matrix(yeniy)
yeni_em<-cbind(yenix[1:n1,], yeniy[1:n1,], yenix[n2:n3,], yeniy[n2:n3,],
yenix[n4:n5,], yeniy[n4:n5,], yenix[n6:n7,], yeniy[n6:n7,])
result_em<-amelia(yeni_em, m = 1)
xt_ems<-matrix(byrow=TRUE)
yt_ems<-matrix(byrow=TRUE)
xt_em<-result_em$imputations$imp1[1,1]
yt_em<-result_em$imputations$imp1[1,2]
xt_ems[sayac_em]=xt_em
yt_ems[sayac_em]=yt_em
#####PCA tabanlı yöntemler ile kayıp değer tahmini
md <- prep(yeni_em, scale="none", center=TRUE)
xt_PPcAs<-matrix(byrow=TRUE)
yt_PPcAs<-matrix(byrow=TRUE)
resPPCA <- pca(md, method="ppca")
xt_PPcA<-resPPCA@completeObs[1,1]
yt_PPcA<-resPPCA@completeObs[1,2]
xt_PPcAs[sayac_ppca]=xt_PPcA
yt_PPcAs[sayac_ppca]=yt_PPcA
xt_BPCAs<-matrix(byrow=TRUE)
yt_BPCAs<-matrix(byrow=TRUE)
resBPCA <- pca(md, method="bpca")
xt_BPCA<-resBPCA@completeObs[1,1]
yt_BPCA<-resBPCA@completeObs[1,2]

```

```

xt_BPCAs[sayac_bpca]=xt_BPCA
yt_BPCAs[sayac_bpca]=yt_BPCA
xt_Nipalss<-matrix(byrow=TRUE)
yt_Nipalss<-matrix(byrow=TRUE)
resNipals <- pca(md, method="nipals")
xt_Nipals<-resNipals@completeObs[1,1]
yt_Nipals<-resNipals@completeObs[1,2]
xt_Nipalss[sayac_nipals]=xt_Nipals
yt_Nipalss[sayac_nipals]=yt_Nipals
xt_NLPCAs<-matrix(byrow=TRUE)
yt_NLPCAs<-matrix(byrow=TRUE)
resNLPCA <- pca(md, method="nlpca")
xt_NLPCA<-resNLPCA@completeObs[1,1]
yt_NLPCA<-resNLPCA@completeObs[1,2]
xt_NLPCAs[sayac_nlpca]=xt_NLPCA
yt_NLPCAs[sayac_nlpca]=yt_NLPCA
#####Min(F) ölçütü için sonuçların düzenlenmesi
ts<-rbind(ls, sayacs, cs, ds, fstatt)
ts<-t(ts)
ts<-as.data.frame(ts)
colnames(ts)<-c("L", "SAYAC", "X", "Y", "F")
minn<-ts %>%
group_by(L) %>%
slice(which.min(F))
maxx<-ts %>%
group_by(L) %>%
slice(which.max(F))
#####Min(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
D=(x2-x1)^2+(y2-y1)^2
A=x2-x1

```

```

B=y2-y1
ub<- minn$X
vb<- minn$Y
C= (ub+0.5)
yj2={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinati
yj<- yorj
xj2={C*D+A*x1-B*yj+B*y1}/A ##### Tahmin edilen x koordinati
xj2<-as.numeric(unlist(xj2))
yj2<-as.numeric(unlist(yj2))
zorj=sqrt(xorj^2+yorj^2)#####Orijinal hipotenüs deęeri
ztah_min=sqrt(xj2^2+yj2^2)#####MinF ölçütüne göre hipotenüs deęeri
error_min=ztah_min-zorj
error_min<-as.numeric(unlist(error_min))
cat(error_min, sep="\n", file="error min.txt", append=TRUE)
#####Max(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ubm<- maxx$X
vbm<- maxx$Y
C= (ubm+0.5)
ym={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinati
yj<- yorj
xm={C*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xm<-as.numeric(unlist(xm))
ym<-as.numeric(unlist(ym))
ztah_max=sqrt(xm^2+ym^2)#####Max(F) ölçütü ile elde edilen hipotenüs deęeri
error_max=ztah_max-zorj
error_max<-as.numeric(unlist(error_max))
cat(error_max, sep="\n", file="error max.txt", append=TRUE)
closeAllConnections()
#####Çoklu regresyon atama yöntemi için Bookstein koordinatlarından orijinal
koordinatlara geri dönme:

```

```

ub_mr<- xt_mr
vb_mr<- yt_mr
C_mr= (ub_mr+0.5)
yj2_mr={(A^2+B^2)*y1+B*C_mr*D+vb*A*D}/(A^2+B^2)#####Tahmin edilen
ykoordinatı
yj<- yorj
xj2_mr={C_mr*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_mr<-as.numeric(unlist(xj2_mr))
yj2_mr<-as.numeric(unlist(yj2_mr))
ztah_mr=sqrt(xj2_mr^2+yj2_mr^2)#####Çoklu regresyon atama yöntemi ile elde
edilen hipotenüs değeri
error_mr=ztah_mr-zorj
error_mr<-as.numeric(unlist(error_mr))
cat(error_mr, sep="\n", file="error mr.txt", append=TRUE)
#####EM algoritması için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub<- xt_em
vb<- yt_em
C= (ub+0.5)
yj2_em={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y
koordinatı
yj<- yorj
xj2_em={C*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_em<-as.numeric(unlist(xj2_em))
yj2_em<-as.numeric(unlist(yj2_em))
ztah_em=sqrt(xj2_em^2+yj2_em^2)
error_em=ztah_em-zorj
error_em<-as.numeric(unlist(error_em))
cat(error_em, sep="\n", file="error em.txt", append=TRUE)
#####BPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme
ub_BPCA<- xt_BPCA
vb_BPCA<- yt_BPCA

```

```

C_BPCA= (ub_BPCA+0.5)
yj2_BPCA={(A^2+B^2)*y1+B*C_BPCA*D+vb_BPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinati
yj<- yorj
xj2_BPCA={C_BPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xj2_BPCA<-as.numeric(unlist(xj2_BPCA))
yj2_BPCA<-as.numeric(unlist(yj2_BPCA))
ztah_BPCA=sqrt(xj2_BPCA^2+yj2_BPCA^2)
error_BPCA=ztah_BPCA-zorj
error_BPCA<-as.numeric(unlist(error_BPCA))
cat(error_BPCA, sep="\n", file="error BPCA.txt", append=TRUE)
#####PPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_PPCCA<- xt_PPCCA
vb_PPCCA<- yt_PPCCA
C_PPCCA= (ub_PPCCA+0.5)
yj2_PPCCA={(A^2+B^2)*y1+B*C_PPCCA*D+vb_PPCCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinati
yj<- yorj
xj2_PPCCA={C_PPCCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xj2_PPCCA<-as.numeric(unlist(xj2_PPCCA))
yj2_PPCCA<-as.numeric(unlist(yj2_PPCCA))
ztah_PPCCA=sqrt(xj2_PPCCA^2+yj2_PPCCA^2)
error_PPCCA=ztah_PPCCA-zorj
error_PPCCA<-as.numeric(unlist(error_PPCCA))
cat(error_PPCCA, sep="\n", file="error PPCCA.txt", append=TRUE)
#####NLPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_NLPCA<- xt_NLPCA
vb_NLPCA<- yt_NLPCA
C_NLPCA= (ub_NLPCA+0.5)
yj2_NLPCA={(A^2+B^2)*y1+B*C_NLPCA*D+vb_NLPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinati

```

```

yj<- yorj
xj2_NLPCA={C_NLPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xj2_NLPCA<-as.numeric(unlist(xj2_NLPCA))
yj2_NLPCA<-as.numeric(unlist(yj2_NLPCA))
ztah_NLPCA=sqrt(xj2_NLPCA^2+yj2_NLPCA^2)
error_NLPCA=ztah_NLPCA-zorj
error_NLPCA<-as.numeric(unlist(error_NLPCA))
cat(error_NLPCA, sep="\n", file="error NLPCA.txt", append=TRUE)
#####INIPALS yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme
ub_Nipals<- xt_Nipals
vb_Nipals<- yt_Nipals
C_Nipals= (ub_Nipals+0.5)
yj2_Nipals={(A^2+B^2)*y1+B*C_Nipals*D+vb_Nipals*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinati
yj<- yorj
xj2_Nipals={C_Nipals*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinati
xj2_Nipals<-as.numeric(unlist(xj2_Nipals))
yj2_Nipals<-as.numeric(unlist(yj2_Nipals))
ztah_Nipals=sqrt(xj2_Nipals^2+yj2_Nipals^2)
error_Nipals=ztah_Nipals-zorj
error_Nipals<-as.numeric(unlist(error_Nipals))
cat(error_Nipals, sep="\n", file="error Nipals.txt", append=TRUE)
}
error_min<- read.table("error min.txt")
error_min<-as.numeric(unlist(error_min))
rmse_f_min<-rmse(error_min)#####Min(F) ölçütü için RMSE değeri
error_max<- read.table("error max.txt")
error_max<-as.numeric(unlist(error_max))
rmse_f_max<-rmse(error_max) #####Max(F) ölçütü için RMSE değeri
error_mr<- read.table("error mr.txt")

```



```
error_mr<-as.numeric(unlist(error_mr))
rmse_mr<-rmse(error_mr) #####Çoklu regresyon atama yöntemi için RMSE değeri
error_em<- read.table("error em.txt")
error_em<-as.numeric(unlist(error_em))
rmse_em<-rmse(error_em) #####EM algoritması için RMSE değeri
error_BPCA<- read.table("error BPCA.txt")
error_BPCA<-as.numeric(unlist(error_BPCA))
rmse_BPCA<-rmse(error_BPCA) #####BPCA yöntemi için RMSE değeri
error_PPCA<- read.table("error PPCA.txt")
error_PPCA<-as.numeric(unlist(error_PPCA))
rmse_PPCA<-rmse(error_PPCA) #####PPCA yöntemi için RMSE değeri
error_NLPCA<- read.table("error NLPCA.txt")
error_NLPCA<-as.numeric(unlist(error_NLPCA))
rmse_NLPCA<-rmse(error_NLPCA) #####NLPCA yöntemi için RMSE değeri
error_Nipals<- read.table("error Nipals.txt")
error_Nipals<-as.numeric(unlist(error_Nipals))
rmse_Nipals<-rmse(error_Nipals) #####INIPALS yöntemi için RMSE değeri
```

Simülasyon çalışmasında 9 landmarklı durumda kullanılan R kodları aşağıda belirtilmiştir.

```
#####Kullanılan paketler
library(mvtnorm)
library(shapes)
library(readxl)
library(mice)
library(pcaMethods)
library(Amelia)
library(Matrix)
library(dplyr)
##RMSE değerlerinin hesaplanmasında kullanılan fonksiyon
rmse <- function(error)
{
  sqrt(mean(error^2))
}
set.seed(100)
t=1000 #####tekrar sayısı
n=30#####örneklem büyüklüğü
n1=n
n2=n1+1
n3=2*n1
n4=n3+1
n5=3*n1
n6=n5+1
n7=4*n1
n8=n7+1
n9=5*n1
n10=n9+1
n11=6*n1
```

```

n12=n11+1
n13=7*n1
la=9####landmark sayısı
l=1
sayac2<-0
sayac_mr<-0
sayac_em<-0
sayac_ppca<-0
sayac_bpca<-0
sayac_nlpca<-0
sayac_nipals<-0
for (l in 1:t) {
sayac_mr<-sayac_mr+1
  sayac_em<-sayac_em+1
  sayac_ppca<-sayac_ppca+1
  sayac_bpca<-sayac_bpca+1
  sayac_nlpca<-sayac_nlpca+1
  sayac_nipals<-sayac_nipals+1 ####Çok değişkenli normal dağılımdan veri
  türetilmesi
KOVARYANS <- read_excel("KOVARYANS.xlsx", col_names = FALSE)
KOVARYANS<-as.matrix(KOVARYANS)
KOVARYANS<-nearPD(KOVARYANS)
cov<-KOVARYANS$mat@x
dim(cov)<-c(la*2,la*2)
data<-rmvnorm(n, mean=c(644.18, 534.72, 747.81, 454.54, 641.90, 824.45, 533.90,
743, 643.18, 751, 646.81, 646.18, 521.72, 534.72, 523.90, 385.81, 390.54, 312.90),
sigma=cov)
my.dat<-array(data, dim=c(la,2,n))
####Kayıp landmark tahmininde kullanılacak landmarkların belirlenmesi
xorj<-my.dat [3,1,1]
x1<- my.dat [1,1,1]
x2<-my.dat [2,1,1]

```

```

yorj<-my.dat [3,2,1]
y1<-my.dat [1,2,1]
y2<-my.dat [2,2,1]
#### Veri setinin Bookstein koordinatlarına dönüştürülmesi
my.d<-bookstein2d(my.dat)
my.d.cor<-bookstein2d(my.dat)$bshpv
####Kayıp landmarkın oluşturulması
my.dat [3,1,1]<-NA
my.dat [3,2,1]<-NA
####Kayıp landmark içeren veri seti üzerinden Bookstein koordinatlarının
oluşturulması
my.dat.book<-bookstein2d(my.dat)
my.dat.book.cor<-my.dat.book$bshpv
veri<-my.dat.book.cor
####Veri setinin F istatistiğini uygulamak için düzenlenmesi
i=1
yeni<-data.frame(matrix(nrow = n, ncol = 2))
colnames(yeni)<-c("x", "y")
for (i in 1:n) {
yeni[i,1]<-veri[3,1,i]
yeni[i,2]<-veri[3,2,i]
}
####F yaklaşımı algoritmasının uygulanması
i=2
d31<-matrix(nrow = n, ncol = 1)
for (i in 2:n) {
deneme2=my.dat.book.cor[,i]
xk1<-deneme2[1,1]
yk1<-deneme2[1,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]

```

```

d31[i,1]=sqrt((xk3-xk1)^2+(yk3-yk1)^2)#####Öklid uzaklığının hesaplanması
}
d31<-d31[-1,]
d31ort=mean(d31)
i=2
d32<-data.frame(matrix(nrow = n, ncol = 1))
for (i in 2:n) {
deneme2=my.dat.book.cor[:,i]
deneme2<-deneme2[1:3,]
xk2<-deneme2[2,1]
yk2<-deneme2[2,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d32[i,1]=sqrt((xk3-xk2)^2+(yk3-yk2)^2) #####Öklid uzaklığının hesaplanması
}
d32<-d32[-1,]
d32ort=mean(d32)
d32ort
c=(-1/2)*(d31ort^2-d32ort^2)
d=sqrt(d31ort^2-(c+0.5)^2)
y4<-yeni
y4<-as.matrix(y4)
y4[1,1]<-c
y4[1,2]<-d
#####Güven aralıklarının hesaplanması:
d2<-d31
d2as<-d31ort-1.96*(sd(d2)/sqrt(length(d2))) ##### d2 alt sınır
d2üs<- d31ort +1.96*(sd(d2)/sqrt(length(d2))) ##### d2 üst sınır
d3<-d32
d3as<- d32ort -1.96*(sd(d3)/sqrt(length(d3))) ##### d3 alt sınır
d3üs<- d32ort+1.96*(sd(d3)/sqrt(length(d3))) ##### d3 üst sınır

```

```

closeAllConnections()
sayac<-0
fstatt<-matrix(byrow=TRUE)
ls<-matrix(byrow=TRUE)
sayacs<-matrix(byrow=TRUE)
cs<-matrix(byrow=TRUE)
ds<-matrix(byrow=TRUE)
closeAllConnections()
#####F istatistiğinin hesaplanması
for ( c in seq(d2as, d2üs, 0.2) ) {
for ( d in seq(d3as, d3üs, 0.2) ) {
sayac<-sayac+1
y4[1,1]<-c
y4[1,2]<-d
gakt=n*(mean(y4[,1])-mean(y4))^2+n*(mean(y4[,2])-mean(y4))^2
cat(gakt, sep="\n", file="gakt.txt", append=TRUE)
sink("gkt.txt")
k=2
for( i in 1:k){
for( j in 1:n) {
t=((y4[j,i]-mean(y4))^2)
cat(t, sep="\n", file="gkt.txt", append=TRUE)
j=j+1
}
i=i+1
}
gkt <- read.table("gkt.txt")
gktson=sum(gkt)
gikt=gktson-gakt
fstat=((gakt/(k-1))/(gikt/(2*n-k)))
fstatt[sayac]<-fstat

```

```

closeAllConnections()
ls[sayac]<-l
sayacs[sayac]<-sayac
cs[sayac]<-c
ds[sayac]<-d
}
closeAllConnections()
}
####Çoklu regresyon atama yöntemi ile kayıp değer tahmini
mr_veri<-yeni
xt_mrs<-matrix(byrow=TRUE)
yt_mrs<-matrix(byrow=TRUE)
colnames(mr_veri)<-c("v1", "v2")
imputed_Data <- mice(mr_veri, method="norm")
xt_mr=imputed_Data$imp$v1[1,1]
yt_mr=imputed_Data$imp$v2[1,1]
xt_mrs[sayac_mr]=xt_mr
yt_mrs[sayac_mr]=yt_mr
####EM algoritması ile kayıp değer tahmini
k=9
j=1
sink("yenix.xls")
while (j <=n) {
i=3
while (i <= k) {
aa<-veri[i,1,j]
cat(aa, sep="\n")
i=i+1
}
j=j+1
}

```

```

j=1
sink("yeni.xls")
while (j <=n) {
i=3
while (i <= k) {
bb<-veri[i,2,j]
cat(bb, sep="\n")
i=i+1
}
j=j+1
}
yenix<-read.table("yenix.xls")
yeni<-read.table ("yeni.xls")
yenix<-as.matrix(yenix)
yeni<-as.matrix(yeni)
yeni_em<-cbind(yenix[1:n1,], yeni[1:n1,], yenix[n2:n3,], yeni[n2:n3,],
yenix[n4:n5,], yeni[n4:n5,], yenix[n6:n7,], yeni[n6:n7,],yenix[n8:n9,],
yeni[n8:n9,], yenix[n10:n11,], yeni[n10:n11,], yenix[n12:n13,], yeni[n12:n13,])
result_em<-amelia(yeni_em, m = 1)
xt_ems<-matrix(byrow=TRUE)
yt_ems<-matrix(byrow=TRUE)
xt_em<-result_em$imputations$imp1[1,1]
yt_em<-result_em$imputations$imp1[1,2]
xt_ems[sayac_em]=xt_em
yt_ems[sayac_em]=yt_em
####PCA tabanlı yöntemler ile kayıp değer tahmini
md <- prep(yeni_em, scale="none", center=TRUE)
xt_PPcAs<-matrix(byrow=TRUE)
yt_PPcAs<-matrix(byrow=TRUE)
resPPCA <- pca(md, method="ppca")
xt_PPCA<-resPPCA@completeObs[1,1]

```



```

yt_PPCA<-resPPCA@completeObs[1,2]
xt_PPcAs[sayac_ppca]=xt_PPCA
yt_PPcAs[sayac_ppca]=yt_PPCA
xt_BPCAs<-matrix(byrow=TRUE)
yt_BPCAs<-matrix(byrow=TRUE)
resBPCA <- pca(md, method="bpca")
xt_BPCA<-resBPCA@completeObs[1,1]
yt_BPCA<-resBPCA@completeObs[1,2]
xt_BPCAs[sayac_bpca]=xt_BPCA
yt_BPCAs[sayac_bpca]=yt_BPCA
xt_Nipalss<-matrix(byrow=TRUE)
yt_Nipalss<-matrix(byrow=TRUE)
resNipals <- pca(md, method="nipals")
xt_Nipals<-resNipals@completeObs[1,1]
yt_Nipals<-resNipals@completeObs[1,2]
xt_Nipalss[sayac_nipals]=xt_Nipals
yt_Nipalss[sayac_nipals]=yt_Nipals
xt_NLPCAs<-matrix(byrow=TRUE)
yt_NLPCAs<-matrix(byrow=TRUE)
resNLPCA <- pca(md, method="nlpca")
xt_NLPCA<-resNLPCA@completeObs[1,1]
yt_NLPCA<-resNLPCA@completeObs[1,2]
xt_NLPCAs[sayac_nlpca]=xt_NLPCA
yt_NLPCAs[sayac_nlpca]=yt_NLPCA
####Min(F) ölçütü için sonuçların düzenlenmesi
ts<-rbind(ls, sayacs, cs, ds, fstatt)
ts<-t(ts)
ts<-as.data.frame(ts)
colnames(ts)<-c("L", "SAYAC", "X", "Y", "F")
minn<-ts %>%
group_by(L) %>%

```

```

slice(which.min(F))
maxx<-ts %>%
group_by(L) %>%
slice(which.max(F))
#####Min(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri dönme
D=(x2-x1)^2+(y2-y1)^2
A=x2-x1
B=y2-y1
ub<- minn$X
vb<- minn$Y
C= (ub+0.5)
yj2={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinatı
yj<- yorj
xj2={C*D+A*x1-B*yj+B*y1}/A ##### Tahmin edilen x koordinatı
xj2<-as.numeric(unlist(xj2))
yj2<-as.numeric(unlist(yj2))
zorj=sqrt(xorj^2+yorj^2)#####Orijinal hipotenüs değeri
ztah_min=sqrt(xj2^2+yj2^2)#####MinF ölçütüne göre hipotenüs değeri
error_min=ztah_min-zorj
error_min<-as.numeric(unlist(error_min))
cat(error_min, sep="\n", file="error min.txt", append=TRUE)
#####Max(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri dönme
ubm<- maxx$X
vbm<- maxx$Y
C= (ubm+0.5)
ym={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinatı
yj<- yorj
xm={C*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xm<-as.numeric(unlist(xm))
ym<-as.numeric(unlist(ym))
ztah_max=sqrt(xm^2+ym^2)#####Max(F) ölçütü ile elde edilen hipotenüs değeri

```

```

error_max=ztah_max-zorj
error_max<-as.numeric(unlist(error_max))
cat(error_max, sep="\n", file="error max.txt", append=TRUE)
closeAllConnections()
#####Çoklu regresyon atama yöntemi için Bookstein koordinatlarından orijinal
koordinatlara geri dönme
ub_mr<- xt_mr
vb_mr<- yt_mr
C_mr= (ub_mr+0.5)
yj2_mr={(A^2+B^2)*y1+B*C_mr*D+vb*A*D}/(A^2+B^2)#####Tahmin edilen
ykoordinatı
yj<- yorj
xj2_mr={ C_mr*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_mr<-as.numeric(unlist(xj2_mr))
yj2_mr<-as.numeric(unlist(yj2_mr))
ztah_mr=sqrt(xj2_mr^2+yj2_mr^2)#####Çoklu regresyon atama yöntemi ile elde
edilen hipotenüs değeri
error_mr=ztah_mr-zorj
error_mr<-as.numeric(unlist(error_mr))
cat(error_mr, sep="\n", file="error mr.txt", append=TRUE)
#####EM algoritması için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub<- xt_em
vb<- yt_em
C= (ub+0.5)
yj2_em={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) ## tahmin değeri
yj<- yorj
xj2_em={ C*D+A*x1-B*yj+B*y1}/A ## tahmin değeri
xj2_em<-as.numeric(unlist(xj2_em))
yj2_em<-as.numeric(unlist(yj2_em))
ztah_em=sqrt(xj2_em^2+yj2_em^2)
error_em=ztah_em-zorj

```

```

error_em<-as.numeric(unlist(error_em))
cat(error_em, sep="\n", file="error em.txt", append=TRUE)

#####BPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme

ub_BPCA<- xt_BPCA
vb_BPCA<- yt_BPCA
C_BPCA= (ub_BPCA+0.5)
yj2_BPCA={(A^2+B^2)*y1+B*C_BPCA*D+vb_BPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı

yj<- yorj
xj2_BPCA={C_BPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_BPCA<-as.numeric(unlist(xj2_BPCA))
yj2_BPCA<-as.numeric(unlist(yj2_BPCA))
ztah_BPCA=sqrt(xj2_BPCA^2+yj2_BPCA^2)
error_BPCA=ztah_BPCA-zorj
error_BPCA<-as.numeric(unlist(error_BPCA))
cat(error_BPCA, sep="\n", file="error BPCA.txt", append=TRUE)

#####PPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme

ub_PPCA<- xt_PPCA
vb_PPCA<- yt_PPCA
C_PPCA= (ub_PPCA+0.5)
yj2_PPCA={(A^2+B^2)*y1+B*C_PPCA*D+vb_PPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı

yj<- yorj
xj2_PPCA={C_PPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_PPCA<-as.numeric(unlist(xj2_PPCA))
yj2_PPCA<-as.numeric(unlist(yj2_PPCA))
ztah_PPCA=sqrt(xj2_PPCA^2+yj2_PPCA^2)
error_PPCA=ztah_PPCA-zorj
error_PPCA<-as.numeric(unlist(error_PPCA))
cat(error_PPCA, sep="\n", file="error PPCA.txt", append=TRUE)

```

#####NLPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri dönme:

```
ub_NLPCA<- xt_NLPCA
```

```
vb_NLPCA<- yt_NLPCA
```

```
C_NLPCA= (ub_NLPCA+0.5)
```

```
yj2_NLPCA={(A^2+B^2)*y1+B*C_NLPCA*D+vb_NLPCA*A*D}/(A^2+B^2)
```

```
#####Tahmin edilen y koordinatı
```

```
yj<- yorj
```

```
xj2_NLPCA={C_NLPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
```

```
xj2_NLPCA<-as.numeric(unlist(xj2_NLPCA))
```

```
yj2_NLPCA<-as.numeric(unlist(yj2_NLPCA))
```

```
ztah_NLPCA=sqrt(xj2_NLPCA^2+yj2_NLPCA^2)
```

```
error_NLPCA=ztah_NLPCA-zorj
```

```
error_NLPCA<-as.numeric(unlist(error_NLPCA))
```

```
cat(error_NLPCA, sep="\n", file="error NLPCA.txt", append=TRUE)
```

#####INIPALS yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri dönme

```
ub_Nipals<- xt_Nipals
```

```
vb_Nipals<- yt_Nipals
```

```
C_Nipals= (ub_Nipals+0.5)
```

```
yj2_Nipals={(A^2+B^2)*y1+B*C_Nipals*D+vb_Nipals*A*D}/(A^2+B^2)
```

```
#####Tahmin edilen y koordinatı
```

```
yj<- yorj
```

```
xj2_Nipals={C_Nipals*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
```

```
xj2_Nipals<-as.numeric(unlist(xj2_Nipals))
```

```
yj2_Nipals<-as.numeric(unlist(yj2_Nipals))
```

```
ztah_Nipals=sqrt(xj2_Nipals^2+yj2_Nipals^2)
```

```
error_Nipals=ztah_Nipals-zorj
```

```
error_Nipals<-as.numeric(unlist(error_Nipals))
```

```
cat(error_Nipals, sep="\n", file="error Nipals.txt", append=TRUE)
```

```
}
```

```
error_min<- read.table("error min.txt")
```

```
error_min<-as.numeric(unlist(error_min))
rmse_f_min<-rmse(error_min)#####Min(F) ölçütü için RMSE değeri
error_max<- read.table("error max.txt")
error_max<-as.numeric(unlist(error_max))
rmse_f_max<-rmse(error_max) #####Max(F) ölçütü için RMSE değeri
error_mr<- read.table("error mr.txt")
error_mr<-as.numeric(unlist(error_mr))
rmse_mr<-rmse(error_mr) #####Çoklu regresyon atama yöntemi için RMSE değeri
error_em<- read.table("error em.txt")
error_em<-as.numeric(unlist(error_em))
rmse_em<-rmse(error_em) #####EM algoritması için RMSE değeri
error_BPCA<- read.table("error BPCA.txt")
error_BPCA<-as.numeric(unlist(error_BPCA))
rmse_BPCA<-rmse(error_BPCA) #####BPCA yöntemi için RMSE değeri
error_PPCA<- read.table("error PPCA.txt")
error_PPCA<-as.numeric(unlist(error_PPCA))
rmse_PPCA<-rmse(error_PPCA) #####PPCA yöntemi için RMSE değeri
error_NLPCA<- read.table("error NLPCA.txt")
error_NLPCA<-as.numeric(unlist(error_NLPCA))
rmse_NLPCA<-rmse(error_NLPCA) #####NLPCA yöntemi için RMSE değeri
error_Nipals<- read.table("error Nipals.txt")
error_Nipals<-as.numeric(unlist(error_Nipals))
rmse_Nipals<-rmse(error_Nipals) #####INIPALS yöntemi için RMSE değeri
```

Simülasyon çalışmasında 12 landmarklı durumda kullanılan R kodları aşağıda belirtilmiştir.

```
#####Kullanılan paketler
library(mvtnorm)
library(shapes)
library(readxl)
library(mice)
library(pcaMethods)
library(Amelia)
library(Matrix)
library(dplyr)
##RMSE değerlerinin hesaplanmasında kullanılan fonksiyon
rmse <- function(error)
{
  sqrt(mean(error^2))
}
set.seed(100)
t=1000 #####tekrar sayısı
n=50#####örneklem büyüklüğü
n1=n
n2=n1+1
n3=2*n1
n4=n3+1
n5=3*n1
n6=n5+1
n7=4*n1
n8=n7+1
n9=5*n1
n10=n9+1
n11=6*n1
n12=n11+1
n13=7*n1
n14=n13+1
n15=8*n1
n16=n15+1
n17=9*n1
n18=n17+1
n19=10*n1
la=12#####landmark sayısı
l=1
sayac2<-0
sayac_mr<-0
sayac_em<-0
sayac_ppca<-0
sayac_bpca<-0
sayac_nlpca<-0
```

```

sayac_nipals<-0
for (l in 1:t) {
sayac_mr<-sayac_mr+1
sayac_em<-sayac_em+1
sayac_ppca<-sayac_ppca+1
sayac_bpca<-sayac_bpca+1
sayac_nlpca<-sayac_nlpca+1
sayac_nipals<-sayac_nipals+1 #####Çok deęişkenli normal daęılımdan veri
türetilmesi
KOVARYANS <- read_excel("KOVARYANS.xlsx", col_names = FALSE)
KOVARYANS<-as.matrix(KOVARYANS)
KOVARYANS<-nearPD(KOVARYANS)
cov<-KOVARYANS$mat@x
dim(cov)<-c(la*2,la*2)
data<-rmvnorm(n, mean=c(644.18, 534.72, 747.81, 454.54, 641.90, 824.45, 533.90,
743, 643.18, 607.81, 675.18, 643.45, 751, 646.81, 646.18, 521.72, 534.72, 523.90,
385.81, 390.54, 312.90, 480, 478.81, 238.54), sigma=cov)
my.dat<-array(data, dim=c(la,2,n))
#####Kayıp landmark tahmininde kullanılacak landmarkların belirlenmesi
xorj<-my.dat [3,1,1]
x1<- my.dat [1,1,1]
x2<-my.dat [2,1,1]
yorj<-my.dat [3,2,1]
y1<-my.dat [1,2,1]
y2<-my.dat [2,2,1]
##### Veri setinin Bookstein koordinatlarına dönüştürülmesi
my.d<-bookstein2d(my.dat)
my.d.cor<-bookstein2d(my.dat)$bshpv
#####Kayıp landmarkın oluşturulması
my.dat [3,1,1]<-NA
my.dat [3,2,1]<-NA
#####Kayıp landmark içeren veri seti üzerinden Bookstein koordinatlarının
oluşturulması
my.dat.book<-bookstein2d(my.dat)
my.dat.book.cor<-my.dat.book$bshpv
veri<-my.dat.book.cor
#####Veri setinin F istatistięini uygulamak için düzenlenmesi
i=1
yeni<-data.frame(matrix(nrow = n, ncol = 2))
colnames(yeni)<-c("x", "y")
for (i in 1:n) {
yeni[i,1]<-veri[3,1,i]
yeni[i,2]<-veri[3,2,i]
}
#####F yaklaşımı algoritmasının uygulanması:
i=2
d31<-matrix(nrow = n, ncol = 1)
for (i in 2:n) {

```



```

deneme2=my.dat.book.cor[:,i]
xk1<-deneme2[1,1]
yk1<-deneme2[1,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d31[i,1]=sqrt((xk3-xk1)^2+(yk3-yk1)^2)#####Öklid uzaklığının hesaplanması
}
d31<-d31[-1,]
d31ort=mean(d31)
i=2
d32<-data.frame(matrix(nrow = n, ncol = 1))
for (i in 2:n) {
deneme2=my.dat.book.cor[:,i]
deneme2<-deneme2[1:3,]
xk2<-deneme2[2,1]
yk2<-deneme2[2,2]
xk3<-deneme2[3,1]
yk3<-deneme2[3,2]
d32[i,1]=sqrt((xk3-xk2)^2+(yk3-yk2)^2) #####Öklid uzaklığının hesaplanması
}
d32<-d32[-1,]
d32ort=mean(d32)
d32ort
c=(-1/2)*(d31ort^2-d32ort^2)
d=sqrt(d31ort^2-(c+0.5)^2)
y4<-yeni
y4<-as.matrix(y4)
y4[1,1]<-c
y4[1,2]<-d
#####Güven aralıklarının hesaplanması:
d2<-d31
d2as<-d31ort-1.96*(sd(d2)/sqrt(length(d2))) ##### d2 alt sınır
d2üs<- d31ort +1.96*(sd(d2)/sqrt(length(d2))) ##### d2 üst sınır
d3<-d32
d3as<- d32ort -1.96*(sd(d3)/sqrt(length(d3))) ##### d3 alt sınır
d3üs<- d32ort+1.96*(sd(d3)/sqrt(length(d3))) ##### d3 üst sınır
closeAllConnections()
sayac<-0
fstatt<-matrix(byrow=TRUE)
ls<-matrix(byrow=TRUE)
sayacs<-matrix(byrow=TRUE)
cs<-matrix(byrow=TRUE)
ds<-matrix(byrow=TRUE)
closeAllConnections()
#####F istatistiğinin hesaplanması
for ( c in seq(d2as, d2üs, 0.2) ) {
for ( d in seq(d3as, d3üs, 0.2) ) {
sayac<-sayac+1

```

```

y4[1,1]<-c
y4[1,2]<-d
gakt=n*(mean(y4[,1])-mean(y4))^2+n*(mean(y4[,2])-mean(y4))^2
cat(gakt, sep="\n", file="gakt.txt", append=TRUE)
sink("gkt.txt")
k=2
for( i in 1:k){
for( j in 1:n) {
t=((y4[j,i]-mean(y4))^2)
cat(t, sep="\n", file="gkt.txt", append=TRUE)
j=j+1
}
i=i+1
}
gkt <- read.table("gkt.txt")
gktson=sum(gkt)
gikt=gktson-gakt
fstat=((gakt/(k-1))/(gikt/(2*n-k)))
fstatt[sayac]<-fstat
closeAllConnections()
ls[sayac]<-l
sayacs[sayac]<-sayac
cs[sayac]<-c
ds[sayac]<-d
}
closeAllConnections()
}
####Çoklu regresyon atama yöntemi ile kayıp değer tahmini
mr_veri<-yeni
xt_mrs<-matrix(byrow=TRUE)
yt_mrs<-matrix(byrow=TRUE)
colnames(mr_veri)<-c("v1", "v2")
imputed_Data <- mice(mr_veri, method="norm")
xt_mr=imputed_Data$imp$v1[1,1]
yt_mr=imputed_Data$imp$v2[1,1]
xt_mrs[sayac_mr]=xt_mr
yt_mrs[sayac_mr]=yt_mr
####EM algoritması ile kayıp değer tahmini
k=12
j=1
sink("yenix.xls")
while( j <=n) {
i=3
while( i <= k) {
aa<-veri[i,1,j]
cat(aa, sep="\n")
i=i+1
}
}

```

```

j=j+1
}
j=1
sink("yeni.xls")
while (j <=n) {
i=3
while (i <= k) {
bb<-veri[i,2,j]
cat(bb, sep="\n")
i=i+1
}
j=j+1
}
yenix<-read.table("yenix.xls")
yeni<-read.table ("yeni.xls")
yenix<-as.matrix(yenix)
yeni<-as.matrix(yeni)
yeni_em<-cbind(yenix[1:n1,], yeni[1:n1,], yenix[n2:n3,], yeni[n2:n3,],
yenix[n4:n5,], yeni[n4:n5,], yenix[n6:n7,], yeni[n6:n7,],yenix[n8:n9,],
yeni[n8:n9,], yenix[n10:n11,], yeni[n10:n11,], yenix[n12:n13,], yeni[n12:n13,],
yenix[n14:n15,], yeni[n14:n15,], yenix[n16:n17,], yeni[n16:n17,], yenix[n18:n19,],
yeni[n18:n19,])
result_em<-amelia(yeni_em, m = 1)
xt_ems<-matrix(byrow=TRUE)
yt_ems<-matrix(byrow=TRUE)
xt_em<-result_em$imputations$imp1[1,1]
yt_em<-result_em$imputations$imp1[1,2]
xt_ems[sayac_em]=xt_em
yt_ems[sayac_em]=yt_em
#####PCA tabanlı yöntemler ile kayıp değer tahmini
md <- prep(yeni_em, scale="none", center=TRUE)
xt_PPcAs<-matrix(byrow=TRUE)
yt_PPcAs<-matrix(byrow=TRUE)
resPPcA <- pca(md, method="ppca")
xt_PPcA<-resPPcA@completeObs[1,1]
yt_PPcA<-resPPcA@completeObs[1,2]
xt_PPcAs[sayac_ppca]=xt_PPcA
yt_PPcAs[sayac_ppca]=yt_PPcA
xt_BPCAs<-matrix(byrow=TRUE)
yt_BPCAs<-matrix(byrow=TRUE)
resBPCa <- pca(md, method="bpca")
xt_BPCa<-resBPCa@completeObs[1,1]
yt_BPCa<-resBPCa@completeObs[1,2]
xt_BPCAs[sayac_bpca]=xt_BPCa
yt_BPCAs[sayac_bpca]=yt_BPCa
xt_Nipalss<-matrix(byrow=TRUE)
yt_Nipalss<-matrix(byrow=TRUE)
resNipals <- pca(md, method="nipals")

```

```

xt_Nipals<-resNipals@completeObs[1,1]
yt_Nipals<-resNipals@completeObs[1,2]
xt_Nipalss[sayac_nipals]=xt_Nipals
yt_Nipalss[sayac_nipals]=yt_Nipals
xt_NLPCAs<-matrix(byrow=TRUE)
yt_NLPCAs<-matrix(byrow=TRUE)
resNLPCA <- pca(md, method="nlpca")
xt_NLPCA<-resNLPCA@completeObs[1,1]
yt_NLPCA<-resNLPCA@completeObs[1,2]
xt_NLPCAs[sayac_nlpca]=xt_NLPCA
yt_NLPCAs[sayac_nlpca]=yt_NLPCA
#####Min(F) ölçütü için sonuçların düzenlenmesi
ts<-rbind(ls, sayacs, cs, ds, fstatt)
ts<-t(ts)
ts<-as.data.frame(ts)
colnames(ts)<-c("L", "SAYAC", "X", "Y", "F")
minn<-ts %>%
group_by(L) %>%
slice(which.min(F))
maxx<-ts %>%
group_by(L) %>%
slice(which.max(F))
#####Min(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
D=(x2-x1)^2+(y2-y1)^2
A=x2-x1
B=y2-y1
ub<- minn$X
vb<- minn$Y
C= (ub+0.5)
yj2={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinatı
yj<- yorj
xj2={C*D+A*x1-B*yj+B*y1}/A ##### Tahmin edilen x koordinatı
xj2<-as.numeric(unlist(xj2))
yj2<-as.numeric(unlist(yj2))
zorj=sqrt(xorj^2+yorj^2)#####Orijinal hipotenüs değeri
ztah_min=sqrt(xj2^2+yj2^2)#####MinF ölçütüne göre hipotenüs değeri
error_min=ztah_min-zorj
error_min<-as.numeric(unlist(error_min))
cat(error_min, sep="\n", file="error min.txt", append=TRUE)
#####Max(F) ölçütü için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ubm<- maxx$X
vbm<- maxx$Y
C= (ubm+0.5)
ym={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) #####Tahmin edilen y koordinatı
yj<- yorj
xm={C*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı

```

```

xm<-as.numeric(unlist(xm))
ym<-as.numeric(unlist(ym))
ztah_max=sqrt(xm^2+ym^2)#####Max(F) ölçütü ile elde edilen hipotenüs değeri
error_max=ztah_max-zorj
error_max<-as.numeric(unlist(error_max))
cat(error_max, sep="\n", file="error max.txt", append=TRUE)
closeAllConnections()
#####Çoklu regresyon atama yöntemi için Bookstein koordinatlarından orijinal
koordinatlara geri dönme:
ub_mr<- xt_mr
vb_mr<- yt_mr
C_mr=(ub_mr+0.5)
yj2_mr={(A^2+B^2)*y1+B*C_mr*D+vb*A*D}/(A^2+B^2)#####Tahmin edilen
ykoordinatı
yj<- zorj
xj2_mr={C_mr*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_mr<-as.numeric(unlist(xj2_mr))
yj2_mr<-as.numeric(unlist(yj2_mr))
ztah_mr=sqrt(xj2_mr^2+yj2_mr^2)#####Çoklu regresyon atama yöntemi ile elde
edilen hipotenüs değeri
error_mr=ztah_mr-zorj
error_mr<-as.numeric(unlist(error_mr))
cat(error_mr, sep="\n", file="error mr.txt", append=TRUE)
#####EM algoritması için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub<- xt_em
vb<- yt_em
C=(ub+0.5)
yj2_em={(A^2+B^2)*y1+B*C*D+vb*A*D}/(A^2+B^2) ## tahmin değeri
yj<- zorj
xj2_em={C*D+A*x1-B*yj+B*y1}/A ## tahmin değeri
xj2_em<-as.numeric(unlist(xj2_em))
yj2_em<-as.numeric(unlist(yj2_em))
ztah_em=sqrt(xj2_em^2+yj2_em^2)
error_em=ztah_em-zorj
error_em<-as.numeric(unlist(error_em))
cat(error_em, sep="\n", file="error em.txt", append=TRUE)
#####BPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_BPCA<- xt_BPCA
vb_BPCA<- yt_BPCA
C_BPCA=(ub_BPCA+0.5)
yj2_BPCA={(A^2+B^2)*y1+B*C_BPCA*D+vb_BPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı
yj<- zorj
xj2_BPCA={C_BPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_BPCA<-as.numeric(unlist(xj2_BPCA))
yj2_BPCA<-as.numeric(unlist(yj2_BPCA))

```

```

ztah_BPCA=sqrt(xj2_BPCA^2+yj2_BPCA^2)
error_BPCA=ztah_BPCA-zorj
error_BPCA<-as.numeric(unlist(error_BPCA))
cat(error_BPCA, sep="\n", file="error BPCA.txt", append=TRUE)
#####PPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_PPCA<- xt_PPCA
vb_PPCA<- yt_PPCA
C_PPCA= (ub_PPCA+0.5)
yj2_PPCA={(A^2+B^2)*y1+B*C_PPCA*D+vb_PPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı
yj<- yorj
xj2_PPCA={C_PPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_PPCA<-as.numeric(unlist(xj2_PPCA))
yj2_PPCA<-as.numeric(unlist(yj2_PPCA))
ztah_PPCA=sqrt(xj2_PPCA^2+yj2_PPCA^2)
error_PPCA=ztah_PPCA-zorj
error_PPCA<-as.numeric(unlist(error_PPCA))
cat(error_PPCA, sep="\n", file="error PPCA.txt", append=TRUE)
#####NLPCA yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_NLPCA<- xt_NLPCA
vb_NLPCA<- yt_NLPCA
C_NLPCA= (ub_NLPCA+0.5)
yj2_NLPCA={(A^2+B^2)*y1+B*C_NLPCA*D+vb_NLPCA*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı
yj<- yorj
xj2_NLPCA={C_NLPCA*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı
xj2_NLPCA<-as.numeric(unlist(xj2_NLPCA))
yj2_NLPCA<-as.numeric(unlist(yj2_NLPCA))
ztah_NLPCA=sqrt(xj2_NLPCA^2+yj2_NLPCA^2)
error_NLPCA=ztah_NLPCA-zorj
error_NLPCA<-as.numeric(unlist(error_NLPCA))
cat(error_NLPCA, sep="\n", file="error NLPCA.txt", append=TRUE)
#####INIPALS yöntemi için Bookstein koordinatlarından orijinal koordinatlara geri
dönme:
ub_Nipals<- xt_Nipals
vb_Nipals<- yt_Nipals
C_Nipals= (ub_Nipals+0.5)
yj2_Nipals={(A^2+B^2)*y1+B*C_Nipals*D+vb_Nipals*A*D}/(A^2+B^2)
#####Tahmin edilen y koordinatı
yj<- yorj
xj2_Nipals={C_Nipals*D+A*x1-B*yj+B*y1}/A #####Tahmin edilen x koordinatı

xj2_Nipals<-as.numeric(unlist(xj2_Nipals))
yj2_Nipals<-as.numeric(unlist(yj2_Nipals))
ztah_Nipals=sqrt(xj2_Nipals^2+yj2_Nipals^2)
error_Nipals=ztah_Nipals-zorj

```

```

error_Nipals<-as.numeric(unlist(error_Nipals))
cat(error_Nipals, sep="\n", file="error Nipals.txt", append=TRUE)
}
error_min<- read.table("error min.txt")
error_min<-as.numeric(unlist(error_min))
rmse_f_min<-rmse(error_min)#####Min(F) ölçütü için RMSE değeri
error_max<- read.table("error max.txt")
error_max<-as.numeric(unlist(error_max))
rmse_f_max<-rmse(error_max) #####Max(F) ölçütü için RMSE değeri
error_mr<- read.table("error mr.txt")
error_mr<-as.numeric(unlist(error_mr))
rmse_mr<-rmse(error_mr) #####Çoklu regresyon atama yöntemi için RMSE değeri
error_em<- read.table("error em.txt")
error_em<-as.numeric(unlist(error_em))
rmse_em<-rmse(error_em) #####EM algoritması için RMSE değeri
error_BPCA<- read.table("error BPCA.txt")
error_BPCA<-as.numeric(unlist(error_BPCA))
rmse_BPCA<-rmse(error_BPCA) #####BPCA yöntemi için RMSE değeri
error_PPCA<- read.table("error PPCA.txt")
error_PPCA<-as.numeric(unlist(error_PPCA))
rmse_PPCA<-rmse(error_PPCA) #####PPCA yöntemi için RMSE değeri
error_NLPCA<- read.table("error NLPCA.txt")
error_NLPCA<-as.numeric(unlist(error_NLPCA))
rmse_NLPCA<-rmse(error_NLPCA) #####NLPCA yöntemi için RMSE değeri
error_Nipals<- read.table("error Nipals.txt")
error_Nipals<-as.numeric(unlist(error_Nipals))
rmse_Nipals<-rmse(error_Nipals) #####INIPALS yöntemi için RMSE değeri

```

9. TEŞEKKÜR

Doktora eğitimim boyunca ve tez aşamasında sonsuz özverisini, desteğini ve sabrını benden hiç esirgemeyen değerli danışmanım Prof. Dr. İlker ERCAN' a, bilimsel gelişimime verdiği emek ve katkılarından dolayı sonsuz teşekkürlerimi sunarım.

Tez çalışmam süresince, tezin değerlendirilmesinde değerli katkılarını benimle paylaşan tez izleme komitesindeki değerli hocalarıma çok teşekkür ederim.

Doktora eğitimim boyunca ve tez çalışmam sırasında göstermiş olduğu destek ve sabrından dolayı, sevgili aileme sonsuz destekleri için teşekkür ederim.

10. ÖZGEÇMİŞ

8 Ağustos 1989 tarihinde Bursa'da doğdum. İlkokulu Dörtçelik İlköğretim Okulu'nda bitirdim. Lise öğrenimimi Bursa Erkek Lisesi'nde tamamladım. 2012 yılında Ege Üniversitesi Fen Fakültesi İstatistik Bölümü'nü bitirdim. 2013 yılında Bursa Uludağ Üniversitesi Sağlık Bilimleri Enstitüsü Biyoistatistik Anabilim Dalı'nda doktora eğitimime başladım. Çek Cumhuriyeti Brno şehrinde bulunan Masaryk Üniversitesi Biyoistatistik ve Analiz Enstitüsü'nde Haziran-Eylül 2014, Haziran-Temmuz 2015, İspanya Valencia şehrinde bulunan Principe Felipe Araştırma Merkezi'nde Haziran-Ağustos 2016 ve İspanya Sevilla şehrinde bulunan Virgen del Rocio Hastanesi'nde kurumunda Temmuz-Eylül 2017 tarihleri arasında Erasmus programı kapsamında staj faaliyetinde bulundum. 2015 yılından itibaren İzmir Katip Çelebi Üniversitesi Tıp Fakültesi Biyoistatistik Anabilim Dalı'nda araştırma görevlisi kadrosundayım.