**İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ**

# ALT SEKANS PROFİL HARİTALARI KULLANILARAK PROTEİN KATLANMASI TANIMA

**YÜKSEK LİSANS TEZİ**

**Ruşen HALEPMOLLASI**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Bilgisayar Mühendisliği Programı**

**Tez Danışmanı: Yrd. Doç. Dr. Ömer Sinan SARAÇ**

**Mayıs 2016**

**İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ**

# ALT SEKANS PROFİL HARİTALARI KULLANILARAK PROTEİN KATLANMASI TANIMA

**YÜKSEK LİSANS TEZİ**

**Ruşen HALEPMOLLASI**
**(504111538)**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Bilgisayar Mühendisliği Programı**

**Tez Danışmanı: Yrd. Doç. Dr. Ömer Sinan SARAÇ**

**Mayıs 2016**

İTÜ, Fen Bilimleri Enstitüsü'nün 504111538 numaralı Yüksek Lisans Öğrencisi Ruşen HALEPMOLLASI, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı " ALT SEKANS PROFİL HARİTALARI KULLANILARAK PROTEİN KATLANMASI TANIMA" başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

| | | |
|---|---|---|
| **Tez Danışmanı :** | **Yrd. Doç. Dr. Ömer Sinan SARAÇ**<br>İstanbul Teknik Üniversitesi | ..... |
| **Jüri Üyeleri :** | **Prof. Dr. Zehra ÇATALTEPE**<br>İstanbul Teknik Üniversitesi | ..... |
| | **Yrd. Doç. Dr. Arzucan ÖZGÜR**<br>Boğaziçi Üniversitesi | ..... |
| | | ..... |

**Teslim Tarihi :** **05 Nisan 2016**
**Savunma Tarihi :** **02 Mayıs 2016**

***Babama,***

## ÖNSÖZ

Böyle bir çalışma yapılması önerisinde bulunan, çalışmamın her aşamasında bilgi ve deneyimleriyle beni yönlendiren ve bu tezi hazırlamamda en büyük katkı sahibi olan değerli hocam ve tez danışmanım Yrd. Doç. Dr. Ömer Sinan SARAÇ'a, kıymetli vakitlerini ayırarak manevi desteğini, bilgisini ve yardımlarını hiç bir zaman esirgemeyen değerli hocam Prof. Dr. Zehra ÇATALTEPE'ye, değerli bilgi ve düşüncelerini benimle paylaşan, çalışmamın her aşamasında destek olan ve fedakarca zaman ayıran değerli arkadaşım İsmail BİLGEN'e katkılarından dolayı teşekkürü bir borç bilirim.

Çalışmam boyunca her türlü desteğini ve sabrını koşulsuz ve her daim hissettiğim sevgili eşime teşekkür ederim.

05 Mayıs 2016

Ruşen HALEPMOLLASI
(Bilgisayar Mühendisi)

## İÇİNDEKİLER

**Sayfa**


**ÖNSÖZ** ..... **vii**
**İÇİNDEKİLER** ..... **ix**
**KISALTMALAR** ..... **xi**
**SEMBOLLER** ..... **xiii**
**ÇİZELGE LİSTESİ** ..... **xv**
**ŞEKİL LİSTESİ** ..... **xvii**
**ÖZET** ..... **xix**
**SUMMARY** ..... **xxi**
**1. GİRİŞ** ..... **1**
1.1 Protein Sentezi ..... 2
1.2 Protein Katlanması ..... 4
1.3 Tezin Yapısı ..... 5
**2. İLGİLİ ÇALIŞMALAR** ..... **7**
2.1 Homoloji Tabanlı Yaklaşım ..... 7
2.2 Taksonomi Tabanlı Yaklaşım ..... 8
**3. YÖNTEMLER** ..... **13**
3.1 Veri Kümesi ..... 13
3.1.1 Ding ve Dubchak ..... 15
3.1.2 EDD - Extended Ding ve Dubchak ..... 16
3.1.3 F95 - Fold95 ..... 16
3.1.4 F194 - Fold194 ..... 16
3.2 Öznitelik Çıkarımı ..... 16
3.2.1 Amino asitlerin fizyokimyasal özellikleri ..... 17
3.2.2 SPMap ..... 21
3.2.2.1 Alt sekans profil haritası oluşturma ..... 22
3.2.2.2 Öznitelik vektörü oluşturma ..... 24
3.3 Destek Vektör Makineleri ..... 24
3.3.1 Doğrusal destek vektör makineleri ..... 25
3.3.2 Doğrusal olmayan destek vektör makineleri ..... 28
3.4 Karar Ağacı ..... 29
3.5 Rastgele Orman ..... 30
**4. DENEYLER** ..... **33**
4.1 Sınıflandırma Modeli Değerlendirme Ölçütleri ..... 35
4.2 Deneyler ve Sonuçları ..... 37
**5. SONUÇLAR** ..... **45**
**KAYNAKLAR** ..... **47**

**EKLER ..... 51**
EK A.1 ..... 53
**ÖZGEÇMİŞ ..... 57**

## KISALTMALAR

| | |
|---|---|
| **DNA** | **:** Deoksiribo Nükleik Asit |
| **RNA** | **:** Ribo Nükleik Asit |
| **mRNA** | **:** Mesajcı RNA |
| **tRNA** | **:** Taşıyıcı RNA |
| **SPMap** | **:** Subsequence Profile Map<br>Alt Sekans Profil Haritası |
| **PDB** | **:** Protein Data Bank<br>Protein Veri Bankası |
| **SCOP** | **:** Structural Classification of Proteins |
| **CATH** | **:** Class, Architecture, Topology and Homologous superfamily |
| **CASP** | **:** Critical Assessment of Structure Prediction |
| **DVM** | **:** Destek Vektör Makineleri |
| **RO** | **:** Rastgele Orman |

## SEMBOLLER

| | |
|---|---|
| $\delta$ | **:** Eşik değeri |
| $\kappa$ | **:** Sahte sayı |
| $\sum$ | **:** Toplam sembolü |
| $\phi$ | **:** Ortalama kare kontenjansı katsayısı |
| $w^T$ | **:** Hiper düzleme dik olan vektör |
| $x$ | **:** Hiper düzlemdeki herhangi bir nokta |
| $\xi$ | **:** Yanlış sınıflandırılan örneğin sınır düzleme uzaklığını gösteren yapay değişken |

## ÇİZELGE LİSTESİ

**Sayfa**

**Çizelge 4.1:** Karışıklık Matrisi ..... 35
**Çizelge 4.2:** EDD veri kümesi kullanılarak SPMap yöntemiyle çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%) ..... 39
**Çizelge 4.3:** EDD veri kümesi kullanılarak SPMap yöntemiyle ve amino asitlerin fizyokimyasal özeliklerine dayanan tanımlayıcılarla çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%) ..... 41
**Çizelge 4.4:** F95 veri kümesi kullanılrak SPMap yöntemiyle çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%) ..... 42
**Çizelge 4.5:** EDD ve F95 veri kümeleri ile elde edilen ortalama doğruluk başarı oranları (%) ..... 43
**Çizelge A.1** EDD veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri ..... 53
**Çizelge A.2** F95 veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri ..... 54

## ŞEKİL LİSTESİ

**Sayfa**


**Şekil 1.1** : Protein Sentezi [1]. ..... 3
**Şekil 1.2** : Protein Yapısının Seviyeleri [2]. ..... 4
**Şekil 3.1** : SCOP Versiyon 1.75 Hiyerarşisi [3]. ..... 15
**Şekil 3.2** : Amino asitlerin özelliklerine göre öznitelik çıkarma. ..... 17
**Şekil 3.3** : Amino asitlerin fizyokimyasal özellikleri [4]. ..... 18
**Şekil 3.4** : Örnek Sekans. ..... 18
**Şekil 3.5** : SPMap Yapısı [5]. ..... 21
**Şekil 3.6** : Destek Vektör Makineleri. ..... 25
**Şekil 3.7** : Doğrusal Destek Vektör Makineleri. ..... 26
**Şekil 3.8** : Doğrusal Olmayan Destek Vektör Makineleri ..... 28
**Şekil 3.9** : Rastgele Orman kararı [6]. ..... 31
**Şekil 4.1** : Tez çalışmasının genel yapısı. ..... 33
**Şekil 4.2** : Veri kümesinin eğitim kümesi ve test kümesi olarak ayrılması. ..... 37

# ALT SEKANS PROFİL HARİTALARI KULLANILARAK PROTEİN KATLANMASI TANIMA

## ÖZET

Yaşamın en temel makro molekülleri olan proteinlerin 3b yapısına ait bilgi bioinformatik çalışmalarında kilit bir rol oynar. Hücrenin karmaşık yapısı içersinde nanometre mesafede ve mikro mili saniyede katlanan proteinlerin katlanma örüntülerini tahmin etmek ne yazık ki oldukça zordur. İki protein aynı düzen ve topoloji ile aynı ikincil yapıya sahipse ortak bir katlanma örüntüsüne sahiptir denilebilir.

Katlandıktan sonra görevlerini yerine getirmek için hazır hale gelen proteinlerin üç boyutlu yapısı fonksiyonlarına uygun olmalıdır. Belli bir sekans bilgisine ve amino asit özelliklerine dayanarak protein katlanmasını tanıma, proteinlerin 3b yapılarının ve fonksiyonlarının belirlenmesinde önemli bir aşama olarak düşünülebilir. Proteinler arasında yakın evrimsel ilişki olduğunda benzerliği tespit etmek için sekans-sekans eşleştirmesi iyi sonuçlar verir. Ancak iki protein yapısal olarak çok benzer olsa da aralarında sekans benzerliği yoksa bu tür bir eşleştirme etkili değildir. Böyle durumlarda sekaslardan çıkarılan özniteliklere makine öğrenme yöntemleri uygulayarak proteinlerin katlanmasını tahmin etmek daha etkili olur. Bunun için proteinlerin doğada sınırlı sayıda olduğunu varsayıp belli bir sayıda katlanma sınıfı üzerinde çalışılmalıdır.

Bu çalışmada, sınırlı sayıda katlanma sınıfı içeren ve literatürde sıkça rastlanan dört adet veri kümesi kullanıldı. Protein katlanmasının tanınmasında amino asitlerin fizyokimyasal özelliklerinden faydalanıldı. Ayrıca ilk kez alt sekans profil haritası (SPMap) kullanılarak makine öğrenme yöntemlerinin uygulanabileceği öznitelikler çıkarıldı. Her katlanma sınıfı için ayrı ayrı elde edilen öznitelikler, iki katmanlı bir yaklaşım ile makine öğrenmesi yöntemlerinden faydalanılarak protein katlanmalarının tahmin edilmesinde kullanıldı. Amino asit özniteliklerine ve alt sekans bilgilerine dayanarak elde edilen öznitelik kümeleri pozitif ve negatif olarak etiketlendi ve ikili sınıflandırma modelleri eğitildi. Oluşturulan modeller ile test veri kümesi sınıflandırıldı. Elde edilen sınıflandırma tahminleri öznitelik vektörleri gibi düşünülerek birleştirildi ve ikinci katmanda kullanılmak üzere yeni bir öznitelik kümesi oluşturuldu. İkinci katmanda çoklu sınıflandırma modeli eğitildi. Bu modeller uygulamanın başında ayrılan test kümesi ile test edildi ve geliştirilen modelin performansı doğruluk oranı, kesinlik, duyarlılık ve F-ölçütü ile değerlendirildi.

Önerilen sistem ile performans ölçümünde kullanılan DD veri kümesi üzerinde ortalama %71.7, EDD veri kümesi üzerinde ortalama%75.7, F95 veri kümesi üzerinde ortalama%75.15 doğruluk başarı oranı elde edildi.

## PROTEIN FOLD RECOGNITION


### SUMMARY

Proteins that are very important macromolecules of life are responsible for some of the most essential functions in an organism such as metabolism, transport, immune system, etc. The analysis of proteins tertiary structure is a difficult task because of the complex structure of the cell. Protein fold recognition helps to understand the tertiary structure of the protein.

Before the folding, there is a single direction of flow from the DNA linear polymer consists of four different bases (Adenosine, Guanine, Cytosine, Thymine) to a protein consists of different 20 amino acids. This process comprises three stages that are DNA replication, transcription and translation. In the DNA replication stage, two identical replicas are created from original molecule of DNA. In the second stage called transcription, a particular segment of DNA is copied into single stranded RNA (mRNA) by the RNA polymerase enzyme. Next, messenger RNA (mRNA) is translated into a specific amino acid chain in a process called translation.

In general, there are three levels of protein structure but in some cases, it can be fourth level. The primary structure is the protein sequence consists of the amino acid chain. The secondary structure is the first stage of protein folding, in which the chain is regulated in regular structures as called "$\alpha$-helix" and "$\beta$-sheet". The tertiary structure formed by the further folding composed of complex and fixed geometric shapes. Three-dimensional proteins folded to form tertiary structure create quaternary structure by coming together.

Information belonging to 3D structures of the proteins, which are the most fundamental macromolecules of life, plays a key role in bioinformatics studies. The 3D structure of the proteins which are ready to fulfill their liabilities after the folding have to fit to the functions (the miss folds would cause the Alzheimer, some types of the cancer and Parkinson's). In other words, the information about the structure of the proteins plays an important role in the determining of the different type diseases and improving the effectiveness of the new medicines.

Protein fold recognition from amino acid sequences plays a critical role in prediction of protein structures and functions. Therefore information of protein 3-dimensional structures is significant for understanding cellular function and the development of drug design and the biomedicine. Unfortunately, it is very difficult to predict the pattern of the folds for the complex structure of the cells during micro milliseconds and in the nanometric distance. In a case, when two proteins with the same order and topology have the same secondary structure, it can be said that they have the same fold pattern.

Protein fold recognition based on the particular sequence information and amino acid properties is the significant step for the determining 3D structure and the functions

of the proteins. Applying the sequence-sequence pairing for the determining the similarity of the proteins when the proteins have the close evolutionary relationship gives good results. But when the proteins do not have the sequence similarity even if such type of the pairing will not be effective. In this cases, fold prediction using the machine learning methods on the extracted features of the sequences will be more effective. Assuming that the number of the proteins in the nature is limited will let us to work on the fold classes which have the certain number.

The purpose of this work it to extract the specific features from the subsequences and psycho-chemical properties of the amino acids of the proteins, and to predict correctly the fold classes of these proteins using the machine learning methods. Protein fold recognition is very difficult subject theoretically and practically because of the complex structure of the proteins. Crystallization of the proteins using the practical methods and analyzing these fold structure is very hard and expensive process. That is why there is need for the theoretical study using the computational techniques. Once the features are extracted from the protein sequences, any machine learning method can be employed.

The recognition process of a query protein sequence in this study can be divided two steps. In the first step, features were extracted from the query sequence. It has benefited from the physicochemical properties of the amino acids for the protein fold recognition. Besides, the attributes on which machine learning methods can be applied are extracted by using subsequence profile map (SPMap) in protein fold recognition for the first time. In the second step, The features exracted from each fold class were used in a two-layer approach to train classifiers to predict correct protein fold belonged to the query sequence. In the first layer, the features, exracted from SPMap and physicochemical properties of amino acids, labeled as positive and negative. Then the feature sets trained binary classifiers and the test set was classified by using these models. The binary classification estimates mind as feature vectors and was combined together. Thus, a new feature set has been created to be used in the second layer and multiple classification models were trained. The developed model was tested with test set separated at the beginning of the application.

We used the binary classifier method on R programming in the fist layer, where as used the multiclass classifier methods on Weka and R programming in the second layer. Random Forest was used for the binary classification and Support Vetor Maachine, Random Forest, Multi-class classifiers and ensemble classifier were tried for multiple classification.

In this work, we have used four datasets with the limited number of the fold classes. The first dataset, called DD set, has been comprehensively employed in several studies for protein fold recognition. We benefit from it as a benchmark dataset. The dataset which has the most popular 27 fold classes in SCOP database is composed of a training set and testing set. The training set includes 313 protein domain sequences, whereas The testing set includes 385 protein domain sequences. The other three datasets created according to the latest version of SCOP, are called EDD, F95 and F194. EDD - Extended DD set comprises 3397 protein domain sequences with the same 27 fold classes of DD set. F95 and F194 sets created to cover more folds, have less than 40% pairwise sequence identity. F95 - Fold 95 set comprises 6364 protein domain sequences from 95 folds. F194 - Fold 194 set comprises 8026 protein domain sequences from 194 folds.

In this study, we used three popular metrics, which are Precision, Recall and F-measure for evaluation of the results.For the evaluate the overall performance of this study, we used the overall accuracy. The classifier performance was evaluated with datasets using our proposed system, and 71.7% for DD dataset, 75.7% for EDD dataset and 75.15% for F95 dataset average accuracy rates were achieved.

In our future work, we will benefit from SCOP database hierarchy for the further improvement of prediction accuracy. Besides, we will try feature selection methods after the extract the features.

# 1. GİRİŞ

Kelime anlamı Yunancada birincil anlamına gelen “proteios” kelimesinden türetilen proteinler gerçektende yaşamın en önemli makro molekülleridir. Hücre içindeki her süreçte yer alan proteinler tüm hücrelerde bulunur ve sudan sonra hücresel ağırlığın ikinci büyük kısmını oluştururlar.Sadece insan vücudunda sayısının yirmibinden fazla olduğu tamin edilen proteinler metabolizma, hücre döngüsü, bağışıklık sistemi, hücresel sinyal iletimi gibi organizmadaki fizyolojik süreçlerin çoğunda önemli rol oynarlar [7].

Her biri ayrı özelliğe sahip ve sayısı birkaç yüz ile yüzbinlerce arasında değişen yirmi çeşit amino asidin birbiri ardına bağlandıktan sonra katlanarak oluşturmuş oldukları proteinin üç boyutlu yapısı çok farklı şekillerde karşımıza çıkabilir. Proteinler, ancak katlandıktan sonra görevini yerine getirmek için hazır hale gelirler. Bu yüzden proteinlerin üç boyutlu yapısı fonksiyonlarına uygun olmalıdır, yani görevlerine uygun şekilde katlanmalıdırlar. Yanlış katlanmalar ise Alzhaimer, bazı kanser türleri ve Parkinson gibi hastalıklara sebep olabilir [8]. Proteinlerin yapısına ait bilgiler, çeşitli hastalıkların belirlenmesi ve hastalıklarla mücadele edebilmek için yeni ilaçların geliştirilmesinde önemli rol oynar.

Protein katlanmasını tanıma, makine öğrenmesi yöntemleri uygulanarak aralarında homolojik benzerlik olmayan ancak aynı katlanma sınıfında yer alan proteinleri modellemek için kullanılan bir yöntemdir. Bu yöntem belli bir sekans bilgisine dayanarak proteinin yapısını tanımlamada oldukça etkindir. Eğer proteinler aynı düzen ve topoloji ile büyük oranda aynı ikincil yapıya sahipse ortak bir katlanma örüntüsüne sahip oldukları düşünülür [4].

Katlanma bilgisi proteinlerin 3b yapısını ve fonksiyonunu belirlemeyi kolaylaştırdığı için amino asit sekanslarından protein katlanmasını tanıma bioinformatikteki önemli ancak zor konulardan bir tanesidir [9]. Bir proteinin yapısını ve fonksiyonunu tahmin etmede sekans sekans eşleştirmesi kritik rol oynar. Ama sekans çiftlerinin hizalanması, proteinler arasındaki yakın evrimsel ilişkiyi bulmada doğru çalışsa da iki protein

yapısal olarak benzerken aralarında önemli ölçüde sekans benzerliği olmadığında etkili değildir [10]. Bunun aksine protein katlanmasını tanıma, sekans benzerliği olmaksızın proteinin yapısını tanımlamada etkili bir yaklaşımdır ve yapılan çalışmalarda umut veren sonuçlar elde edilmektedir. Bu bilgilerden yola çıkarak bu çalışmada amino asitlerden ve alt sekanslardan elde edilen bilgilerin proteinlerin katlanmasındaki etkisi incelenecektir.

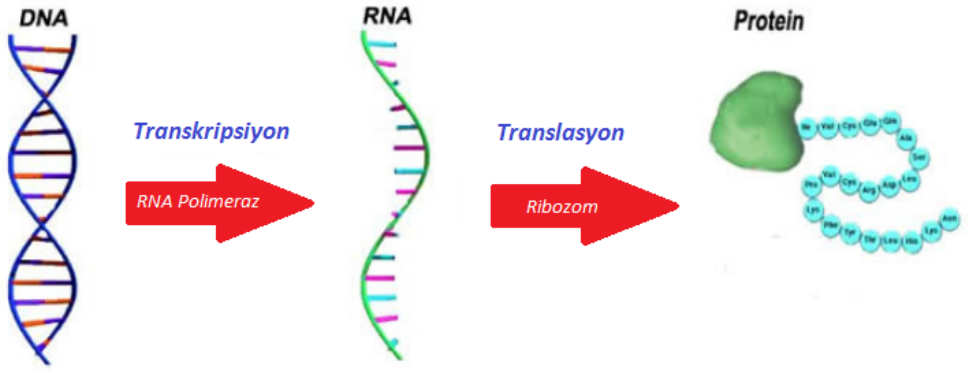
Bu çalışmanın amacı bir proteine ait alt sekanslardan ve amino asitlerin fizyokimyasal özelliklerinden belli öznitelikler çıkararak makine öğrenmesi yöntemleri ile bu proteinin katlanma sınıfını doğru tahmin etmektir. Protein katlanması hem deneysel olarak hem de teorik olarak oldukça zor konulardan biridir. Çünkü proteinler hücrenin karmaşık yapısı içersinde, nanometre mesafede ve mikromili zaman aralığında sentezlenerek katlanmaktadır. Dolayısıyla deneysel yöntemlerle proteinleri kristalleştirip katlanmış haldeki yapılarını incelemek oldukça maliyetli ve meşakatli bir süreçtir. Bu yüzden hesaplamalı yöntemlerle teorik çalışmalara ihtiyaç duyulur. Makine öğrenme yöntemleri kullanılarak proteinlerin katlanmasını tahmin edebilmek için proteinlerin doğada sınırlı sayıda olduğunu varsayıp belli bir sayıda katlanma sınıfı üzerinde çalışılmalıdır. Hazırlanan tez çalışmasında sınırlı sayıda katlanma sınıfı içeren ve literatürde sıkça kullanılan dört adet veri kümesinden faydalanılmıştır. Makine öğrenme yöntemleri, giriş parametresi olarak öznitelik vektörü alırlar. Bu sebeple bir protein, amino asitlerin fizyokimyasal özelliklerini ve alt sekans bilgilerini temsil eden sayısal değerlere dönüştürülerek öznitelikler elde edilmiştir. Bu işlemlerin ardından elde edilen özniteliklerle iki katmanlı bir sınıflandırma yöntemi ile protein katlanma sınıflarını tahmin edecek sınıflandırıcılar eğitilmiştir. Sınıflandırıcı eğitiminden sonra sınıflandırıcının performansı test edilmiş ve doğruluk oranı, kesinlik, duyarlılık ve F-ölçütü ile değerlendirilmiştir.

Bu bölümde proteinlerin katlanmadan önce nasıl sentezlendiği ile ilgili bilgi verilecek ve protein katlanması ve seviyeleri daha detaylı açıklanacaktır. Ardından tezin yapısı ile ilgili genel bilgi verilecektir.

## 1.1 Protein Sentezi

Protein katlanarak üç boyutlu yapısına kavuşmadan önce DNA (DeoksiriboNükleik Asit) adı verilen ve canlıya ait tüm genetik bilgileri kodlanmış olarak taşıyan yapıdan

proteine tek yönlü bir akış gözlemlenir. DNA'da bulunan genetik bilgi kopyalanarak mRNA üretildikten sonra mRNA'daki koda uygun olarak amino asit zinciri şeklinde protein sentezlenir. Yani DNA'yı oluşturan Adenin (A), Timin (T), Guanin (G) ve Sitozin (C) bazlarından proteini oluşturan 20 amino aside dönüşüm vardır. Şekil 1.1'de görülen üç aşamalı bu süreç aşağıda daha detaylı bir şekilde anlatılacaktır.




**Şekil 1.1** : Protein Sentezi [1].

Protein sentezinin ilk aşaması olan DNA replikasyonu bir çift sarmal DNA molekülünden birbirinin kopyası iki DNA'nın oluşması sürecidir. Orijinal DNA'nın her biri ipliği, DNA'yı okuyarak her nükleotidin karşısına uygun nükleotidi taşıyıp hidrojen bağları ile bağlayan ve polimeraz olarak adlandırılan bir enzim tarafından sentezlenen tamamlayıcı ipliğin üretiminde kalıp görevi görür. Replikasyon sürecindeki hata oranı 109 bazda 1 tanedir.

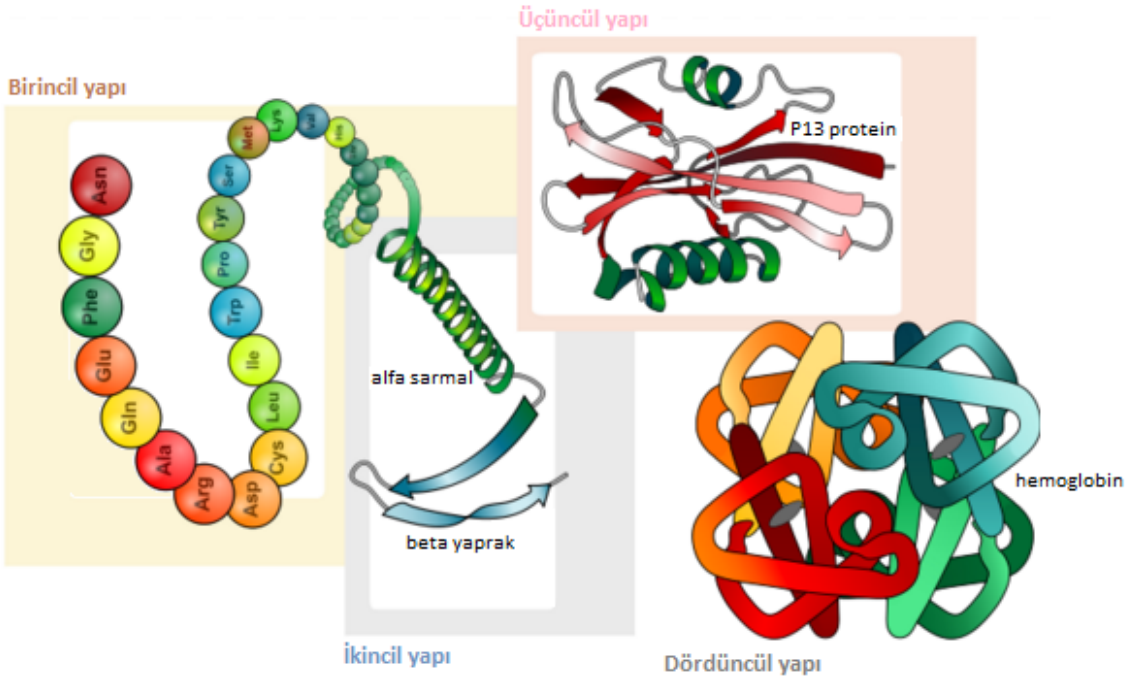
İkinci aşama olan transkripsiyon, DNA'nın gene sinyal geldikten sonra belli bir kısmındaki taban dizisinin RNA (RiboNükleik Asit) polimeraz tarafından tek ipli bir RNA dizi olarak kopyalanması sürecidir. Transkripsiyon aşamasında DNA'daki polimer 3'-5' yönünde okunur ve yeni tümleyici polimer 5'-3' doğrultusunda oluşturulur.

Son aşama ise transkripsiyon surecinde oluşturulan mRNA'nin ribozomlar tarafından proteinlere dönüştürülmesi sürecidir ve stoplazmada gerçekleşir. Ribozom, üzerindeki bağlanma bölgelerinde mRNA'daki kodonları tRNA'daki tamamlayıcıları olan antikodonlara bağlar. mRNA'daki kodonlara karşılık gelen antikodonlari bulunduran tRNA'ların art arda eklenmesi sırasında tRNA'nın 3' ucuna bağlanmış olan aminoasitler birbirine bağlanarak polipeptit zinciri oluştururarak protein sentezlenir ve

ardından protein, saniyenin milyonda birinden çok daha kısa bir sürede katlanarak üç boyutlu yapısına kavuşur.

## 1.2 Protein Katlanması

Protein, birbirine zincir halinde bağlanmış olan aminoasit dizisinin oluşturduğu organik biopolimerdir. Şekil 1.2'de görüldüğü gibi proteinin yapısı bileşenlerin etkileşiminden dolayı üç boyutlu uzayda şekil alır. Genellikle protein yapısı üç seviye ile belirtilebilir, ancak bazı durumlarda dördüncü seviyeden de söz edilebilir [11].




**Şekil 1.2** : Protein Yapısının Seviyeleri [2].

***Primer (birincil) yapı :*** Protein sentezi sonucunda amino asitlerden birinin karboksil grubuyla diğerinin amino grubunun peptit bağlarıyla birbirine bağlanması sonucu oluşan zincir şeklindeki yapı katlanmanın ilk aşamasını oluşturur. Proteinlerin yapıtaşları olan ve doğada 20 çeşit bulunan amino asitlerin hangi sırada dizileceği ise DNA'dan gelen genetik bilgiye dayanır.

***Sekonder (ikincil) yapı :*** Uzamsal olarak birbirine yakın aminoasitlerin hidrojen bağları ile bükülmesi sonucu üç boyutlu uzayda protein sekansının şekli etkilenir ve katlanma süreci başlar. Bu süreçte "$\alpha$-helices" adı verilen sarmal ve "$\beta$-sheets" adı

verilen yaprak alt yapıları oluşur. Düzenli bir şekilde tekrarlayan bu alt yapıların birleşimi proteinin ikincil yapısını oluşturur.

***Tersiyer (üçüncül) yapı :*** Katlanmanın bir sonraki aşamasında ikincil yapıyı oluşturan "$\alpha$-helices ve $\beta$-sheets" alt yapıları arasındaki itici ve çekici kuvvetler katlanmalara sebep olur ve karmaşık olmasına rağmen sabit bir protein yapısı olan üçüncül yapıyı oluşturur. Protein üçüncül yapısına kavuştukan sonra görevini yerine getirmeye hazırdır.

***Kuarterner (dördüncül) yapı :*** Spectrin (proteini) gibi bazı proteinler tek bir amino asit sekansından değil de birden fazla amino asit sekansının birleşmesinden oluşur. Bu sekansların her biri kendi üçüncül yapısını oluşturmak için katlanırken bir araya gelerek dördüncül yapıdaki proteinleri oluştururlar. Üçüncül yapısına kavuşmuş iki veya daha fazla proteinin bir araya gelerek oluşturdukları dördüncül yapıdaki proteinler görevlerini artık bu şekilde yerine getirirler.

## 1.3 Tezin Yapısı

Hazırlanan tez çalışması 5 bölümden oluşmaktadır. Giriş bölümünde proteinler ve protein katlanmasını tahmin etmenin önemi anlatılmış ve bu doğrultuda tez çalışmasının amacı açıklanmıştır. Ardından proteinlerin katlanmadan önce nasıl sentezlendiğinden kısaca bahsedildikten sonra proteinlerin yapısı ve katlanması daha detaylı bir şekilde anlatılmıştır. Son olarak tezin yapısı hakkında bilgi verilmiştir.

2.Bölüm'de proteinlerin yapısının tahmin edilmesi ile ilgili yapılan çalışmalar iki kategoride incelenmiştir. Bu çalışmalardan proteinlerin doğada sınırlı sayıda olduğunu varsayıp makine öğrenme yöntemleri ile proteinlerin katlanma sınıfını tahmin eden taksonomi tabanlı yaklaşımlar özetlenmiştir.

3.Bölüm'de bu çalışmada kullanılan veri kümelerine ve yöntemlere yer verilmiştir. Bu bağlamda amino asitlerin fizyokimyasal özelliklerinden ve alt sekanslara dayalı bilgilerden öznitelik vektörlerinin oluşturulması anlatılmıştır. Ayrıca kullanılan sınıflandırma yöntemlerinden bahsedilmiştir.

4.Bölüm'de tez çalışmasına ait deneyler yapılırken uygulanan aşamalar ve deneysel sonuçlar anlatılmıştır. Ayrıca bu deneyler için kullanılan değerlendirme ölçütlerine de bu bölümde yer verilmiştir.

Son bölümde ise sonuçlardan bahsedilmiş ve gelecek çalışmalar için öneriler sunulmuştur.

## 2. İLGİLİ ÇALIŞMALAR

Hücrenin karmaşık yapısı içinde saniyenin milyonda biri hızla gerçekleştiği için çok zor tahmin edilen protein katlanmasına ait bilgi, proteinin üç boyutlu yapısının ve fonksiyonunun belirlenmesine ciddi ölçüde katkı sağlar. Bu yüzden amino asit sekanslarından bu bilgilere ulaşmak bioinformatik alanında oldukça önemli bir yere sahiptir. Dolayısıyla proteinin katlaması da bu alanda birçok araştırmacının ilgisini çekmiştir. Özellikle doksanlı yıllardan beri bu konu ile ilgili önemli çalışmalar ortaya konulmuştur.

Bu çalışmalar temel olarak taksonomi tabanlı ve homoloji tabanlı olmak üzere iki kategoride değerlendirilmiştir [9]. Ancak son yıllarda taksonomi tabanlı çalışmalar, umut veren performanslarından ötürü daha çok ilgi görmektedir. Bir sonraki başlık altında bu iki yaklaşım daha detaylı bir şekilde anlatılacaktır.

### 2.1 Homoloji Tabanlı Yaklaşım

Bioinformatikte sekans hizalama(sequence alignment) yöntemi ile proteinler arasındaki evrimsel ilişkilerin analizi yapılabilmekte ve proteinler arasındaki homolojiye dayanarak benzer proteinlerin yapısı ve fonksiyonu tahmin edilebilmektedir. Farklı türlerde proteinler arasında homoloji varsa bu proteinler yüksek seviyede dizilim benzerliğine sahiptirler ve aynı atadan geldikleri düşünülür. Buradan yola çıkarak homoloji tabanlı yaklaşım, aynı ataya sahip ve aralarında sekans benzerliği olan iki proteinin benzer üçüncül yapıya sahip olduklarını varsayar. Şablon tabanlı(template-based) olarak da bilinen homoloji tabanlı yaklaşımda deneysel yöntemlerle 3b yapısı çözülmüş benzer proteinlerin şablon olarak kullanılmasıyla yapısı bilinmeyen hedef proteinlerin modellenmesi amaçlanır. Bu yaklaşımı benimseyen çalışmaların sonuçlarına sekans hizalama ve şablon yapı nitelikleri yön vermektedir. Protein sekansları arasındaki benzerlik %25'in altında olduğunda bu proteinlerin 3b yapısının benzer olmadığı öngörülür. Ancak bazen aralarında sekans benzerliği olmayan proteinler yapısal olarak benzer olabilirler. Bu yüzden son yıllarda

protein katlanmasını tahmin ederek proteinlerin yapısı ve fonksiyonunu belirlemede taksonomi tabanlı çalışmalara olan ilgi hızla artmaktadır

## 2.2 Taksonomi Tabanlı Yaklaşım

Taksonomi tabanlı yaklaşımı benimseyen çalışmalarda protein katlanmasını tahmin ederken sınıflandırma yapılabilmesi için makine öğrenmesi yöntemlerinden faydalanılmaktadır. Sınırlı sayıda protein katlanma sınıfı düşünüldüğünde ve bir protein amino asitlerin fizyokimyasal özelliklerini ve sekans bilgilerini temsil eden sayısal değerlere dönüştürüldüğünde artık makine öğrenmesine ait parametreler elde edilebilir. Bu aşamadan sonra protein katlanmasını tanıma problemi bir sınıflandırma problemi olarak ele alınabilir ve taksometrik yaklaşımlar uygulanabilir.

Taksonomi tabanlı çalışmaların öncülüğünü Dubchak ve diğ.(1995) yapmıştır [12]. Protein katlanmalarının 83 tanesini inceledikleri için protein katlanmasını makine öğrenme yöntemleri uygulanabilen sınıflandırma problemi olarak ele almışlardır. Veri kümesinde toplam 254 tane protein domaini içeren 83 katlanma sınıfının, 38 tanesi iki veya daha fazla protein örneğine sahipken 45 tanesi sadece bir tane örneğe sahiptir. Global tanımlayıcılar ile amino asitlerin fizyokimyasal özelliklerinden öznitelik çıkarma ilk defa bu çalışmada önerilmiştir. Sonraki yıllarda bu özniteliklere yenileri de eklenerek geliştirilmiştir. Faydalanılan amino asit fizyokimyasal özelliklerinden ilki üç gruptan oluşan hidrofobidir. Amino asitlerin hidrofobi özelliğine ait gruplar; hidrofilik (Arg, Lys, Glu, Asp, Gln ve Asn), nötral (Gly, Ala, Ser, Tht, Pro, His ve Tyr) ve hidrofobik (Cys, Val, Leu, Ile, Met, Phe ve Trp) gruplarıdır. Kullanılan diğer bir özellik ise tahmin edilen ikincil yapıdır. Bu özelliğe dayalı olarak amino asitler sarmal (helix), yaprak (sheet) ve sarım (coil) olmak üzere üç gruba ayrılmıştır. Amino asitlerin tahmin edilen solvent erişilebilirliği kullanılan üçüncü özelliktir. Öznitelik çıkarırken bir proteindeki belli bir amino asit özelliğinin global birleşimini tanımlamak için kompozisyon (Composition-C), geçiş (Transition-T) ve dağılım (Distribution-D) tanımlayıcıları kullanılmıştır. C tanımlayıcısı ile amino asit fizyokimyasal özelliğine ait her bir grubun sekanstaki yüzdesi hesaplanmıştır. T tanımlayıcısı ile gruplar arasındaki geçiş frekanslarının yüzdesi karakterize edilmiştir. Bu sayede sekans boyunca grupların değişimin oranı tespit edilmiştir. Son olarak D tanımlayıcısıyla protein zinciri amino asit özelliğinin her bir bileşeni için yüzde olarak beş parçaya

(%0, %25, %50, %75, %100) ayrılmış ve bu parçalardaki amino asitlerin yüzdesi ayrı ayrı hesaplanmıştır. Böylece tüm sekans boyunca amino asit özelliğinin örüntü dağılımı bulunmuştur. Ardından C,T ve D tanımlayıcıları ile hesaplanan bu değerler birleştirilerek giriş parametreleri olarak kullanılmıştır.

Ding ve Dubchak (2001), Destek Vektör Makineleri (DVM, Support Vector Machine) ve Yapay Ağlar (YA, Neural Network) makine öğrenme yöntemlerini kullanarak sekans benzerliği olmaksızın yapısal benzerliği tespit etmeye odaklanmıştır [10]. PDB veri tabanına dayanan veri kümesinde 27 popüler katlanma kullanmıştır. Eğitim veri kümesinde en fazla %35 sekans benzerliğine sahip 313 örnek, test kümesinde ise %40 veya daha az sekans benzerliğine sahip 385 örnek vardır. Hazırlanan bu veri kümeleri sonraki yıllarda birçok araştırmacı tarafından kullanılmış ve halen de performans ölçümü için kullanılmaktadır. Protein sekansından öznitelik seçerken 20 çeşit amino asidin miktarını hesaplayarak 20b boyutunda ilk parametre setini oluşturmuştur. Buna ilaveten kompozisyon (Composition-C), geçiş (Transition-T) ve dağılım (Distribution-D) tanımlayıcıları ile amino asitlerin yapısal veya fizyokimyasal özelliklerinden öznitelik çıkarmıştır. Tahmin edilen ikincil yapı (predicted secondary structure-S), hidrofobi (hydrophobicity-H), normalize edilmiş Van der Waals şiddeti (normalized van der Waals volume-V), polarite (polarity-P) ve polarizabilite (polarizability-Z) faydalanılan özelliklerdir. Bu özelliklerin her biri kendi içinde üç gruptan oluşan amino asit bileşenlerine ayrılmış ve her özellik için 21b (C için 3b, T için 3b, D için 3x5=15b) boyutunda parametre seti oluşturulmuştur. Yani altı tane parametre seti birbirinden bağımsız hesaplanmıştır. Böylece protein katlanmasının tahmininde makine öğrenme teknikleri parametre setlerinin her birine ayrı ayrı veya farklı kombinasyonlarla uygulanabilmiştir. En iyi sonuçlar ise tümü kullanıldığında elde edilmiştir. Elde ettikleri 125d boyutundaki öznitelik vektörü ile çok sınıflı sınıflandırma yöntemleri kullanılmıştır. Eşsiz bir sınıfa karşı diğerleri (Unique one versus others) olarak adlandırdıkları yöntemde, bir sınıfa karşı diğerleri (one versus others) yönteminde karşılaşılan yanlış pozitif tahmin hatalarını azaltmak için bu yönteme yeni bir adım eklenmiştir. Bu adımda yanlış pozitif tahmin veren sonuçlara tekrar bir sınıfa karşı diğerleri yöntemi uygulanmış ve sonuçta en yüksek skora sahip değer doğru sınıf olarak belirlenmiştir. Bu şekilde başarı oranında %8 ile %29 arasında

artış sağlansa da en yüksek sonuçlar tüm sınıflar karşı tüm sınıflar (all versus all) yöntemi ile elde edilmiştir.

Nanni, 2006 yılında yaptığı çalışmasında HKNN (K-local hyperplane distance nearest neighbour) sınıflandırma metodunu kullandığı ve Specialized Ensemble adını verdiği bir yöntem tasarlamıştır [13]. Uzaklık hesaplamalarında ise Mahalanobis uzaklık metodundan faydalanmıştır. Bu yöntemde sınıf sayısı kadar sınıflandırıcı kullanmıştır. Her bir sınıflandırıcı bir favori sınıfa sahiptir. En iyi sonuçları elde etmek için sınıflandırıcıları birleştirirken toplama veya oylama kurallarını uygulamıştır. Toplam kuralında, final skoru olarak sınıflandırıcıların skorlarının toplamı kabul edilmiştir. Oylama kuralında ise tüm sınıflandırıcıların oyları sayılmış ve en yüksek oya sahip sınıf final tahmin olarak belirlenmiştir. Ding ve Dubchak'ın çalışması ile aynı veri kümesini ve öznitelikleri kullanmış ve sonuçlarını karşılaştırmıştır. Sonuçta ise %61.1 oranında bir başarı elde etmiştir.

Shen ve Chou (2006), tarafından gerçekleştirilen çalışmada sınıflandırıcı toplulukları (ensemble classifier) kullanılarak proteinlerin katlanma örüntüsü tahmin edilmektedir [14]. Ding ve Dubchak (2001) tarafından geliştirilen veri kümesini revize ederek kullanmıştır. Öznitelikleri çıkarırken amino asitlerin tahmin edilen ikincil yapı (S), hidrofobi (H), normalize edilmiş Van der Waals şiddeti (V), polarite (P) ve polarizabilite (Z) özelliklerini kulanmıştır. Sekans dizilimini gözardı etmemek için 20 çeşit amino asidin birleşim (composition) yüzdelerinin yerine $(20+2\lambda)$b boyutundaki sahte amino-asit (pseudo-amino acid) birleşimlerini hesaplamıştır. Burada ilk 20 değer amino asitlerin ağırlığına(composition) karşılık gelirken, sonraki $2(\lambda)$ değeri protein zincirindeki amfifilik (amphiphilic) sekans korelasyonunu göstermektedir. Böylece toplamda$((21*5)+(20+2\lambda))$b boyutunda öznitelik vektörü elde etmiştir. $(\lambda)$ 'ya 1, 4, 12 ve 15 sayılarını atayarak dört faklı giriş parametresi oluşturmuştur. Toplamda ise dokuz tane giriş parametresi döndürmüştür. Her parametre seti için bir sınıflandırıcı eğitmiş ve böylece 9 tane sonuç elde etmiştir. Bu sonuçlar için The Optimized Evidence Theoretic K-Nearest Neighbors (OET-KNN) sınıflandırıcı toplulukları geliştirmiştir. Sistem, protein katlanmasındaki örüntüleri %62 oranında doğru tahmin etmiştir.

Chen Lin ve diğ. (2013), protein katlanma örüntülerini tahmin etmenin etkisini artırmak için hiyerarşik bir yapıya dayanan yeni bir sınıflandırıcı toplulukları metodu

önermiştir [4]. Kompozisyon (C), frekans (F) ve dağılılm (D) tanımlayıcılarından faydalanarak amino asitlerin fizyokimyasal özelliklerinden öznitelik çıkarmış ve toplamda 188b boyutunda öznitelik vektörü elde etmiştir. Kullanılan amino asit fizyokimyasal özellikler; tahmin edilen ikincil yapı, hidrofobi, normalize edilmiş Van der Waals şiddeti, polarite, polarizabilite, yük (charge), yüzey gerilimi (surface tension) ve solvent erişebilirlik (solvent accessibility) özellikleridir. Sınıflandırma aşamasında 18 tane sınıflandırıcı kullanmıştır ( 1. Logistic Regression, 2. SMO, 3. SVM, 4. IB1, 5. IB5, 6. IB10, 7. OneR, 8. Conjunctive Rule, 9. Decision Table, 10. JRip, 11. ZeroR, 12. Simple Cart, 13. Naïve Bayes, 14. Random Tree, 15. FT Tree, 16. RF, 17. Decision Stump, ve 18. J48). Elde edilen sonuçlara K=9 değeri verilerek K-Merkezli öbekleme (K-Means clustering) metodu uygulanmış ve sınıflandırıcı sayısı 18'den 9'a düşürülmüştür. Bu dokuz sınıflandırıcı için geliştirdiği Circulation Combination of EFSS (Ensemble Forward Sequential Selection) algoritmasını uygulamıştır. Bu algoritma ile elde edilen sınıflandırıcılar sonuç olarak bildirilmiştir. Ding ve Dubchak tarafından oluşturulan veri kümesi (DD), sınırlı sayıda protein katlamasını kapsadığı için yeni bir veri kümesi kulanmış ama DD veri kümesiniden yine de performans ölçümü için faydalanmıştır.

Yang ve Chen, 2011 yılında TAXFOLD adını verdikleri yeni bir taksonomi tabanlı yöntem geliştirmişlerdir [9]. Yaptıkları çalışmada öznitelik çıkarmanın sınıflandırmanın en önemli adımı olduğunu öngörerek PSI-BLAST ve PSIPRED profillerinden ayırımcı bilgiler içeren yeni öznitelikler çıkarmışlardır. TAXFOLD yöntemi PSI-BLAST profillerinden sekans evrim bilgisini ve PSIPRED profillerinden ikincil yapı bilgisini yoğun bir şekilde kullanmıştır. Çıkarılan öznitelikler bu profillerin hem global hem de lokal karakteristiklerini gösterir. Bir amino asit sekansının bir Pozisyon Spesifik Skor Matrisi (Position Specific Score Matrix, PSSM) olarak da bilinen PSI-BLAST profili oluşturulurken PSI-BLAST E-value 0,001 kesim değeri ile üç yinelemede çalıştırılmıştır. Evrimsel öznitelikler çıkarılırken orjinal sekans bir konsensüs sekansa dönüştürülmüştür. Bunun için bir sekansın PSSM matrisi değerleri 20 çeşit amino asitten her birinin sekansta görülme sıklığına göre sıklık matrisi değerlerine dönüştürülmüştür. Bir sekansın PIPRED profili ise PSIPRED ile tahmin edilen ikincil yapı bilgisini içerir. PSI-BLAST profillerinden 82 ve PSIPRED profilleriden 55 olmak üzere her bir sekans için toplamda 137 öznitelik içeren öznitelik

vektörü elde edilmiştir. Bu çalışmada 5 tane veri kümesi kullanılmıştır. Bunlardan ilki Ding ve Dubchak(2001) tarafından geliştirilen performans ölçümünde kullanılan veri kümesidir. Bu veri kümesinin Shen ve Chou(2006) tarafından revize edilen şekli kullanılan ikinci veri kümesidir. Bunların yanı sıra yeni SCOP versiyonuna göre ve daha fazla protein katlanmasını içeren 3 tane veri kümesi daha kullanmıştır. Sınıflandırma aşamasında DVM sınıflandırıcıları eğitilmiş ve yapılan testler sonucunda uygulanan en iyi yöntemlerin elde ettiği sonuçlar %6.9 iyileştirilmiştir.

# 3. YÖNTEMLER

## 3.1 Veri Kümesi

Proteinin üçüncül yapısını belirlemek için X-ray ışın kristalografisi [15] ve NMR (Nükleer Manyetik Rezonans) [16] gibi deneysel yöntemler tasarlanmıştır. Bu yöntemlerin maliyetinden dolayı PHD [17], PROF [18] ve NNSSP [19] gibi otomatik yapı tahmin tekniklerinin geliştirilmesi hızlı bir şekilde artış göstermiştir. Bu aletlerin çoğu protein tahmin sunucusu içine entegre edilmiştir [20].

Bu deneysel sistemlere ek olarak protein yapı tahmini alanındaki ilginç bir girişim de CASP (Critical Assessment of Structure Prediction) olmuştur. CASP 1994 yılından beri iki yılda bir dünya çapında gerçekleşen bir yarışmadır [21] [22]. Katılımcılar yapısı deneysel yöntemlerle çözülmüş ama herhangi bir veri tabanında bulunmayan protein veri kümesi için olası yapıları sunarlar. Sunulan tahminler deneysel yöntemlerle elde edilen verilerle karşılaştırılarak yarışmanın kazananları belirlenir. Bu organizasyonun birincil amacı amino asit sekanslarından proteinin üçüncül yapısını belirleyen yöntemlerin gelişmesine yardım etmek olsa da, bilimin bu alanındaki en önemli yarışmalardan biri haline gelmiştir. Tüm dünyadan yüzden fazla araştırmacı grup CASP'a düzenli olarak katılırlar. Bu grupların çoğu diğer araştırmalarını aylarca askıya alırlar ve sunucularını deneyler için hazır hale getirerek detaylı tahminler gerçekleştirmeye odaklanırlar.

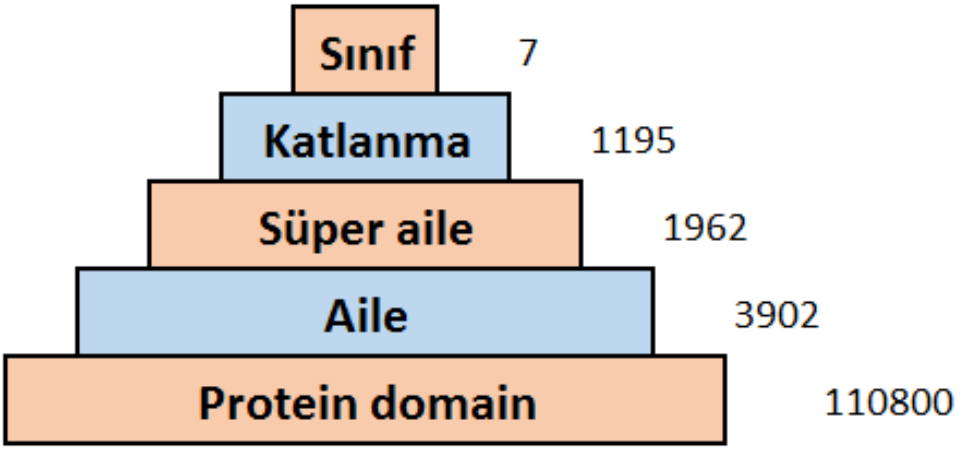
Protein yapısının öneminden dolayı bu hesaplamalı ve deneysel yöntemler kullanılarak elde edilen yapısal veriler çeşitli standartlarda veri tabanlarında toplanmıştır. Böylece verilerin saklanması, düzenlenmesi ve kolayca erişilmesi sağlanmıştır. Bu veri tabanlarından en popüler olanı ise PDB(Protein Data Bank) veri tabanıdır [23]. PDB, 1971 yılında Brookhaven National Laboratories (BNL) tarafından kurulmuşur. Bu tarihten itibaren sürekli olarak yeni protein yapıları eklense de 1980'lerden sonra teknik imkanların gelişmesiyle birlikte artış daha hızlı olmuştur ve Aralık, 2015 itibariyle X-ray kristallografisi, elektron mikroskopu ve NMR spektroskopisi gibi cihazlarla yapısı belirlenen biyolojik makromolekül sayısı 114080'e ulaşmıştır.

Her çarşamba günü bu veriler güncellenerek artmaya devam etmektedir. PDB'ye verilerin girişi, değerlendrilmesi ve onaylanarak kullanıma sunulması RCSB (Research Collaboratory for Structural Bioinformatics) tarafından yürütülmektedir.

Proteinin yapısal verileriyle ilgili popüler veri tabanlarından diğer ikisi de PDB'nin birer uzantısı olarak düşünebileceğimiz CATH (Class, Architecture, Topology and Homologous superfamily) [24] ve SCOP (Structural Classification of Proteins) [25] veri tabanlarıdır. Bu sistemler PDB'de yapısı belirlenmiş olan proteinleri belli bir hiyerarşide düzenlemektedir.

Dört katmanlı ve yarı otomatik bir yapı olan CATH, kısaltması açıldığında aslında veri tabanının amacı da anlaşılmış oluyor. Birinci katman olan sınıf (class) katmanı, domanin ikincil yapısının genel içeriğinin bulunduğu katmandır. Mimari (architecture) katmanında yüksek yapısal benzerlik var ama belli bir homoloji yok. Topoloji (topology) katmanı, özel yapısal özellikleri paylaşan geniş ölçekli topoloji grubudur. Son katman olan Homolog süperaile (homologous superfamily) ise evrimsel ilişkinin göstergesidir.

SCOP, PDB'ye girişi yapılan tüm proteinlerin yapısal ve evrimsel ilişkilerini öğrenmek için onları belli bir hiyerarşide organize eder. Şekil 3.1'de görüldüğü üzere SCOP mimarisinin dört katmanı vardır. Aile (family) katmanında iki koşul vardır; ya proteinler arasında %30 veya daha fazla sekans benzerliği olmalı ya da proteinler benzer yapı ve fonksiyonlara sahip olmalıdırlar. Süper aile (super family) katmanında proteinler arasında sekans benzerliği az ancak fonksiyon benzerliği yüksektir. Katlanma (fold) katmanında ise proteinler aynı düzenleme ve topolojik yapı ile aynı ikincil yapıya sahiptirler. En üst katman olan sınıf (class) katmanında ise tüm proteinler sadece alfa (all $\alpha$), sadece beta (all $\beta$), $\alpha/\beta$, $\alpha\&\beta$, çoklu domain proteinler (multi domain proteins), membran ve hücre yüzey proteinler ve peptitler (membrane and cell surface proteins and peptides) ve küçük proteinler (small proteins) olmak üzere yedi sınıfa ayrılırlar. SCOP veri tabanın belli bir hiyerarşik düzene sahip olması ve bu düzen içersinde katlanma katmanının da yer almasından dolayı protein katlanmasının tahmini ile ilgili yapılan çalışmalarda bu veri tabanından oldukça yaygın bir şekilde faydalanılmaktadır.




**Şekil 3.1** : SCOP Versiyon 1.75 Hiyerarşisi [3].

Bu çalışmada protein katlanmasını tanıma için önerilen yöntemin başarısını değerlendirmek için 4 adet veri kümesi kullanıldı. Bu veri kümelerinden DD veri kümesi ilk olarak Ding ve Dubchak tarafından hazırlanmıştır ve taksonomi tabanlı yaklaşımlarda en çok kullanılan veri kümelerinden bir tanesidir. Sonraki yıllarda daha gerçekçi sonuçlar elde edebilmek için daha fazla katlanma sınıfı içeren üç adet yeni veri kümesi daha oluşturulmuştur. Bu veri kümeleri Ding ve Dubchak tarafından da kullanılan Dong Q. [26] ve arkadaşlarının oluşturduğu prosedürün SCOP 1.75 versiyonuna göre güncellenmesiyle oluşturulmuştur. Bunlardan EDD olarak adlandırılan veri kümesi DD veri kümesinin genişletilmiş versiyonudur. Yani DD veri kümesi ile aynı 27 katlanma için daha fazla örnek protein sekansı içermektedir. F95 ve F194 veri kümelerinde ise daha fazla katlanma sınıfı ve protein sekansı yer almaktadır. Bu veri kümeleri aşağıda daha detaylı bir şekilde açıklanmıştır.

### 3.1.1 Ding ve Dubchak

SCOP veri tabanından faydalanılarak Ding ve Dubchak (DD) tarafından hazırlanan veri kümesi uzun yıllar birçok çalışmada kullanılmış, halen de performans ölçümü için yaygın bir şekilde kullanılmaktadır. Eğitim veri kümesinde iki protein hizalandığında 80 amino asitten uzun olan sekanslar en fazla %35 sekans benzerliği göstermektedir. Yedi veya daha fazla protein içeren ve dört temel sınıf (sadece alfa, sadece beta, alfa/beta ve alfa&beta) ile ifade edilen 27 katlanma için 313 adet örnek kullanılmıştır. Test veri kümesinde ise aynı 27 katlanma için 385 adet örnek kullanılmış ve proteinler

arasındaki sekans benzerliği en fazla %40'tır. Hazırlanan tez çalışmasında önerilen yöntemin performansını literatürde yer alan diğer çalışmalarla karşılaştırabilmek için performans ölçümü veri kümesi olarak bu veri kümesinden faydalanıldı.

### 3.1.2 EDD - Extended Ding ve Dubchak

Extended Ding ve Dubchak (EDD) olarak adlandırılan veri kümesi oluşturulurken SCOP versiyon 1.75 veri tabanından en fazla %40 sekans benzerliği olan proteinler seçilmiş ve bunların içersinden 31 amino asitten kısa olanları silinmiştir. Geriye kalan 10.493 domain sekansından Ding ve Dubchak veri kümesindeki 27 katlanma ile sınıflandırılabilen 3.397 tanesini kullanmıştır. Özet olarak EDD veri kümesi, DD veri kümesi ile aynı 27 katlanma sınıfı için 31 amino asitten uzun ve aralarında en fazla %40 sekans benzerliği olan 3397 adet örnek içermektedir.

### 3.1.3 F95 - Fold95

Daha fazla katlanmayı kapsaması amacıyla oluşturulan veri kümesinde SCOP versiyon 1.75 veri tabanından seçilen 95 katlanma sınıfı yer almaktadır. Her bir katlanma sınıfında en az 26 sekans olan veri kümesinin toplam sekans sayısı 6364'tür. 95 katlanma sınıfı içerdiği için F95 olarak adlandırılmıştır.

### 3.1.4 F194 - Fold194

Daha fazla katlanmayı kapsaması amacıyla oluşturulan diğer bir veri kümesi de F194'tür. 194 katlanma sınıfı içeren F194 veri kümesi de yine SCOP versiyon 1.75 veri tabanından seçilmiştir. Toplam 8026 örnek sekansa sahip ve her bir katlanma sınıfında en fazla 11 sekans vardır. 194 katlanma sınıfı içerdiği için F194 olarak adlandırılmıştır

## 3.2 Öznitelik Çıkarımı

Makine öğrenmesinde öznitelik çıkarımı, öznitelik kümesinden asıl öznitelikleri almak yerine orjinal veriyle en alakalı bilgiyi elde etmek için, çeşitli yöntemler kullanarak bilgiyi sabit uzunlukta sayısal vektör olarak daha düşük uzayda temsil etmektir.

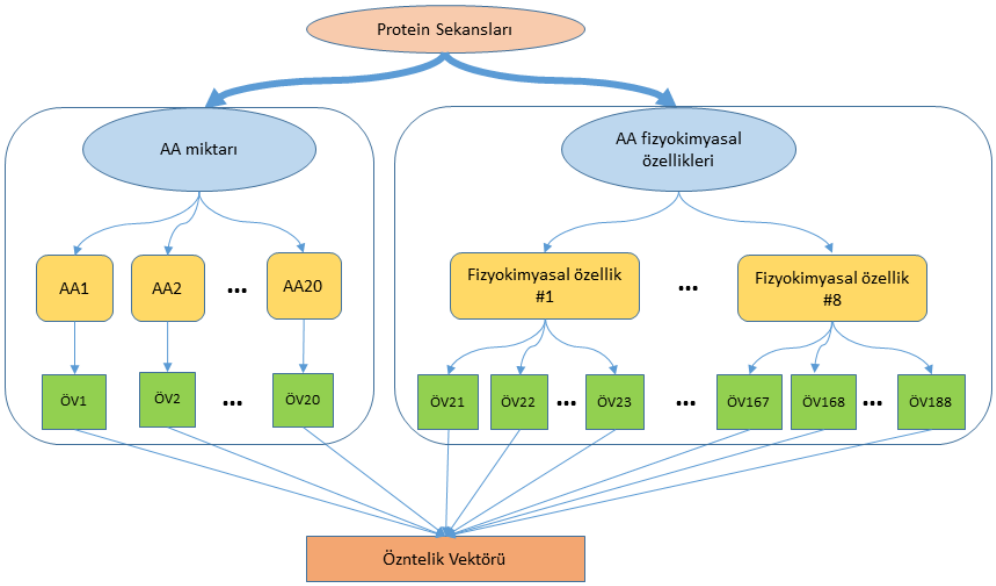
Sekans eşleştirmesi yapılan proteinler arasında benzerlik tespit edilmesi durumunda bu proteinlerin yakın evrimsel ilişkili olduğu sonucuna varılması mümkündür. Ancak yapısal olarak çok benzer olmasına rağmen aralarında sekans benzerliği olmayan

proteinler için bu tür bir eşleştirme etkili sonuç üretmeyecektir. İşte böyle bir durumda öznitelik çıkarımı, protein sekansının öznitelik uzayı ile ifade edilmesini sağlayan ve sınıflandırma başarısı için önem teşkil eden bir ara adım olarak düşünülebilir. Böylece protein sekanslarından belli öznitelikler çıkarılarak çeşitli tahmin yöntemleri uygulanabilir.

Önemli ölçüde sekans benzerliğine sahip olmayan iki proteinin yapısal olarak benzer olduğu durumlarda daha rasyonel tahminler, bir proteine ait amino asitlerin yapısal özelliklerine dayanan bilgilerle elde edilir. Buna binaen bu çalışmada, öznitelik vektörü oluşturulurken amino asitlerin fizyokimyasal özellikleri ve proteinlerin alt sekanslarından gelen bilgiden faydalanan alt sekans profil haritası (subsequence profile map, SPMap) kullanılmıştır. Alt başlıklarda bu yaklaşımlar daha detaylı bir şekilde anlatılacaktır.

### 3.2.1 Amino asitlerin fizyokimyasal özellikleri

Protein sekanslarından çeşitli fizyokimyasal özelliklere göre öznitelik çıkarılırken amino asitlerin kompozisyon, dağılım ve frekans bilgilerinden faydalanılmıştır. Bunun yanı sıra 20 çesit amino asitten her birinin sekanstaki miktarı da ayrı ayrı hesaplanmıştır. Şekil 3.2'de amino asitlerin miktarına ve fizyokimyasal özelliklerine göre özniteliklerin elde edilmesi gösterilmiştir.




**Şekil 3.2** : Amino asitlerin özelliklerine göre öznitelik çıkarma.

Kullanılan amino asit fizyokimyasal özellikleri şunlardır; hidrofobi(H), tahmin edilen ikincil yapı (S), normalize edilmiş Van der Waals şiddeti (V), polarite (P), polarizabilite (Z), yük-Charge(C), yüzey gerginliği-surface tension (T) ve solvent erişebilirlik-solvent accessibility (A). Şekil 3.3'de görüldüğü üzere amino asitler her bir fizyokimyasal özellik için üç gruba ayrılmıştır.

| Fizyokimyasal Özellik | Grup 1 | Grup 2 | Grup 3 |
|---|---|---|---|
| Hidrofobi | RKEDQN | GASTPHY | CVLIMFW |
| Normalize Van der Waals şiddeti | GASCTPD | NVEQIL | MHKFRYW |
| Polarite | LIFWCMVY | PATGS | HQRKNED |
| Polarizibilite | GASDT | CPNVEQIL | KMHFRYW |
| Yük | KR | ANCQGHILMFPSTWYV | DE |
| Yüzey gerginliği | GQDNAHR | KTSEC | ILMFPWYV |
| Solvent erişebilirlik | ALFCGIVW | RKQEND | MPSTHY |
| Tahmin edilen ikincil yapı | EALMQKRH | VIYCWFT | GNPSD |

**Şekil 3.3** : Amino asitlerin fizyokimyasal özellikleri [4].

Bu özellikleri tanımlamak için üç adet global tanımlayıcı kullanılmıştır. Bu tanımlayıcılar kompozisyon (C), frekans (F) ve dağılımdır (D). İlk tanımlayıcı C ile her bir özelliğe ait amino asit gruplarının yüzde olarak sorgu sekanstaki miktarları hesaplanır. İkinci tanımlayıcı F ile gruplar arasındaki geçiş sıklığı hesaplanır. Yani herhangi bir gruba ait amino asitlerden başka bir gruptaki amino asitlere geçişlerin sıklığı hesaplanır. Son olarak üçüncü tanımlayıcı D ile her bir gruba ait başlangıç, %25'inci, %50'nci, %75'inci ve %100'üncü amino asitlerin konumuna göre sorgu sekans 5 adet alt sekansa ayrılır ve bu alt sekanslardaki amino asitlerin miktarı yüzde olarak hesaplanır. Yukarıda bahsedilen tanımlayıcılarla yapılan hesaplamaların yanı sıra her amino asidin sekanstaki miktarı da yüzde olarak hesaplanır ve öznitelik vektörüne eklenir.

| Sekans | R | G | A | C | V | K | E | S | L | I | N | W | S | C | D | V | F | Q | G | R | Y | H | E | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sekans Numaralandırma | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 1. Grup Numaralandırma | 1 | | | | | 2 | 3 | | | | 4 | | | | 5 | | | 6 | | 7 | | | 8 | |
| 2. Grup Numaralandırma | | 1 | 2 | | | | | 3 | | | | | 4 | | | | | | 5 | | 6 | 7 | | |
| 3. Grup Numaralandırma | | | | 1 | 2 | | | | 3 | 4 | | 5 | | 6 | | 7 | 8 | | | | | | | 9 |
| 1->2 veya 2->1 Geçişleri | | → | | | | | → | | | | | | | | | | | → | → | → | | → | | |
| 1->3 veya 3->1 Geçişleri | | | | | → | | | | | → | → | | | → | → | | → | | | | | | → | |
| 2->3 veya 3->2 Geçişleri | | | → | | | | | → | | | | → | → | | | | | | | | | | | |

**Şekil 3.4** : Örnek Sekans.

Amino asitlerin fizyokimyasal özelliklerine dayanan öznitelik vektörünün elde edilmesini daha iyi anlamak için Şekil 3.4'te gösterilen farazi bir protein sekansı örnek olarak ele alınabilir. Öncelikle alfabetik sıraya göre her bir amino asidin sekanstaki miktarı hesaplanır ve öznitelik vektörünün $OV_i \ \ (i = 1,2,...,20)$ değerleri aşağıdaki gibi hesaplanır:

$$OV_i = \frac{n_j}{L} \qquad (i = 1,2,...,20; j = 1,2,...,20) \tag{3.1}$$

Denklem 3.1'de $OV_i$ öznitelik vektörünün $i$. elemanını, $L$ protein sekansının uzunluğunu, $n_j$ ise $j$ amino asidinin sekanstaki toplam sayısını gösterir.

Daha sonra amino asitler her bir fizyokimyasal özellik için üç gruba ayrılır ve bu özellikleri tanımlamak için C, F ve D tanımlayıcıları kullanılarak hesaplamalar yapılır. Örnek olarak hidrofobi özelliği ele alındığında 20 çeşit amino asit, bu özelliğe göre üç gruba ayrılır. Birinci grup hidrofilik özelliğe sahip R,K,E,D,Q ve N amino asitlerinden oluşur, ikici grup nötr G,A,S,T,P,H ve Y amino asitlerinden oluşur ve üçüncü grup hidrofobik özelliğe sahip C,V,L,I,M,F ve W amino asitlerinden oluşur.

C tanımlayıcısı ile her bir grubun sekanstaki miktarı bulunur ve gruplara ait bu boyut bilgileri kullanılarak öznitelik vektörünün $OV_i \ \ (i = 21,22,23)$ değerleri Denklem 3.2'deki gibi hesaplanır:

$$OV_i = \frac{CH_j}{L} \qquad (i = 21,22,23; j = 1,2,3) \tag{3.2}$$

Bu eşitlikte $CH_j$, hidrofobi özelliği için protein sekansındaki $j$ grubuna ait amino asit sayısını gösterir. Örnek sekansta hidrofobi özelliğine göre birinci gruptaki amino asitlerden 8 adet vardır ve $CH_1 = 8$ olur. İkinci gruptaki amino asitlerden 7 adet vardır ve $CH_2 = 7$ olur. Son olarak üçüncü grupta bulunan amino asitlerin sayısı ise 9'dur ve $CH_3 = 9$ olur. Bu doğrultuda örnek sekans için Denklem3.2'deki hesaplamalar yapılırsa aşağıdaki sonuçlar elde edilir:

$$(OV_{21}, OV_{22}, OV_{23}) = (\tfrac{8}{24}, \tfrac{7}{24}, \tfrac{9}{24}) = (0.33, 0.29, 0.37)$$

Denklem 3.3 ile F tanımlayıcısı kullanılarak iki gruba ait amino asitler arasındaki geçişler toplanır ve toplam geçişlere bölünür. Böylece öznitelik vektörünün $OV_i \ \ (i = 24,25,26)$ değerleri aşağıdaki gibi hesaplanır:

$$OV_i = \frac{FH_j}{L-1} \qquad (i = 24,25,26; j = 1,2,3) \tag{3.3}$$

$FH_j$, iki grup arasıdaki geçişlerin sayısını, $L-1$ ise $L$ uzunluğundaki protein sekansı için tüm amino asitler arasındaki toplam geçişlerin sayısını gösterir. Şekil 3.4'de görüldüğü gibi örnek sekansta, 1. grupta yer alan amino asitlerden 2. gruptakilere veya 2. grupta yer alan amino asitlerden 1. gruptakilere geçişlerin sayısı 6'dır. O halde $FH_1 = 6$ olur. Aynı şekilde 1. gruptan 3. gruba veya 3. gruptan 1. gruba geçişlerin sayısı 7'dir ve $FH_2 = 7$ olur. Son olarak 2. gruptan 3. gruba veya 3. gruptan 2. gruba geçişlerin sayısı 4'tür ve $FH_3 = 4$ olur. Toplam geçiş sayısı ise 23'tür. Bu değerlerle Denklem 3.3 hesaplandığında şu sonuçlar elde edilir:

$$(OV_{24}, OV_{25}, OV_{26}) = (\tfrac{6}{23}, \tfrac{7}{23}, \tfrac{4}{23}) = (0.26, 0.30, 0.17)$$

Amino asitlerin fizyokimyasal özelliklerine ait her bir grubun sekanstaki dağılımı D tanımlayıcısı ile hesaplanır. Öncelikle birinci grupta yer alan amino asitlerden başlangıç, %25., %50., %75. ve %100. amino asitler tespit edilir. Ardıdan bunların sekanstaki yeri bulunur ve o amino aside kadar birinci gruba ait amino asitlerin miktarı hesaplanır. Bu işlemler 2. ve 3. gruplar için de yapılır. Bu hesaplamalar sonucunda her bir grup için 5 değer ve toplamda üç grup olduğu için 15 değer elde edilir. Bu bağlamda, öznitelik vektörünün $OV_i$ $(i = 27-31; 32-36; 37-41)$ değerleri Denklem 3.4'teki gibi hesaplanır:

$$OV_i = \frac{DH_{jk}}{L} \qquad (i = 27, 28, ..., 41; j = 1, 2, 3; k = 1, 2, 3, 4, 5) \tag{3.4}$$

$DH_{jk}$, j grubundaki amino asitlerin k yüzdesindeki dağılımını gösterir. Örnek sekansta, 1. grubun ilk elemanı olan R amino asidi aynı zamanda sekansın da ilk elemanı olduğu için $DH_{11} = 1$ olur. Sekansta 1. grupta yer alan amino asitlerin sayısı 8 ve 8'in %25'i de 2'dir. O halde 1. grubun 2. elemanına bakılmalıdır. Bu eleman K amino asididir ve bu amino aside kadar sekanstaki toplam amino asit sayısı 6 olduğu için $DH_{12} = 6$ olur. Aynı hesaplamalar diğer değerleri bulmak için yapıldığında 1. grubun %50. amino asidi için $DH_{13} = 11$, %75. amino asidi için $DH_{14} = 18$ ve %100. amino asidi için $DH_{15} = 23$ olur. Bu değerlere Denklem 3.4 uygulandığında aşağıdaki sonuçlar elde edilir:

$$(OV_{27}, OV_{28}, OV_{29}, OV_{30}, OV_{31}) = (\tfrac{1}{24}, \tfrac{6}{24}, \tfrac{11}{24}, \tfrac{18}{24}, \tfrac{23}{24}) = (0.04, 0.25, 0.45, 0.75, 0.95)$$

Denklem 3.4'e göre anlatılan işlemler 2. ve 3. grup için de uygulandığında aşağıdaki sonuçlar elde edilir:

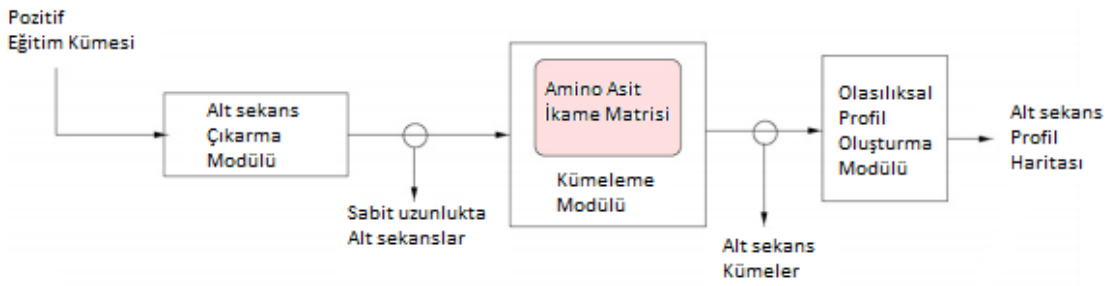
$$(OV_{32}, OV_{33}, OV_{34}, OV_{35}, OV_{36}) = (\frac{2}{24}, \frac{2}{24}, \frac{8}{24}, \frac{13}{24}, \frac{22}{24}) = (0.08, 0.08, 0.33, 0.54, 0.92)$$

$$(OV_{37}, OV_{38}, OV_{39}, OV_{40}, OV_{41}) = (\frac{4}{24}, \frac{5}{24}, \frac{10}{24}, \frac{14}{24}, \frac{24}{24}) = (0.16, 0.21, 0.42, 0.58, 1)$$

Özetle, öncelikle 20 çeşit amino asitten her birinin sekanstaki miktarı hesaplanarak 20 adet öznitelik ($OV_{1-20}$) elde edilir. Ardından amino asitlerin hidrofobi özelliğine göre C tanımlayıcısı ile 3 adet, F tanımlayıcısı ile 3 adet ve D tanımlaycısı ile 15 adet olmak üzere toplamda 21 adet öznitelik ($OV_{21-41}$) elde edilir. Yine C,F ve T tanımlayıcıları ile amino asitlerin diğer fizyokimyasal özelliklerine göre öznitelikler çıkarılarak öznitelik vektörü oluşturulur.

### 3.2.2 SPMap

Protein sekanslarından SPMap ile öznitelik çıkarılırken proteinlerin alt sekanslarından gelen bilgileri kullanan bir öznitelik uzayı eşleme (feature space mapping) tanımlanır. Bu yaklaşım hem sekans benzerliğini hem de proteinler üzerinde korunan önemli alt bölgelerden gelen bilgiyi içerir. Belirli işlevsel motiflere odaklanmak yerine, ayrıştırılarak ve boyut indirgenerek bütün alt sekanslar kullanılır [5].




**Şekil 3.5** : SPMap Yapısı [5].

Protein sekanslarının genel benzerliğine bakmak yerine alt sekansların dağılımını kullanan SPMap, protein katlanmasını tahmin etmek için pozitif örnekleri kullanan ayırt edici bir sistemdir. Bu yöntemde, öncelikle pozitif eğitim kümesindeki protein sekansları sabit uzunlukta alt sekanslara ayrılır ve bu uzunluktaki olası tüm alt sekansları kullanan bir profil oluşturulur. Daha sonra benzer alt sekanslar birlikte

kümelenerek olasılık profilleri oluşturulur ve her bir küme için bir model tanımlanır. Böylece proteinler, bu profiller üzerinde alt sekanslarının dağılımlarına göre sabit boyutlu öznitelik vektörleri ile temsil edilir ve öznitelik uzayında anlamlı bir şekilde boyut indirgenmiş olur (şekil 3.5).

İki önemli aşamadan oluşan SPMap yapısının ilk aşamasında alt sekans profil haritası oluşturulur, ikinci aşamada ise sınıflandırma işleminin yapılabilmesi için öznitelik vektörü üretilir. Aşağıda bu adımlar daha detaylı olarak anlatılacaktır.

#### 3.2.2.1 Alt sekans profil haritası oluşturma

SPMap ile öznitelik çıkarılırken bir protein sekansı olasılık profilleri üzerinde alt sekanslarının dağılımı olarak ifade edilir. Üç aşamda gerçekleşen bu sürecin ilk aşamasında pozitif eğitim kümesindeki protein sekanslarından sabit uzunluktaki olası tüm alt sekanslar çıkarılır. İkinci aşamada, uygun bir kümeleme yöntemi ile benzer alt sekanslar birlikte kümelenir. Son aşamada ise her bir küme için bir model tanımlanır. Burada en önemli adımlardan biri alt sekansların kümelenmesidir. Çünkü $l$ uzunluğundaki olası tüm alt sekansların uzayı 20 çeşit amino asit olduğu için $20^l$ dir. Pozitif eğitim kümesindeki alt sekanslar kümelenerek bu yüksek boyutlu uzayın yerine daha düşük boyutlu bir uzay elde edilir.

**Algoritma 1:** SPMap Kümeleme Algoritması

X <- Pozitif eğitim kümesinin sabit uzunluktaki tüm alt sekansları;
C <- ;
**for** *tüm* $x_i \varepsilon X$ **do**
  **for** *tüm* $C_k$ *kümeleri* **do**
    $S_k = \frac{\sum_{x_i \varepsilon C_k} s(x_i, x_j)}{|C_k|}$
  $m = argmax_{k=1...|C|} s_k$
  **if** $s_m > \delta$ **then**
    $x_i'$*yi* $C_m$ *kumesine ekle*;
  **else**
    *Yeni bir* $C_{|C|+1}$ *kumesi ekle ve* $x_i'$*yi* $C_{|C|+1}$ *kumesine ekle*;

Pozitif egitim kümesindeki protein sekanslarından belli bir $l$ uzunluğundaki olası tüm alt sekanslar çıkarıldıktan sonra Algoritma 1'de verilen kümeleme algoritması uygulanır. Algoritmada $X$, pozitif eğitim kümesindeki sabit uzunlukta alt sekanslara karşılık gelmektedir. $C$ ise küme sayısını göstermekte ve başlangıçta 0 olarak atanır. Kümeleme yapılırken her alt sekans var olan tüm kümelerle ayrı ayrı karşılaştırılır.

Her kümenin elemanları için ortalama benzerlik hesaplanırak maksimum benzerlik değerine sahip küme bulunur. İki alt sekans ($x$ ve $y$) arasındaki benzerlik Denklem 3.5 ile hesaplanır.

$$s(x,y) = \sum_{i=1}^{l} M(x(i), y(i)) \tag{3.5}$$

Denklem 3.5'te $l$ alt sekans uzunluğu ve $M(x(i),y(i))$, $x$ ve $y$'nin $i.$ elemanları için benzerlik matrisindeki değerdir. $M$ değeri için amino asit benzerlik matrisi kullanılmıştır. Alt sekans, tüm kümelerle karşılaştırılıp her küme elemanları için ortalama benzerlik değeri hesaplanır ve maksimum benzerlik ortalaması gönderen küme ($C_{max}$) belirlenir. Eğer $C_{max}$, $\delta$ (eşik-threshold) değerinden büyükse alt sekans bu kümeye atanır, aksi takdirde yeni bir küme oluşturulur ve alt sekans ona eklenir. Burada $\delta$ değeri küme sayısını belirleyen etkendir. Eğer $\delta$ değeri büyük seçilirse çok benzer alt sekanslar birlikte kümelenir ve küme sayısı çok olur. $\delta$ değerinin düşük belirlenmesi durumunda ise küme sayısı daha az olur ama biyolojik olarak birbiri ile ilgili olmayan alt sekanslar aynı kümede yer alabilir. Bu yüzden $\delta$ değeri belirlenirken hem küme sayısının çok olmamasına hem de ilgili sekansların birlikte kümelenmesine dikkat edilmelidir.

Kümeleme işleminden sonra her bir küme için olasılıksal profil oluşturulur. Bir $C_k$ kümesi için $PP_k$ olasılıksal profili $lx20$ boyutunda bir matristir. Burada $l$ bir alt sekansın uzunluğudur. $C_k$ kümesi verildiğinde bu kümenin olasılıksal profili Denklem 3.6 ile hesaplanır;

$$PP_k = log\frac{\phi_k(i,j) + \kappa}{|C_k|} \tag{3.6}$$

Denklem 3.6'da $\phi_k(i,j)$, $C_k$ kümesindeki alt sekansların $i.$ pozisyonundaki $j$ amino asitlerinin sayısıdır. $C_k$ kümesi için oluşturulan $PP_k$ olasılık profiline ait matrisin $P_k(i,j)$ girişi, alt sekansın $i.$ pozisyonunda $j$ amino asidinin olma olasılığıdır. Ayrıca her pozisyondaki amino asitler için $\kappa$ sahte-sayı (pseudo-count) eklenerek aşırı-uyum(over-fitting) ve sıfır olasılığı (zero probability) problemleri engellenmiş olur. Ardından profillerin logaritması alınır ve öznitelik vektörü oluşturma aşamasına geçilir.

#### 3.2.2.2 Öznitelik vektörü oluşturma

Bir proteinin tüm alt sekansları öznitelik vektörü oluşturmak için çıkarılır ve protein, öznitelik uzayında alt sekanslarının dağılımı olarak ifade edilir. Bunun için her bir $x$ alt sekansı her bir $PP_k$ olasılık profili ile karşılaştırılır ve olasılık Denklem 3.7 ile hesaplanır.

$$PP(x|PP_k) = \sum_{i=0}^{l} PP_k(i, x(i)) \tag{3.7}$$

Öznitelik vektörü V'nin $k$. elemanı ise Denklem 3.8'te gösterildiği gibi oluşturulur.

$$V(k) = max_{x_i \in S} P(x_i|PP_k) \tag{3.8}$$

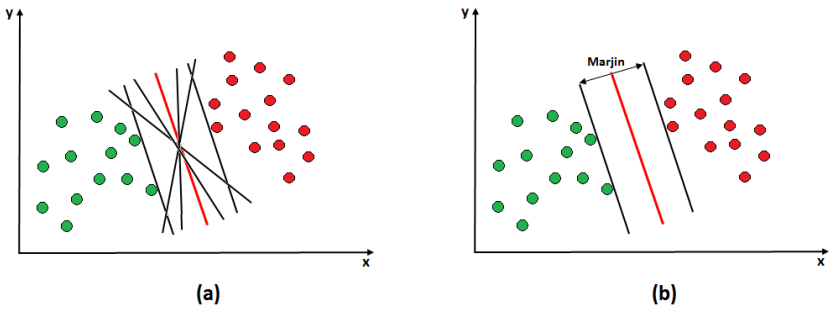
Burada öznitelik vektörü V'nin $k$. elemanına $PP_k$ olasılık profilinde $S$ proteinine ait en yüksek skorlu alt sekansın olasılık değeri atanır. Daha detaylı bilgi için [5] incelenebilir.

### 3.3 Destek Vektör Makineleri

Destek Vektör Makineleri (DVM/SVM-Support Vector Machines) algoritması ilk olarak 1963'te Vladimir Vapnik ve Alexey Chervonenkis tarafından doğrusal ayrılabilen sınıflandırma problemlerinin çözümü için önerildi. Temel olarak istatistiksel öğrenme teorisine dayanan DVM'nin günümüzde de kullanılan standart versiyonu Vapnik ve Cortes tarafından 1995'te ortaya konuldu [27].

DVM, sınıflandırma ve regresyon problemlerinde kullanılan gözetimli öğrenme modelleridir. Yani eğitim veri kümesi ile öğrenme yaparak test veri kümesi ile doğru tahmin etmeye çalışan makine öğrenmesidir. İki sınıfın yer aldığı sınıflandırma probleminde veriler doğrusal olarak ayrılabiliyorsa; Şekil 3.6'de görüldüğü gibi DVM bu sınıfları birbirinden ayırırken sınıflar arasında yer alan sonsuz sayıdaki hiper düzlemden bu sınıflara ait destek vektörler arasındaki uzaklığı maksimize eden optimal hiper düzlemi bulmaya çalışır. Sınıfların destek vektörleri hiper düzleme en yakın öğrenme verilerdir.

DVM ile doğrusal olarak ayrılamayan verileri ait oldukları boyutta bir düzlem ile ayırmak mümkün değildir. Bu yüzden n adet örnekten oluşan veri kümesi, doğrusal




**Şekil 3.6** : Destek Vektör Makineleri.

olmayan çekirdek (kernel) fonksiyonları kullanılarak p boyutlu girdi uzayından f>p olacak şekilde f boyutlu yeni bir uzaya taşınır ve bu uzayda örnekler optimal hiper düzlem ile sınıflara ayrılır.

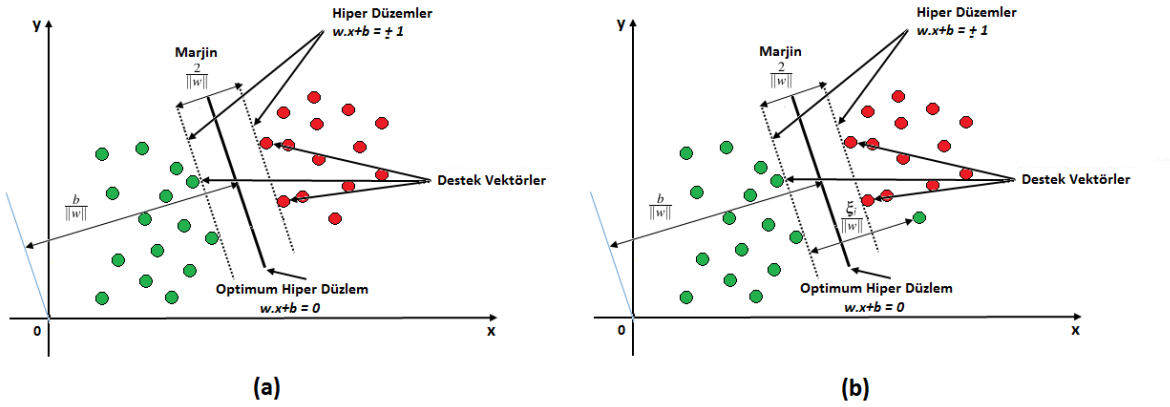
Verilerin doğrusal olarak ayrılabilen veya ayrılamayan bir yapıya sahip olmasına göre DVM temelde iki kategoride ele alınabilir. Doğrusal DVM ve Doğrusal olmayan DVM başlıkları altında bu kavramlar daha detaylı bir şekilde açıklanmıştır.

### 3.3.1 Doğrusal destek vektör makineleri

Verilerin doğrusal olarak ayrılabildiği ve $\{x_i, y_i\}$ $(i = 1, 2, ..., n)$ ikililerinin oluşturduğu veri kümesinde her bir $x_i \in R^p$ $p$ boyutlu öznitelik vektörünü, $y_i \in \{-1, 1\}$ ise $x_i$ örneğinin ait olduğu sınıfı temsil eden etiket değerini gösterir. Bu şekilde bir veri kümesi verildiğinde $y = 1$ sınıfı örneklerini $y = -1$ sınıfı örneklerinden ayıracak sonsuz sayıda hiper düzlem vardır. DVM bu hiper düzlemler arasından iki sınıfa da eşit mesafede olan hiper düzlemi bulmayı amaçlar. Böylece sınıfları birbirinden ayırt edecek en büyük marjinli doğrusal fonksiyon bulunmuş olur ve yeni bir veri ile karşılaşıldığında yanlış sınıflandırma hatası azalır. Hiper düzlem üzerindeki her $x$ noktası Denklem 3.9 ile yazılabilir.

$$w^T.x + b = 0 \tag{3.9}$$

Bu eşitlikte $w^T$, hiper düzleme dik olan normal vektörü (ağırlık vektörü), $b$ sabit değeri (bias) ve $x$ hiper düzlem üzerindeki herhangi bir noktayı gösterirken $\frac{b}{\|w^T\|}$ değeri hiper




**Şekil 3.7** : Doğrusal Destek Vektör Makineleri.

düzlemden orjine olan dik uzaklığa karşılık gelir ve hiperdüzlemin toleransı (offset) olarak adlandırılan iki sınıfa ait örnekler arasındaki mesafe farkının bulunmasını sağlar. Şekilde görüldüğü üzere bu mesafeyi maksimuma çıkararak en iyi ayrımı yapan hiper düzleme *optimal ayırıcı hiper düzlem* denir. Optimum hiper düzlem belirlenirken bu düzleme eşit mesafede ve paralel olan $y=-1$, $y=1$ sınıflarının sınırlarını belirleyen iki hiper düzlem oluşturulur. Şekilde kesikli çizgilerle gösterilen bu hiper düzlemlerin fonksiyon gösterimleri aşağıdaki gibidir.

$$w^T.x+b=-1, \qquad y=-1 \ ise \tag{3.10}$$

$$w^T.x+b=1, \qquad y=1 \ ise \tag{3.11}$$

Bu iki eşitliği sağlayan hiper düzlemler arasındaki mesafeye *marjin* denir. Bu hiper düzlemler üzerinde bulunan ve sınırı belirleyen noktalara ise destek vektörleri adı verilmektedir. Marjini maksimum yapan hiper düzlem optimal ayırıcı hiper düzlemdir. $y=-1$ sınıfını ayıran hiper düzlem için Denklem 3.10 kullanılır ve bu düzlemin orjine dik uzaklığı $\frac{|-1-b|}{\|w^T\|}$ olur. $y=1$ sınıfını ayıran hiper düzlem içinse Denklem 3.11 kullanılır ve bu düzlemin orjine dik uzaklığı $\frac{|1-b|}{\|w^T\|}$ olur. Optimum hiper düzlemin orjine uzaklığı $\frac{b}{\|w^T\|}$ olduğu göz önünde bulundurulduğunda her iki hiper düzlemin de optimal hiper düzleme uzaklığı $\frac{1}{\|w^T\|}$ olur. Optimum hiper düzleme ve birbirlerine paralel olan bu hiper düzlemler arasındaki uzaklık yani marjin değeri $\frac{2}{\|w^T\|}$'dir. Dolayısıyla

marjinin maksimum değeri, $\left\|w^T\right\|$ ağırlık vektörünün minimize edilmesiyle bulunur. Bu durumda herhangi bir veri noktasının marjin içinde yer almaması ve marjinin maksimum olması için aşağıdaki koşullu optimizasyonun çözülmesi gerekir:

$$min\frac{1}{2}\left\|w^T\right\|^2 \tag{3.12}$$

Buna bağlı koşullar ise aşağıdaki gibidir:

$$w^T.x_i + b \leq -1, \qquad y_i = -1 \ ise \tag{3.13}$$

$$w^T.x_i + b \geq 1, \qquad y_i = 1 \ ise \tag{3.14}$$

Eğitim veri kümesi için bu eşitsizlikler aşağıdaki gibi bir arada ifade edilebilir:

$$y_i(w^T.x_i + b) \geq 1, \qquad 1 \leq i \leq n \tag{3.15}$$

Bu optimizasyon probleminin çözülmesi ile sınıflar arasındaki uzaklığı maksimize edecek optimal ayırıcı hiper düzlem elde edilir [28] .

Doğrusal ayrılma durumunda verilerin birbirinden tamamen doğrusal olarak ayrılması gerekmektedir. Ancak pratikte bu durum çoğu kez geçerli olmamakta ve Şekil 3.7'de görüldüğü gibi veriler birbirinden ancak belli bir hata ile doğrusal olarak ayrılabilmektedir. Yani verilerin bir kısmı hiper düzlemin diğer tarafında kalmaktadır. Bu durum genellikle veri kümesinin çok boyutlu ve karmaşık bir yapıya sahip olması veya gürültülü veri içermesinden kaynaklanmaktadır. Böyle bir veri kümesinde iki sınıfı birbirinden ayırırken yumuşak marjin (soft margin) yaklaşımı kullanılır ve $\xi_i$ yapay değişkeni tanımlanır. Marjinin maksimum olması ve yanlış sınıflandırma hatalarının minimum olması arasındaki dengenin sağlanması için bir düzenleme parametresi tanımlanır. Bu düzenlemeler ile optimizasyon problemi aşağıdaki yeni şeklini alır:

$$min\frac{\left\|w^T\right\|^2}{2} + C\sum_{i=1}^{r}\xi_i \tag{3.16}$$

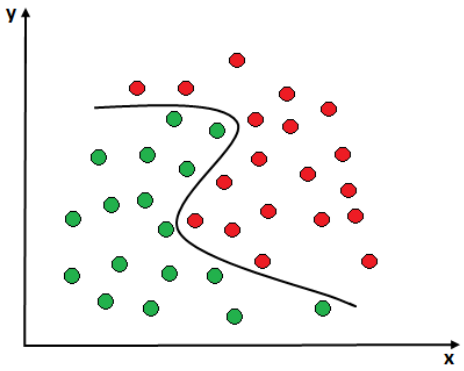
$\xi_i$, yanlış sınıflandırılan örneğin sınır hiper düzlemine olan uzaklığı gösteren yapay değişken, $C$ ise pozitif değerler alan düzenleme parametresi. $C$'nin değerinin yüksek olması hata beklentisinin yüksek olduğu anlamına gelir. Bu optimizasyon problemine bağlı koşullar ise aşağıdaki gibidir:

$$< w^T .x_i > +b \geq +1 - \xi_i \qquad y_i = -1 \ ise \tag{3.17}$$

$$< w^T .x_i > +b \leq -1 + \xi_i \qquad y_i = 1 \ ise \tag{3.18}$$

### 3.3.2 Doğrusal olmayan destek vektör makineleri

Doğrusal olmayan DVM algoritmaları, veri kümesinde yer alan verilerin doğrusal olarak ayrılmalarının mümkün olmadığı durumlarda kullanılır (Şekil 3.8). Bu veriler ait oldukları boyutta bir düzlem ile ayrılamayacağı için doğrusal olmayan haritalama yaklaşımı ile daha yüksek boyutlu bir uzaya taşınır. Ardından DVM bu uzayda doğrusal DVM gibi çalışır ve verileri ayırmak için optimal ayırıcı hiper düzlemi bulmayı amaçlar. DVM'den farklı olarak dönüştürülmüş uzayda $x$ yerine $\phi(x)$ kullanılır. Bu durumda doğrusal olmayan DVM için karar fonksiyonu Denklem 3.19'daki gibi yazılır.




**Şekil 3.8** : Doğrusal Olmayan Destek Vektör Makineleri.

$$y_i(w^T .\phi(x)_i + b) \geq 1, \qquad x \in R^p \rightarrow \phi(x) \in R^f \tag{3.19}$$

$\phi$, doğrusal olmayan haritalamayı ve $\phi(x)$, yeni nitelik uzayındaki $x$ vektörünü göstermektedir. $x \in R^p$, $p$ boyutlu $x$ öznitelik vektörünü ve $\phi(x) \in R^f$ ise $f$ boyutlu

$\phi(x)$ öznitelik vektörünü temsil eder. Doğrusal olmayan çekirdek fonksiyonları kullanılarak $f > p$ olacak şekilde $p$ boyutlu girdi uzayından $f$ boyutlu nitelik uzayına geçiş yapılır [29].

### 3.4 Karar Ağacı

Karar ağacı, sınıflandırma ve regresyon modellerinde etkin bir şekilde kullanılan, dağılımdan bağımsız gözetimli öğrenme modelidir. Akış diyagramı şeklindeki karar ağaçları iç karar düğümleri, dallar ve uç yapraklardan oluşur. İç karar düğümleri, verilere test fonksiyonunun uygulandığı ve ne tarafa dallanacaklarını belirleyen karar düğümleridir. Dallar ise uygulanan testlerin sonucunu temsil eder. Uç yapraklar sınıflandırmada sınıf etiketlerine, regresyonda sayısal değerlere karşılık gelir.

Karar ağacı öğrenmesi, iç karar düğümleri, dallar ve uç yapraklardan oluşan bir ağaç yapısı ile veri kümesine bağlı olarak denetim işlevleri gerçekleştirilerek ağacın yaprakları seviyesinde sınıf etiketlerini ifade etmektir. Bir eğitim kümesini öğrenen birden fazla karar ağacı olabilir. Böyle bir durumda basitlik ilkesinden dolayı en küçük boyutlu karar ağacı tercih edilir. Bir ağacın boyutu düğüm sayısına ve düğümlerdeki denetim işlevlerinin karmaşıklığına bağlıdır. Bu karmaşıklığa göre ağaca yeni dallar ve yapraklar eklenerek ağaç büyütülebilir.

Karar ağacı oluşturulurken veri kümesi eğer-ise (if-then) kuralına göre yinelemeli olarak daha küçük alt kümelere ayrılır. İlk yineleme eğitim kümesinin tamamı ile kökte başlar. Ağaç yapısındaki her karar düğümü $m$, sonucu dallara karşılık gelen bir $f_m(x)$ test fonksiyonu gerçekler ve karar düğümüne ilişkin tüm olası değerlere karşılık gelen dallardan biri seçilir. Kökte başlayan bu yinelemeli süreç yaprak düğümüne gelip çıktı oluşturulana kadar devam eder. İç karar düğümleri belirlenirken bilgi kazancı yöntemleri uygulanır ve bilgi kazancı en büyük olan nitelik, örnekleri en iyi bölecek olan karar düğümü niteliği olarak seçilir [30].

Karar ağacı yönteminin alt yöntemleri sınıflandırma ağaçları ve regresyon ağaçlarıdır. Sınıflandırma ağacı öğrenmesinde her yaprak bir sınıf etiketine karşılık gelir ve hedef küme bu sınıflara atanır. Regresyon ağacı öğrenmesinde ise yapraklarda sınıf yerine sayısal değerler vardır. Sınıflandırma ağacında bir bölmenin iyliği katışıklık ölçütü (impurity measure) ile hesaplanır. Eğer bölme sonrasında o dala düşen örneklerin hepsi

aynı sınıfa aitse o bölme saftır(pure) denilebilir. Katışıklık ölçütü olarak genellikle entropi kullanılır [31]. m düğümünün entropisi Denklem 3.20'deki gibi hesaplanır.

$$I_m = -\sum_{i=1}^{K} (\frac{N_m^i}{N_m}) \log_2 (\frac{N_m^i}{N_m}) = -\sum_{i=1}^{K} p_m^i \log_2 p_m^i \tag{3.20}$$

$N_m$, $m$ düğümüne ulaşan örnek sayısını ve $N_m^i$, $N_m$ örnekten $C_i$ sınıfında olanların sayısını gösterir. Bu durumda $p_m^i$, $m$ düğümüne ulaşan bir örneğin $C_i$ sınıfında olma olasılığına karşılık gelir. Eğer $m$ düğümüne ulaşan örneklerin hiçbiri $C_i$ sınıfından değilse $p_m^i$ 0 olur, hepsi $C_i$ sınıfından ise 1 olur. m düğümünde tüm $p_m^i$ değerleri 0 veya 1 ise bu düğüm saf bir düğümdür ve artık bölmeye gerek yoktur. Böyle bir durumda $p_m^i$'in 1 olduğu sınıfla etiketlenen bir yaprak eklenerek işlem sonlandırılır. Bir bölmeden sonra o düğümün dallarına düşen verinin toplam katışıklılıkğı yine entropi ile hesaplanabilir (Denklem 3.21).

$$I'_m = -\sum_{i=1}^{n} (\frac{N_{mj}}{N_m}) \sum_{i=1}^{K} p_{mj}^i \log_2 p_{mj}^i \tag{3.21}$$

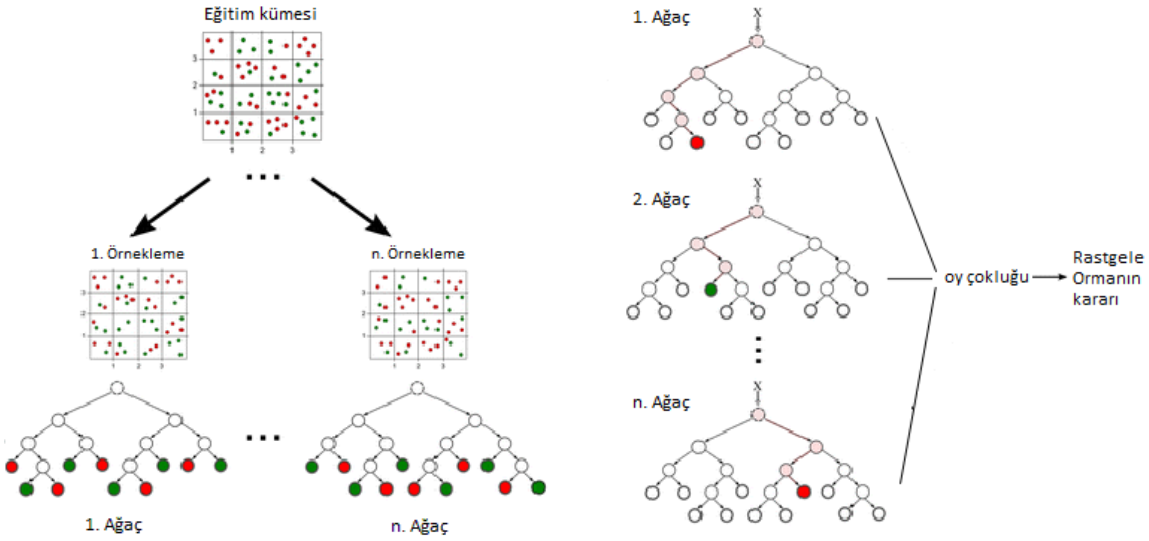
$N_{mj}$, $m$ düğümündeki $N_m$ örnekten $j$ dalına düşenlerin sayısı ve $p_{mj}^i$, $j$ dalına düşen bir örneğin $C_i$ sınıfında olma olasılığıdır. $m$ düğümüne ulaşan verinin katışıklılığı $(I_m)$ ile bölmeden sonra o düğümün dallarına düşen verinin toplam katışıklılığı $(I'_m)$ arasındaki farka bakılarak katışıklılığı en fazla azaltan bölme seçilir. Yani saflığı en düşük olan girdi bölme işlemi için seçilir [30].

### 3.5 Rastgele Orman

Rastgele Orman (RO), sınıflandırıcı olarak karar ağaçlarını kullanan ve sınıflandırma doğruluğunu önemli ölçüde artıran bir toplu öğrenme (ensemble learning) yöntemidir. Classification and Regression Tree (CART) algoritması ile ağaç yapısı maksimum olacak şekilde büyütülüp budama yapılmamasına rağmen yöntem, aşırı uyum (overfitting) problemine karşı dayanıklıdır [32] [33]. Oldukça hızlı çalışan RO yönteminde, ağaç sayısı istenilen sayıda ayarlanabilir.

RO yönteminde önyüklemeli örneklem (bootstrap sampling) tekniği ve rastgele özellik seçimi (random feature selection) ile her biri farklı eğitim kümesinde eğitilen karar ağaçları oluşturulur. Yani her bir karar ağacı için orjinal veri kümesinden önyükleme

tekniği ile örneklem yapılarak yeni veri kümeleri üretilir. Ağaçlar eğitilirken tüm değişkenler içersinden en iyi değişkeni seçip her bir düğümü dallara ayırmak yerine her bir düğümde rastgele değişkenler seçilir.




**Şekil 3.9** : Rastgele Orman kararı [6].

RO yönteminde belirlenen parametreler; $n$, oluşturulacak ağaç sayısı ve $m$, her düğümde rastgele seçilen tahmin değişkenlerinin sayısıdır. Veri kümesindeki toplam değişken sayısı $M$ olmak üzere, $m < M$ koşuluna göre seçilen tahmin değişkenlerinin sayısı tüm orman için sabittir. $m$ değeri, $M^{1/2}$ olarak seçildiğinde genellikle optimum sonuca en yakın sonucu verir [34]. Şekil 3.9'te görüldüğü üzere $n$ tane karar ağacı yapısı büyütmek için orjinal veri kümesinden $n$ tane önyüklemeli örnekleme yapılır. Oluşturulan veri kümeleri için eğitim ve test kümeleri önceden belirlenmemişse her bir veri kümesinin $2/3$'si ağaç yapısını oluştumak için eğitim verisi (inBag), kalan $1/3$'i ise oluşturulan ağacın performansını ölçmek için test verisi (OutOfBag, OOB) olarak ayrılır [35]. Her bir ağaç yapısının inBag veri kümesinden rastgele $m$ tane tahmin değişkeni seçilir. Bu değişkenlerden hangisi ile dallanmanın başlayacağı bilgi kazancı hesaplaması ile belirlenir. Belirlenen değişkenin en iyi dallanma kriteri Gini indeksi ile hesaplanır. Bir değişkenin dallara ayırmadaki gücünü ölçen Gini indeksindeki azalış o sınıfın saflılığının arttığı anlamına gelir. Bu şekilde ormandaki ağaçların hepsi oluşturulduktan sonra her ağacın performansı, ayrılan OOB test verisi ile test edilir ve her bir ağaç için hata oranı hesaplanır ve bu değerle ters orantılı olarak da ağaç ağırlıklandırılır. Yeni bir veri sınıflandırılırken bu veri, oluşturulan $n$ tane

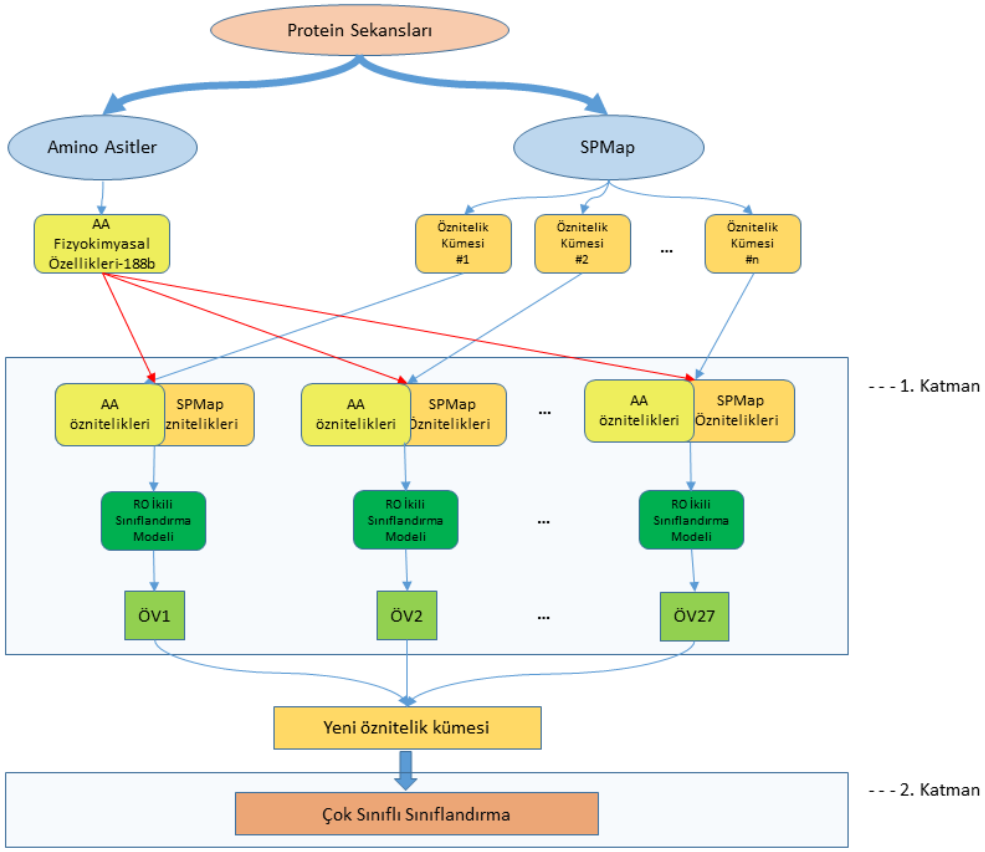
ağacın hepsine gönderilir. Her ağaç ağırlık değerine göre bir sınıflandırma tahmininde bulunur. Dolayısıyla *n* tane tahmin yani oy elde edilmiş olur. Veri, en yüksek oyu alan sınıf ile sınıflandırılır.

Bir değişkenin Gini önem derecesi hesaplanırken, o değişkenin dallara ayrılmadan önceki gini değeri bulunur. Dallara ayrıldıktan sonra, ayrılan verilerin gini değerlerinin toplamı bulunur ve bu iki değer arasındaki fark hesaplanır. Bu işlem ormanda bu değişkeni kullanan tüm ağaçlarda yapılır ve hesaplanan farkların toplamı o değişkenin gini önem derecesi olur.

# 4. DENEYLER

Bu bölümde hazırlanan tez çalışmasının genel süreci, uygulama adımları ve yapılan deneylerin sonuçları anlatılacaktır. Ayrıca sınıflandırıcı performansını ölçerken kullanılan başarı değerlendirme ölçütlerinden bahsedilecektir. Çalışmanın akış diyagramı Şekil 4.1'de gösterilmektedir.

Proteinlerden makine öğrenmesi yöntemlerinin uygulanabileceği anlamlı veriler elde etmek için öznitelik çıkarımı yapılırken iki yöntem kullanıldı. Yöntemler bölümünde detaylı bir şekilde anlatılan öznitelik çıkarma yöntemlerinden birincisi amino asitlerin fizyokimyasal özelliklerine; ikincisi ise SPMap yöntemi ile proteinlerin alt sekanslarından çıkarılan bilgilere dayanır.




**Şekil 4.1** : Tez çalışmasının genel yapısı.

Şekil 4.1'de görüldüğü gibi proteinlerin katlanma örüntülerinin tahmininde iki katmanlı bir yaklaşım önerildi. İlk katman öncesinde amino asitlerin fizyokimyasal özelliklerine dayanarak çıkarılan 188b boyutlu öznitelik vektörü, her katlanma sınıfı için ayrı ayrı üretilen SPMap öznitelik kümelerinin her birine eklendi. İlk katmanda, bu öznitelik kümeleri, pozitif ve negatif olarak yeniden etiketlendi. Bu katmanda kullanılan veri kümesini, eğitim kümesi ve test kümesi oluşturacak şekilde ayırmak için 5 kat çapraz doğrulama (5-fold Cross Validation) yöntemi kullanıldı. Daha sonra eğitim veri kümesi ile ikili sınıflandırma modelleri eğitildi. Oluşturulan modeller ile test veri kümesi sınıflandırıldı. Bu sınıflandırma sonucu elde edilen sayısal sınıflandırma değerleri öznitelik vektörleri gibi düşünülerek birleştirildi ve ikinci katmanda kullanılmak üzere n boyutlu yeni bir öznitelik kümesi oluşturuldu. İkinci katmanda, çoklu sınıflandırma modeli eğitilerek protein katlanmaları tahmin edildi.

Birinci katmanda sınıflandırma yöntemi olarak RO kullanılırken, ikinci katmanda çeşitli sınıflandırma yöntemleri denendi. Birinci katmanda R ortamında ikili sınıflandırma yapıldı. İkinci katmanda çoklu sınıflandırma yaparken hem R hem de Weka [36] kullanıldı. R ortamında RO ve DVM yöntemleri kullanıldı. RO yöntemi için orjinali Breiman tarafından yazılan algoritmanın R uyarlaması olan randomForest R paketi kullanılırken DVM için e1071 R paketi kullanıldı. Weka ortamında ise DVM, çoklu sınıflandırma (multi-class classifier) ve sınıflandırıcı toplulukları (ensemble classifier) yöntemleri kullanıldı. Weka ile yapılan denemelerde Weka arayüzünden faydalanıldı. RO yönteminde ağaç sayısı ilk katman için 100; ikinci katman için 500 olarak belirlendi. DVM için kernel olarak hem R hem de Weka ortamında Polykernel seçildi. Sınıflandırıcı toplulukları yönteminde iki tane sınıflandırıcı kullanıldı. Bunlardan birisi DVM ile çoklu sınıflandırma, diğeri ise Lojistik Regresyon (Logistic Regression) ile çoklu sınıflandırmadır. Çoklu sınıflandırma yaparken birine karşı hepsi (1 vs all) yöntemi uygulandı. İkinci katmanda, geliştirilen modelin test edilmesi için eğitim veri kümesinden bağımsız test veri kümesi kullanıldı. Test veri kümesi ile geliştirilen yöntemin başarısı değerlenirildi.

İlk katmanda, her katlanma sınıfı için SPMap öznitelik kümeleri, birine karşı hepsi yöntemi kullanılarak pozitif ve negatif olarak etiketlendi. Yani veri kümesindeki n tane katlanma sınıfından bir tanesi pozitif, diğerleri negatif olarak etiketleme yapıldı. Dolayısıyla negatif örneklerin sayısı, pozitif örneklerin sayısından çok fazla

olduğu için dengesiz sınıf dağılımı problemi ortaya çıktı. Bu problemin önüne geçmek için kullanılan yöntemlerden biri örneklemedir. İlk katmanda kullanılan RO yönteminde, dengesiz sınıf dagılımının olumsuz etkisini azaltmak için örnekleme boyutu parametresi olarak en küçük sınıfın örnek sayısı baz alındı. Pozitif sınıfın boyutunun m olduğu varsayılırsa örnekleme boyutu parametresi [m,m] olur. Yani m boyutlu pozitif sınıfındaki örneklerin tamamı kullanılırken, negatif sınıfındaki örneklerden m tanesi seçilir. Böylece veri kümesindeki dengesizlik probleminin önüne geçilmiş olur.

## 4.1 Sınıflandırma Modeli Değerlendirme Ölçütleri

Eğitim kümesi ile sınıflandırma modelleri geliştirildikten sonra, test kümesi ile geliştirilen bu modellerin performansı değerlendirilir. Performansın değerlendirilmesinde çeşitli ölçütler kullanılır. Bu çalışmada, her bir katlanma sınıfı için geliştirilen modelin performansını değerlendirmede dört ölçüt kullanıldı. Bu ölçütler; kesinlik (precision), duyarlılık (recall), F-ölçütü (F-measure) ve doğruluk oranı (accuracy)'dır.

**Çizelge 4.1** : Karışıklık Matrisi

| | | **Tahmin Edilen Sınıf** | |
|---|---|---|---|
| | | **Pozitif Sınıf** | **Negatif Sınıf** |
| **Gerçek Sınıf** | **Pozitif Sınıf** | DP | YN |
| | **Negatif Sınıf** | YP | DN |

Pozitif ve negatif sınıflardan oluşan ikili sınıflandırma problemlerinde model değerlendirme ölçütleri, Çizelge 4.1'de gösterilen Karışıklık matrisine (Confusion matrix) dayanmaktadır. Karışıklık matrisinde, satırlar test kümesindeki örneklerin gerçek sınıflarını temsil ederken; sütunlar geliştirilen modelin tahmin ettiği sınıfları temsil eder. Bu çizelge yapısı ile sınıflandırıcıların pozitif ve negatif sınıflara ait örnekleri nasıl tahmin ettiğine bakılabilir ve geliştirilen modelin başarısı değerlendirilebilir. Karışıklık matrisinde kullanılan kavramlar şunlardır; DP(Doğru Pozitif), YP(Yanlış Pozitif), DN(Doğru Negatif) ve YN(Yanlış Negatif). DP, gerçek sınıfı pozitif olan yani pozitif olarak etiketlenen örneklerin pozitif tahmin edilerek doğru sınıflandırıldığını; YP, gerçek sınıfı negatif olan örneklerin pozitif tahmin edildiğini; DN, gerçek sınıfı negatif olan örneklerin negatif tahmin edildiğini; ve YN, gerçek sınıfı pozitif olan örneklerin negatif tahmin edildiğini ifade eder.

Model değerlendirme ölçütlerinden en basit olan ve en yaygın kullanılanı doğruluk oranıdır. Karışıklık matrisi kullanılarak Denklem 4.1'de gösterildiği gibi hesaplanan doğruluk oranı, doğru sınıflandırılan örnek sayısının toplam örnek sayısına oranıdır. Denklem 4.2'de verilen hata oranı ise yanlış sınıflandırılan örnek sayısının toplam örnek sayısına oranıdır. Doğruluk oranı ve hata oranı toplamı 1'e eşittir.

$$DogrulukOranı = \frac{DP + DN}{DP + DN + YP + YN} \tag{4.1}$$

$$HataOranı = \frac{YP + YN}{DP + DN + YP + YN} \tag{4.2}$$

Dengesiz veri kümesi ile sınıflandırma yaparken model başarısını sadece doğruluk oranı ile değerlendirmek yeterli değildir. Bunun yanı sıra kesinlik, duyarlılık ve F-ölçütü değerlerine de bakılması gerekir. Kesinlik, gerçek sınıfı pozitif olan ve pozitif tahmin edilen örnek sayısının, pozitif tahmin edilen örneklerin tamamına bölünmesi ile bulunur (Denklem 4.3). Duyarlılık ise gerçek sınıfı pozitif olan ve pozitif tahmin edilen örnek sayısının, gerçek sınıfı pozitif olan örneklerin tamamına oranıdır (Denklem 4.4). Başka bir ifade ile kesinlik ile pozitif tahmin edilen örnekler içersinde gerçekte sınıf etiketi pozitif olan örneklerin oranı ölçülürken; duyarlılık ile gerçekte sınıf etiketi pozitif olan örnekler içersinde doğru tahmin edilen örneklerin oranı ölçülür. Kesinlik ve duyarlılık ölçütlerinin birlikte değerlendirilmesi ile Denklem 4.5'de verilen F-ölçütü hesaplanır. F-ölçütü, kesinlik ve duyarlılık değerlerinin harmonik ortalamasının alınması ile bulunur. Geliştirilen model ne kadar iyi ise F-ölçütü o kadar büyük bir değer alır.

$$Kesinlik = \frac{DP}{DP + YP} \tag{4.3}$$

$$Duyarlılık = \frac{DP}{DP + YN} \tag{4.4}$$

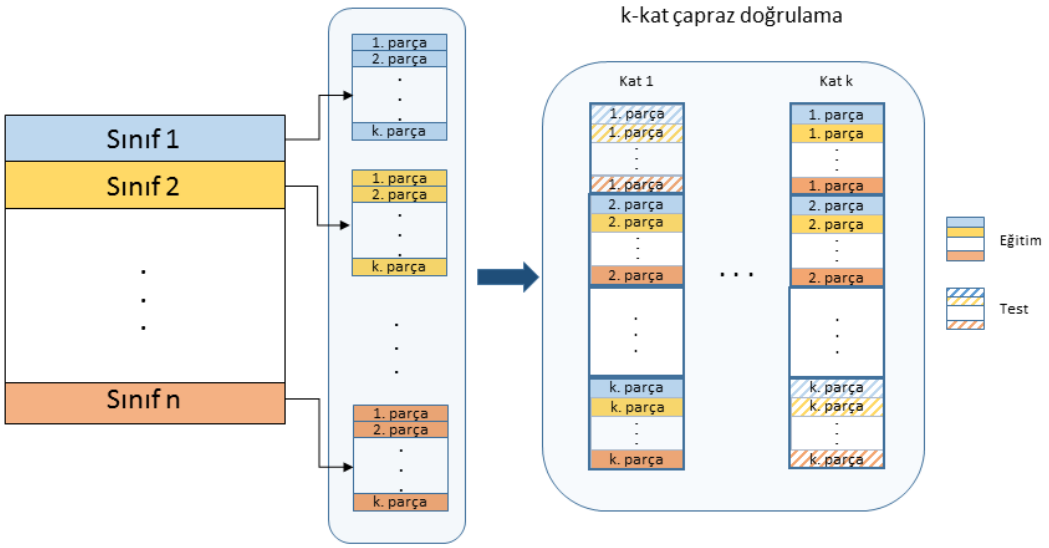
$$F - ölçütü = 2\frac{duyarlılık * kesinlik}{duyarlılık + kesinlik} \tag{4.5}$$

İkili sınıflandırma için kullanılan F-ölçütü, kesinlik ve duyarlılık ölçütleri, çoklu sınıflandırmada kullanılırken sınıflardan biri pozitif diğerleri ise negatif olarak düşünülür ve hesaplamalar ona göre yapılır.

## 4.2 Deneyler ve Sonuçları

Deneyler DD, EDD, F95 ve F194 veri kümeleri ile yapıldı. Orjinal DD veri kümesi, birbirinden bağımsız eğitim ve test kümesinden oluşmaktadır. Eğitim kümesi, ilk katmanda geliştirilen ikili sınıflandırma modelleri için veri kümesi olarak düşünüldü ve ikili sınıflandırma yaparken bu veri kümesi, 5-kat çapraz doğrulama yöntemi ile eğitim kümesi ve test kümesi olarak ayrıldı. İlk katmanda geliştirlen model test edildikten sonra elde edilen sonuçlarla ikinci katmanda kullanılmak üzere yeni bir öznitelik vektörü oluşturuldu. Ardından eğitim kümesinden bağımsız test kümesi ile ikinci katmandaki model test edildi.

EDD, F95 ve F194 veri kümeleri ile uygulanan yöntemin başarısı test edilirken, veri kümesi eğitim ve test kümesi olarak ayrıldı. Bu ayırma işleminin nasıl yapıldığı Şekil 4.2'de gösterilmektedir.




**Şekil 4.2** : Veri kümesinin eğitim kümesi ve test kümesi olarak ayrılması.

Şekil 4.2'de n, veri kümesindeki katlanma sınıflarının sayısını ve k, veri kümesinin kaç parçaya ayrılacağını gösteren değerdir. k değeri belirlenirken veri kümesindeki her katlanma sınıfının örnek sayısına bakıldı ve örnek sayısı en az olan sınıf belirlendi.

Eğer bu sınıfın örnek sayısı 20'den küçükse k'ye bu sayı atandı. 20'den büyük olması durumda ise k'ye 20 değeri atandı. k değeri belirlendikten sonra veri kümesindeki her sınıfın örnekleri k parçaya ayrıldı. Ardından şekilde görüldüğü gibi her bir sınıfın 1.,2.,...,k. parçaları bir araya gelecek şekilde ayrı ayrı birleştirildi. Bu işlem sonucunda k parçaya ayrılan veri kümesinin ilk (k-1) tanesinin örnek sayısı eşit olur, k. parçasında ise her sınıfın kalan örnekleri yer alır. Bu aşamadan sonra, k kat çapraz doğrulama ile her defasında bir parça hariç bırakılarak sistem eğitilir ve hariç bırakılan parça ile eğitilen sistem test edilir. Bu işlem, sistemi her parça ile sınayana kadar yani k defa yapılır. Böylece her sınıfın performansı ölçülmüş olur. Örneğin; EDD veri kümesinde 27 katlanma sınıfı vardır ve en az örneğe sahip olan 24. sınıfın örnek sayısı 16'dır (Yani n=27 ve k=16). O halde her bir sınıfın örneklerini 16 parçaya ayırmalıyız. Mesela 1. sınıfın örnek sayısı 41'dir ve floor(41/16)=2 sonucunu verir. Dolayısıyla ilk 15 parçada 2'şer örnek, son parçada ise 41-(15*2)=11 örnek olacak şekilde 1. sınıf 16 parçaya ayrılır. Aynı işlemler diğer sınıflar için de yapılır ve sonuçta 27 tane 16 parçaya bölünmüş veri kümeleri elde edilir. Ardından bu veri kümelerinin her birinin 1.,2.,...,16. parçaları ayrı ayrı birleştirilir. Bu aşamada artık elimizde 16 parçaya ayrılmış bir veri kümesi vardır. Bu veri kümesini eğitim kümesi ve test kümesi olarak ayırırken 16 parçadan biri test diğerleri eğitim kümesi olarak 16 farklı konfigürasyon ile yöntemin başarısı test edilir.

DD veri kümesi 27 sınıftan oluşmaktadır. Eğitim kümesinde 313 ve test kümesinde 385 örnek vardır. Bu veri kümesinde eğitim ve test kümesi birbirinden bağımsız olduğu için veri kümesi ayırma işlemi gerçekleştirilmedi. DD veri kümesi ile uygulanan yöntem test edildi. İkinci katmanda çoklu sınıflandırma yöntemi olarak RO kullanıldığında ortalama %64.2, DVM kullanıldığında ise **%71.7** başarı elde edildi.

DD veri kümesi ile aynı 27 katlanma sınıfına sahip olan EDD veri kümesinde toplam 3397 örnek vardır. Önerilen yöntemi ikinci katmanda test etmek için, uygulamanın başında veri kümesi eğitim ve test kümesi olarak ayrıldı.Bu işlem örnek sayısı en az olan sınıfa göre gerçekleştirildi. En az örneği olan sınıfın örnek sayısı 16'dır. Bu yüzden Şekil 4.2'de gösterildiği gibi veri kümesi 16 eşit parçaya ayrıldı ve tüm sınıfların aynı numaraya sahip parçaları birleştirildi. Ardından deneyler 16 kat-çapraz doğrulama ile yapıldı. Bu işlemler uygulandığında ilk 15 kat için eğitim kümesinde 3197, test kümesinde 200 örnek vardır. 16. kat için eğitim kümesinde 3000, test

**Çizelge 4.2** : EDD veri kümesi kullanılarak SPMap yöntemiyle çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%)

| KAT | RO(R) | DVM(R) | DVM(Weka) | ÇS(LR) | ÇS(DVM) | ST |
|---|---|---|---|---|---|---|
| **1** | 68 | 70,5 | 71,5 | 76 | 78,5 | 79 |
| **2** | 70,5 | 79 | 79,5 | 76,5 | 78,5 | 78,5 |
| **3** | 69 | 73,5 | 75,5 | 79 | 80 | 79,5 |
| **4** | 74 | 77,5 | 78 | 82,5 | 84,5 | 84 |
| **5** | 80 | 85,5 | 86 | 89 | 87,5 | 89 |
| **6** | 67,5 | 74,5 | 72,5 | 78,5 | 75 | 79 |
| **7** | 73 | 77,5 | 76,5 | 76,5 | 77 | 78,5 |
| **8** | 67,5 | 71 | 73 | 73,5 | 74,5 | 74,5 |
| **9** | 68,5 | 72,5 | 73,5 | 77 | 76 | 77 |
| **10** | 71,5 | 75 | 73 | 73,5 | 75 | 74,5 |
| **11** | 65,5 | 67 | 69 | 67,5 | 67,5 | 67,5 |
| **12** | 66 | 68,5 | 70,5 | 73,5 | 72,5 | 73,5 |
| **13** | 65 | 68,5 | 70,5 | 72,5 | 73 | 74 |
| **14** | 55,5 | 63 | 62,5 | 64,5 | 64,5 | 65 |
| **15** | 55,5 | 61,5 | 58,5 | 61,5 | 63 | 62 |
| **16** | 57,4 | 54,4 | 58,7 | 57,4 | 58,9 | 59,7 |
| **Ort** | 67,2 | 71,2 | 71,8 | 73,7 | 74,1 | **75,7** |

kümesinde ise 397 örnek vardır. EDD veri kümesi ile her kat için ayrı ayrı elde edilen sonuçlar ve bunların ortalaması Çizelge 4.2'de ve Çizelge 4.3'te verilmiştir.

EDD veri kümesindeki sınıf sayısı 27 olduğu için, ilk katmandaki ikili sınıflandırma sonucunda 27 öznitelikten oluşan öznitelik vektörleri elde edildi. Bu öznitelik vektörlerinin birleştirilmesi sonucunda da ikinci katmanda kullanılan yeni bir öznitelik kümesi oluşturuldu. İkinci katmanda çeşitli sınıflandırma yöntemleri kullanıldı. R ortamında RO ve DVM yöntemleri ile; Weka ortamında ise DVM, DVM ile Çok Sınıflı Sınıflandırma(ÇS), LR ile Çok Sınıflı Sınıflandırma(ÇS) ve sınıflandırıcı topluluğu yöntemleri ile deneyler yapıldı. Sınıflandırıcı topluluğu yönteminde DVM ve LR yöntemleri birlikte kullanıldı.

Çizelge 4.2'deki değerler sadece SPMap yöntemi ile çıkarılan özniteliklere dayanan sonuçlardır. En yüksek doğruluk başarı oranı Weka ortamında LR ve DVM yöntemlerinin birlikte kullanıldığı sınıflandırıcı topluluğu ile elde edildi. 16 kat çapraz doğrulama sonucunda ortalama **%75,7** doğruluk başarı oranına ulaşıldı. Weka ortamında yapılan uygulamalar, R ortamında yapılan uygulamalarla karşılaştırıldığında ise genel olarak weka ortamında yapılan uygulamaların daha iyi sonuçlar ürettiği görülür.

Çizelge 4.3'teki değerler ise SPMap ile çıkarılan özniteliklerle amino asit özelliklerine dayanarak kompozisyon, frekans ve dağılım tanımlayıcıları ile çıkarılan özniteliklerin birleştirilmesiyle elde edilen sonuçlardır. Her kat için ayrı ayrı sonuçların gösterildiği çizelgede hesaplanan ortalama doğruluk başarı oranları incelendiğinde en yüksek doğruluk başarı oranına **%74,9** ile yine Weka ortamında sınıflandırıcı topluluğu yönteminin kullanılması ile ulaşılmıştır. SPMap'e ve amino asit özelliklerine dayanarak ayrı ayrı çıkarılan özniteliklerin birleştirilmesiyle yapılan uygulamalar ile sadece SPMap'e dayanarak çıkarılan özniteliklerle yapılan uygulamalar karşılaştırıldığında, sadece SPMap'e dayanan yöntemin az farkla dahi olsa daha düşük sonuçlar ürettiği görülmektedir. Gelecek çalışmalarda SPMap'e ve amino asitlerin fizyokimyasal özelliklerine dayanarak ayrı ayrı çıkarılan öznitelikler birleştirildikten sonra çeşitli öznitelik seçme yöntemlerinin denenmesi amaçlanmakta ve başarının artırılması hedeflenmektedir.

**Çizelge 4.3** : EDD veri kümesi kullanılarak SPMap yöntemiyle ve amino asitlerin fizyokimyasal özeliklerine dayanan tanımlayıcılarla çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%)

| KAT | RO (R) | DVM (R) | DVM (Weka) | ÇS (LR) | ÇS (DVM) | ST |
|---|---|---|---|---|---|---|
| **1** | 62 | 72 | 73,5 | 72 | 74 | 74,5 |
| **2** | 67,5 | 76,5 | 82 | 77,5 | 81,5 | 82 |
| **3** | 65,5 | 74 | 78 | 79 | 85 | 84,5 |
| **4** | 69,5 | 75,5 | 78 | 77,5 | 79 | 78,5 |
| **5** | 75 | 83,5 | 83,5 | 82 | 83 | 83,5 |
| **6** | 60,5 | 71,5 | 71,5 | 73,5 | 75 | 76,5 |
| **7** | 67 | 76 | 77,5 | 75 | 79 | 80,5 |
| **8** | 59,5 | 68,5 | 70,5 | 75,5 | 76 | 75,5 |
| **9** | 58,5 | 67,5 | 68 | 73 | 70,5 | 72,5 |
| **10** | 67 | 70 | 75,5 | 77,5 | 78,5 | 78 |
| **11** | 66 | 69 | 70,5 | 70 | 73 | 72,5 |
| **12** | 64 | 70,5 | 75 | 75 | 74,5 | 75 |
| **13** | 62 | 71,5 | 71,5 | 73,5 | 73 | 75 |
| **14** | 59,5 | 59 | 65 | 64,5 | 65 | 65,5 |
| **15** | 54 | 59,5 | 60,5 | 61,5 | 59,5 | 61 |
| **16** | 55,6 | 56,6 | 59,6 | 63,7 | 61,9 | 63,6 |
| **Ort** | 63,3 | 70,1 | 72,5 | 73,2 | 74,3 | **74,9** |

Daha fazla katlanma ile daha gerçekçi sonuçlar elde etmek için oluşturulan F95 veri kümesinde 95 katlanma sınıfına ait toplam 6364 örnek vardır. İkinci katmanda test adımını gerçekleştirmek için veri kümesi, örnek sayısı en az olan sınıfa göre eğitim ve test kümesi olarak ayrıldı. En az örneği olan sınıfın örnek sayısı 24'tür. Bu sayı 20'den büyük olduğu için veri kümesi 20 eşit parçaya ayrıldı ve tüm sınıfların aynı numaradaki parçaları birleştirildi. Ardından deneyler 20 kat-çapraz doğrulama ile yapıldı. Bu işlemler uygulandığında ilk 19 kat için eğitim kümesinde 6092, test kümesinde 272 örnek vardır. 20. kat için eğitim kümesinde 5168, test kümesinde ise 1196 örnek vardır.

F95 veri kümesindeki sınıf sayısı 95'tir.Dolayısıyla ilk katmandaki ikili sınıflandırma sonucunda 95 öznitelikten oluşan öznitelik vektörleri elde edildi. Bu öznitelik vektörlerinin birleştirilmesi sonucunda da ikinci katmanda kullanılan yeni bir öznitelik kümesi oluşturuldu. İkinci katmanda uygulanan sınıflandırma yöntemleriyle F95 veri kümesine ait sonuçlar, Çizelge 4.4'de her kat için ayrı ayrı gösterilmiştir. Hesaplanan

**Çizelge 4.4** : F95 veri kümesi kullanılrak SPMap yöntemiyle çıkarılan öznitelikler ile her kat için elde edilen sınıflandırma sonuçları (Doğruluk Başarı Oranı%)

| KAT | RO (R) | DVM (R) | DVM (Weka) | ÇS (DVM) | ÇS (LR) | ST |
|---|---|---|---|---|---|---|
| **1** | 66,9 | 62,1 | 69,48 | 70,58 | 76,1 | 76,47 |
| **2** | 71,3 | 68 | 77,94 | 78,3 | 81,61 | 82,35 |
| **3** | 70,9 | 63,6 | 76,1 | 72,42 | 78,67 | 79,41 |
| **4** | 77,9 | 66,5 | 73,89 | 72,79 | 77,94 | 78,3 |
| **5** | 76,4 | 70,2 | 81,61 | 77,94 | 83,08 | 81,98 |
| **6** | 76,4 | 69,8 | 76,83 | 76,47 | 82,35 | 79,7 |
| **7** | 71,6 | 65 | 76,47 | 79,41 | 81,25 | 83,08 |
| **8** | 76,1 | 69,8 | 77,94 | 74,26 | 79,04 | 79,04 |
| **9** | 69,8 | 64,3 | 74,26 | 70,95 | 75,36 | 75,52 |
| **10** | 69,1 | 65,8 | 71,69 | 69,11 | 72,42 | 71,32 |
| **11** | 69,8 | 64,7 | 77,94 | 75,73 | 75 | 77,5 |
| **12** | 72,4 | 61,3 | 72,05 | 69,48 | 78,3 | 77,5 |
| **13** | 74,2 | 66,5 | 74,63 | 70,95 | 76,83 | 75,73 |
| **14** | 71,6 | 63,2 | 73,52 | 69,85 | 76,1 | 76,43 |
| **15** | 66,5 | 63,2 | 73,52 | 70,95 | 74,26 | 76,4 |
| **16** | 70,9 | 59,9 | 73,89 | 69,48 | 75 | 72,79 |
| **17** | 66,5 | 57,7 | 66,17 | 63,97 | 71,32 | 69,48 |
| **18** | 61,7 | 53,6 | 60,29 | 59,55 | 67,27 | 65,42 |
| **19** | 59,5 | 55,8 | 65,44 | 63,6 | 69,48 | 70,58 |
| **20** | 50,8 | 48,9 | 45,48 | 39,63 | 51,67 | 45,99 |
| **Ort** | 69,5 | 62,9 | 71,9 | 69,8 | **75,2** | 74,7 |

ortalama doğruluk başarı oranları incelendiğinde F95 veri kümesi için **%75,2** en yüksek ortalama başarı oranına LR yöntemi ile çoklu sınıflandırma yapıldığında ulaşıldığı görülmektedir.

Daha fazla katlanma ile daha gerçekçi sonuçlar elde etmek için oluşturulan diğer bir veri kümesi de F194 veri kümesidir. Bu veri kümesinde 194 katlanma sınıfına ait toplam 8026 örnek vardır. En az örneği olan sınıfın örnek sayısı 11'dir. Bu yüzden ikinci katmanda test adımını gerçekleştirmek için veri kümesi, örnek sayısı en az olan sınıfa göre eğitim ve test kümesi 11 parçaya ayrıldı ve tüm sınıfların aynı numaradaki parçaları birleştirildi. Ardından deneyler 11 kat-çapraz doğrulama ile yapıldı. Yani her defasında 11 parçadan biri hariç bırakılarak model eğitildi, hariç bırakılan parça ile de geliştirilen model test edildi.

194 katlanma sınıfına ait F194 veri kümesinde ilk katmandaki ikili sınıflandırma sonucu oluşturulan 194 adet öznitelik vektörü birleştirildi ve ikinci katmanda kullanılmak üzere 194b boyutlu yeni bir öznitelik kümesi elde edildi. İkinci katmanda çeşitli sınıflandırma yöntemleri denendi

Deneylerde elde edilen sonuçlar incelendiğinde en son kat için doğruluk başarı oranlarının genel olarak düşük olduğu görülmektedir. En son kat test edilirken eğitim kümesinin örnek sayısı diğer katlara göre daha az, test kümesindeki örnek sayısı ise diğer katlara göre daha fazla olması başarının bu kat için düşmesine sebep olmaktadır. Bunun önüne geçebilmek için gelecek çalışmalarda en son kata ait test kümesindeki fazla verilerin, diğer katlara ait test kümelerine dengeli bir şekilde dağıtılması planlanmaktadır.

**Çizelge 4.5** : EDD ve F95 veri kümeleri ile elde edilen ortalama doğruluk başarı oranları (%)

| | **RO(R)** | **DVM(R)** | **DVM(Weka)** | **ÇS(SVM)** | **ÇS(LR)** | **ST** |
|---|---|---|---|---|---|---|
| **EDD (SPMap)** | 67,1 | 71,2 | 71,8 | 74,1 | 73,7 | **75,7** |
| **EDD (SPMap+AA)** | 63,3 | 70,1 | 72,5 | 74,2 | 73,2 | **74,9** |
| **F95** | 69,5 | 62,9 | 71,9 | 69,8 | **75,2** | 74,7 |

Yapılan deneylerin ortalama doğruluk başarı oranları Çizelge 4.5'te listelenmiştir. EDD veri kümesi ile en yüksek doğruluk oranına sadece SPMap metoduyla çıkarılan özniteliklere sınıflandırıcı toplulukları yönteminin uygulanması sonucunda ulaşılmıştır. F95 veri kümesi ile yapılan deneylerde en yüksek doğruluk oranı, LR ile çoklu sınıflandırma yönteminin uygulanmasıyla elde edilmiştir.

# 5. SONUÇLAR

Peptit bağları ile birbirine bağlanan amino asitlerden oluşan proteinlerin yapısı, üç boyutlu uzayda katlanarak şekil alır ve fonksiyonlarını da katlandıktan sonra yerine getirirler. Dolayısıyla bu katlanma bilgileri proteinlerin 3b yapılarının ve fonksiyonlarının belirlenmesinde önemli bir aşamadır. Bu yüzden bioinformatik alanındaki önemli çalışma konularından biridir.

Protein katlanması tanıma probleminin sınıflandırma problemi olarak ele alınabilmesi için farklı uzunluktaki protein sekanslarından makine öğrenme yöntemlerinin uygulanabileceği öznitelikler çıkarılmalıdır. Bu doğrultuda hazırlanan çalışma ile protein sekansları, amino asitlerin fizyokimyasal özelliklerii ve sekans bilgilerini temsil eden sayısal değerlere dönüştürülmüştür. Bu aşamada öznitelik çıkarırken iki yöntem kullanılmıştır. İlk olarak çeşitli tanımlayıcılarla amino asitlerin fizyokimyasal özelliklerinden; sonrasında ise SPMap ile proteinlerin alt sekanslarındaki bilgilerden faydalanılmıştır. SPMap yöntemi ilk olarak bu çalışmada protein katlanması tanıma için kullanılmıştır.

Bu çalışmada, protein sekansları sayısal değerlere dönüştürüldükten sonra iki katmanlı bir yöntem geliştirilmiştir. İlk katmanda amino asitlerin fizyokimyasal özelliklerinden ve SPMap'ten gelen öznitelikler birleştirilmiş ve her bir sınıfa ait öznitelik kümesi ile ikili sınıflandırma modelleri oluşturulmuştur. İkili sıınıflandırma sonucunda gelen tahmin değerleri öznitelik vektörü olarak düşünülmüş ve birleştirilerek ikinci katmanda kullanılmak üzere yeni bir öznitelik kümesi oluşturulmuştur. İkinci katmanda hem R hem de Weka ortamında çeşitli sınıflandırma yöntemleri uygulanarak geliştirilen modelin performansı değerlendirilmiştir. Model performansı değerlendirme ölçütü olarak doğruluk oranı, duyarlılık, kesinlik ve F-ölçütü kullanılmıştır.

Makine öğrenmesi yöntemlerinin kullanılabileceği sınırlı sayıda katlanma sınıfı içeren 4 adet veri kümesi ile deneyler yapılmıştır. Bu veri kümeleri DD (performans ölçümü veri kümesi), EDD, F95 ve F194'tür. Yapılan deneyler sonucunda en yüksek doğruluk

başarı oranları DD veri kümesi için %71,7; EDD veri kümesi için %75,7; F95 veri kümesi için %75,2'dir.

Gelecek çalışmalarda başarı oranının artırılması hedeflenmektedir. Bunun için öncelikli yapılacak adım öznitelik seçme yöntemlerinin denenmesidir. Ayrıca veri kümesi, k-kat çapraz doğrulama ile eğitim kümesi ve test kümesi olarak ayrılırken, en son katta test kümesindeki veri sayısı diğer katlara oranla daha çok olduğundan bu kat için başarı oranının düştüğü görülmüştür. Bunun önüne geçebilmek için bu kata ait test kümesindeki veri diğer katlara dengeli bir şekilde dağıtılacaktır. Bu şekilde başarı oranının artması beklenmektedir. Başarı oranını artırabilmek için yapılması planlanan uygulamalardan birisi de SCOP veri tabanının hiyerarşisinden faydalanmak olacaktır. SCOP veri tabanında katlanma katmanından önce sınıf katmanı bulunmaktadır. Bu yapı göz önünde bulundurularak öncelikle sınıf katmanında bir etiketleme yapılacaktır.

## KAYNAKLAR


[1] Protein Sentezi, http://aisbiology.pbworks.com/w/page/49012795/DNA,%20RNA,%20and%20Proteins, erişildi: 06.12.2015.

[2] Protein Yapısı, https://en.wikipedia.org/wiki/Protein_structure, erişildi: 17.01.2016.

[3] SCOP, http://www.slideshare.net/pebourne/bioinformatics-in-the-bourne-lab, erişildi: 12.11.2015.

[4] **Lin, C.**, **Zou, Y.**, **Qin, J.**, **Liu, X.**, **Jiang, Y.**, **Ke, C. ve Zou, Q.** (2013). Hierarchical classification of protein folds using a novel ensemble classifier, *PloS one*, *8*(2), e56499.

[5] **Sarac, O.S.**, **Gürsoy-Yüzügüllü, Ö.**, **Cetin-Atalay, R. ve Atalay, V.** (2008). Subsequence-based feature map for protein function classification, *Computational biology and chemistry*, *32*(2), 122–130.

[6] RO, https://www.researchgate.net/, erişildi: 26.02.2015.

[7] **Tramontano, A. ve Lesk, A.M.** (2006). Protein structure prediction, *John Wiley and Sons, Inc, Weinheim*.

[8] **Dobson, C.M.** (2003). Protein folding and misfolding, *Nature*, *426*(6968), 884–890.

[9] **Yang, J.Y. ve Chen, X.** (2011). Improving taxonomy-based protein fold recognition by using global and local features, *Proteins: Structure, Function, and Bioinformatics*, *79*(7), 2053–2064.

[10] **Ding, C.H.Q. ve Dubchak, I.** (2001). Multi-class Protein Fold Recognition Using Support Vector Machines and Neural Networks, *Bioinformatics*, *17*, 349–358.

[11] **Schulz, G.E. ve Schirmer, R.H.** (2013). *Principles of protein structure*, Springer Science & Business Media.

[12] **Dubchak, I.**, **Muchnik, I.**, **Holbrook, S.R. ve Kim, S.H.** (1995). Prediction of protein folding class using global description of amino acid sequence, *Proceedings of the National Academy of Sciences*, *92*(19), 8700–8704.

[13] **Nanni, L.** (2006). A novel ensemble of classifiers for protein fold recognition, *Neurocomputing*, *69*(16), 2434–2437.

[14] **Shen, H. ve Chou, K.** (2006). Ensemble classifier for protein fold pattern recognition, *Bioinformatics*, *22*(14), 1717–1722.

[15] **Wood, C. ve Drenth, J.** (1999). Principles of Protein X-ray Crystallography, *International Journal of Biochemistry and Cell Biology*, *31*(9), 979.

[16] **Cavanagh, J.**, **Fairbrother, W.J.**, **Palmer III, A.G. ve Skelton, N.J.** (1995). *Protein NMR spectroscopy: principles and practice*, Academic Press.

[17] **Rost, B.** (1996). [31] PHD: Predicting one-dimensional protein structure by profile-based neural networks, *Methods in enzymology*, *266*, 525–539.

[18] **Ouali, M. ve King, R.D.** (2000). Cascaded multiple classifiers for secondary structure prediction, *Protein Science*, *9*(06), 1162–1176.

[19] **Salamov, A.A. ve Solovyev, V.V.** (1995). Prediction of protein secondary structure by combining nearest-neighbor algorithms and multiple sequence alignments, *Journal of molecular biology*, *247*(1), 11–15.

[20] **Pandey, G.**, **Kumar, V. ve Steinbach, M.** (2006). Computational approaches for protein function prediction: A survey, *Twin Cities: Department of Computer Science and Engineering, University of Minnesota*.

[21] **Moult, J.**, **Pedersen, J.T.**, **Judson, R. ve Fidelis, K.** (1995). A large-scale experiment to assess protein structure prediction methods, *Proteins: Structure, Function, and Bioinformatics*, *23*(3).

[22] CASP, `http://predictioncenter.org/`, erişildi: 16.12.2015.

[23] PDB, `http://www.rcsb.org/pdb/home/home.do`, erişildi: 05.11.2015.

[24] CATH, `http://www.cathdb.info/`, erişildi: 14.11.2015.

[25] SCOP, `http://scop.mrc-lmb.cam.ac.uk/scop/`, erişildi: 05.11.2015.

[26] **Dong, Q.**, **Zhou, S. ve Guan, J.** (2009). A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation, *Bioinformatics*, *25*(20), 2655–2662.

[27] **Cortes, C. ve Vapnik, V.** (1995). Support-vector networks, *Machine learning*, *20*(3), 273–297.

[28] **Gunn, S.R.** *ve diğerleri* (1998). Support vector machines for classification and regression, *ISIS technical report*, *14*.

[29] **Boser, B.E.**, **Guyon, I.M. ve Vapnik, V.N.** (1992). A training algorithm for optimal margin classifiers, *Proceedings of the fifth annual workshop on Computational learning theory*, ACM, s.144–152.

[30] **Alpaydin, E.** (2014). *Introduction to machine learning*, MIT press.

[31] **Quinlan, J.R.** (1986). Induction of decision trees, *Machine learning*, *1*(1), 81–106.

[32] **Breiman, L.**, **Friedman, J.**, **Stone, C.J. ve Olshen, R.A.** (1984). *Classification and regression trees*, CRC press.

[33] **Breiman, L.** (2001). Random forests, *Machine learning*, *45*(1), 5–32.

[34] **Breiman, L.** (2002). Manual on setting up, using, and understanding random forests v3. 1, *Statistics Department University of California Berkeley, CA, USA*.

[35] **Liaw, A. ve Wiener, M.** (2002). Classification and regression by randomForest, *R news*, *2*(3), 18–22.

[36] **Hall, M.**, **Frank, E.**, **Holmes, G.**, **Pfahringer, B.**, **Reutemann, P. ve Witten, I.H.** (2009). The WEKA data mining software: an update, *ACM SIGKDD explorations newsletter*, *11*(1), 10–18.

**EKLER**

**EK A.1 :** EDD veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri
**EK A.2 :** F95 veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri

## EK A.1

**Çizelge A.1** : EDD veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri

| | **Kesinlik** | **Duyarlılık** | **F-Ölçütü** |
|---|---|---|---|
| **1** | 0,979188 | 0,954563 | 0,951813 |
| **2** | 0,958375 | 0,9375 | 0,933375 |
| **3** | 0,570688 | 0,801125 | 0,659063 |
| **4** | 0,94375 | 0,81425 | 0,866625 |
| **5** | 0,722188 | 0,845813 | 0,750063 |
| **6** | 0,895813 | 0,8765 | 0,875875 |
| **7** | 0,759875 | 0,882875 | 0,811125 |
| **8** | 0,979188 | 0,966938 | 0,965 |
| **9** | 0,884563 | 0,75425 | 0,781063 |
| **10** | 0,932313 | 0,9115 | 0,905938 |
| **11** | 0,775438 | 0,505188 | 0,580438 |
| **12** | 0,701 | 0,625 | 0,644688 |
| **13** | 1 | 0,983313 | 0,990375 |
| **14** | 0,979188 | 0,970813 | 0,9685 |
| **15** | 0,945875 | 0,910688 | 0,912563 |
| **16** | 0,71525 | 0,833313 | 0,764563 |
| **17** | 0,661 | 0,64225 | 0,644063 |
| **18** | 0,866688 | 0,84975 | 0,836375 |
| **19** | 0,849875 | 0,695313 | 0,74925 |
| **20** | 0,831438 | 0,844 | 0,832188 |
| **21** | 0,871063 | 0,924 | 0,886 |
| **22** | 0,7765 | 0,710938 | 0,728375 |
| **23** | 0,863125 | 0,753125 | 0,769875 |
| **24** | 0,8125 | 0,8125 | 0,8125 |
| **25** | 0,761438 | 0,61375 | 0,665 |
| **26** | 0,578438 | 0,348625 | 0,419188 |
| **27** | 0,947 | 0,979188 | 0,955375 |

**Çizelge A.2** : F95 veri kümesine ait her sınıf için Kesinlik, Duyarlılık ve F-ölçütü değerleri

| Sınıf | Kesinlik | Duyarlılık | F-Ölçütü |
|---|---|---|---|
| **1** | 1 | 0,89165 | 0,92505 |
| **2** | 0,595 | 0,55835 | 0,54175 |
| **3** | 0,875 | 0,88 | 0,87085 |
| **4** | 0,47175 | 0,6266 | 0,5215 |
| **5** | 0,87505 | 0,7643 | 0,78 |
| **6** | 0,71255 | 0,625 | 0,64195 |
| **7** | 0,67195 | 0,7126 | 0,68035 |
| **8** | 0,71255 | 0,70455 | 0,67555 |
| **9** | 0,71665 | 0,77275 | 0,72945 |
| **10** | 1 | 0,9594 | 0,9658 |
| **11** | 0,69165 | 0,715 | 0,68145 |
| **12** | 0,88335 | 0,85475 | 0,8354 |
| **13** | 0,875 | 0,88 | 0,87085 |
| **14** | 0,7973 | 0,6861 | 0,7117 |
| **15** | 0,9 | 0,86665 | 0,875 |
| **16** | 0,55835 | 0,53215 | 0,4951 |
| **17** | 0,63915 | 0,555 | 0,5676 |
| **18** | 0,7111 | 0,72665 | 0,70835 |
| **19** | 0,7078 | 0,8385 | 0,75625 |
| **20** | 0,76675 | 0,775 | 0,75005 |
| **21** | 1 | 0,83055 | 0,87675 |
| **22** | 0,975 | 0,95715 | 0,94585 |
| **23** | 0,694 | 0,49995 | 0,545 |
| **24** | 0,90005 | 0,86665 | 0,8534 |
| **25** | 0,78755 | 0,7434 | 0,741 |
| **26** | 0,8 | 0,775 | 0,78335 |
| **27** | 0,51295 | 0,4316 | 0,4574 |
| **28** | 0,86675 | 0,92235 | 0,882 |
| **29** | 0,611 | 0,4411 | 0,48975 |
| **30** | 1 | 0,90355 | 0,9364 |
| **31** | 0,92505 | 0,87 | 0,8653 |
| **32** | 0,9 | 0,95 | 0,9167 |
| **33** | 1 | 0,8643 | 0,9056 |
| **34** | 0,8267 | 0,90485 | 0,8473 |
| **35** | 0,925 | 0,9028 | 0,8886 |
| **36** | 0,6625 | 0,71665 | 0,67315 |
| **37** | 0,74165 | 0,7139 | 0,72085 |
| **38** | 0,7 | 0,75525 | 0,70875 |
| **39** | 0,65 | 0,65 | 0,65 |
| **40** | 0,7409 | 0,6809 | 0,67865 |
| **41** | 0,84165 | 0,89445 | 0,8554 |
| **42** | 0,85 | 0,85385 | 0,82385 |
| **43** | 0,4938 | 0,6893 | 0,56355 |

| Sınıf | Kesinlik | Duyarlılık | F-Ölçütü |
|---|---|---|---|
| **44** | 0,5519 | 0,6417 | 0,581 |
| **45** | 0,6961 | 0,6511 | 0,64105 |
| **46** | 0,875 | 0,8643 | 0,85555 |
| **47** | 0,7 | 0,6611 | 0,6682 |
| **48** | 0,7651 | 0,7053 | 0,71965 |
| **49** | 0,7583 | 0,56 | 0,61985 |
| **50** | 0,8917 | 0,9861 | 0,922 |
| **51** | 0,71665 | 0,8111 | 0,7266 |
| **52** | 0,6875 | 0,76365 | 0,7034 |
| **53** | 0,7879 | 0,81715 | 0,79505 |
| **54** | 0,86825 | 0,815 | 0,8154 |
| **55** | 1 | 0,9643 | 0,9722 |
| **56** | 0,8792 | 0,7893 | 0,80225 |
| **57** | 0,7893 | 0,64045 | 0,6871 |
| **58** | 0,625 | 0,6059 | 0,57215 |
| **59** | 0,875 | 0,89375 | 0,88 |
| **60** | 0,76895 | 0,69125 | 0,70435 |
| **61** | 0,535 | 0,48955 | 0,48315 |
| **62** | 0,525 | 0,5577 | 0,5134 |
| **63** | 0,69235 | 0,55895 | 0,58485 |
| **64** | 0,5375 | 0,55455 | 0,5117 |
| **65** | 0,81675 | 0,6893 | 0,71345 |
| **66** | 0,825 | 0,64165 | 0,7001 |
| **67** | 0,88335 | 0,83335 | 0,85005 |
| **68** | 0,5586 | 0,3881 | 0,44345 |
| **69** | 0,9437 | 0,78975 | 0,8447 |
| **70** | 0,85 | 0,81665 | 0,825 |
| **71** | 0,91665 | 0,9231 | 0,9066 |
| **72** | 0,98335 | 0,9423 | 0,95255 |
| **73** | 0,7 | 0,65715 | 0,6625 |
| **74** | 0,4594 | 0,465 | 0,44985 |
| **75** | 0,92 | 0,90555 | 0,9087 |
| **76** | 0,85 | 0,80625 | 0,8111 |
| **77** | 0,8 | 0,7559 | 0,76055 |
| **78** | 0,54165 | 0,60555 | 0,5584 |
| **79** | 0,91875 | 0,7949 | 0,83 |
| **80** | 0,9417 | 0,885 | 0,89625 |
| **81** | 0,9 | 0,8818 | 0,8889 |
| **82** | 0,9792 | 0,8844 | 0,9127 |
| **83** | 0,775 | 0,76365 | 0,7548 |
| **84** | 0,95 | 0,95 | 0,95 |
| **85** | 0,52045 | 0,5263 | 0,5167 |
| **86** | 0,95 | 0,9611 | 0,9349 |
| **87** | 0,9 | 0,9147 | 0,88945 |

| Sınıf | Kesinlik | Duyarlılık | F-Ölçütü |
|---|---|---|---|
| **88** | 0,8909 | 0,8909 | 0,8909 |
| **89** | 0,57415 | 0,725 | 0,60525 |
| **90** | 0,93815 | 0,875 | 0,88885 |
| **91** | 0,8 | 0,7722 | 0,78075 |
| **92** | 0,7894 | 0,85 | 0,78725 |
| **93** | 0,66675 | 0,55265 | 0,56135 |
| **94** | 0,5158 | 0,505 | 0,4792 |
| **95** | 0,65 | 0,65625 | 0,6278 |

## ÖZGEÇMİŞ


**Ad-Soyad** : **Ruşen HALEPMOLLASI**

**Doğum Tarihi ve Yeri** : **13 Ocak 1984 Şanlıurfa**

**E-Posta** : **halepmollasi@itu.edu.tr**


**ÖĞRENİM DURUMU:**

- **Lisans : Mersin Üniversitesi (2006)**
- **Y. Lisans: : İstanbul Teknik Üniversitesi (2016)**

**TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR**

- **Halepmollası R.**, Bilgen İ., Saraç Ö.S. 2016: Subsequence Profile Map for Protein Fold Recognition *24th IEEE Conference on Signal Processing and Communications Applications (SİU 2016)* , May 16-19, 2016 Zonguldak, Turkey.