



T.C.
ULUDAĞ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



YAPAY ÖĞRENME İLE GEN – MUTASYON – HASTALIK ANALİZİ

Bilal GÜNDEN, 031690045

Aleyna ER, 031790058

BİTİRME PROJESİ

BURSA 2021
T.C.
ULUDAĞ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

YAPAY ÖĞRENME İLE GEN – MUTASYON – HASTALIK ANALİZİ

Bilal GÜNDEN, 031690045

Aleyna ER , 031790058

Proje Danışmanı : Doç. Dr. Gıyasettin Özcan

İÇİNDEKİLER

İÇİNDEKİLER	3
ŞEKİLLER DİZİNİ	5
ÖZET	6
1. GİRİŞ	7
2. COSMIC VERİ TABANINDAKİ VERİLERİN İNCELENMESİ	8
3. ML ALGORİTMALARININ ARAŞTIRILMASI	9
3.1. Makine Öğrenmesi Nedir?	9
3.2. Makine Öğrenmesi Yöntemleri Nelerdir?	9
3.2.1. Denetimli (supervised) Öğrenme Algoritmaları	10
3.2.2. Denetimsiz (unsupervised) Öğrenme Algoritmaları	12
3.2.3. Takviyeli (Reinforcement) Öğrenme Yöntemleri	14
3.3. Yapay Sinir Ağları	14
4. SINIFLANDIRMA ALGORİTMALARININ KARŞILAŞTIRILMASI	15
4.1. KNN algoritmasının Avantaj (+) ve Dezavantajları (-)	15
4.2. Lojistik Regresyon Algoritmasının Avantaj (+) ve Dezavantajları (-)	16
4.3. Random Forest Algoritmasının Avantaj (+) ve Dezavantajları (-)	16
4.4. Naive Bayes Algoritmasının Avantaj (+) ve Dezavantajları (-)	17
5. VERİ SETLERİNİN MANİPÜLASYONU	18
5.1. Non – Coding Variants (NCV) Veri Seti	18
5.1.1. Non – Coding Variants (NCV) Ön İşleme	19
5.2. All Mutations in Census Genes (MutExCen) Veri Seti	26
5.2.1. All Mutations in Census Genes (MutExCen) Veri Ön İşleme	27
5.3. Copy Number Variants (CNA) Veri Seti	29
5.3.1. Copy Number Variants (CNA) Veri Ön İşleme	29
6. ANALİZ	31
6.1. Deneysel (Experimental) Analiz	31
6.1.2. Mutation Export Census ile Deneysel Analiz	31
6.1.3. NCV ile Deneysel Analiz	34
6.2. UI ANALİZ	36
6.2.1. KNN ile Kromozom Tahminleme	36
6.2.2. Naive Bayes ile Primary Histology Tahminleme	36
7. KULLANICI ARAYÜZÜ	38

7.1.	Arayüz Kapsamında Oluşturulan Dosyalar	38
8.	TAHMİNLEME UYGULAMASININ ÇALIŞTIRILMASI	43
8.1.	Mutation Export Census Verisi ile Tahminleme	44
8.2.	Copy Number Variant (CNA) Verisi ile Tahminleme	45
9.	SONUÇ	46
10.	KAYNAKÇA	47
11.	TEŞEKKÜR	48
12.	ÖZGEÇMİŞ	49

ŞEKİLLER DİZİNİ

Şekil 1. Makine öğrenmesi yöntemleri ve kullanım alanları.....	9
Şekil 2. Rastgele ormanlar.....	10
Şekil 3. K-NN ile Sınıflandırma.....	11
Şekil 4. Lineer Regresyon.....	11
Şekil 5. Lojistik Regresyon.....	12
Şekil 6. Kmeans ile kümeleme.....	13
Şekil 7. Hiyerarşik kümeleme algoritmaları.....	13
Şekil 8. Boyut Azaltma.....	13
Şekil 9. Yapay sinir ağı modeli.....	14
Şekil 10. NCV veri setinin örnek gösterimi.....	19
Şekil 11. WT_SEQ değerleri.....	19
Şekil 12. MUT_SEQ değerleri.....	19
Şekil 13. Veri setinde eksik değerler bulunmaktadır.....	20
Şekil 14. Veri setinde eksik değerlerin dağılımı	20
Şekil 15. Null değerler silindikten sonra WT_SEQ ve MUT_SEQ için değerler dağılımı.....	21
Şekil 16. WT_SEQ pasta grafiği.....	21
Şekil 17. MUT_SEQ pasta grafiği.....	22
Şekil 18. FATHMM Non Coding Skoru sütunu için encoding yapan kod bloğu.....	22
Şekil 19. FATHMM Non Coding Skoru sütunu için encoding işlemi sonucu.....	22
Şekil 20. Dataframedeki veri tipleri.....	23
Şekil 21. NCV veri setini düzenlemek için kod bloğu.....	24
Şekil 22. NCV setinde kalan sütunları otomatik kategorilendiren ve Label bilgilerini birleştiren kod bloğu.....	24
Şekil 23. Veri setinin eğitime hazır, sütunların kategorilerine etiketlenmiş hali.....	25
Şekil 24. NCV’de kullanılan diğer sütunların değer dağılımı.....	25
Şekil 25. Cosmic Mutant Export Census veri setinin örnek gösterimi.....	26
Şekil 26. Veri seti hakkında genel bilgi.....	27
Şekil 27. Mutant Export Census veri setinde düzenleme yapan kod bloğu.....	27
Şekil 28. Genom pozisyonu sütununun Kromozom ID olarak değiştirilmesi.....	28
Şekil 29. Kullanımı kolaylaştıracak değişiklikler yapılmıştır.....	28
Şekil 30. MutExCen için sütunların eşsiz değer sayıları.....	28
Şekil 31. CNA veri seti örnek gösterimi.....	30
Şekil 32. Random Forest ile FATHMM tahminleme.....	32
Şekil 33. KNN ile FATHMM tahminleme.....	33
Şekil 34. Logistic Regression ile FATHMM tahminleme.....	33
Şekil 35. Naive Bayes ile FATHMM tahminleme.....	34
Şekil 36. Naive Bayes ile FATHMM skoru tahminleme.....	35
Şekil 37.. KNN ile Kromozom Tahminleme (MutExCen).....	36
Şekil 38. Naive Bayes ile Primary Histology Tahminleme (CNA).....	37
Şekil 39. Primary histology değerleri ve karşılık gelen sınıflar.....	37
Şekil 40. serverFunctions.py.....	38
Şekil 41. settings.py.....	39
Şekil 42. urls.py.....	39
Şekil 43. CNA veri seti için index.html.....	40
Şekil 44. Formda doldurulan değerler değişkenlere atılır.....	41
Şekil 45. Sonuç değeri bastırılır.....	41
Şekil 46. MutExpCen veri seti için index.html.....	42
Şekil 47. MutExpCen için serverFunctions.py.....	42
Şekil 48. MutExCen için örnek test verisi.....	44
Şekil 49. Kromozom bilgisi tahminleme örneği.....	44
Şekil 50. CNA için örnek test verisi.....	45
Şekil 51. Primary histology bilgisi tahminleme örneği.....	45

ÖZET

Gen – mutasyon – hastalık analizi projesi süresince, COSMIC [1] veri tabanında bulunan veri setleri incelenmiş, bu verilerin analizi yapılabilmesi için gerekli yapay öğrenme metotları araştırılmıştır. Projeye uygun olduğu öngörülen veri setleri üzerinde kümeleme ve sınıflandırma olmak üzere çeşitli analizler yapılmış, sonuçları raporlanmıştır. Öğrenme algoritmalarının kıyaslamasının yapılabilmesi için seçilen veri setlerinden birine (CosmicMutantExportCensus) sırayla uygulanmış ve accuracy'leri karşılaştırılmıştır.

CosmicCompleteCNA ve CosmicMutantExportCensus veri setleri için tahminleme arayüzü geliştirilmiştir. Geliştirilen arayüzlerde kullanıcıdan alınan bilgilere göre CosmicCompleteCNA'da hastalık, CosmicMutantExportCensus'ta ise mutasyonun gerçekleştiği kromozom tahmini yapılabilmektedir.

1. GİRİŞ

COSMIC veri tabanı [1], projemiz için önemli kaynaklar içermektedir. Kanserde Somatik Mutasyonlar Kataloğu olan COSMIC (Catalogue Of Somatic Mutations In Cancer), insan ırkında oluşan kanserlerde somatik mutasyonların etkisini araştırmak için oluşturulmuş kapsamlı bir kaynaktır. Wellcome Sanger Institute tarafından desteklenmiş ve oluşturulmuştur.

Proje süresince, COSMIC’te bulunan veriler incelenmiş, projeye uygun olabilecek aday veri setleri incelenmiştir. Analizlerde kullanılacak yapay öğrenme algoritmaları araştırılmış ve karşılaştırılmış, elde edilen bilgiler ışığında çeşitli analizler yapılmıştır. Yapılan çalışmalar COSMIC Veri Tabanındaki Verilerin İncelenmesi, ML Algoritmalarının Araştırılması, Sınıflandırma Algoritmalarının Karşılaştırılması, Veri Setlerinin Manipülasyonu, Analiz ve Kullanıcı Arayüzü olarak altı başlıkta toplanmıştır.

2. COSMIC VERİ TABANINDAKİ VERİLERİN İNCELENMESİ

COSMIC internet sayfasında, insanlarda gelişen ve kansere sebep olan birçok mutasyon hakkında çeşitli bilgilerin sunulduğu tablolar bulunmaktadır. GRCh37 ve GRCh38 isimli iki adet referans gen, yapılan çalışmalarda kullanılmıştır. GRCh38 referansı diğerine göre daha günceldir.

Cosmic'te bulunan veri setleri, insanlardan (hastalardan) alınan örneklerle referans geni kıyaslar ve farklılıkları, bu farklılıkların getirdiği sonuçları raporlar. Tablolar genellikle referans genle kıyaslanmış ve referanstan farklı olan (mutasyona uğramış) bölgeleri, mutasyon konumunu, meydana gelen hastalığı ve hastalığın insan vücudundaki yeri gibi bazı genel bilgilerle beraber o tabloya (çalışmaya) özgü farklı bilgiler içerir. Var olan tablolarla ilgili kısa bilgi vermek gerekirse:

- COSMIC Mutation Data: DNA'nın protein kodlayan bölgelerinde oluşan nokta mutasyonların incelendiği çalışma bilgileri içerir. Örnek ID'si (insanı temsilen), birincil hastalık, birincil bölge gibi genel bilgilerin yanısıra nükleotit sekansta veya amino asit düzeyinde gerçekleşen değişimleri de sunar.
- All Mutations in Census Genes: Üstteki tabloya benzer bilgiler içermekle beraber, sadece CGC (Cancer Gene Census) çalışmasında bulunan genler toplanmıştır.
- Complete Fusion Export: DNA'da füzyon mutasyonuna uğramış bölgeler (genler) hakkında bilgi içerir. Kopyalama sayısı arttıkça genlerin uçlarında (transcription factor binding sites) erime gerçekleşmekte ve bu da daha fazla kopyalanmasını engellemektedir.
- Non coding variants: DNA'nın protein kodlamayan bölgelerinde oluşan mutasyonlar hakkında yapılmış çalışmanın ürünüdür.
- Copy Number Variants: DNA'da gerçekleşen çoklu mutasyonların incelendiği çalışmadır.

Kullanılan veri setleri ilerleyen bölümlerde daha detaylı açıklanacaktır.

3. ML ALGORİTMALARININ ARAŞTIRILMASI

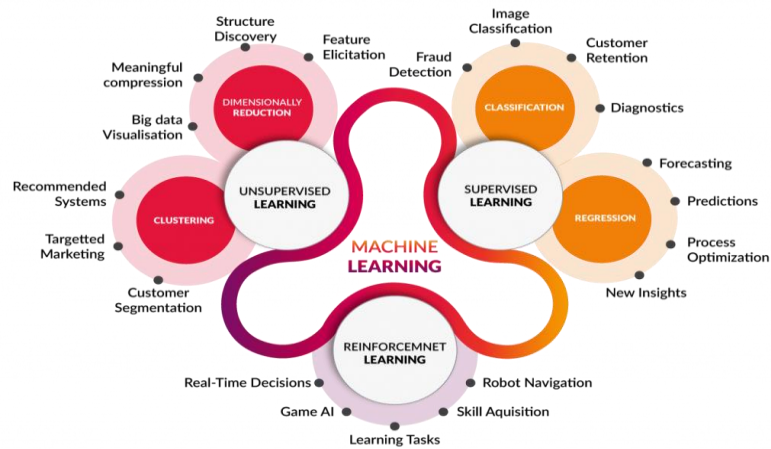
3.1. Makine Öğrenmesi Nedir?

Son yıllarda popülerliği bir hayli artmış olan makine öğrenmesi genelde yapay zeka ve derin öğrenme kavramlarıyla karıştırılmaktadır. Yapay zeka denilen kavram bilgisayarların verilen problemleri insan aklını taklit ederek çözmesine olanak veren uygulamaların genel ismidir.

Makine öğrenmesi ise çeşitli öğrenme yöntemleri ve bu yöntemlerin barındırdığı çeşitli algoritmaları kullanarak bilgisayardan mantıklı bir çıktı oluşturulması için kullanılır. Diğer yapay zeka uygulamalarından farklı olarak makine öğrenmesinde bilgisayara kurallar tanımlayarak oluşturacağı çıktıyı değiştirmek yerine bilgisayarın o kuralları kendisinin oluşturmasını sağlarız. Tıpkı insan beyninde olduğu gibi. Örneğin insan bir hayvan gördüğünde beyninde herhangi bir işlem gerçekleştirmeden veya belirli kurallara uyup uymadığına bakmadan o hayvanın hangi tür olduğunu söyleyebilir. Makine öğrenmesinde de ulaşılmak istenen budur. Bu seviyeye ulaşabilmesi için de bilgisayara çok sayıda veriler girdi olarak verilir ve bu veriler çıktıya en uygun olan öğrenme algoritmalarıyla bir çıktı oluşturulur.

3.2. Makine Öğrenmesi Yöntemleri Nelerdir?

Makine öğrenmesi yöntemleri temelde 3'e ayrılır. Bu yöntemler gözetimli (supervised), gözetimsiz (unsupervised) ve takviyeli(reinforced) yöntemlerdir. Bu yöntemler kendi başlarına kullanılabilir veya birlikte kullanılarak daha karmaşık öğrenimlerin gerçekleştirilmesinde kullanılabilir. Örneğin yapay sinir ağları algoritmaları bu yöntemlerden üçünü de kullanarak kompleks ilişkilerin çözümlenmesinde kullanılır.



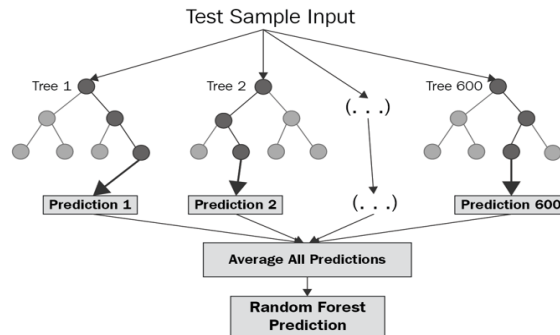
Şekil 1. Makine öğrenmesi yöntemleri ve kullanım alanları

3.2.1.Denetimli (supervised) Öğrenme Algoritmaları

Bu öğrenme yönteminde girilecek verilerin çıktıları bellidir. Girdiler ve çıktılar makine öğrenimine sokularak bilgisayar tarafından girdi ve çıktı arasında bir eşleşme fonksiyonu oluşturulur. Böylece yeni gelecek girdiler hakkında mantıklı bir çıktı oluşturabilir. Denetimli öğrenme algoritmaları da kendi aralarında ikiye ayrılmaktadır. Eğer oluşan çıktılar numerik değil ise (bu çıktılar bir veya birden fazla obje, string olabilir.) sınıflandırma (classification) algoritmaları kullanılır. Eğer çıktılar nümerik değerler ise regresyon(regression) algoritmaları kullanılır.

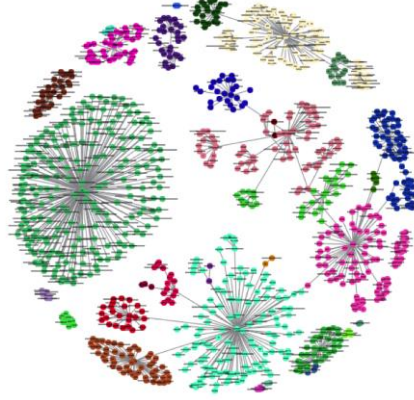
3.2.1.1.Sınıflandırma yöntemi algoritmaları

- Naif Bayes Sınıflandırıcı : Kurulum açısından diğer algoritmalara göre daha basittir ve çok büyük verilerin olduğu durumlarda kullanışlıdır. Bayes teorime dayanır.
- Karar ağaçları : Diğer algoritmalarından farklı olarak karar ağaçları hem regresyon algoritması hem de sınıflandırma algoritması olarak kullanılabilir. Karar ağaçlarında veriler küçük karar kümelerine dönüşür. Bu kümelerden karar düğümleri oluşturulur. Her karar düğümünün birden fazla cevabı olabilir. Bu cevaplar karar veya yaprak düğümlere karşılık gelir. En üstteki düğüme kök düğüm denir.
- Rastgele ormanlar : Rastgele ormanlar algoritmasını birden fazla karar ağacının birlikte çalışması olarak düşünebiliriz. Verilen veriler orman diyebileceğimiz hayali bir kümede dağıtılarak birden fazla karar ağacı oluşturur. Daha sonra verinin doğruluğunu ve istikrarını artırmak için bu ağaçlar birleştirilir. Karar ağaçlarından oluşturulduğu için bu algoritma da hem regresyon hem de sınıflandırma algoritması olarak kullanılabilir.



Şekil 2. Rastgele ormanlar

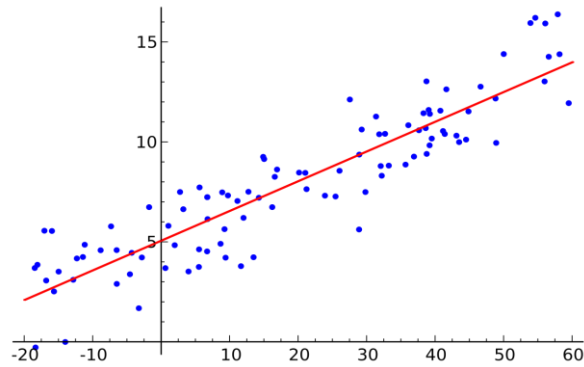
- K-NN: Kullanması ve kurulması basit algoritmalarından bir diğeri de K-NN algoritmasıdır. Yapı itibariyle lazy denilen bir öğrenim algoritmasıdır. Lazy denmesinin nedeni verilerin özelliklerini öğretmekten çok ezberletir. Böylece yeni bir veri girildiğinde veya başka bir veri bütünüyle karşılaştığında en yakınındaki komşuların özelliklerine bakarak en uygun bulduğu küme içerisine yerleştirir. Buradaki K değeri bakılacak elemanların toplam sayısıdır. Yeni değer geldiğinde k kadar uzağındaki komşularına bakılarak uzaklık hesabı yapılır.



Şekil 3. K-NN ile Sınıflandırma

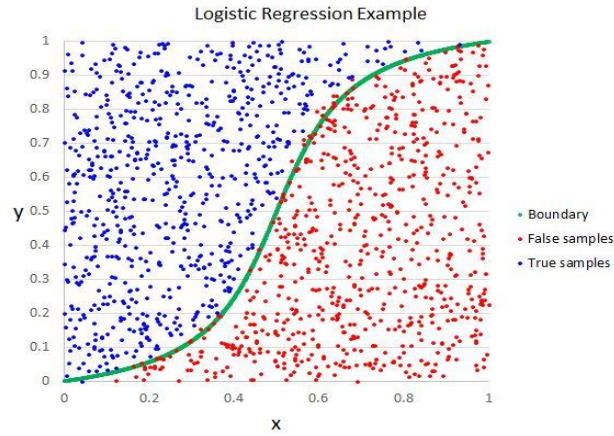
3.2.1.2.Regresyon yöntemi algoritmaları

- Lineer regresyon: Lineer regresyon algoritmasında bir bağımlı ve bir bağımsız değişken üzerinden 2 boyutlu grafik üzerinde olabilecek en uygun lineer çizgiyi bulan algoritmadır. Bu grafikte y eksenini bağımlı değişken, x ise bağımsız değişkendir. Örneğin cinsiyetin maaş üzerindeki etkisini belirlemek için lineer regresyon algoritması kullanılabilir. Burada cinsiyet bağımsız değişken, maaş ise bağımlı değişkendir. Buna göre bir sonraki erkek çalışan ile kadın çalışan arasındaki maaş farkı hesaplanabilir.



Şekil 4. Lineer Regresyon

- Lojistik regresyon : Lojistik regresyon algoritması sınıflandırma algoritmasıdır. Lineer regresyon algoritması gibi 2 boyutlu bir grafik oluşturur. Yine x bağımsız ve y bağımlı değişkenler vardır. Bu algortmada lineer regresyon algoritmasından farklı olarak lineer bir çizgi yerine bir eğri oluşturur ve cevapları bu eğriler arasında kümeler. Nümerik cevaplar yerine boolean denilen Evet/Hayır cevaplarının olduğu durumlarda kullanılması daha iyi bir sonuç verecektir.



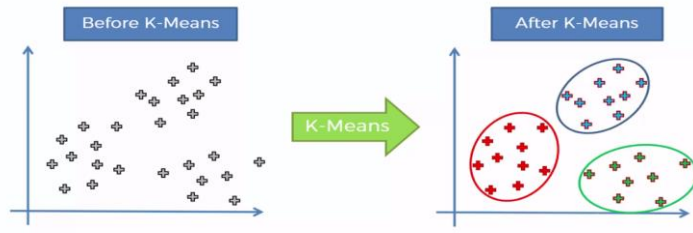
Şekil 5. Lojistik Regresyon

3.2.2.Denetimsiz (unsupervised) Öğrenme Algoritmaları

Denetimli öğrenme yöntemleri etiketlenmiş veriler üzerinde etkilidir. Eğer elimizde kategorize edilmemiş etiketsiz büyük oranda veriler var ise burada denetimsiz öğrenme yöntemleri kullanılması daha etkili çözümler üretir. Denetimsiz öğrenme algoritmalarında 2 ana yöntem vardır; bunlar Kümeleme (Clustering), Boyut azaltma (Dimensionality Reduction) yöntemleridir.

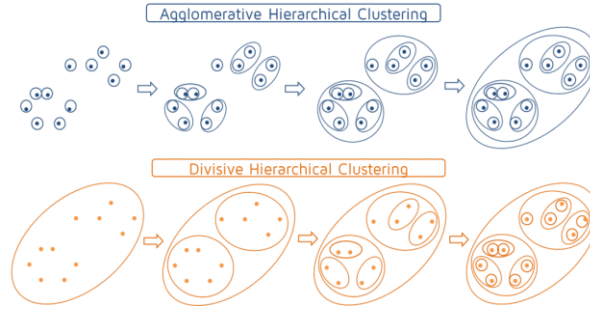
3.2.2.1.Kümeleme (Clustering) yöntemleri

- K-Means Algoritması : K-means algoritmasındaki K değeri küme sayısını verir. Bu algoritma parametre olarak kaç kümeye bölüneceği belirlenir. Bazı durumlarda bu avantaja döneceği gibi eğer küme sayısı fazla veya az olursa verim düşer. Bunun önüne geçebilmek için K küme sayısını kendi belirleyen X-Means adı altında başka bir algoritma da vardır. Elimizdeki verileri K kadar kümeye bölerek kategorize eder.



Şekil 6. Kmeans ile kümeleme

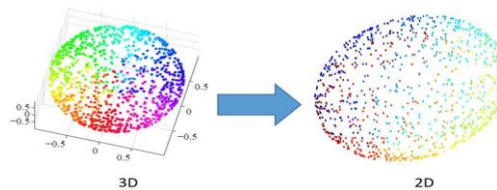
- **Hiyerarşik Kümeleme Algoritması :** Kümeleme algoritmalarından biri olan hiyerarşik kümeleme algoritması 2 şekilde yapılabilir. Bu yöntemler agglomerative (parçadan bütüne) ve divisive (bütünden parçaya) olarak adlandırılır. Agglomerative yöneliminde eleman sayısı (N) kadar küme oluşturulur ardından bu kümeler birleştirilerek tek bir küme altında toplanır. Divisive yönelimi ise agglomerative yöneliminin tam tersi olarak tüm elemanlar genel bir küme içerisine alınır ve bu küme kendi içinde bölünerek daha küçük kümeler oluşturur.



Şekil 7. Hiyerarşik kümeleme algoritmaları

3.2.2.2. Boyut Azaltma (Dimensionality Reduction) yöntemleri

- Makine öğrenmesinde girdi olarak verilen verilerde bazı ayırt edici özellikler aranır. Bu sebeple veride ayırt edici özelliklerin olması genelde performans açısından iyi olarak gösterilir. Ancak bu özelliklerin gerektiğinden fazla olması makine öğreniminin performansına negatif yönde etkiler. Bu gibi durumlarda boyut azaltma yöntemine başvurulur. Böylece veri girdilerinin özellikleri daha basite indirgenir ve performansa artı yönde bir katkı sağlanır.



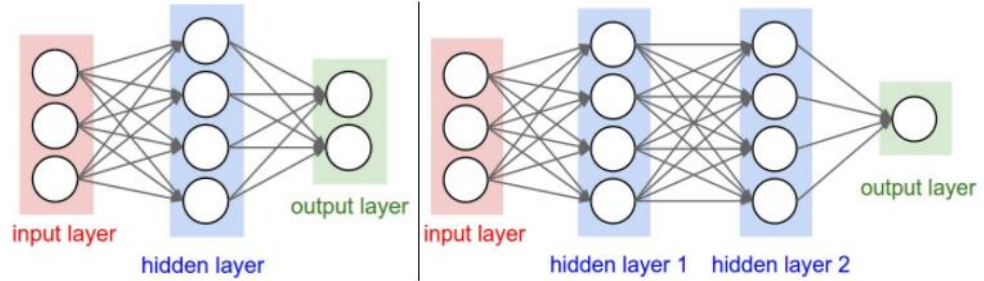
Şekil 8. Boyut Azaltma

3.2.3.Takviyeli (Reinforcement) Öğrenme Yöntemleri

Takviyeli öğrenme yönteminde ana üç anahtar kelime vardır. Bunlar ajan (agent) çevre (environment) ve ödül (reward) kelimeleridir. Burada ajan bizim öğretmek istediğimiz makine öğrenimi sistemine verilen genel isimdir. Ajanı optimum seviyede eğitmek için ödül/ceza sistemi kullanılmaktadır. Ajanın her doğru kararı için bir ödül çıktısı üretilir, tersi durumda ise bir ceza çıktısı üretilir. Bu çıktılarına göre ajan kendini eğitir ve olabilecek en yüksek ödül sayısını almaya çalışır. Ödül/cezanın belirleyen ve ajanın eğitilmesi için gerekli ortamı oluşturan sisteme de çevre genel ismi verilmiştir.

3.3.Yapay Sinir Ağları

Yapay sinir ağları bir makinenin insan beynine en yakın davrandığı algoritmadır.(şu an kabul edilen) Bu algoritma sayesinde makine yeni bilgiler türetebilir, yeni bilgiler oluşturabilir. Girdi ve çıktı değerleri kümelenirilmiş bir şekilde verildiği için gözetimli sınıflandırma algoritması kategorisine girer fakat yeni çıktılar oluşturabildiği için aynı zamanda gözetimsiz öğrenim algoritmasıdır. Verilen görselden anlaşılacağı üzere karar aşaması tek katmanlı veya çok katmanlı olabilir. İnsan beyni taklit edildiği için yapay sinir ağları birden fazla hücreden oluşur; bu hücreler tek başına çalışabilir veya daha karmaşık işlemleri gerçekleştirebilir.



Şekil 9. Yapay sinir ağı modeli

4. SINIFLANDIRMA ALGORİTMALARININ KARŞILAŞTIRILMASI

4.1. KNN algoritmasının Avantaj (+) ve Dezavantajları (-)

- + Basitlik: Kabul edilmiş bir fikir olarak KNN algoritması gerçekleştirilmesi en kolay algoritmalarından biridir. Bu nedenle makine öğrenimine yeni başlayan insanlar için kolaylıkla yapılabilir.
- + Çok yönlülük: KNN çoğunlukla sınıflandırmak için kullanılsa da bunun yanında regresyon işlemleri için de kullanılabilen çok yönlü bir algoritmadır.
- + Parametrik Olmama: Eğer kullanılacak veri seti üzerinde fazla bilgi bilinmiyorsa KNN algoritmasını kullanmak mantıklı bir seçenektir. KNN algoritması veri seti üzerine varsayımlarda bulunmaz bu sebeple parametre gerektirmez.
- + Hassasiyet: Kullanılacak veri setinde aykırı/manasız değerler var ise (null, NaN, unknown vb.) KNN algoritması diğer algoritmalara kıyasla bu veri setlerinde yüksek doğruluk oranıyla çalışabilmektedir.
- Pahalı Hesaplama: KNN algoritması her komşu değeri için ayrı ayrı çalışıp optimum sonucu geri döndürdüğü için bilgisayarı donanımsal olarak zorlayabilir. Bu durum KNN algoritmasını işlevsiz bir algoritma haline getirmez fakat yüksek boyutlu veri setlerinde KNN kullanmak zaman ve kullanılan masraf bakımından en iyi çözüm olmayabilir.
- Aşırı RAM Kullanımı: Büyük verilerde işlem yapabilmek için CPU'yu çok kullanmasının yanı sıra komşu sayısından gelen sonuçları karşılaştırmak için RAM'e sonuçları saklar. Veri seti büyüdükçe performans kaybı yaşanmasının bir diğer sebebi de budur.
- Eşit Değerler: KNN algoritmasının parametrik olmaması çoğunlukla avantaj olarak görülse de bazı durumlarda bu dezavantaja dönüşebilir. Projelerde bazı değerler diğer değerlerden daha öncelikli olabilir fakat KNN bu farkı algılayamaz ve her değere aynı önemi verir. Bu sebeple doğruluk değerinde düşüş yaşanabilir.
- Eksik değerler: Veri setinde eksik veya dikkate alınmaması gereken veriler olduğunda KNN bunları seçemez ve onları da bir sınıf gibi düşünür. Örneğin boş verilerden oluşan değerler geldiğinde bunu da bir sınıf olarak ayırabilir ve boş bir değer gönderildiğinde

bu sınıfın bir üyesi olarak görebilir. Bunun çözümü olarak programcının biraz daha efor harcayarak boş değerleri ayıklaması gerekir.

4.2. Lojistik Regresyon Algoritmasının Avantaj (+) ve Dezavantajları (-)

- + Lojistik regresyon algoritması kullanımı ve öğrenimi en kolay algoritmalarından biridir. Kullanımı basit olmasına rağmen bazı durumlarda yüksek performans ve doğruluk oranı sağlar.
- + KNN algoritmasında bir dezavantaj olarak bahsedilen bazı değerlerin önemlerinin farklı olmasını algılamama durumu Lojistik regresyon için geçerli değildir. Öğrenim sırasında verilere otomatik olarak bir önem değeri atar.
- + Düşük boyutlu veri setlerinde veya verinin az olduğu bilinmeyen verinin çok olduğu veri setlerinde over-fit yapmaya daha az meyillidir.
- + Eğitim süresi Yapay sinir ağı gibi algoritmalara göre çok çok kısadır.
- Lojistik regresyon algoritması kesin sonuçlar çıkarmaya yönelik geliştirilen bir öğrenme algoritmasıdır. Bu yüzden yoruma açık yüksek boyutlu veri setlerinde performans ve doğruluk kaybı yaşanmaktadır.
- Lineer sonuçlar çıkarma üzerine geliştirildiği için lineer olmayan sonuçların tahmini mümkün değildir.
- Veriler arasında karmaşık bir ilişki var ise çoğu algoritma bu algorithmadan daha iyi sonuçlar üretebilir. Basit ilişkilerin kurulduğu modellerde kullanılmalıdır.
- Algoritma aykırı verilere yüksek hassasiyet duymakta. Bu sebeple programcıya ek iş düşmekte. Programcı veri setindeki aykırı verileri manuel olarak temizlemelidir. Aksi takdirde sağlıklı bir sonuç çıkmayacaktır.

4.3. Random Forest Algoritmasının Avantaj (+) ve Dezavantajları (-)

- + Karar ağaçlarındaki overfitting sorununa topluluk öğrenimi (Ensemble Learning) tekniğini uygulayarak çözüm bulmuştur.
- + Hem regresyon öğrenme yöntemi hem de sınıflandırma öğrenmesi olarak kullanılabilir.

- + Hem kategorik hem de numerik verilerle başarılı sonuçlar verebilir.
- + Lineer regresyon algoritmasının aksine lineer olmayan verilerin öğreniminde yüksek performans verir.
- + Çoğu algoritmada sorunlar yaratan aykırı/eksik verilerin üstesinden gelebilir çünkü tek bir sonuca değil birden çok sonucun birleşmesiyle oluşan ortak bir sonuç çıkarır.
- + Yeni verilerin girmesi, girmiş olan verilen yok olması gibi durumlardan çok az etkilenmektedir. Bunun sebebi yine tek bir sonuca değil birden fazla sonuca bakmasıdır. Bu sebeple kararlı bir algoritmadır diyebiliriz.

- Random forest algoritmasının overfitting sorununu çözmek için çok fazla alt karar ağacı oluşturur. Bu yöntem overfitting sorunu için avantaj olarak görünse de eğitimin karmaşıklığını artırır ve normal karar ağacı yöntemine göre çok daha fazla bilgisayar gücü gerektirir.
- Çok fazla karar ağacı oluşturduğu için ve daha fazla bilgisayar gücü gerektirdiği için normal karar ağacı yöntemine göre eğitim süresi daha uzun sürer.

4.4. Naive Bayes Algoritmasının Avantaj (+) ve Dezavantajları (-)

- + Algoritmanın öğrenme ve tahmin etme süreci çok hızlı gerçekleşmektedir.
- + Çok fazla sınıf olduğu durumlarda yüksek performans göstermektedir.
- + Bağımsız verilerin tutarlılığı yüksek ise daha kısa eğitim süreci ile diğer algoritmalara göre daha yüksek performans göstermektedir.

- Naive Bayes algoritmasında öğrenmenin kalitesi bağımsız verilerin tutarlılık oranı aşırı bağlıdır. Eğer veriler arasında bu oran düşük ise eğitimin kalitesi ciddi oranda düşer. Eğer eğitime sokulan kategoriler ile teste tabi tutulan kategoriler arasında bir farklılık var ise yani daha önce gözlemlenmeyen bir kategori girdisi olursa algoritma bu kategoriye 0 değerini atar. Bu sorun Sıfır frekansı olarak da isimlendirilir. Bu sorunun ortadan kaldırılması için yumuşatma yöntemleri kullanılır. En bilinen yöntemlerden biri Laplace tahmini yöntemidir.

5. VERİ SETLERİNİN MANİPÜLASYONU

COSMIC veri tabanında yapılan incelemeler sonucu, analizini yapmak üzere üç veri seti seçilmiştir: Non – Coding Variants, All Mutations in Census Genes ve Copy Number Variants.

Veri setlerinde ön işleme (manipülasyon) yapılarak sadece istenen sütunlar alınmış, veriler projeye uygun hale getirilmiştir. Aynı zamanda yapılan bu işlemler kullanıcının uygulama arayüzünü daha rahat kullanmasını sağlamıştır. İşlenen veri setleri kaydedilmiş, sonrasında öğrenme algoritmaları ile analiz edilmiştir.

5.1. Non – Coding Variants (NCV) Veri Seti

Veri setine ait dosya, [1]'de “CosmicNCV” adıyla bulunabilir. Non-Coding terimi, herhangi bir proteini kodlamayan (üretmeyen) DNA parçaları için kullanılmaktadır. Geçmişte DNA'nın bu bölgeleri anlamsız görülse de, buralarda oluşan mutasyonların hastalıklara sebep olabileceği keşfedilmiştir. NCV veri seti 40 sütun (feature) bulundurmaktadır ancak projede sadece gerekli görülenler kullanılmıştır.

Kullanılan sütunlar:

"Primary site" : Örneğin (sample) kaynaklandığı birincil doku/kanser bilgisini içerir.

"Primary histology" : Örneğin ana histolojik sınıflandırması, hastalığın çeşididir.

"Histology subtype 1": Örneğin histolojik alt sınıflarıdır. Sadece bir alt seviye göz önünde bulundurulmuştur.

"zygosity": Örnekte gerçekleşen mutasyona dair bilgileri verir: homozigot, heterozigot veya bilinmeyen.

"genome position": Mutasyonun gerçekleştiği genomik koordinat bilgisidir.

Gerçekleşebilecek konum çeşitliliğinden dolayı eğitim aşamasında kayda değer bir rolü olmamış, aksine öğrenme oranını düşürmüştür.

"WT_SEQ": Mutasyon yaşanmadan önce, yani sağlıklı bireyde bulunan sekanstır.

"MUT_SEQ": Mutasyon gerçekleşikten sonra, olması gereken sekansın yerine gelen sekanstır.

"FATHMM_MKL_NON_CODING_SCORE": Orijinal veri setinde bu sütundaki değerler 0-1 arasında görülmektedir. COSMIC veri tabanında bu feature ile ilgili, p-değeri $\geq 0,7$ işlevsel olarak anlamlı olduğu belirtilmiştir. Bu bilgiler ışığında, değerler önemli-önemsiz olarak ayrıştırılmıştır. İşlevsel olarak anlamlı değerler ($p \geq 0,7$) için 1, diğer değerleri göstermek için 0 kullanılmıştır.

Index	Primary site	Primary histology	Histology subtype 1	zygosity	genome position	WT_SEQ	MUT_SEQ	FATHMM_MKL_NON_CODING_SCORE
0	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	1:29327428-29327428	G	C	0.09452
1	breast	carcinoma	NS	Unknown	17:7675161-7675161	G	C	0.99074
2	urinary_tract	carcinoma	NS	Unknown	17:82032399-82032399	G	C	0.95301
3	lung	carcinoma	NS	Unknown	1:173867913-173867913	G	A	0.78875
4	breast	carcinoma	lobular_carcinoma	Unknown	7:92956657-92956657	C	T	0.07532
5	central_nervous_system	glioma	astrocytoma_Grade_IV	Unknown	12:55958041-55958041	G	C	0.98048
6	prostate	carcinoma	adenocarcinoma	Unknown	19:16898133-16898133	C	T	0.07569
7	prostate	carcinoma	NS	Unknown	10:127776797-127776797	C	A	0.03701
8	oesophagus	carcinoma	adenocarcinoma	Unknown	4:17819411-17819411	T	A	0.10287
9	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	4:188947523-188947523	C	A	0.06258
10	haematopoietic_and_lymphoid_tissue	lymphoid_neoplasm	NS	Unknown	18:45723428-45723428	A	T	0.12629
11	oesophagus	carcinoma	adenocarcinoma	Unknown	15:52547184-52547184	A	T	0.25327
12	oesophagus	carcinoma	adenocarcinoma	Unknown	15:52547184-52547184	A	T	0.25327

Şekil 10. NCV veri setinin örnek gösterimi

5.1.1. Non – Coding Variants (NCV) Ön İşleme

Veri setinde keşif amaçlı uygulamalar yapılmış, WT_SEQ ve MUT_SEQ değerlerinin çeşitliliği incelenmiş, aşağıdaki sonuçlar elde edilmiştir.

```
In [8]: data["WT_SEQ"].value_counts()
Out[8]:
G          5460048
C          5427712
T          3396363
A          3376443
CA           12833
...
GTCCTGGACGGTACTCAG          1
ACATCAATCATCAATGGAA          1
AGAGCGCTGG          1
GCAGGC          1
GTCCCAGCTACTAGGAAGACATAGG          1
Name: WT_SEQ, Length: 54709, dtype: int64
```

Şekil 11. WT_SEQ değerleri

```
In [9]: data["MUT_SEQ"].value_counts()
Out[9]:
T          5686893
A          5635238
G          3125240
C          3118409
TT           12682
...
GCCATACTCATTA          1
AGGATGCTTTA          1
GTCCATCT          1
CAGCCAAAAGACACAT          1
CGTGTGTGCACGTGTGTG          1
Name: MUT_SEQ, Length: 21409, dtype: int64
```

Şekil 12. MUT_SEQ değerleri

Şekil 11 ve 12’ye bakılarak, örneklerde (satır) WT_SEQ sütunu için 54709, MUT_SEQ için 21409 farklı (unique) değer bulunduğu söylenebilir. Örneğin, WT_SEQ sütununda G bulunan 5460048 tane kayıt vardır.

Veri seti analiz edilmeden önce, kayıtlarda eksik bilginin bulunup bulunmadığı kontrol edilmiştir.

Index	Primary site	Primary histology	Histology subtype 1	zygosity	genome position	WT_SEQ	MUT_SEQ	FATHMM_MKL_NON_CODING_SCORE
18497312	skin	malignant_melanoma	NS	Unknown	16:69748993-69748993	G	A	0.91719
18497313	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	28:7579484-7579484	C	T	0.88839
18497314	upper_aerodigestive_tract	carcinoma	squamous_cell_carcinoma	Unknown	9:62843579-62843579	G	A	0.37839
18497315	oesophagus	carcinoma	adenocarcinoma	Unknown	9:108825195-108825198	TACT	nan	nan
18497316	stomach	carcinoma	NS	Unknown	15:101603498-101603498	G	T	0.09235
18497317	liver	other	neoplasm	Unknown	11:101891156-101891156	A	G	0.28812
18497318	liver	carcinoma	NS	Unknown	23:91563030-91563030	T	A	0.05936
18497319	breast	carcinoma	ductal_carcinoma	Unknown	10:9549090-9549090	C	T	0.09752
18497320	breast	carcinoma	ductal_carcinoma	Unknown	9:75181043-75181043	T	C	0.7875
18497321	pancreas	carcinoma	ductal_carcinoma	Unknown	9:94531317-94531317	A	C	0.3594
18497322	prostate	carcinoma	adenocarcinoma	Unknown	8:109787973-109787973	C	T	0.16534
18497323	endometrium	carcinoma	endometrioid_carcinoma	Unknown	15:71771092-71771092	G	A	0.98112
18497324	oesophagus	carcinoma	adenocarcinoma	Unknown	5:30113288-30113288	A	T	0.07385
18497325	prostate	carcinoma	adenocarcinoma	Unknown	10:99604271-99604271	C	T	0.05749
18497326	pancreas	carcinoma	ductal_carcinoma	Unknown	16:35395470-35395470	G	T	0.08773
18497327	haematopoietic_and_lymphoid_tissue	lymphoid_neoplasm	chronic_lymphocytic_leukaemia-small_lymphocytic_lymphoma	Unknown	21:26311382-26311382	C	T	0.06827
18497328	central_nervous_system	primitive_neuroectodermal_tumour-medulloblastoma	NS	Unknown	12:131511252-131511252	T	A	nan
18497329	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	23:84520576-84520576	G	A	0.04516
18497330	pancreas	carcinoma	NS	Unknown	10:125452876-125452876	G	A	0.62332
18497331	endometrium	carcinoma	endometrioid_carcinoma	Unknown	13:30952766-30952766	G	T	0.19542
18497332	liver	carcinoma	NS	Unknown	11:80876988-80876988	T	C	0.11239
18497333	endometrium	carcinoma	carcinosarcoma-malignant_mesodermal_mixed_tumour	Unknown	5:103687876-103687876	A	C	0.18285
18497334	biliary_tract	carcinoma	NS	Unknown	3:193957837-193957838	nan	TAT	nan
18497335	liver	carcinoma	NS	Heterozygous	8:70088767-70088767	T	G	0.15998
18497336	liver	carcinoma	NS	Unknown	14:51244688-51244688	A	G	0.76162

Şekil 13. Veri setinde eksik değerler bulunmaktadır.

```
In [7]: data.isnull().sum()
Out[7]:
Primary site                0
Primary histology           0
Histology subtype 1         0
zygosity                    0
genome position             0
WT_SEQ                     545493
MUT_SEQ                     739108
FATHMM_MKL_NON_CODING_SCORE 1312356
dtype: int64
```

Şekil 14. Veri setinde eksik değerlerin dağılımı

Görölmüştür ki, WT_SEQ, MUT_SEQ ve FATHMM sütunlarında null değerler (eksikler) vardır. Veri setinin tutarlı analiz edilebilmesi için bu satırlar tablodan silinmiştir. Örn. WT_SEQ sütununda 545493 null değer bulunmaktadır.

Null kayıtlar silindikten sonra, WT_SEQ ve MUT_SEQ sütunları tekrar incelenmiş, sekans çeşitliliğinin azaldığı görülmüştür. Kalan değerler sadece temel nükleotitlerdir (adenin, timin vb)

```
In [16]: #droptan sonra

In [17]: data['WT_SEQ'].value_counts()
Out[17]:
G    5395202
C    5363262
T    3222200
A    3204345
Name: WT_SEQ, dtype: int64

In [18]: data['MUT_SEQ'].value_counts()
Out[18]:
T    5510581
A    5488480
G    3096389
C    3089559
Name: MUT_SEQ, dtype: int64
```

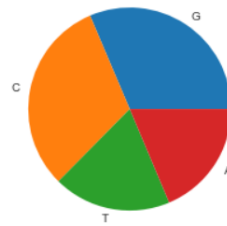
Şekil 15. Null değerler silindikten sonra WT_SEQ ve MUT_SEQ için değerler dağılımı

Mutasyondan önce (WT_SEQ) ve sonraki (MUT_SEQ) alfabe değerler dağılımı görsellenmiştir.

Yandaki Pasta (dairesel) grafikte, mutasyondan önce alfabenin dağılımı görülmektedir. Guanin ve Sitozin'in daha yoğun görüldüğü anlaşılabilir.

```
In [21]: beforeMut = NCV_df["WT_SEQ"].value_counts()
         alphabet = NCV_df["WT_SEQ"].value_counts().index
         plt.pie(beforeMut, labels = alphabet)

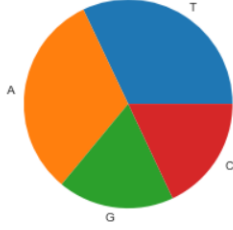
Out[21]: ([<matplotlib.patches.Wedge at 0x1ce9ee777b8>,
<matplotlib.patches.Wedge at 0x1ce9ee77cc0>,
<matplotlib.patches.Wedge at 0x1ce9ee97198>,
<matplotlib.patches.Wedge at 0x1ce9ee97630>],
[Text(0.6069598017129868, 0.9173874858011373, 'G'),
Text(-1.080506883709639, 0.20616710274940825, 'C'),
Text(-0.20755789084182127, -1.080240585216689, 'T'),
Text(0.916605516059877, -0.6081400561783501, 'A')])
```



Şekil 16. WT_SEQ pasta grafiği

```
In [20]: afterMut = NCV_df["MUT_SEQ"].value_counts()
MutAlphabet = NCV_df["MUT_SEQ"].value_counts().index
plt.pie(afterMut, labels = MutAlphabet)
```

```
Out[20]: ([<matplotlib.patches.Wedge at 0x1ce9eea1898>,
<matplotlib.patches.Wedge at 0x1ce9f328358>,
<matplotlib.patches.Wedge at 0x1ce9f07e4a8>,
<matplotlib.patches.Wedge at 0x1ce9f07ec18>],
[Text(0.5874763011994089, 0.9299847286536815, 'T'),
Text(-1.0916268331132442, 0.13546533588763432, 'A'),
Text(-0.13698883164129697, -1.0914366953724584, 'G'),
Text(0.9291639686315637, -0.5887735722642813, 'C')])
```



Şekil 17. MUT_SEQ pasta grafiği

Mutasyon gerçekleştikten sonraki dağılım ise Şekil 17’deki gibidir. Mutasyondan sonra Adenin ve Timin’in daha baskın olduğu görülmüştür.

FATHMM_MKL_NON_CODING_SCORE sütunundaki değerlerin 0-1 olarak encode edildiğinden bahsedilmişti. Düzenleme işlemi aşağıdaki kod satırı ile yapılmıştır.

```
# FATHMM_MKL_NON_CODING_SCORE column will be encoded 0-1
# given that, FATHMM non-coding score value > .7 is functionally significant (for further info: COSMIC database)
# 1: significant 0: non-significant

for i in range(len(data)):
    if(data["FATHMM_MKL_NON_CODING_SCORE"].values[i] > 0.7):
        data["FATHMM_MKL_NON_CODING_SCORE"].values[i] = 1
    else:
        data["FATHMM_MKL_NON_CODING_SCORE"].values[i] = 0
```

Şekil 18. FATHMM Non Coding Skoru sütunu için encoding yapan kod bloğu

Düzenlemeden sonra veri setinin görünümü şu şekildedir:

Index	Primary site	Primary histology	Histology subtype 1	zygosity	genome position	WT_SEQ	MUT_SEQ	FATHMM_MKL_NON_CODING_SCORE
0	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	1:29327428-29327428	G	C	0
1	breast	carcinoma	NS	Unknown	17:7675161-7675161	G	C	1
2	urinary_tract	carcinoma	NS	Unknown	17:82032399-82032399	G	C	1
3	lung	carcinoma	NS	Unknown	1:173867913-173867913	G	A	1
4	breast	carcinoma	lobular_carcinoma	Unknown	7:92956657-92956657	C	T	0
5	central_nervous_system	glioma	astrocytoma_Grade_IV	Unknown	12:55958841-55958841	G	C	1
6	prostate	carcinoma	adenocarcinoma	Unknown	19:16898133-16898133	C	T	0
7	prostate	carcinoma	NS	Unknown	10:127776797-127776797	C	A	0
8	oesophagus	carcinoma	adenocarcinoma	Unknown	4:17819411-17819411	T	A	0
9	ovary	carcinoma	mixed_adenosquamous_carcinoma	Unknown	4:188947523-188947523	C	A	0
10	haematopoietic_and_lymphoid_tissue	lymphoid_neoplasm	NS	Unknown	18:45723428-45723428	A	T	0

Şekil 19. FATHMM Non Coding Skoru sütunu için encoding işlemi sonucu

NCV veri seti üzerinde sınıflama (classification) algoritması olan Logistic Regression kullanılmıştır. Bu algoritma ile, mutasyon sonucu gelecek harfin ne olacağının tahmini yapılması amaçlanmaktadır.

NCV veri setinde bulunan veri tipleri değişkenlik göstermekle beraber yoğun olarak object tipindedir (Şekil 20). Bu verilerin eğitime sokulabilmesi için str veya int formatına çevrilip, sonrasında LabelEncoding yapılması gereklidir. LabelEncoding, kategorik değişkenleri etiketlemek amacıyla kullanılmış, veri seti bilgisayarın işleyebileceği hale getirilmiştir.

FATHMM sütunundaki değerleri 0 ve 1 ile göstermek de Label Encoding sayılabilir. Bu etiketleme manuel yapılabileceği gibi Python'da bulunan kütüphanelerden de faydalanılabilir.

```
In [5]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18497365 entries, 0 to 18497364
Data columns (total 8 columns):
#   Column                                Dtype
---  -
0   Primary site                          object
1   Primary histology                     object
2   Histology subtype 1                   object
3   zygoty                                object
4   genome position                       object
5   WT_SEQ                                object
6   MUT_SEQ                               object
7   FATHMM_MKL_NON_CODING_SCORE           float64
dtypes: float64(1), object(7)
memory usage: 1.1+ GB
```

Şekil 20. Dataframedeki veri tipleri

```

#%%% encode the alphabet (A,T,G,C) manually for mut_seq and wt_seq, so encoding will be same
# A:0 T:1 G:2 C:3

data["MUT_SEQ"].replace({"A": 0, "T": 1, "G": 2, "C":3}, inplace=True)
data["WT_SEQ"].replace({"A": 0, "T": 1, "G": 2, "C":3}, inplace=True)

#print(data.head())

#%%% converting dtype for label encoding

convert_dict = {'Primary site': 'string',
                'Primary histology': 'string',
                'Histology subtype 1' : 'string',
                'zygosity' : 'string',
                'genome position' : 'string',
                "WT_SEQ" : int,
                "MUT_SEQ": int,
                "FATHMM_MKL_NON_CODING_SCORE" : int
                }

data = data.astype(convert_dict)
# data.info()

```

Şekil 21. NCV veri setini düzenlemek için kod bloğu

Şekil 21’de, üst blokta alfabedeki harfler yerine sayılar getirilerek WT_SEQ ve MUT_SEQ sütunları üzerine manuel olarak LabelEncoding yapılmıştır.Alt blokta ise, dataframe’de bulunan sütunların veri tipleri şekildeki gibi değiştirilmiştir.

```

#%%% encode the remaining columns automatically
data_copy = pd.DataFrame(data.iloc[:,5])

from sklearn import preprocessing

le = preprocessing.LabelEncoder()
data_copy = data_copy.apply(le.fit_transform)

#%%% drop the columns in data that not-encoded

cols = data.iloc[:,5]
data.drop(cols, axis=1, inplace = True)

#%%% combine the encoded columns

frames = [data_copy,data]
final_df = pd.concat(frames, axis = 1)

#%%% save the final df, encoded version

final_df.to_csv("ncv_encoded.csv",index = False)

```

LabelEncoding, sklearn kütüphanesinde bulunan preprocessing sınıfının LabelEncoder() fonksiyonu ile otomatik olarak yapılabilir.

Manuel ve otomatik olarak etiketlenen veriler birleştirilerek veri setinin son hali eğitimde kullanılmak üzere yeni bir .csv dosyasına kaydedilmiştir.

Şekil 22. NCV setinde kalan sütunları otomatik kategorilendiren ve Label bilgilerini birleştiren kod bloğu

dfhead - DataFrame

Index	Primary site	Primary histology	Histology subtype 1	zygosity	genome position	WT_SEQ	MUT_SEQ	FATHMM_MKL_NON_CODING_SCORE
0	21	24	334	2	5818194	2	1	0
1	5	24	35	2	4111478	2	1	1
2	41	24	35	2	4153485	2	1	1
3	18	24	35	2	5364376	2	0	1
4	5	24	295	2	12962446	1	3	0
5	6	50	102	2	1700123	2	1	1
6	31	24	72	2	4611588	1	3	0
7	31	24	35	2	142264	1	0	0
8	20	24	72	2	9916050	3	0	0
9	21	24	334	2	10001611	1	0	0
10	14	70	35	2	4338252	0	3	0
11	20	24	72	2	3106368	0	3	0
12	20	24	72	2	10268721	3	0	0
13	31	24	72	2	9525166	1	2	0
14	5	24	187	2	1235764	3	1	0
15	18	24	480	2	7904652	1	2	1
16	22	23	35	2	13415133	0	2	0
17	14	70	148	2	2533211	2	0	0
18	22	24	187	2	13633576	2	0	0
19	17	24	35	2	8902976	3	1	0
20	33	72	35	0	2509401	2	0	0
21	22	24	35	2	4544528	1	3	0
22	20	24	72	2	11692818	2	0	0
23	20	24	72	2	10260233	1	0	0
24	17	24	35	0	12006153	3	0	0

Şekil 23. Veri setinin eğitime hazır, sütunların kategorilerine etiketlenmiş hali

Veri setindeki diğer özelliklerin (feature/sütun) çeşit sayısına bakacak olursak;

Name	Type	Size	Value
genomepos_count	Series	(14282800,)	Series object of pandas.core.series module
histSub_count	Series	(518,)	Series object of pandas.core.series module
prim_hist_count	Series	(113,)	Series object of pandas.core.series module
prim_site_count	Series	(44,)	Series object of pandas.core.series module


```

In [5]: x = df['Primary site'].value_counts()
In [6]: prim_site_count = df['Primary site'].value_counts()
In [7]: prim_hist_count = df['Primary histology'].value_counts()
In [8]: genomepos_count = df['genome position'].value_counts()
In [9]: histSub_count = df['Histology subtype 1'].value_counts()
In [10]:

```

Şekil 24. NCV’de kullanılan diğer sütunların değer dağılımı

Genome position sütunu için 14282800, Histology subtype için 518, Primary histology için 113, Primary site için ise 44 unique değer vardır. (WT_SEQ = MUT_SEQ = 4, zygosity = 3 farklı değer vardı)

5.2. All Mutations in Census Genes (MutExCen) Veri Seti

Veri setine ait dosya, [1]'de "CosmicMutantExportCensus" adıyla bulunabilir NCV'nin aksine, DNA'nın protein kodlayan (coding) kısımlarında meydana gelen mutasyonlar hakkında bilgilerin bulunduğu veri setidir. Orijinal dataset'te 41 sütun bulunmaktadır ancak projede sadece aşağıdakiler üzerine çalışılmıştır:

"Gene name", : Mutasyona uğrayan gen

"ID_sample", : Alınan örneğin (kişinin) ID'si (bkz. Bir örnekte birden çok gende birçok mutasyon oluşabilir)

"Primary site" : Örneğin (sample) kaynaklandığı birincil doku/kanser bilgisini içerir.

"Primary histology" : Örneğin ana histolojik sınıflandırması, hastalığın çeşididir.

"Histology subtype 1" : Örneğin histolojik alt sınıfıdır.

"Mutation Description" : Mutasyon tipi bilgisidir; substitution, deletion, insertion, complex, fusion vb

"Mutation genome position" : Mutasyonun gerçekleştiği genomik koordinat bilgisidir.

"FATHMM prediction" : Oluşan mutasyonun patojen (kanser,zararlı) veya nötr olması bilgisidir.

"Tumour origin" : Tümör tipi bilgisi içerir (NS, primary,metastasis, recurrent, secondary vb)

Kullanılan Veri setinin örnek gösterimi:

Index	Gene name	D_sample	Primary site	Primary histology	Histology subtype 1	Mutation Description	Mutation genome position	FATHMM prediction	Tumour origin
744711	BRAF	683058	pancreas	carcinoma	ductal_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1312096	BRAF	683066	pancreas	carcinoma	ductal_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1331007	BRAF	683114	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1367545	BRAF	683115	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1294082	BRAF	683116	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
368469	BRAF	683117	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
894314	BRAF	683118	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1418313	BRAF	683119	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
88503	BRAF	683120	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
610443	BRAF	683121	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1459583	BRAF	683122	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
695801	BRAF	683123	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
158958	BRAF	683124	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
209165	BRAF	683125	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
486330	BRAF	683126	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
933999	BRAF	683127	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
539612	BRAF	683128	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1534995	BRAF	683129	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
1420416	BRAF	683130	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary
599919	BRAF	683131	thyroid	carcinoma	papillary_carcinoma	Substitution - Missense	7:140753336-140753336	PATHOGENIC	primary

Şekil 25. Cosmic Mutant Export Census veri setinin örnek gösterimi

5.2.1. All Mutations in Census Genes (MutExCen) Veri Ön İşleme

Veri daha iyi incelenmek adına “ID_sample” sütunundaki değerlere göre azalan sırada sıralanmıştır.

Veri seti kabaca incelenmiş, null içerip içermediği kontrol edilmiştir (Şekil 30).

```
In [4]: df.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1666170 entries, 744711 to 726132
Data columns (total 9 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Gene name                            1666170 non-null object
1   ID_sample                            1666170 non-null int64
2   Primary site                         1666170 non-null object
3   Primary histology                    1666170 non-null object
4   Histology subtype 1                  1666170 non-null object
5   Mutation Description                 1666170 non-null object
6   Mutation genome position             1539367 non-null object
7   FATHMM prediction                    1353732 non-null object
8   Tumour origin                        1666170 non-null object
dtypes: int64(1), object(8)
memory usage: 127.1+ MB
```

Şekil 26. Veri seti hakkında genel bilgi

Görüldüğü üzere, Mutation genome position ve FATHMM prediction sütunları eksik bilgi içermektedir. Mutation genome position sütunu hem eksik bilgi içermesi, hem de eğitimde yeterince etkili olmaması sebebiyle (çok fazla unique değer içerir) tablodan çıkarılmıştır. FATHMM prediction sütunundaki null değerler ise yerine “unknown” konularak halledilmiştir. Sonraki adımda ise, Label Encoding yapılabilmesi için dataframe’deki veri tipleri object’ten string’e dönüştürülmüştür (Şekil 31).

```
##%%
print(df.info()) # mutation genome position ve FATHMM predictionda null variablelar var
# genome position ı sil, FATHMM için üçüncü bir değer oluştur = unknown

df.drop(["Mutation genome position"], inplace = True, axis = 1)

df["FATHMM prediction"].replace({np.nan:"unknown"}, inplace = True)

print(df.info())
print(df["FATHMM prediction"].value_counts())

##%%

df = df.astype("string")
print(df.info())
```

Şekil 27. Mutant Export Census veri setinde düzenleme yapan kod bloğu

Genome position sütunu, mutasyonun genomik konumunu 13:335-534 şeklinde tutmaktadır. Genomik konum fazlaca detaylı olduğundan ve çok fazla değer içerdiğinden, analizde kullanılabilmesi için sadece mutasyonun gerçekleştiği kromozom bilgisi tutulması tercih edilmiş, değerler buna göre dönüştürülmüştür.

```
### changing genome position column into chromosome id
for i in range(len(data)):
    chro = data["Mutation genome position"].values[i]
    chro = chro.split(":")
    data["Mutation genome position"].values[i] = chro[0]

data = data.rename(columns = {"Mutation genome position": "Chromosome ID"})
```

Şekil 28. Genom pozisyonu sütununun Kromozom ID olarak değiştirilmesi

Veri setinde bazı bilgiler xx_yy gibi noktalama işaretleri ile birlikte tutulmaktadır, analiz açısından bir sorun teşkil etmese de, kullanıcının tahminleme sistemini rahat kullanaabilmesi için bu işaretler kaldırılmıştır.

```
### replace "-" mark with space for ui friendly design
for i in range(len(data)):
    primS = data["Primary site"].values[i]
    primS = primS.replace("_", " ")
    #print(primS)
    data["Primary site"].values[i] = primS

    primH = data["Primary histology"].values[i]
    primH = primH.replace("_", " ")
    data["Primary histology"].values[i] = primH

    histS1 = data["Histology subtype 1"].values[i]
    histS1 = histS1.replace("_", " ")
    data["Histology subtype 1"].values[i] = histS1
```

Şekil 29. Kullanımı kolaylaştıracak değişiklikler yapılmıştır

Yine aynı sebeple, tablonun içindeki tüm veriler str.lower() fonksiyonu ile küçük harfe çevrilmiştir.

```
### unique value counts / keşif için -> to select the predict (y) variable
#genes = data["gene_name"].value_counts()
genes = len(data["Gene name"].unique()) # 710
sites = len(data["Primary site"].unique()) # 45
hists = len(data["Primary histology"].unique()) # 127
hist_subs = len(data["Histology subtype 1"].unique()) # 587
mut_desc = len(data["Mutation Description"].unique()) # 13
chromos = len(data["Chromosome ID"].unique()) # 24
fathmms = len(data["FATHMM prediction"].unique()) # 3
tumour_org = len(data["Tumour origin"].unique()) # 7
```

Şekil 30. MutExCen için sütunların eşsiz değer sayıları

Veri setinin barındırdığı eşsiz değer sayılarına bakılarak (Şekil 30) analizi yapılacak (tahmin ettirilecek) değer seçimi yapılmış, Kromozom ID'si ve FATHMM değeri uygun görülmüştür. Eşsiz değer sayısı sınıf sayısını temsil ettiğinden, tahmini yapılmak üzere çeşitliliği fazla olan bir sütun seçmek, hesaplama maliyetini arttıracaktır.

5.3. Copy Number Variants (CNA) Veri Seti

Veri setine ait dosya, [1]'de "CosmicCompleteCNA" adıyla bulunabilir. DNA'da gerçekleşen çoklu mutasyonların ve alel sayılarının incelendiği çalışmadır. Veri setinin kullanılan sütunları aşağıdaki gibidir:

- "Gene name": Kopya numarası segmentiyle örtüşen genin ismi
- "Primary site": Numunenin kaynaklandığı birincil doku/kanser bilgisidir
- "Primary histology": Örneğin ana histolojik sınıflandırması, hastalığın çeşididir.
- "TOTAL_CN": Majör ve Minör alel sayılarının toplamını barındırır
- "MINOR_ALLELE": En az görülen alelin kopya sayısıdır, çok fazla eksik bilgi içerdiğinden sütun çalışmamızdan çıkarılmıştır
- "MUT_TYPE": Gain yahut Loss olarak değerler alır
- "ID_STUDY": Çalışmanın benzeriz kimliğidir, projede index olarak kullanılmıştır
- "Chromosome:G_Start..G_Stop": Varyasyonun genomik koordinatı bilgisini içerir

5.3.1. Copy Number Variants (CNA) Veri Ön İşleme

Önceki veri setleri gibi, null değerlerden ayıklanmış, verilerin içerdiği noktalama işaretleri kaldırılmış ve tüm veriler küçük harf ile saklanmak üzere manipüle edilmiştir.

Varyasyonun genomik koordinat bilgisi, kromozom bilgisi olarak değiştirilmiş, MINOR_ALLELE sütunu çok fazla eksik değer barındırdığından tablodan silinmiştir.

Şekil.xx'te veri setinin eğitime hazır hali gösterilmiştir:

CNA_head - DataFrame

Index	gene_name	Primary site	Primary histology	TOTAL_CN	MUT_TYPE	Chromosome
0	tnn11 enst00000555948	pancreas	carcinoma	4	gain	1
1	ube3a enst00000232165	pancreas	carcinoma	4	gain	15
2	vps35	pancreas	carcinoma	4	loss	16
3	znf423 enst00000567169	pancreas	carcinoma	4	gain	16
4	bcar1 enst00000420641	pancreas	carcinoma	4	gain	16
5	zpbp2	pancreas	carcinoma	4	gain	17
6	rps6kb1 enst00000443572	pancreas	carcinoma	4	gain	17
7	lrrc61 enst00000493307	pancreas	carcinoma	4	gain	7
8	myl10	pancreas	carcinoma	4	gain	7
9	thrb enst00000356447	pancreas	carcinoma	4	gain	3
10	il17rc enst00000455057	pancreas	carcinoma	4	gain	3
11	heca	pancreas	carcinoma	4	gain	6
12	micu1 enst00000418483	pancreas	carcinoma	4	gain	10
13	majin enst00000432175	pancreas	carcinoma	4	gain	11
14	dpp3 enst00000532677	pancreas	carcinoma	4	gain	11
15	igdcc4	pancreas	carcinoma	3	gain	15
16	dcaf15	pancreas	carcinoma	4	gain	19
17	snpc2	pancreas	carcinoma	4	gain	19
18	zswim9	pancreas	carcinoma	4	gain	19
19	steap3	pancreas	carcinoma	4	loss	2
20	ctnna2 enst00000496558	pancreas	carcinoma	4	gain	2
21	mgme1 enst00000377704	pancreas	carcinoma	4	gain	20
22	tgif2-rab5if	pancreas	carcinoma	4	gain	20
23	lipi	pancreas	carcinoma	3	gain	21
24	ap4s1 enst00000334725	pancreas	carcinoma	4	gain	14

Şekil 31. CNA veri seti örnek gösterimi

6. ANALİZ

Analiz aşamasında yapılan çalışmalar kabaca iki bölümde toplanmıştır: algoritmaların performansını ölçen ve arayüzde kullanılıp kullanılmamasına ilişkin fikir edinilmesini sağlayan Deneysel (Experimental) Analiz ve tasarlanan arayüzde tahminleme yapmada kullanılan UI Analiz.

Üç veri seti de çeşitli algoritmalarla analiz edilmiş, eğitimler sonucu oluşturulan ve uygun görülen modeller kullanıcılar için tasarlanan tahminleme arayüzünün arka planında kullanılmıştır.

6.1. Deneysel (Experimental) Analiz

Seçilen veri setlerinin tamamı bu aşamada analiz edilmiştir. Analizlerde sınıflandırma algoritmaları tercih edilmiştir (veri setleri sınıflandırmaya elverişlidir).

6.1.2. Mutation Export Census ile Deneysel Analiz

Mutation Export Census (MutExpCensus) veri seti, makine öğrenmesi algoritmaları ile analiz edilmiş, algoritmaların performansı incelenirken uygulaması yapılacak tahminleme sistemi için de fikir oluşturmuştur.

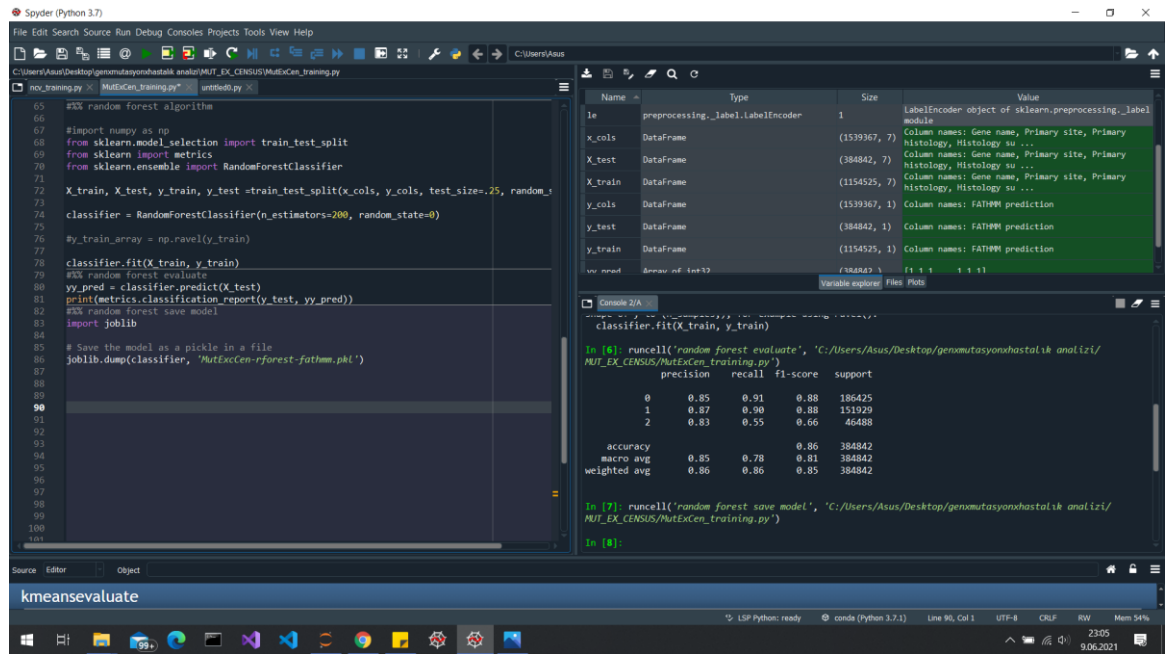
Veri seti kategorik (non-numeric) bilgiler içermesi sebebiyle, eğitime sokulmadan önce Scikit-Learn kütüphanesindeki preprocessing fonksiyonlarından biri olan LabelEncoder() ile sayısallaştırılarak bilgisayarın anlayabileceği hale getirilmiştir. Hesaplama kullanılan bilgiler (features) ile tahmini yapılan değer (predict) ayrı ayrı etiketlenmiştir, çıktının (prediction) decode edilerek doğruluğunun kontrol edilmesi amaçlanmıştır.

Eğitim sonucu oluşturulan modeller joblib kütüphanesi vasıtasıyla tekrar kullanılabilirlik üzere kaydedilmiştir.

Veri seti incelemesi ve manipülasyonu aşamalarında yapılan çalışmalar sonucu MutExpCensus verisi için tahminlemesi yapılacak değerler FATHMM prediction ve Chromosome ID seçilmiştir. Bu iki değere yönelik eğitimler gerçekleştirilmiş ve

Chromosome ID arayüzde kullanılmak üzere seçilmiştir (bkz. 6.1.3). FATHMM prediction 3 değer almakta ve mutasyonun etkisini göstermektedir: Pathogenic, Neutral ve Unkown.

Aşağıda FATHMM prediction değişkeni için çeşitli algoritmalarla tahminlemeler verilmiştir, 0 etiketi (sınıfı) neutral, 1: pathogenic, 2: unknown sınıflarını göstermektedir.



```
65 #%% random forest algorithm
66
67 #import numpy as np
68 from sklearn.model_selection import train_test_split
69 from sklearn import metrics
70 from sklearn.ensemble import RandomForestClassifier
71
72 X_train, X_test, y_train, y_test = train_test_split(x_cols, y_cols, test_size=.25, random=
73
74 classifier = RandomForestClassifier(n_estimators=200, random_state=0)
75
76 #y_train_array = np.ravel(y_train)
77
78 classifier.fit(X_train, y_train)
79
80 #%% random forest evaluate
81 yy_pred = classifier.predict(X_test)
82 print(metrics.classification_report(y_test, yy_pred))
83
84 # Save the model as a pickle in a file
85 joblib.dump(classifier, "MutExcCen-rforest-fathmm.pkl")
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
```

```
In [6]: runcell('random forest evaluate', 'C:/Users/Asus/Desktop/genmutasyonhastalik analizi/
Mut_EX_CENSUS/MutExcCen_training.py')
precision    recall  f1-score   support

0           0.95      0.91      0.88      186425
1           0.87      0.90      0.88      151929
2           0.83      0.55      0.66       46488

accuracy          0.85      0.78      0.86      384842
macro avg          0.86      0.78      0.85      384842

In [7]: runcell('random forest save model', 'C:/Users/Asus/Desktop/genmutasyonhastalik analizi/
Mut_EX_CENSUS/MutExcCen_training.py')

In [8]:
```

Name	Type	Size	Value
le	preprocessing_label.LabelEncoder	1	LabelEncoder object of sklearn.preprocessing_label module
x_cols	DataFrame	(1539367, 7)	Column names: Gene name, Primary site, Primary histology, Histology su ...
X_test	DataFrame	(384842, 7)	Column names: Gene name, Primary site, Primary histology, Histology su ...
X_train	DataFrame	(1154525, 7)	Column names: Gene name, Primary site, Primary histology, Histology su ...
y_cols	DataFrame	(1539367, 1)	Column names: FATHMM prediction
y_test	DataFrame	(384842, 1)	Column names: FATHMM prediction
y_train	DataFrame	(1154525, 1)	Column names: FATHMM prediction

Variable explorer: File: Plot

Console 2/A

```
classifer.fit(X_train, y_train)
```

```
In [6]: runcell('random forest evaluate', 'C:/Users/Asus/Desktop/genmutasyonhastalik analizi/
Mut_EX_CENSUS/MutExcCen_training.py')
```

	precision	recall	f1-score	support
0	0.95	0.91	0.88	186425
1	0.87	0.90	0.88	151929
2	0.83	0.55	0.66	46488
accuracy	0.85	0.78	0.86	384842
macro avg	0.86	0.78	0.85	384842

```
In [7]: runcell('random forest save model', 'C:/Users/Asus/Desktop/genmutasyonhastalik analizi/
Mut_EX_CENSUS/MutExcCen_training.py')
```

```
In [8]:
```

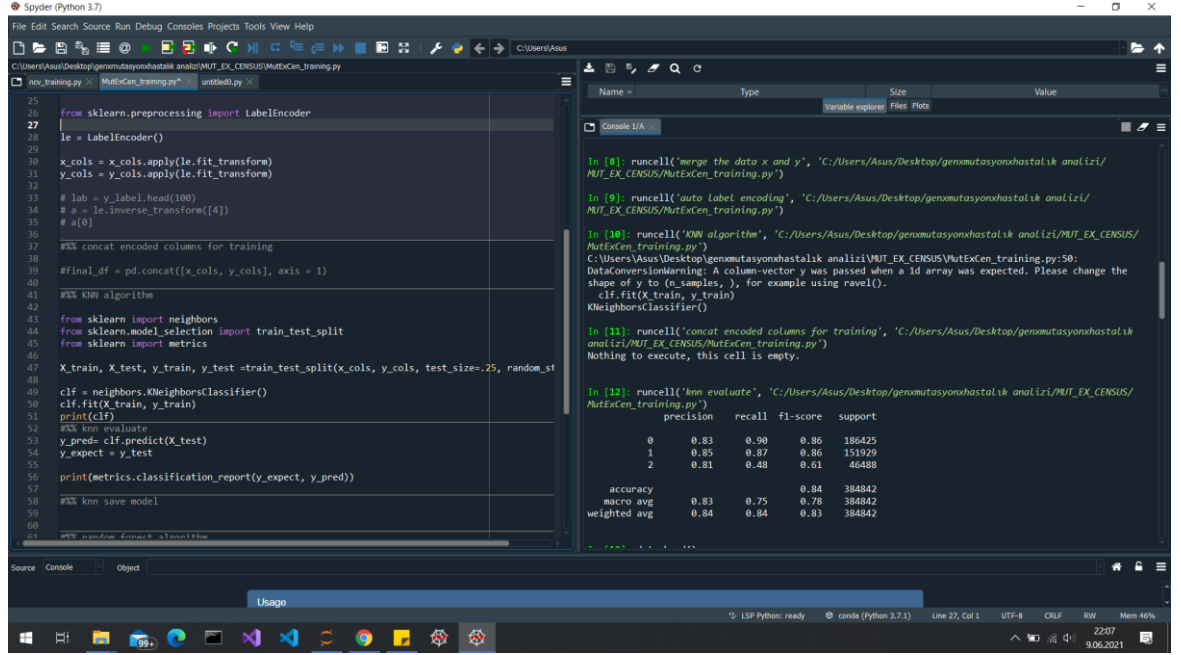
Source: Editor: Object

kmeansevaluate

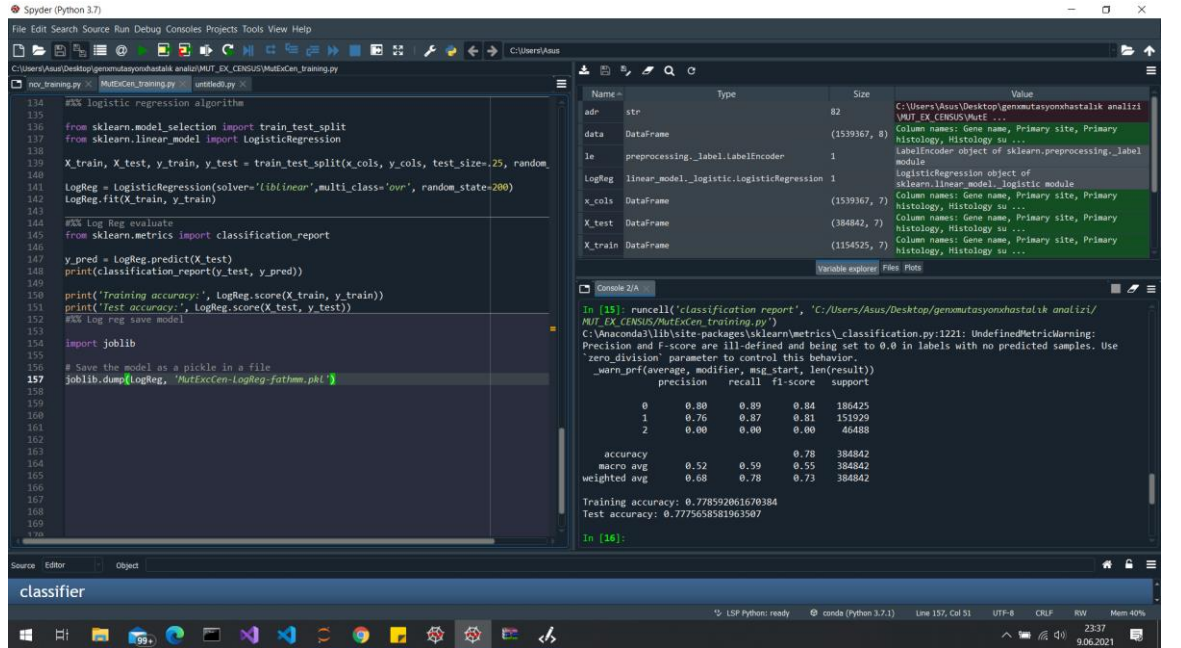
LSP Python: ready | conda (Python 3.7.1) | Line 96, Col 1 | UTF-8 | CRLF | RW | Mem 54%

2305 | 9.06.2021

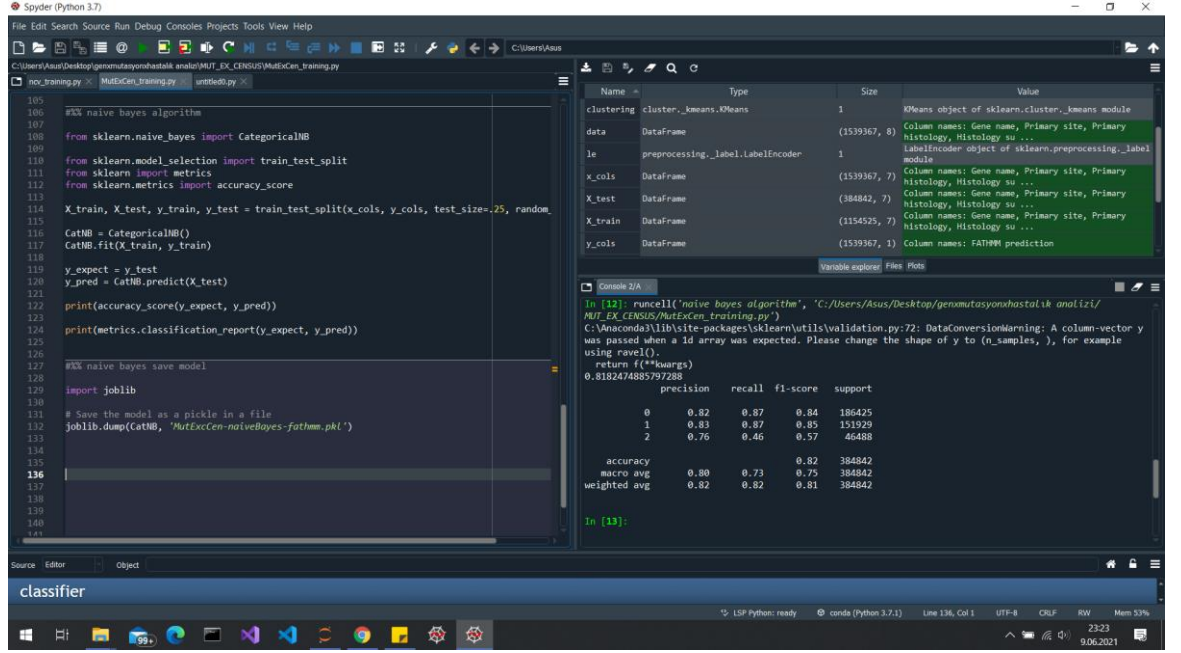
Şekil 32. Random Forest ile FATHMM tahminleme



Şekil 33. KNN ile FATHMM tahminleme



Şekil 34. Logistic Regression ile FATHMM tahminleme

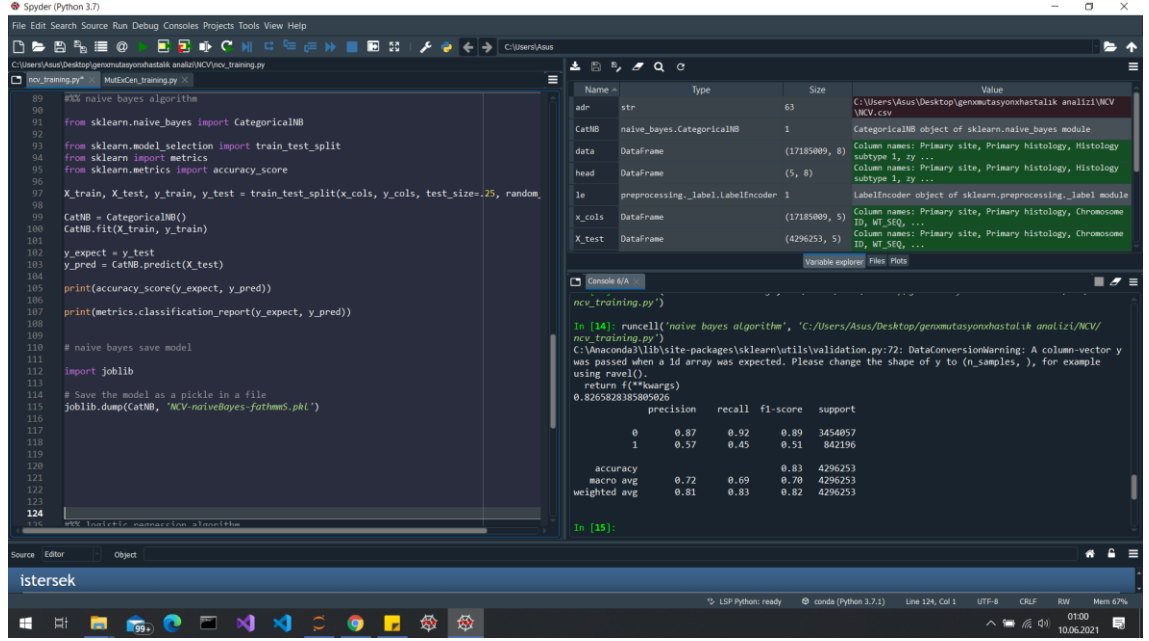


Şekil 35. Naive Bayes ile FATHMM tahminleme

3 sınıfımız olduğu için Logistic Regression algoritmasında multiclass parametresi gönderilmiştir. Veriler kategorik tipte olduğundan Categorical Naive Bayes tercih edilmiştir. FATHMM için yapılan tahminlerde, %86 doğruluk oranı ile en iyi çalışan algoritma Random Forest'tır.

6.1.3. NCV ile Deneysel Analiz

NCV veri setinde tahmin edilmesi uygun görülen değerler WT_SEQ ve Chromosome ve FATHMM Score'dur. Bu değerleri tahminlemek üzere KNN, Naive Bayes ve Logistic Regression algoritmaları kullanılmıştır. Çalışmalarda istenilen performans elde edilememiş, bu sebeple veri seti arayüzde kullanılmamıştır.



Şekil 36. Naive Bayes ile FATHMM skoru tahminleme

Yukarıdaki şekilde, Naive Bayes ile FATHMM skoru tahminleme sonuçları verilmiştir. Model doğruluğu %83'tür ancak kurulan model sınıf-1'i yeterince iyi hesaplayamamaktadır.

Aynı algoritma ile Kromozom bilgisi de tahmin ettirilmeye çalışılmış, %8 gibi çok düşük bir doğruluk elde edilmiştir.

WT_SEQ sütunu, nokta mutasyonundan önceki nükleotid (A-T-G-C) bilgisini barındırır. Logistic Regression (accuracy: %46), ve Naive Bayes (accuracy: %54) algortimaları ile bu değer tahmin ettirilmeye çalışılmıştır.

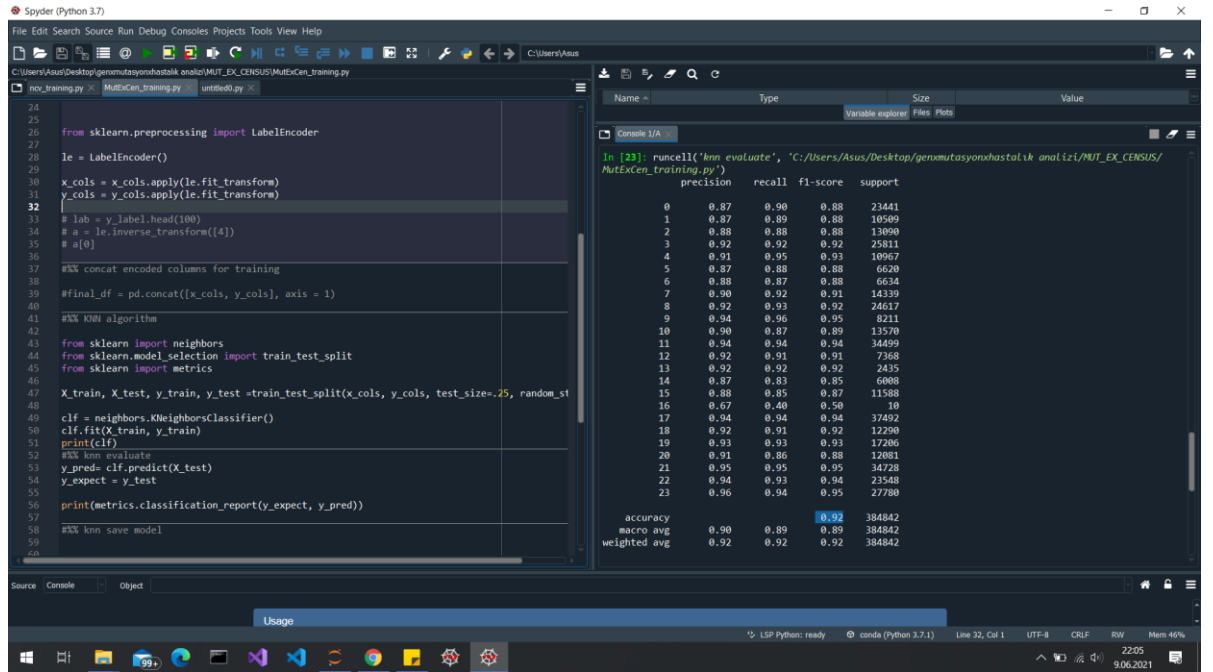
Yapılan analizler sonucunda, veri seti arayüzde kullanılmak için elverişli bulunmamıştır.

6.2. UI ANALİZ

Deneysel aşamada eğitilip elde edilen modellerden ikisi, kullanıcıların hizmetine sunulmuştur. Analizi yapılan modeller “pickle” kütüphanesi ile kaydedilip arayüzde çağırılarak kullanılmıştır.

6.2.1. KNN ile Kromozom Tahminleme

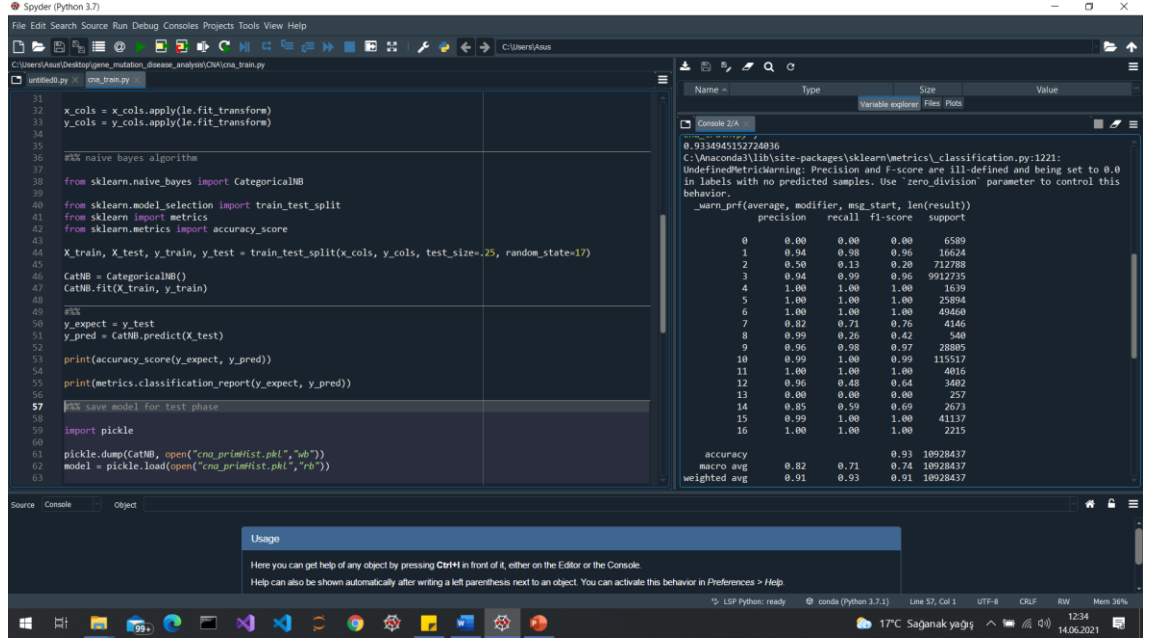
Cosmic Mutant Export Census verisi kullanılarak eğitilen model, kullanıcıdan çeşitli bilgiler alınarak mutasyonun gerçekleştiği kromozomu tahmin etmektedir. KNN ile eğitilen bu modelin öğrenme doğruluğu %92’dir. Homo sapiens türü (insanlar) 24 adet kromozom içerdiğinden, sınıf sayısı 24’tür. Sınıf isimleri temsilidir, etiketlemeye tabii tutulduğu için 5 numaralı sınıf direkt olarak 5.kromozoma karşılık gelmemektedir.



Şekil 37.. KNN ile Kromozom Tahminleme (MutExCen)

6.2.2. Naive Bayes ile Primary Histology Tahminleme

Cosmic Complete CNA verisi kullanılarak eğitilen model, kullanıcıdan çeşitli bilgiler alınarak mutasyonun sebep olduğu hastalığı (Primary histology) tahmin etmektedir. Naive Bayes ile eğitilen bu modelin öğrenme doğruluğu %93’tür.



Şekil 38. Naive Bayes ile Primary Histology Tahminleme (CNA)

Veri seti 17 çeşit hastalık içerdiğinden modeldeki sınıf sayısı 17’dir. Görseldeki sayısal değerler hastalık isimlerine denk gelmektedir.

Kullanılan iki veri seti de kategorik değerler içerir ve LabelEncoder() ile etiketlenmeye tabii tutulur. Hesaplama sonuçlarının anlaşılabilmesi için döndürülen sayısal değerlerin decode edilmesi gerekmektedir. Bu işlem test.py projelerinde oluşturduğumuz ve label_encode_decode klasörlerinde sakladığımız tablolar ile yapılmıştır.

A	B	C
unlabeled	labeled	
carcinoma,3		
glioma,6		
ns,12		
other,13		
haematopoietic neoplasm,7		
chondrosarcoma,4		
malignant melanoma,10		
carcinoid-endocrine tumour,2		
adrenal cortical carcinoma,1		
lymphoid neoplasm,9		
sarcoma,15		
leiomyoma,8		
aberrant crypt foci,0		
mesothelioma,11		
pheochromocytoma,14		
germ cell tumour,5		
thymoma,16		

Şekil 39. Primary histology değerleri ve karşılık gelen sınıflar

Örneğin primary histology için tutulan tablo yandaki gibidir:

Unlabeled sütunu hastalık isimlerini, labeled sütunu ise LabelEncoder() uygulanıp şifrelenen sayısal karşılıkları tutmaktadır. Bu dosyalar, sonraki aşama olan test fazında kullanılacaktır.

7. KULLANICI ARAYÜZÜ

Proje arayüzü python Django framework'ü kullanılarak oluşturulmuştur. Django seçilmesindeki temel etken eğitim ve tahmin etme adımlarının python kodu ile çalıştırılmasıdır. 2 veri seti üzerinde işlem yapıldığı için ve bu girdiler birbirinden farklı olduğu için 2 farklı html dosyası oluşturulmuştur, arayüzde oluşturulan diğer dosyalar bu iki veri seti için ortak (benzer) kullanılmıştır. Bu dosyalardaki form elementleri uygun şekilde doldurulduğunda kullanıcı farklı bir ekrana yansıtılacak ve burada girilen veriler sonucunda çıkarılan tahmin kullanıcıya gösterilecektir.

7.1. Arayüz Kapsamında Oluşturulan Dosyalar

Bir django projesi başlattığımızda dosyalardan bazıları framework tarafından oluşturulmakta. Daha sonra proje kapsamında dosya ekleyebilir veya dosyalar üzerinde oynamalar yapılabilir. Oluşturulan veya değiştirilen dosyalar:

- serverFunctions.py

```
import sys
from subprocess import run, PIPE
from django.shortcuts import render

# buraya fonksiyonlarımızı yazıyoruz

# home.html'i yüklüyor
def showWebPage(request):
    return render(request, 'index.html')

def returnPredictValue(request):
    geneName = request.POST.get('geneName').lower()
    PrimarySite = request.POST.get('PrimarySite').lower()
    TotalCN = request.POST.get('TotalCN').lower()
    mutationType = request.POST.get('mutationType').lower()
    Chromosome = request.POST.get('Chromosome').lower()
    out = run([sys.executable, "C://Users//Bilal Günden//Desktop//CNA//cna_test.py", geneName, PrimarySite, TotalCN, mutationType, Chromosome], shell=False, stdout=PIPE)
    print(out)

    return render(request, 'index.html', {'data1': out.stdout.decode('UTF-8')})
```

Şekil 40. serverFunctions.py

Bu dosyada 2 fonksiyon kullanıldı.

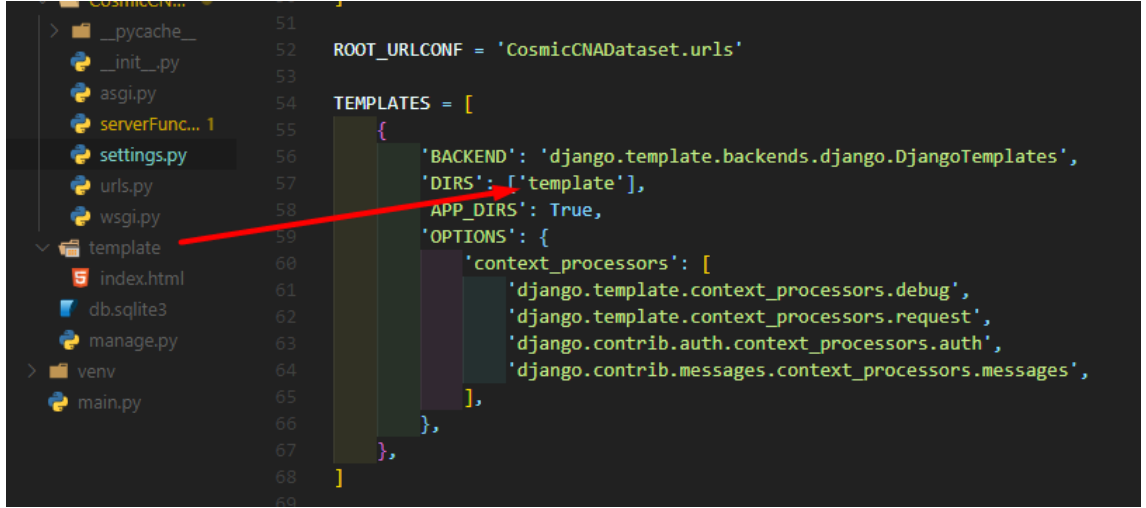
ShowWebPage() fonksiyonu oluşturduğumuz index.html dosyasını sunucu üzerinde göstermek için oluşturuldu.

ReturnPredictValue() fonksiyonu ise kullanıcıdan alınan girdileri test dosyasında çalıştırmak üzere uygun hale getirip bunları argüman olarak dosyaya göndermekte.

Fonksiyon içerisinde kullanılan run() fonksiyonu dizindeki dosyaya argümanları

gönderir ve dosyayı çalıştırır. Argümanlar test dosyasında uygun şekilde gönderildiğinde dosya çalışır ve bir çıktı oluşturur. Bu çıktı UTF-8 tipine uygun olarak şekilde decode edilir ve html dosyasında belirtilen yerde yazdırılmak üzere sonucu geri döndürür.

- Settings.py



Şekil 41. settings.py

Bu dosyada Django tarafından otomatik oluşturulmakta fakat projenin sağlıklı çalışması için dosya üzerinde değişiklikler yapıldı. Index.html dosyasının sunucu üzerinde görülebilmesi için bu dosyanın sunucunun template yoluna eklenmesi gerekli.

- urls.py



Şekil 42. urls.py

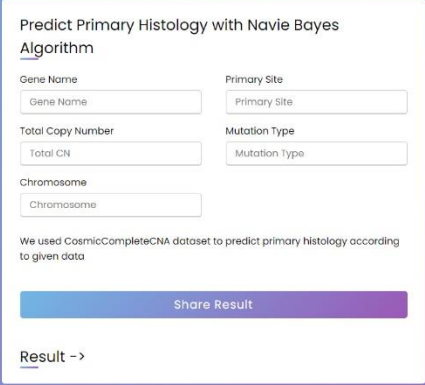
Bu dosyada sayfa üzerinde hangi işlemler gerçekleştiğinde hangi olayların çalıştırılacağı belirlenmekte.

^\$ ifadesi sayfanın yüklenmesini temsil etmektedir. Yani sunucunun yüklenmesi tamamlandığında serverFunctions dosyasındaki showWebPage fonksiyonu çalıştırılır.

^formComplete ise html dosyasında tanımlanan bir olaydır. Kullanıcı bütün girdileri girdikten sonra form başarılı bir şekilde dolduruldu sayılır ve serverFunctions dosyasındaki returnPredictValue fonksiyonu çalıştırılır.

- **Index.html**

CNA veri seti kullanılarak hastalık tahmini yapılmıştır



Predict Primary Histology with Navie Bayes Algorithm

Gene Name

Primary Site

Total Copy Number

Mutation Type

Chromosome

We used CosmicCompleteCNA dataset to predict primary histology according to given data

[Share Result](#)

[Result ->](#)

Şekil 43. CNA veri seti için index.html

Kullanıcıdan verileri aldığımız ve kullanıcıya sonucu yazdırdığımız html dosyası.

Design için CSS kullanıldı. Sayfa üzerindeki inputların her birinin kendi özgü bir name etiketi var. Bu etiket üzerinden inputlara girilen değerler toplanır ve bir değişkende saklanır.


```

<body>
  <div class="container">
    <div class="title">Predict Primary Histology with Navie Bayes Algorithm</div>
    <form action="/formComplete/" method="post">
      {% csrf_token %}
      <div class="user-details">
        <div class="input-box">
          <span class="details">Gene Name</span>
          <input type="text" placeholder="Gene Name" name="geneName" required>
        </div>
        <div class="input-box">
          <span class="details">Primary Site</span>
          <input type="text" placeholder="Primary Site" name="PrimarySite" required>
        </div>
        <div class="input-box">
          <span class="details">Total Copy Number</span>
          <input type="text" placeholder="Total CN" name="TotalCN" required>
        </div>
        <div class="input-box">
          <span class="details">Mutation Type</span>
          <input type="text" placeholder="Mutation Type" name="mutationType" required>
        </div>
        <div class="input-box">
          <span class="details">Chromosome</span>
          <input type="text" placeholder="Chromosome" name="Chromosome" required>
        </div>
      </div>
      <span class="details">We used CosmicCompleteCNA dataset to predict primary histology according to given data</span>
      <div class="button">
        <input type="submit" value="Share Result">
      </div>
    </form>
  </div>

```

Şekil 44. Formda doldurulan değerler değişkenlere atılır

Form başarılı bir şekilde doldurulduktan sonra fonksiyondan dönen tahmin değeri sayfa üzerine basılır. Sayfa üzerinden formun aşağısında gönderilen veri belirtilmiştir.

```

<div class="title">Result - {{data1}}
</div>
</div>

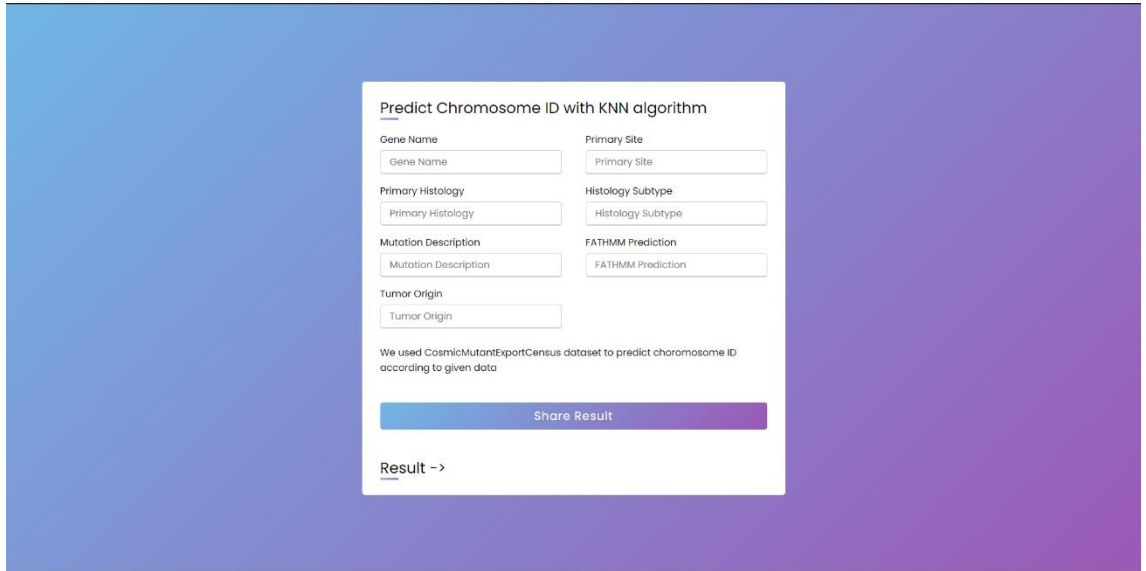
```

Şekil 45. Sonuç değeri bastırılır

Ekran görüntüleri Cosmic CNA veri seti için oluşturulan dosyalara aittir. Cosmic MEC veri seti için fonksiyonlar ve index.html üzerinde çeşitli değişiklikler yapılmıştır. Mantık açısından aynı olmasına rağmen kullanıcı girdi sayısı ve girdi isimleri farklı olduğu için belirtilen fonksiyonlarda modifikasyon gerçekleştirildi.

- **Index.html**

Mutant Export Census veri seti kullanılarak mutasyonun gerçekleştiği kromozom tahmini yapılmıştır



Şekil 46. MutExpCen veri seti için index.html

- **serverFunctions.py**

```
import sys
from subprocess import run, PIPE
from django.shortcuts import render

# buraya fonksiyonlarımızı yazıyoruz

# home.html'ı yükleyon
def showWebPage(request):
    return render(request, 'index.html')

def returnPredictValue(request):
    geneName = request.POST.get('geneName').lower()
    PrimarySite = request.POST.get('PrimarySite').lower()
    primaryHistology = request.POST.get('primaryHistology').lower()
    histologySubtype = request.POST.get('histologySubtype').lower()
    mutationDescription = request.POST.get('mutationDescription').lower()
    FATHMWPrediction = request.POST.get('FATHMMPrediction').lower()
    tumorOrigin = request.POST.get('tumorOrigin').lower()
    out = run([sys.executable, "C://Users//Bilal Gunden/Desktop//MUT_EX_CENSUS//mutexcen_test.py", geneName, PrimarySite, primaryHistology, histologySubtype, mutationDescription, FATHMWPrediction, tumorOrigin], capture_output=True, text=True)
    print(out)
    return render(request, 'index.html', {'data1':out.stdout.decode("UTF-8")})
```

Şekil 47. MutExpCen için serverFunctions.py

Web uygulamasının çalıştırılması için, Python manage.py runserver 127.0.0.1:8002 kodu girilmelidir.

8. TAHMİNLEME UYGULAMASININ ÇALIŞTIRILMASI

Web uygulamamız kullanılarak tahmin yapma aşaması bu başlıkta gösterilmiştir. Form yapısı ile kullanıcıdan girdi değerleri alınır ve “Share Results” butonuna basıldığında uygulama arka plandaki test.py dosyasını çalıştırarak girilen değerlere özgü tahminlerde bulunur.

Verilerimizin kategorik olduğundan (non-numeric) bahsetmiştik. Test.py dosyaları, formdan gelen sözel değerlerini bilgisayarın anlayabileceği şekilde sayısallaştırır (encodelar), karşılık gelen sayısal değerleri UI Analiz aşamasında oluşturulan modele (pickle dosyası) atar ve tahminleme yapmasını sağlar. Model analiz sonucu sayısal bir değer (sınıf) döndürür, bu sınıfı kullanıcının anlayabilmesi için çıktı değer decode’lanarak bastırılır. Test.py’deki encode-decode işlemlerinde eğitim aşamasında oluşturulup kaydedilen excel tabloları (.csv) dosyaları kullanılmıştır.

Test aşamasında LabelEncoder() fonksiyonunun tercih edilmeme sebebi veri setlerinin çok büyük olması dolayısıyla okuma ve kodlama işlemlerinin zaman almasıdır. İstenirse sklearn kütüphanesi çağırılarak LabelEncoder() da kullanılabilir, aynı sayısal etiket değerleri döndürülecektir, sayısal karşılık atama işlemi fonksiyon içinde random (rastgele) yapılmadığından herhangi bir sorun teşkil etmeyecektir.

Modellerin etkin kullanılabilmesi için kullanıcının forma modelin daha önce gördüğü feature değerlerini görmesi gerekir. Örn. Gen ismi yerine saçma bir değer verilirse, bu saçma değerın etiketsel karşılığı olmadığından modele gönderilemeyecektir.

İki veri seti için de örnek kullanım aşağıda bulunmaktadır, modelin doğruluğunu gösterebilmek amacıyla veri setlerinden rastgele bir kayıt (satır) seçip tahmin etmesi istenmiştir.

Proje kodlarına [2] ve [3]’den ulaşılabilir. [2]’de yaptığımız deneysel analizlere ait kodları, [3]’te ise arayüze erişilebilir.

8.1. Mutation Export Census Verisi ile Tahminleme

Girdide kullanılmak üzere rastgele seçilen kayıt şeklindeki gibidir. Gen ismi, birincil bölge, birincil hastalık, hastalık alt dalı, mutasyon tanımı, FATHMM (mutasyonun etkisi), ve tümör değerleri girilerek Kromozom bilgisi tahmini istenmektedir.

Kullandığımız modelin öğrenme doğruluğunun (accuracy) %92 olduğunu analiz aşamasında belirtmiştik. Aşağıdaki değerlerin girilmesi sonucu modelin Chromosome ID'yi 3 olarak bulması gerekmektedir.

test_mec - DataFrame								
Index	Gene name	Primary site	Primary histology	Histology subtype 1	Mutation Description	Chromosome ID	FATHMM prediction	Tumour origin
33442	pik3ca	stomach	carcinoma	adenocarcinoma	substitution - missense	3	pathogenic	ns

Şekil 48. MutExCen için örnek test verisi

Predict Chromosome ID with KNN algorithm

Gene Name

pik3ca

Primary Site

stomach

Primary Histology

carcinoma

Histology Subtype

adenocarcinoma

Mutation Description

substitution - missense

FATHMM Prediction

pathogenic

Tumor Origin

ns

We used CosmicMutantExportCensus dataset to predict chromosome ID according to given data

Share Result

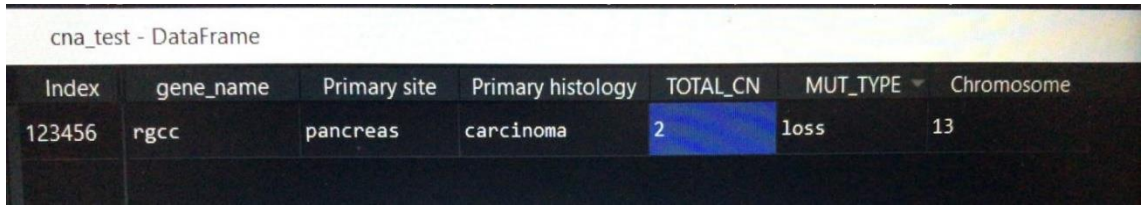
Result -> kromozom (chromosome ID) : 3
possibility : % 100.0

Şekil 49. Kromozom bilgisi tahminleme örneği

Şekil xx'e bakılarak modelin Kromozom bilgisini doğru olarak tahmin etmiştir. Possibility değeri, tahmin edilen sınıfın ne kadar olasılıkla kazandığını gösterir. Buradaki örneğe göre, 3'ten başka kromozom olma olasılığı yoktur/ çok düşüktür.

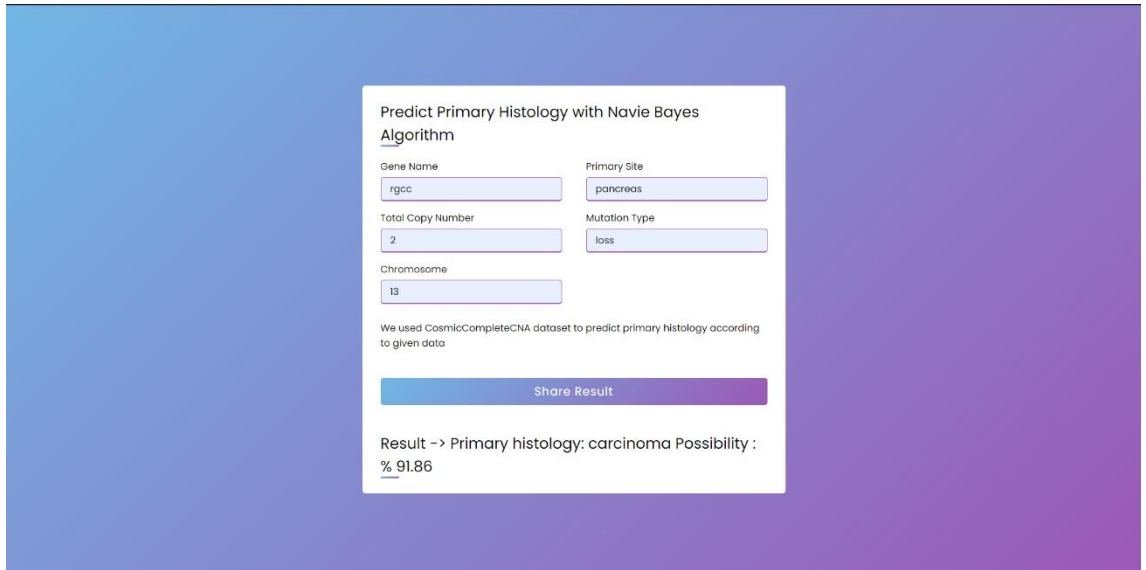
8.2. Copy Number Variant (CNA) Verisi ile Tahminleme

Girdide kullanılmak üzere rastgele seçilen kayıt şeklindeki gibidir. Gen ismi, birincil bölge, kopya sayısı, mutasyon tipi ve kromozom değerleri girilerek birincil hastalık (primary histology) bilgisi tahmini istenmektedir. Kullandığımız modelin öğrenme doğruluğunun (accuracy) %93 olduğunu analiz aşamasında belirtmiştik. Aşağıdaki değerlerin girilmesi sonucu modelin Primary histology’yi “carcinoma” olarak bulması gerekmektedir.



Index	gene_name	Primary site	Primary histology	TOTAL_CN	MUT_TYPE	Chromosome
123456	rgcc	pancreas	carcinoma	2	loss	13

Şekil 50. CNA için örnek test verisi



Predict Primary Histology with Navie Bayes Algorithm

Gene Name: rgcc Primary Site: pancreas

Total Copy Number: 2 Mutation Type: loss

Chromosome: 13

We used CosmicCompleteCNA dataset to predict primary histology according to given data

Share Result

Result -> Primary histology: carcinoma Possibility : % 91.86

Şekil 51. Primary histology bilgisi tahminleme örneği

Modelin Primary histology bilgisini doğru olarak tahmin etmiştir. Possibility değeri, tahmin edilen sınıfın ne kadar olasılıkla kazandığını gösterir. Buradaki örneğe göre, tahmin edilen carcinoma sınıfının olasılığı yaklaşık %92’dir.

9. SONUÇ

Mutation Export Census ve Copy Number Variants eğitim setleri için kurduğumuz modeller başarıyla tahmin yapmaktadır. Tahminleme sistemini kullanmak isteyen herkes, geliştirdiğimiz web uygulamasını kullanabilmektedir. Web uygulaması şu an sadece lokalde çalışmaktadır, ancak istenmesi durumunda internet sitesi haline dönüştürülebilir.

10.KAYNAKÇA

1. <https://cancer.sanger.ac.uk/cosmic>
2. <https://github.com/aleynaer/FinalProject-Experimental>
3. <https://github.com/BilalGunden-Insider/FinalProject>
4. <http://www.prowmes.com/blog/makine-ogrenmesi/>
5. <https://www.veribilimiokulu.com/makineler-nasil-ogrenir/>
6. <https://medium.com/türkiye/makine-öğrenmesi-nedir-20dee450b56e>
7. <https://evrimagaci.org/yapay-zeka-makine-ogrenmesi-ve-derin-ogrenme-kavramlari-arasindaki-fark-nedir-8889>
8. <http://muratsakal.com/?p=230>
9. <https://sengul-krdrl.medium.com/makine-öğrenmesinde-siniflandirma-algoritmasi-türleri-5e0f32245889>
10. <https://bilimfili.com/dunyayi-degistirmekte-olan-yapay-sinir-aglari-nedir>
11. <https://www.ibm.com/tr-tr/analytics/machine-learning>
12. <https://medium.com/@ekrem.hatipoglu/machine-learning-clustering-kümeleme-k-means-algorithm-part-13-be33aeef4fc8>
13. <https://ichi.pro/tr/boyut-azaltma-temel-bilesen-analizi-230073535364785>
14. <https://www.datascienceearth.com/boyutsal-kucultme-dimensionality-reduction/>
15. <https://medium.com/deep-learning-turkiye/pekiştirmeli-öğrenmeye-giriş-serisi-1-8f5c35b6044>
16. <https://corporatefinanceinstitute.com/resources/knowledge/other/random-forest/>
<https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/>
17. <https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a>
18. <https://helloacm.com/a-short-introduction-logistic-regression-algorithm/>
19. <https://medium.datadriveninvestor.com/k-means-clustering-4a700d4a4720>
20. <https://iq.opengenus.org/advantages-and-disadvantages-of-logistic-regression/>
21. <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/>
22. <http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-random.html>
23. <https://www.mygreatlearning.com/blog/random-forest-algorithm/>
24. <https://holypython.com/rf/random-forest-pros-cons/>
25. <https://www.upgrad.com/blog/naive-bayes-classifier/>
26. <http://theprofessionalspoint.blogspot.com/2019/03/advantages-and-disadvantages-of-naive.html>

11.TEŞEKKÜR

Bitirme Projesi Dersi kapsamında gerçekleştirdiğim bu projede Genomik Veri Analizi konusu üzerinde yetkinliğimi ve bilgimi arttırmış bulunuyorum. Pandemi şartları dolayısıyla yüz yüze toplantılar gerçekleştirmesek de dijital çağın sunduğu fırsatlardan sonuna kadar yararlanarak iş birliği ve uyum içinde projeyi gerçekleştirmiş bulunuyoruz. Bu süreç boyunca bilgi birikimi ve deneyimleriyle yardımcı olan proje danışmanım Sayın Doç. Dr. Gıyasettin Özcan'a ve projeyi gerçekleştiren beraber emek harcadığım takım arkadaşım Bilal Günden'e teşekkürlerimi sunarım.

12.ÖZGEÇMİŞ

Aleyna ER

Mail: 031790058@ogr.uluda.edu.tr

LinkedIn : linkedin.com/in/AleynaER

GitHub: github.com/aleynaer

Eğitim

- Bursa Uludağ Üniversitesi, Nilüfer / Bursa

Eylül 2017 - Haziran 2021, 4.Sınıf öğrencisi (devam etmekte)

Mühendislik Fakültesi / Bilgisayar Mühendisliği Bölümü

Deneyim

- Ağustos 2020 - Ekim 2020 Stajyer, FineSci Teknoloji A.Ş. , İstanbul

Türkçe Metinlerin NLP Yöntemleri ile Özetleme Çalışmaları

Geçmişten günümüze Türkçe dilinde çalışılmış metin özetleme sistemleri üzerine akademik araştırma ve saha araştırmaları yaptım. Araştırılan sistemlerin çalışma mekanizmaları diyagramlarla raporladım. Sistemlerin avantajları / dezavantajları üzerine inceleme yazdım. Seçtiğim Semantic-based (LSA) ve Graph-based (TextRank) yöntemleriyle çalışan iki sistemi aynı metinler vererek eşit şartlarda çalıştırdım, özet çıkarma performanslarını karşılaştırıp rapor yazdım.

- Temmuz 2020 - Ağustos 2020 Stajyer, Xtinge Teknoloji A.Ş. , İstanbul

3B Model Formatları, Karşılaştırması ve Dönüşümü

Unity projesinde farklı dosya türleri kullanmanın yararları ve sakıncaları üzerine araştırma yaptım. Aynı uzantılı dosyaların kullanılmasının daha avantajlı görülmesi sonucunda Unity platformunda kullanılan 3 boyutlu modellerin dosya formatları ile özellikleri inceleyip kıyasladım. Çalışmalarım ışığında hedef dosya formatı seçildi. Piyasada var olan dosya format dönüştürme uygulamaları ile beraber, yazılmış dönüştürücü scriptleri araştırdım. AutoDesk FBX Converter uygulamasının kullanımını otomatikleştirecek bir script üzerine çalıştım.