YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

FİZYOLOJİK SÜREÇLERDE MODEL TABANLI YENİ ÖĞRENME YAKLAŞIMLARI

Sistem ve Kontrol Yüksek Mühendisi, Uğur AYAN

FBE Elektrik Anabilim Dalı Kontrol ve Otomasyon Programında Hazırlanan

DOKTORA TEZİ

Tez Savunma Tarihi : 22 Temmuz 2010
Tez Danışmanı : Prof. Dr. Galip CANSEVER (Yıldız Teknik Üniversitesi)
Jüri Üyeleri : Prof. Dr. Murat TAYLI (T.C. Maltepe Üniversitesi)
Prof. Dr. Bekir KARLIK (Mevlana Üniversitesi)
Doç. Dr. Abdullah BAL (Yıldız Teknik Üniversitesi)
Yrd. Doç. Dr. Kayhan GÜLEZ (Yıldız Teknik Üniversitesi)

İSTANBUL, 2010

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ

$\boldsymbol{x}$	Giriş vektörü
$\boldsymbol{y}$	Etiket/çıkış vektörü
b	Orijinden kayıklık oranı
d_m	Çoklu çekirdek ağırlık parametresi
$\phi(.)$	Öznitelik uzayına dönüşüm işlevi
Σ	Kovaryans matrisi
$L(.)$	Lagrangian işlevi
L_1	Birincil Lagrangian
L_2	İkincil Lagrangian
α	Lagrangian çarpanı
α_l	Öğrenilmemiş yöneylerin Lagrangian katsayıları
α_k	Sınır yöneylerin Lagrangian katsayıları
Δb	KKT durumlarını sağlayacak dizgisel sapma
μ	Doğrusal ayrılamayan çekirdekler için ikinci Lagrangian çarpanı
Δp	Sarsım artırımı
$\boldsymbol{\mathcal{K}}$	Çekirdek modeli
C	Düzenleştirme sabiti
$\boldsymbol{\mathcal{H}}$	Hessian matrisi/ Hilbert Uzayı
ξ	Ödünleşme parametresi
ε	Duyarsız bölgenin genişliği
$\varsigma_m(.)$	Geçitleme modeli
σ	Gauss işlevinin genişliği
∇	Gradyen matrisi
β	Ölçeklenmiş ağırlıklar
$\boldsymbol{w}$	Gerçel ağırlık vektörü
$\frac{\partial y}{\partial x}$	y'nin x'e göre kısmi türevi
$P(.,.)$	Pearson korelasyon katsayısı

KISALTMA LİSTESİ

ADMET	Emilim, Dağılım, Metobolizma, Köken, Zehirlilik (Absorption, Distribution, Metabolism, Extraction, Toxicity)
BLOSUM	Blok Yerine Koyma Matrisi (Block Substitution Matrix)
BofW	Kelimeler Torbası Çakirdeği (Bag-of-words Kernel)
DDAG	Yönlü Çevrimsiz Çizge Kararları (Decision Directed Acyclic Graph)
DSSP	The Dictionary of Protein Secondary Structure (İkincil Yapı Ansiklopedisi)
DVM	Destek Vektör Makinaları (Support Vector Machines)
FN	Yanlış Negatif(False Negative)
FP	Yanlış Pozitif (False Positive)
GA	Genetik Algoritma
GRNN	Genel Bağlanım Sinir Ağı (General Regression Neural Network)
IEEE	Elektrik-Elektronik Müh. Enstitüsü (Institute of Electrical and Electronics Engineers)
IKL	Artımlı Çekirdek Öğrenimi (Incremental Kernel Learning)
IMKL	Artımlı Çoklu Çekirdek Öğrenimi(Incremental Multiple Kernel Learning)
kNN	k-En Yakın Komşuluk (k-Nearest Neighbour)
LSO	En Küçük Kareler En İyilemesi(Least Square Optimization)
MAE	Ortalama Mutlak Hata (Mean Absolute Error)
MCMKL	Çok Sınıflı Çoklu Çekirdek Öğrenimi (Multi Class Multiple Kernel Learning)
MKL	Çoklu Çekirdek Öğrenimi (Multiple Kernel Learning)
ML	Makine Öğrenmesi (Machine Learning)
MLP	Çok Katmanlı Algılayıcı (Multi Layer Perceptron)
mRMR	Enküçük Artıklık Enfazla İlişki (Minimum Redundancy Maximum Relevance)
PAM	Nokta Kabul Mutasyon (Point Accepted Mutation)
PCA	Temel Bileşenşer Analizi (Principal Component Analysis)
PCR	Temel Bileşen Tabanlı Regresyon (Principal Component Regression)
PDB	Protein Veri Bankası (Protein Data Bank)
PNN	Olasılıksal Yapay Sinir Ağları (Probabilistic Neural Network)
QCQP	İkinci Dereceden Kısıtlamalı İkinci Dereceden Programlama (Quadratic Constraint Quadratic Programming)
QP	İkinci Dereceden Programlama (Quadratic Programming)
QSAR	Niceliksel Yapı-Özellik İlişki Analizi (Quantitative Structure-Property Relationship Analysis)

RBF	Radyal Tabanlı İşlev (Radial Basis Function)
RGB	Red Green Blue
RMSE	Ortalama Karesel Hatanın Karekökü (Root-Mean Squared Error)
RO3	3 Kuralları (Rule of 3)
RO5	5 Kuralları (Rule of 5)
SL	Öğreticili Öğrenme (Supervised Learning)
SMM	Saklı Markov Modelleri (Hidden Markov Models)
SDP	Yarı Tanımlı Programlama(Semi Definite Programming)
SMO	Ardışık En Azlama En İyilemesi (Sequential Minimal Optimization)
SOM	Kendi Kendini Düzenleyen Haritalar (Self Orginizing Map)
sSVM	Yarı Destek Vektör Makinaları (Semi-Support Vector Machines)
TFIDF	Terim Frekans Ters Belge Frekansı (Term Frequency Inverse Document Frequency)
TN	Doğru Negatif (True Negative)
TP	Doğru Pozitif (True Pozitive)
TÜBİTAK	Türkiye Bilimsel ve Teknik Araştırma Kurumu
USL	Öğreticisiz Öğrenme (Unsupervised Learning)
VTYS	Veri Tabanı Yönetim Sistemi
WDI	Dünya İlaç İndexi (World Drug Index)
YSA	Yapay Sinir Ağları

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

ÖNSÖZ

Yıldız Teknik Üniversitesi ailesine katıldığım ilk günden bu yana bana her anlamda desteğini esirgemeyen tez danışmanım Sayın Prof. Dr. Galip CANSEVER hocama dekanlık gibi ağır bir görevi olmasına rağmen tez çalışmam ve doktora dönemim boyunca bana gösterdiği sabır ve ilgiden dolayı, bunun yanında yönlendirmeleri ve değerli bilgi ve deneyimlerini bana aktardığı için çok teşekkür ederim.

Amarika'nın Texas eyaletinde bulunan Houston Üniversitesi Bilgisayar Bilimleri Bölümündeki Eckhard Pfeiffer Professor ünvanlı Sayın Prof. Dr. Ioannis PAVLIDIS'e 2008-2009 yılında yöneticisi olduğu Computational Physiology Laboratuarında bana sağlamış olduğu araştırma olanaklarından ve araştırma bursundan dolayı, lab çalışanlarından Sayın Yrd. Doç. Dr. Dvijesh SHASTRI ve Dr. Peggy LINDER'e araştırmalarımdaki yardımlarından dolayı teşekkür ediyorum.

2002 yılından bu yana çalışmayı sürdürdüğüm İstanbul Kültür Üniversitesi Bilgisayar Mühendisliği Bölümü akademik kadrosu ve iş arkadaşlarıma severek çalıştığım bir ortam oluşturdukları için minnettarım.

Evliliğimizin ilk gününden itibaren desteğini hiçbir zaman eksik etmeyen, benim için her türlü fedakârlıkta bulunan, bana güvenen, akademik çalışmam sırasında kendisine ve çocuklarıma zamana ayıramama rağmen hep sabırla arkamda duran ve çalışmalarımın her aşamasında beni yüreklendiren sevgili eşim Elif Bozkurt AYAN'a çok teşekkür ederim. Buna ek olarak akademik hayatım boyunca fazla zaman ayıramama rağmen sevgi ve sabırla babalarını bekleyen oğullarım Utku Efe ve Ömer Eren'e sonsuz sevgilerimi sunuyorum.

Buraya gelmemde ilkokuldan bu zamana ellerinden geldiği kadar destek olan, beni ve kardeşimi gözetip büyüten çok değerli anne ve babama da tüm anne ve babalarımızın adına teşekkür ediyorum.

.

ÖZET

Bu tezde; fizyolojik veriler üzerinde önerdiğimiz yeni çekirdek tabanlı gözetimli öğrenme algoritmaları örnek olarak dizi çekirdekleri ve artımlı azalımlı çekirdek modelleri, çizge tabanlı yeni yarı gözetimli öğrenme modelleri, gürültülü ve aykırı verilerden kurtulmak için iki farklı öznitelik kalitesini ölçen methodu birleştirdiğimiz bir boyut indirgeme modeli, protein yapı tahmininde ve fonksiyon tanımada kullanılabilecek Saklı Markov Modeli baz alınarak oluşturulan yeni bir çekirdek modeli önerilmiş olup, önerilen modeller fizyolojik veriler üzerinde uygulanmıştır.

Bu çalışmada ilk olarak ilaç tasarımı ile ilgili veri kümeleri, yapılan çalışmalar, ve akademik yazımdaki bilgisayarlı ilaç tasarımı ile ilgili yöntemler hakkında bilgi verilmiş olup, hemen devamında kullanılan diğer protein veri bankaları, hastalık ve kanser veri kümeleri ile akademik yazımdaki bazı yüksek boyutlu veriler tanıtılmıştır. Sonraki bölümde ise kısaca öğrenme modelleri ele alınmıştır. Dördüncü bölümde, gözetimli öğrenme yöntemlerinden çekirdek tabanlı DVM modellerinden doğrusal ayrılabilen ve doğrusal olarak ayrılamayan öğrenme yöntemlerinin matematiksel alt yapısı tanıtılmış ve yeni önerilen modeller iki temel başlıkta; metin tabanlı çekirdek öğrenme ile artımlı çekirdek öğrenme algoritmaları, detayları ile verilmiş ve başarımları ikinci bölümde tanıtılan veri kümeleri üzerinde incelenmiştir. Özellikle bu bölümde önerilen yeni çekirdek modellerinin başarımlarını fizyolojik verilerin haricinde diğer makine öğrenme verileri üzerindede denenmiş ve başarımları incelenmiştir. Bir sonraki bölümde ise deneylerimizde karşılaştırmak için kullanılan üç faklı yarı gözetimli öğrenme modeli ve etkin öğrenme ile birleştirdiğimiz etkin yarı gözetimli öğrenme modeli detayları ile ele alınmıştır. Son olarak ele aldığımız yöntem ise akademik yazımda protein yapılarının sınıflandırılmasında sıkça kullanılan Saklı Markov Modeli ile dinamik programlama modellerinden yola çıkılarak proteinlerin SMM yapısı üzerinden eşleşen durumlara dayanan, protein dizilerinin metinsel yapısı yerine sonlu durumları kullanılarak geliştirilen ikili saklı Markov durumlarının skorlaması ile oluşturulan PHA-çekirdek modelinin matematiksel alt yapısı tanıtılmıştır.

Önerilen tüm algoritmalarda, çekirdek düzenlileştirme sabiti, ceza parametresi gibi farklı çekirdek paremetreleri ele alınarak başarımları karşılaştırmalı olarak verilmiştir. Algoritmaların, bilimsel yazındaki diğer birçok yöntemle eğitim ve test hataları açısından karşılaştırılmıştır.

Anahtar kelimeler : Gözetimli öğrenme, yarı gözetimli öğrenme, çekirdek makinaları, ilaç tasarımı, etkin yarı gözetimli öğrenme

ABSTRACT

In this thesis, new kernel based supervised learning algorithms such string kernels, incremental kernel learning moedels, new graph based semi-supervised learning models, a feature reduction model which is combination of different feature selaction methods measuring the quality of features in order to get rid of noisy and redundant data and Hidden Markov Model based new kernel machine method to predict the structure of protein and function classification is proposed.

Firstly, four different drug datasets, methodologies and the studies related with computer aided drug design are given, and immediately other protein databases, disease data, cancer datasets, some high-dimensional data on literature are presented. In the next section, the learning models is discussed briefly. In fourth chapter, matematical background of linearly separable and soft margin kernel based Support Vector Machines are introduced. Newly proposed kernel models are given in two areas as string kernels and incremental kernel learning algorithms. The performance of these methods on above datasets introduced in the second chapter is examined in details. Especially in this chapter, the performance of the proposed kernel models on other machine learning repository is also tested and analyzed. In the next chapter, three main semi-supervised learning model to compare in our experiments and new active semi-supervised learning model dealt with details. Finally, PHA-kernel, Pairwse Hidden Markov Models Alignment kernel which is based on Hidden Markov Models mostly used in protein classification and alignment scoring by dynamic programming is mathematically defined. We use finite state machines instead of using protein sequence structures in this model.

The accuracy of all proposed learning models are given by using different kernel regularization parameter, penalty parameter, slack variables and other kernel parameters Training and test errors of our algorithms are compared with other learning models in details.

Keywords: Supervised learning, semi-supervised learning, kernel machines, drug design, active semi-supervised learning

.

1. GİRİŞ

Öğrenme yaklaşımlarını incelediğimizde akademik yazımda çok farklı başlıklar altında model ve yöntemler bulmak mümkündür. Tüm problemlere uygulanabilecek ve en iyi çözümü oluşturacak bir model yada yaklaşım bulmak oldukça güçtür. Bundan dolayı da "*en iyi çözümü üreten yöntem, uygulandığı veri kümesine ve probleme göre farklılıklar göstermektedir*". Alpaydın (2004) kitabında öğrenme yaklaşımlarının üzerinde çalıştığı konuları; "*sınıflandırma, kümeleme, eğri uydurma, özellik çıkarımı ve ilişki belirleme*" gibi alt başlıklarda incelemektedir.

Öğrenme kavramı değişik şekillerde tanımlanmakla birlikte öğrenmeyi: "*Zaman içinde yeni bilgilerin keşfedilmesi yoluyla davranışların iyileştirilmesi süreci*" olarak tanımlayabiliriz. Makine öğrenmesi (ML) ise bu öğrenme işinin zaman içerisinde iyileştirilerek bilgisayarlar tarafından gerçekleştirilmesinin sağlanmasıdır. Burada zaman içerisinde iyileşme kavramına dikkat çekmek gerekmektedir. Bilgisayarın da insan gibi zaman içerisinde tecrübe kazanması istenmektedir. Diğer bir deyişle makine öğrenmesi "*bilgisayarın bir olay ile ilgili bilgileri ve tecrübeleri öğrenerek gelecekte oluşacak benzeri olaylar hakkında kararlar verebilmesi ve problemlere çözümler üretebilmesidir*" denilebilir. Aslında temelde bilgisayarların akıllanması şeklinde ifade edebiliriz. 20. yüzyılın en büyük bilim adamlarından nobel ödüllü bilim adamı Alexis Carrel "*Akıl bu dünyanın en büyük gücüdür*" demiştir. Eğer biz bu aklı makinelere öğretirsek işte o zaman makineler kendi kendini geliştirebilir ve gerçekten yapay us denilen kavram gerçekleşmiş olur.

Makine öğrenme, aslında yapay us'un bir alt alanı olarak değerlendirilmekle birlikte birçok alanda özellikle veri madenciliği, bilgisayar ağları, işaret işleme, sınıflandırma, kümeleme gibi belirli modeller yada teknikler altında bilgisayarın öğrenmesine izin veren yapılar yada algoritmalardır.

Tezde uygulama alanı olarak sürekli ve ayrık fizyolojik işaretler üzerinde yeni metod ve yaklaşımları uygulamaktayız. Özellikle protein ve DNA dizileri, ilaç tasarımı ve kanser verileri üzerinde geliştirmiş olduğumuz yeni algoritmaları karşılaştırmalı olarak göstereceğiz. Bununla birlikte geliştirilen metodları sadece yukarıdaki alanlarda değil her türlü sınıflandırma, kümeleme ve eğri uydurma problemlerinde uygulanabilecektir.

1.1 Fizyoloji ve Fizyolojik Süreç Nedir?

Fizyoloji (işlevbilim) ingilizcesi *physiology* olarak ifade edilen, doğa, köken ve origin anlamına gelen yunanca "*physis*" kelimesi ile nizam(doğal şeylerin kuralları) anlamına gelen "*-logia, -logos*" kelimelerinin birleşmesiyle oluşmuştur. Yaşayan organizmaların yani hayvan, bitki ve tüm canlılardaki hücre, doku ve organların işleyişini, mekanik, fiziksel ve biokimyasal fonksiyonlarını inceleyen bir bilim dalıdır. Bitki, hayvan ve insan fizyolojisi gibi alt branşlarının olmasının yanında fizyoloji bilimi canlıların iç mekanizmalarını ve işleyişlerini incelediği için bakteri, doku ve hücre fizyolojisi gibi daha birçok alt branşları bulunmaktadır. Canlı işleyişleri sadece mekanik ve fiziksel bir olaydan ibaret olmadığı için kimya ve biyoloji gibi diğer bilim dalları ile çok sıkı ilişki içindedir. Bu ilişkilerden dolayı biyokimya ve biyofizik gibi başka bilim dalları da oluşmuştur.

Hücrelerde meydana gelen kimyasal reaksiyonları, ilaçlara karşı tepki mekanizması, sinir sisteminin çalışma şekil ve prensipleri, uyarıların vücut tarafından nasıl alınıp, nasıl değerlendirildiğini, kasların çalışma mekanizmalarını, kanın damarlarda dolaşmasını, dokularda kanın kullanılma özelliklerini, kalbin ve beş duyumuzun nasıl çalıştığını, böbreklerin idrar meydana getirme kabiliyetini ve vücudun dış şartlarından nasıl etkilendiğini ve bunun gibi daha birçok vücut fonksiyonunun nasıl yapıldığını, hücresel, hatta moleküler seviyeye inerek araştırıp, gözler önüne sermeye çalışır. Ayrıca atmosferin üst tabakaları ve uzaydaki vücud fonksiyonlarını inceleyen "*hava fizyolojisi*", su altında meydana gelen değişiklikleri inceleyen "*sualtı fizyolojisi*" gibi daha ilginç fizyoloji dalları da kurulmuştur.

Fizyolojik süreçler üzerinde çalışılması zor fakat içerdiği gizli bilgiler ile araştırmaya açık alanları barındırmaktadır. Tezimizde bu süreçlerden vücudun hastalıklara karşı savunma mekanizmasında önemli bir rol oynayan ilaç adayı olabilecek moleküllerin çıkarımını bilgisayar ortamında (*bilgisayar destekli ilaç tasarımı*) inceleyeceğiz. Genelde zor modellenen veri tabanları olmasından dolayı var olan makine öğrenme yaklaşımlarına ek olarak yeni geliştirilen metodlara gereksinim duymaktadır. Bir başka uygulanacak fizyolojik süreç de protein yada amino asit dizilerinin işlevlerine göre sınıflandırılması olacaktır. En son olarak da kanserli verilerden (örüntü, işaret vb...) elde edilerek sürekli işaretlerin öğrenilmesi ve sınıflandırılması olacaktır. Tezin 2. bölümünde ise kullanılan verilerin detaylı tanıtımı yapılacaktır.

1.2 Makine Öğrenmesi

Öğrenme, son yüzyılda bilgisayar bilimlerinin yanısıra elektronik, kontrol, işaret işleme, bilişsel bilim, psikoloji, fizyoloji, biyofizik, fizikokimya gibi bilim dallarının da araştırma konusu olmaya başlamıştır. Birçok bilim adamı tarafından farklı şekillerde ifade edilmekle birlikte öğrenme kavramı, Turing'in (1950) "*Bir makinenin insan beynini simüle edebilecek şekilde programlanıp, öğrenme programı yüklenerek bir çocuk gibi konuşmasının mümkün olabileceği*" ni iddia ettiği yıllara kadar uzanır. O zamandan beri öğrenme kavramı bilgisayar bilimleri ve yapay us alanındaki etkisini giderek arttırmaktadır. Bilgi Teorisi (*Information Theory*) adı altında Jaynes 1957'de bir makale yayınlayarak ön ayak olmuştur. Rosenblatt'in (1958) makalesinde bir psikolog olarak perseptronları keşfi ve Widrow ve Hoff (1960) "*en küçük ortalamalı kareler*" olarak isimlendirilen bir denetçi öğrenme prosedürü geliştirilmesi süreç daha da hızlanmıştır. En yakın komşuluk kuramının geliştirilmesi (Cover ve Hart, 1967). Bu yıllarda yapay sinir ağları ile birlikte "*perceptron*" kavramı da gelişmeye başlamıştır. Uyarlanabilir doğrusal nöron ve bunun türevleri olan çoklu uyarlanabilir doğrusal nöron kavramları da 1960'lı yıllarda ortaya konmuştur (Parzen, 1962; Widrow, 1992). Bu süreçte "*backpropagation*" olarak bilinen öğrenme algoritmasının 1974'de Webos, 1982 ve 1985'de Parker ve 1986'da Rumelhart tarafından geliştirilmesi, "*genetik algoritmalar*"ın 1970 yılından itibaren John Holland tarafından alt yapısının tanımının ortaya konması, makine öğrenme ve akıllı öğrenmenin alt yapısını hazırlamada etkili olmuştur. Daview ve Bouldin(1979)'da kümeleme(gözetimsiz öğrenme) yöntemlerini detaylı el almışlardır. Kohonen (1990) kendi kendini örgütleyen haritalar metodunu ileri sürmüştür. Aslında makine öğrenmesi ile ilgili olarak gerçekleştirilen ilk başarılı pratik çalışma daha eskilere dayanmaktadır. Arthur Samuel(1959)'in geliştirdiği dama oyunu tecrübeli dama oyuncularını yenebilecek kadar öğrenebilen ilk programlardan birisidir.

Bu yıllarda geliştirilen bu yeni teknikler sonucunda öğrenmenin tanımı tekrar ortaya konmaya başlanmıştır. Simon'un (1983) "*benzer görevleri tekrarlarken, bir sistemdeki daha etkili olabilecek herhangi bir değişimi öğrenme olarak*" tanımlaması, Langley vd.(1995) "*deneyimlerden bilginin kazanılması ve performansı geliştirmek için sayısal metodların çalıştırılması ve otomatik olarak bilginin öğrenilmesi*", Holte (1993) "*öğrenme sistemleri tanımlanmış sınıflandırma kuralları ile bu kuralları doğrulayann test kümelerinden oluşur*" şekilinde ifade etmesi, Mitchell(1997) "*deneyimleri otomatik olarak geliştirebilen, veri madenciliği gibi uygulama alanlarından genel kuralları keşfeden ve kullanıcıların ilgilerini*

öğrenip filtreleyen ve otomatik olarak kendini geliştirebilen sistemlerdir" ve Hu vd.(2003) "*eğitim örneklerinden sınıflandırma kuralları veya yararlı örneklerin keşfedilmesi için sınıflandırma işlemlerinin kullanılması*" gibi birçok farklı tanım getirilmiştir. Ama genel olarak tüm tanımlara bakıldığında öğrenmeyi sistemin her türlü girişine göre otomatik olarak beklenen çıktıyı oluşturacak bir model olarak tanımlamak mümkündür.

1.3 Çalışmanın Amacı

Çalışmamız aslında temel bilimlerin yetersiz kaldığı ve hesaplamalı bilimlerin katkı sağlayacaği bilgisayar destekli ilaç tasarımı, protein yapılarının öngörüsü, işlevleri ile kanser verilerinin sınınflandırılması konuları üzerinde ve diğer birçok farklı sınıflandırma problemleri üzerinde başarıyla kullanılabilecek yeni öğrenme model ve metodları geliştirmek ve başarımlarını var olan diğer makine öğrenme algoritmaları ile başarıyla karşılaştırmak şeklinde özetlenebilir.

1.4 Tez İçeriği

Tezin 2. bölümünde deneylerde kullanılan fizyolojik verileri, protein yapılarını, ilaç tasarımı için gerekli format ve altyapı ile makine öğrenme grubları tarafından elde edilen kanser veri kümeleri tanıtılmaktadır. 3. bölümde makine öğrenme yaklaşımları hakkında kısa bir ön bilgi vereceğiz. 4. bölümde ise var olan çekirdek öğrenme yaklaşımlarının yanında kendimizin önerdiği metin tabanlı çekirdekler ile artımlı çekirdek modellerini sunacağız. 5. bölüm yarı gözetimli öğrenme hakkında bilgi verip önerdiğimiz modeli inceleyeceğiz ve 6. bölüm ise protein dizileri ve işaretlerin sayısallaştırılması üzerine durulacak ve yeni ortaya attığımız ve savunduğumuz SMM üzerinden yeni bir çekirdek modeli ortaya konulacaktır.

2. KULLANILAN VERİ KÜMELERİ VE HEDEFLENEN AMAÇLAR

Dünya'nın ve Türkiye'nin gelecek vizyonunda disiplinlerarası çalışmalar gelişmekte ve önem kazanmaktadır. Mühendislik ve hesaplamalı bilimlerin, tıp, moleküler biyoloji, genetik, kimya, fiziko-kimya, eczacılık, fizyolji, kemoenformatik(Brown, 1998; Gasteiger ve Engel, 2003), biyoenformatik (Wishart, 2005) gibi doğal bilimlerin yetersiz kaldığı matematiksel modelleme, yüksek boyutlu veriler üzerinde hesaplama, verilerden özellik çıkarımı gibi konularda yetkin olarak kullanılması bu çalışmamızın önemini bir kat daha artırmaktadır. Tezimizde bu kapsam içinde bilgisayar destekli ilaç tasarımı, protein işlev öngörüsü, enzim ve bakteri sınıflandırması, kanser ve hastalık oluşturacak durumların önceden yada otomatik tespiti gibi birçok fizyolojik konularda yeni modeller ve öğrenme algoritmaları ortaya konulmuş olup modellerin başarımları detaylarıyla verilmiştir.

Makine öğrenmesi metotlarının elde ettikleri birçok başarı olmasına rağmen genelde zor modellenebilen veri setleri için yeni özellik ve alt uzay seçimi gibi çalışmaların hala yoğun bir şekilde devam etmektedir. Geliştirilen metotların sadece fizyolojik veriler için değil her türlü yüksek boyutlu yada küçük boyutlu sınıflandırma, kümeleme ve eğri uydurma problemlerine uygulanabilecek yapıda olması tezimizin önemini artırmaktadır.

2.1 İlaç Tasarımı

İlaç oluşumunu, yapımı zor ve geliştirilmesi uzun ve pahalı bir süreç olmaktan çıkarmak için deneme-yanılma metodolojisi yerine içerdiği kimyasal bileşiklerin ilaç olabilme olasılıklarını hesaplama, kullanılacağı ortamdaki etkisinin katma değeri ve bu bileşiklerin ilaç adayı olup olmamasına göre sınıflandırma gelecek yüzyılların en önemli araştırma alanı olarak karşımıza çıkmaktadır.

İlaç olabilecek molekülün keşfi 2 ie 10 yıl arası süre almaktadır. Klinik öncesi ve deneme fazları ile birlikte bu süre 7-15 yıl arası değişmektedir (Amasyalı, 2007).

İlaç olabilecek molekülün literatürde bazı şartları sağlaması gerekmektedir. Bu şartlardan en önemlisi ADMET (**A**bsorbtion, **D**istribution, **M**etabolism, **E**xtraction, **T**oxicity) olarak söylenebilir. Tezin ilerleyen kısımlarında bu şartlar detaylı anlatılacaktır.

İlaç adayı moleküllerin istenen hedefe bağlanması gerçekleşse bile aşağıdaki durumların gözlenmesi durumunda molekül ilaç özelliğini yitirmekte yada etkisi çok düşük olmaktadır.

Bu özellikler :

- Toksik özellik göstermesi
- Yan etki (side-effect) göstermesi
- Proteine sıkı ve gevşek bağlanması
- Kan dolaşımına yayılamaması
- Hedef bölge haricinde başka bölgelere de etki etmesi
- Vücuttan erken atılma durumu
- Vücudun içine girdiğinde etkisini yitirmesi

şeklinde söylenebilir. Bu yukarıdaki özelliklere ek olarak WDI tarafından yönlendirici ek bilgi olarak ilaç yapımında Lipinski'nin (1997, 2000, 2003, 2004) "Beş Kuralı" (Rule of Five) :

- Molekül Ağırlığı (MWT) ≤ 500
- OH ve NH Toplamı ≤ 5 (H-bağ vericisi)
- ve N atomlar Toplamı (H-bağ alıcısı) ≤ 5
- Lipophilicity ClogP ≤ 5 (yada Moriguchi logP ≤ 4.15)

ve Veber vd. (2002)'nin ek iki kuralı :

- Polar yüzey alanı ≤ 140 A^0 (yada H-bağ verici ve alıcı toplamı ≤ 12)
- Dönebilen bağ sayısı ≤ 10

yönlendirici bilgiler olarak karşımıza çıkmaktadır. Çizelge 2.1'de Lipinski'nin Beş kuralını belirlerken kullandığı ilaç ve özellikleri yer almaktadır. Bu kural tablosundan yola çıkılarak yukarıdaki sonuçlara erişilmiş olup böylece bir proteinin ilaç olma olasılığı kurallara bağlanarak artırılmış olur. Toplam bu altı kural temel kurallar olarak ele alınmaktadır.

Bemis ve Murcko (1996, 1999), Ajay vd. (1998), Clark ve Pickett(2000), Wagener ve Geerestein (2000), Frimurer (2000), Brüstle vd.(2002), Geneste vd. (2002), Walters ve Murcko(2002), Gasteiger (2003), Weston vd.(2003), Bayram vd.(2004), Wang vd.(2004), Lipinski ve Veber'in bu temel tanımlayıcı öznitelikleri haricinde özellikle QSAR(Niceliksel Yapı-Aktivite İlişki Analizi) ve QSPR(Niceliksel Yapı-Özellik İlişki Analizi) özniteliklerini kullanarak makine öğrenme yöntemlerinden yapay sinir ağları,(Bybatov vd., 2003) hiyerarşik karar ağaçları, Bayesian tabanlı sınıflandırıcılar(Frank vd., 2003), genetik algoritmalar gibi modeller yardımı ile bilgisayarlı ilaç tasarımına katkıda bulunmuşlardır.

Çizelge 2.1 Beş kuralı tablosu (Lipinski vd., 1997)

İlaç	MLogP	OH+NH	MWT	N+O	Alert
Aciclovir	0.009	4	225.21	8	0
Alprazolam	4.74	0	308.77	4	0
Aspirin	1.70	1	180.16	4	0
Atenolol	0.92	4	266.34	5	0
Azithromycin	0.14	5	749.00	14	1
AZT	-4.38	2	267.25	9	0
Benzyl-penicillin	1.82	2	334.40	6	0
Caffeine	0.20	0	194.19	6	0
Candoxtril	3.03	2	515.65	8	0
Captopril	0.64	1	217.29	4	0
Carbomazepine	3.53	2	236.28	3	0
Chloramphenicol	1.23	3	323.14	7	0
Cimetidine	0.82	3	252.34	6	0
Clonidine	3.47	2	230.10	3	0
Cyclosporine	-0.32	5	1202.6	23	1
Desipramine	3.64	1	266.39	2	0
Dexamethasone	1.85	3	392.47	5	0
Diazepam	3.36	0	284.5	3	0
Diclofenac	3.99	2	296.15	3	0
Diltiazem-HCI	2.67	0	414.53	6	0
Doxorubicin	-1.33	7	543.53	12	1
Enalapril-meleate	1.64	2	376.46	7	0
Erythromycin	-0.14	5	733.95	14	0
Famotidine	-0.18	8	337.45	9	0
Felodipine	3.22	1	384.26	5	0
Fluorouracil	-0.63	2	130.08	4	0
Flurbiprofen	3.90	1	244.27	2	0
Furosemide	0.95	4	330.75	7	0
Glycine	-3.44	3	75.07	3	0
Hydrochlorth.	-1.08	4	297.74	7	0
Ibuprofen	3.23	1	206.29	2	0
Imipramine	3.88	0	280.42	2	0
Itraconazole	5.53	0	705.65	12	1
Ketaconazole	4.45	0	380.92	1	0
Ketoprofen	3.37	1	254.29	3	0
Labetalol-HCI	2.67	5	328.42	5	0
Lisinopril	1.11	5	405.50	8	0
Mannitol	-2.50	6	182.18	6	0
Methotrexate	1.60	7	454.45	13	1
Metoprolol-tartrate	1.65	2	267.37	4	0
Nadolol	0.97	4	309.41	5	0
Naloxone	1.53	2	327.38	5	0
Naproxen-sodium	2.76	1	230.27	3	0
Nortritylene-HCI	4.14	1	263.39	1	0
Omeprazole	-4.38	2	267.25	9	0
Phenytoin	2.20	2	451.49	10	0
Piroxicam	0.00	2	331.35	7	0
Prazosin	2.05	2	383.41	9	0
Propranolol-HCI	2.53	2	259.35	3	0
Quinidine	2.19	1	324.43	4	0
Ranitidine-HCI	0.66	2	314.41	7	0
Scopolamine	1.42	1	303.36	5	0
Tenidap	1.95	2	320.76	5	0
Terfenadine	4.94	2	471.69	3	0
Testesterone	3.70	1	288.43	2	0
Trovafloxac.	2.81	3	416.36	7	0
Valproic-acid	2.06	1	144.22	2	0
Vinblastine	2.96	3	811.00	13	1
Ziprasidone	3.71	1	412.95	5	0

Bu tabloda *Alert* sütunu beş kuralına göre, 0: bir problem olmadığını, 1: zayıf emilim yada zayıf nüfuz etme bilgisini göstermektedir.

2.1.1 İlaç Tasarım Aşamaları

Dünya ülkeleri ilaç yapım ve geliştirme alanında dört temel gruba ayrılır. Çok gelişmiş ilaç endüstrisine sahip olan ve yeni ilaç geliştiren ülkeler grubu (ABD, İngiltere, İsviçre, Japonya, Hollanda, Almanya, İsveç, Belçika ve Fransa), araştırma kapasitesi olan ülkeler (Arjantin, Avustralya, Avusturya, Çin, Danimarka, Hindistan, İrlanda, İsrail, İtalya, Kore, Macaristan, Meksika, Portekiz ve Slovenya) grubu, mamul ilaç ve etkin madde üretebilen ülke grubu (Türkiye bu gruptadır.) ve sadece mamul ilaç üreten ülkeler (Baykara vd., 2004). Önümüzdeki 20-30 sene için Tübitak'ın da yayınlamış olduğu rapor ve yayınlara göre ülkemizin gelecek vizyonu içinde önem verdiği ve vereceği alanlardan birisi olarak karşımıza çıkmaktadır.

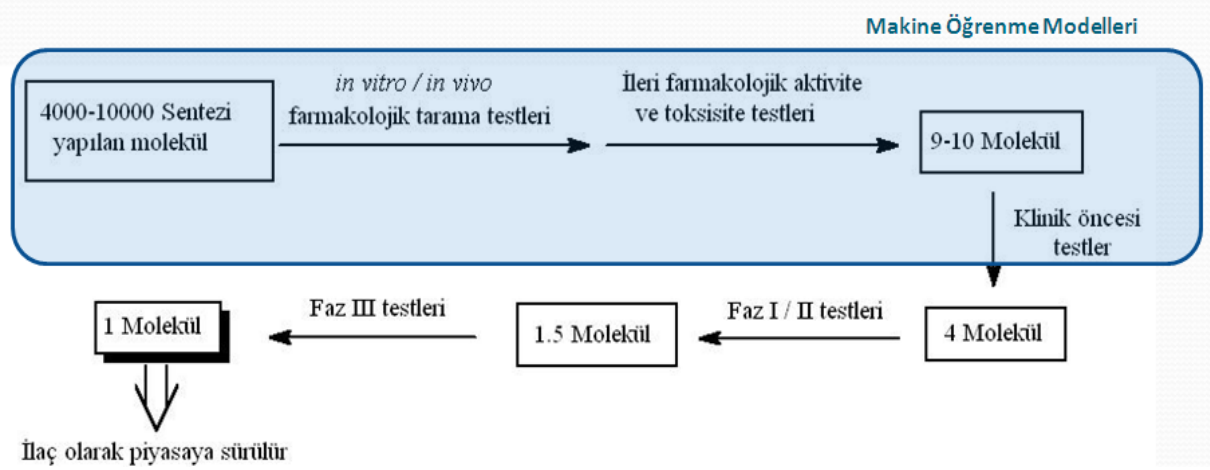
2.1.2 Yeni İlaç Geliştirme Süreci

İlaç geliştirme süreci birkaç ana bölümden oluşur [*]:

a) **Keşif (discovery) ve araştırma :** Geliştirilmesi düşünülen ilacın kullanılabileceği hastalık/hastalıklar/bulgular ile ilgili yeterli ve gerekli bilgi edinilme en temel şartlardan birisidir. Bu bilgiler uzun yıllar alan çalışmalar sonucunda elde edilmektedir. Bu çalışmalar sırasında hem ekonomik yükü hemde manevi yönü yani hastalığın etyolojisi, patogenezi, görülme sıklığı incelenir. Bu aşamada hedef endikasyon üzerinde biyolojik etkinliği ile hayvan ve insanlar üzerindeki yüksek emniyet ve güvenirlik profili oluşturması beklenmektedir. Hastalığın etyolojisi ve patogenezine yönelik araştırmalar, geliştirilmesi düşünülen ilaçla ilgili planların yapılmasına yardımcı olmaktadır. Hastalığın nedeninin elimine edilmesi, hastalık nedeni ile bozulan fizyolojik fonksiyonların yerine konulması, hastalığın olası komplikasyonlarının önlenmesi, hastalığın semptomlarının azaltılması gibi temel bilgiler ışığında geliştirilecek ilaçla ilgili önemli stratejik kararlar alınabilir. Böylece bu ilacın asıl amacının ne olacağına karar verilir. Rasyonel ilaç geliştirme; hastalık veya biyolojik prosesin temel mekanizmalarının anlaşılması, bilinen tedavi araçlarının farmakolojik etkilerinin anlaşılması ve rastgele tarama ve geniş biyolojik tarama işlemlerinin yapılması ile olur. Genel olarak kabul edilen 10.000 - 50.000 kimyasal

[*] http://www.pfizer.com.tr/

bileşikten ancak birinin hastaya ulaşabileceğidir. Keşif safhasında moleküler biyoloji, biyokimya, süper bilgisayar kullanımı ve "medicinal" kimya önemli rolü olan bilim dallarıdır. Gelişen tıp, biyoloji, biyoteknoloji, moleküler biyoloji ve nanoteknoloji gibi bilim dalları sayesinde uygulama ve test süreci içindeki hayvan kullanımı da giderek azalmakta. Diğer beşeri bilimlerin özellikle bilgisayar ve elektronik mühendisliği gibi gelişmesiyle de geliştirilecek yada ilaç olabilecek maddenin keşfi sırasındaki maliyet giderek azaltılmaktadır.



Şekil 2.1 İlaç tasarım aşamaları

b) Klinik Öncesi çalışmalar : Klinik öncesi faz çalışmalarının amacı potansiyel ilaç etkinlik ve güvenilirliğinin insanlarda denenmeden önce değerlendirilmesidir. Keşif döneminde seçilen kimyasal bileşikler "*klinik öncesi faz*"a uygulanmaya başlanır. Uluslararası Uyumlandırma Konferansı süreci dahilinde "*klinik öncesi ve klinik fazlar*" harmonize edilmeye çalışılır. Bu çalışmalar hayvanlarda ve laboratuvar modellerinde gerçekleştirilir. Ürünün kimyasal yapısı, üretim ile ilgili detaylı bilgileri, hayvanlar üzerinde edilen sonuçların detayları, klinik çalışmayı yürütecek araştırmacıların detaylı bilgileri ve klinik plan ve protokolleri tam olan ürünler bir sonraki aşama olan klinik aşamasına geçmeden önce birde ülkenin İlaç ve/veya Sağlık Bakanlığına (Amerika Birleşik Devletleri'ndeki İlaç ve Gıda Dairesi gibi) ***Yeni Araştırma İlacı*** başvurusu yapma zorunluluğu vadır. Yeni ilaç için akut, subakut ve kronik toksisite çalışmaları, genel ve spesifik organlara olan etkileri, reprodüktif toksisite testleri, mutajenisite ve karsinojenisite araştırmaları yapılır. Hayvanlarda yapılan bu tüm çalışmalar sırasında ***Uluslarası Hayvan Koruma ve Kullanma Komitesi*** kurallarına

uyulması, ***İyi Laboratuvar Uygulamaları*** kılavuzuna ve ***İyi Üretim Uygulamaları*** kurallarına uygun olması zorunludur.

c) Klinik çalışmalar : Tüm klinik çalışmalarda ***İyi Klinik Araştırmalar*** kurallarına uyulması zorunludur. Klinik çalışmalar dört fazda yapılır.

- **Faz I :** Bu fazın ana amacı "*güvenilirlik*"tir. Çalışmalar genellikle sağlıklı gönüllülerde yapılır. Denek sayısı 20-100 arasındadır. Bu çalışmalar ortalama 1-1.5 yılda tamamlanır. Düzenli olarak artan tek doz uygulamaları yapılır ve güvenirliği kontrol edilir.
- **Faz II** : Bu fazın ana amacı "*etkinlik ve güvenilirlik*"tir. Doz-cevap verilerinin toplanması, etkinliğinin hastalarda belirlenmesi ve yan etkilerinin araştırılması yapılmaktadır. Genellikle açık ve çok katı protokollerle uygulanır. Çalışmalar 100-300 hasta gönüllü üzerinde 2 yıl süreyle deneme yapılır.
- **Faz III** : Bu fazın ana amacı "*etkinliğin kanıtlanması ve yan etkilerin izlenmesi*"dir. Yeni ilaç adayının klinik etkinliğinin ve yan etkilerinin daha geniş bir hasta popülasyonu (genelde 1000-3000) üzerinde değerlendirilmesidir. Hedef hastalığı olan 1000-3000 hasta gönüllü 3-4 yıl süreyle bu çalışmalarda yer alır. İlaç adayının ilaç olarak kullanılabilmesi için "*onay*" alınması gerekir. Bunlar dışında ise her ülkenin yasal olarak sorumlu olan kuruluşuna gerekli başvuruyu yaparak onay alması gerekir. Ürünün onayı alındıktan sonra ilaç olarak kullanımına başlanabilir.
- **Faz IV** : Bu çalışmaların ana amacı "*uzun süreli güvenilirlik*" verilerinin toplanmasıdır. Ürün ilaç olarak kullanılmaya başlandıktan sonra yapılan klinik çalışmalar tümü olarak kabul edilir. İlacın tüm bilimsel aşamaları, etkinlik durumları, yan etkileri, edilebilirilaçla veya kullanıldığı hastalık ve hasta grubu ile ilgili ekonomik çalışmalar ve yaşam kalitesi çalışmaları bu fazda uygulanabilir.

Şekil 2.1'de de belirtildiği üzere, 4.000-10.000 sentezlenecek molekül bileşiğinden 9-10 moleküle indiregeme aşaması uzun ve maliyetli bir süreçtir. İlaç geliştirme süreci ilacın patent ömrü boyunca sürer. Bu süre 7 - 15 yıl kadar 500 – 900 milyon dolar civarındadır. "*Evergreening*" adı altında yapılan tüm çalışmalar aslında ilaç kullanıma girdikten sonra yeni endikasyonlarda kullanılması için yapılan çalışmalar olarak kabul edilir ve Faz III kuralları uyularak yapılır.

d) Onay (approval) : Tüm fazlar başarıyla tamamlandığında ilacın onay aşaması oluşmuş olur. İlacın onaylanması ülkenin bu işten sorumlu bakanlığın izniyle olur.

2.1.3 İlaç Veri Formatları

Kimyasal, biyolojik ve moleküler veri tabanlarının birçok farklı formatı bulunmaktadır. İlaç tasarımında ve protein yapılarında kullanılmakta olan temel bazı veri seti formatları aşağıda örneklenerek verilmiştir.

i. **PDB (P**rotein **D**ata **B**ank**)**

Protein veri bankası moleküllerin üç boyutlu yapılarını belirten; araştırmacılar, öğrenciler ve eğitimciler için hizmet veren küresel bir topluluğun (wwPDB) oluşturmuş olduğu bir format şeklidir. Bu topluluk bünyesinde RCSB PDB (ABD), PDBe(Avrupa), PDBj (Japonya) ve BMRB(ABD) gruplarını birleştirerek ortak bir veritabanı ve yapı oluşturmuşlardır. Atomik koordinatlar, atom ve kristal yapı faktörleri, NMR deneysel verileri, molekül adı, bulan kişinin bilgileri, birincil ve ikincil yapıların özellikleri, referans bilgileri gibi tüm gerekli olan bilgileri içerisinde barındırmaktadır. 2003 yılında Nature dergisinde Berman, H.M., Henrick, K. ve Nakamura, H. tarafından yayınlanarak dünya çapında kullanılan bir veri bankası olarak lansmanı edilmiştir. 2009 Eylül ayı itibariyle bünyesinde her türlü özelliği tanımlanmış 60.173 adet onaylanmış protein barındırmaktadır. (EK3'de bir pdb dosyası örneği gösterilmiştir.)

ii. **FASTA (**Pearson Formatı**)**

Biyoenformatik bilim dalında sıkça kullanılan metin tabanlı bir DNA dizisi yada aminoasit dizileri şeklinde temsil eden bir protein yapısı formatıdır. İlk satırda (">") büyüktür sembolü ile başlamakta olup tek bir satır açıklama yapılarak, her satırda en çok 80 karakterlik amino asit dizilişleri kullanılarak ifade edilen veri tipi şeklidir (Şekil 2.2).

```
>1FO4:A|PDBID|CHAIN|SEQUENCE
MTADELVFFVNGKKVVEKNADPETTLLAYLRRKLGLRGTKLGCGEGGCGACTV
```

Şekil 2.2 Bir fasta formatlı protein yapısı dizilişi

iii. **SDF (S**tructure **D**ata **F**ile**)**

Bir metin dosya formatı olduğundan herhangi bir kelime-işlemci programla açılıp işlenebilmektedir. Moleküler Dizayn Şirketi tarafından moleküllerin özelliklerini ifade etmek için geliştirilmiş ve standart haline dönüşmüştür. Molekülün adını, özelliklerini, atomlar arası bağların türlerini ve koordinatlarını, istenen diğer moleküler özellikleri içermektedir (Şekil 2.3).

```
1 benzene
2 ACD/Labs0812062058

3  6  6  0  0  0  0  0  0  0  0  1 V2000
4    1.9050   -0.7932    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
5    1.9050   -2.1232    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
6    0.7531   -0.1282    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
7    0.7531   -2.7882    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
8   -0.3987   -0.7932    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
9   -0.3987   -2.1232    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
10  2  1  1  0  0  0  0
11  3  1  2  0  0  0  0
12  4  2  2  0  0  0  0
13  5  3  1  0  0  0  0
14  6  4  1  0  0  0  0
15  6  5  2  0  0  0  0
 M END
 $$$$
```

Şekil 2.3 Örnek bir sdf formatlı dosya[*]

iv. **SMILES (S**implified **M**olecular **I**nput **L**ine **E**ntry **S**pecification**)**

"*Basitleştrilmiş Moleküler Giriş Hattı Madde Özellikleri*" deyiminin kısaltmasıdır. Molekül yapılarının kısa ASCII dizileri şeklinde tanımlanabilmesi için geliştirilmiş bir söz dizimidir. Atomlar elementlerin standart kısaltmaları ile temsil edilirler, örneğin altın için [Au]. B, C, N, O, P, S, F, Cl, Br ve I'dan oluşan "organik altgrup" için köşeli parantezler kullanılmayabilir, diger elementler için kullanılır. Eğer köşeli parantez kullanılmamışsa yeterli sayıda hidrojen atomunun bulunduğu varsayılır, örneğin su (H_2O) için SMILES kodu O'dur, etanol (C_2H_6O) için CCO'dur yada etylene (C_2H_4) için C=C'dir. Çift bağlı karbon dioksit O=C=O, üçlü bağlı hidrojen siyanür ise C#N olarak gösterilir. Dallar parantezlerle gösterilir, propionik asit için CCC(=O)O ve fluoroform için C(F)(F)F (veya FC(F)F olarak da) gibi. Sikloheksan C1CCCCC1 olarak gösterilir, burda "1" numara'nın molekülde aynı konumu işaretlediği anlaşılır, böylece 6 karbonlu bir halka oluşur. Burda kullanılan işaret numaradır (bu örnekte 1), C1 bileşimi değildir. Aromatik C, O, S ve N atomları küçük harfli olarak, 'c', 'o', 's', ve 'n' olarak gösterilir. Böylece benzen c1ccccc1'dir. İzomerlerde, çift bağların etrafındaki atomların konumlarını belirtmek için "/" ve "\" karakterleri kullanılır. Örneğin F/C=C/F trans-difluoroeten'i temsil eder, F'ler çift bağın zıt taraflarındadır. F'lerin çift bağın aynı tarafında olduğu cis-difluoroeten için F/C=C\F kullanılabilir.

[*] Wikipedia

Çizelge 2.2 Bazı smiles formatında moleküller

Ethanol	CCO
Acetic acid	CC(=O)O
Cyclohexane	C1CCCCC1
Pyridine	c1cnccc1
Trans-2-butene	C/C=C/C
L-alanine	N[C@@H](C)C(=O)O
Sodium chloride	[Na+].[Cl-]
Displacement reaction	C=CCBr>>C=CCI

v. SMARTS

Daylight Chemical Information Systems şirketi tarafından geliştirilmiş olan moleküler motifleri nitelendiren bir dil yapısıdır. Fonksiyonel gruplara (C:alkane, C&O:carbonyl, H:hydrogen atoms, N:amide, O:hydroxyl, alcohol, phenol, P:phosphoric bileşikler, S: thio grubu, X: halide), yapısal özelliklere, orbital konfigurasyona, rotasyona, bağlantılara, elektron, proton özelliklere ve amino asidlik durumlarına göre nitelendirilen bir yapıdır.

Çizelge 2.3 Bazı smart formatında yapılar

1-methyl-2-hydroxy benzene	[c;$([*Cl]),$([*H1])]1ccc(O)c(C)c1 or Cc1:c(O):c:c:[$(cCl),$([cH])]:c1
Imidazolium Nitrogen	[nX3r5+]:c:n
Carbonic Acid	[CX3](=[OX1])(O)O

vi. MDL / MOL (Molecular Design Limited)

Symyx Techonoloji şirketi tarafından bilgisayar destekli ilaç tasarımı amacıyla geliştirilmiş bir formattır. İlk olarak 1979'da etkileşimli moleküler modelleme (PRXBLD), 1982'de kimyasal reaksiyonlar (REACCS), 1985 de kimya veritabanı sistemi, 1986'da kimya çizim ve kelime işleme sistemi (ChemText), 1988'de 3-boyutlu yapı veritabanı (MACCS-3D), 1991'de kimyasal ve biyolojik istemci-sunucu sistemi (MDL ISIS) oluşturarak günümüzdeki Symyx MDL yazılımının alt yapıları hazırlanmış oldu.

```
# chemdig rzepa example_mol2  Name: hexane.mol2

@<TRIPOS>MOLECULE
hexane.mol2
  8  7  0  0  0
SMALL
NO_CHARGES


@<TRIPOS>ATOM
   1       C        -0.663    0.365    0.000  C.3
   2       C         0.663   -0.365    0.000  C.3
   3       H        -0.492    1.465    0.000  H
   4       H        -1.250    0.090    0.905  H
   5       H        -1.250    0.090   -0.905  H
   6       H         0.492   -1.465    0.000  H
   7       H         1.250   -0.090    0.905  H
   8       H         1.250   -0.090   -0.905  H
@<TRIPOS>BOND
   1    1    2    1
   2    1    3    1
   3    1    4    1
   4    1    5    1
   5    2    6    1
   6    2    7    1
   7    2    8    1
```

Şekil 2.4 Örnek bir mol formatlı dosya

vii. **Diğer Veri Tipi Formatları**

Yukarıdaki formatlardan farklı olarak Çizelge 2.3'de literatürde kullanılan fizyolojik, kimyasal, biyolojik ve protein yapı formatları verilmiştir.

Çizelge 2.4 Diğer veri tipi formatları

alc	Alchemy Format
csf	CAChe MolStruct CSF
cbin, cascii, ctab	CACTVS format
cdx	ChemDraw eXchange file
cer	MSI Cerius II format
c3d	Chem3D Format
chm	ChemDraw file
cif	Crystallographic Information File,
cmdf	CrystalMaker Data format
cml	Chemical Markup Language
cpa	Compass program of the Takahashi
bsd	Crossfire file
csm, csml	Chemical Style Markup Language

ctx	Gasteiger group CTX file format
cxf, cef	Chemical eXchange Format
emb, embl	EMBL Nucleotide Format
spc	SPC format for spectral and chromatographic data
inp, gam, gamin	GAMESS Input format
fch, fchk	Gaussian Checkpoint Format
cub	Gaussian Cube (Wavefunction) Forma
gau, gjc, gjf	Gaussian Input Format
gcg	Protein Sequence Format
gen	ToGenBank Format
istr,ist	IsoStar Library of Intermolecular Interactions
jdx, dx	JCAMP Spectroscopic Data Exchange Format
kin	Kinetic (Protein Structure) Images
mcm	MacMolecule File Format
mmd, mmod	MacroModel Molecular Mechanics

2.1.4 Kullanılan Veri Setleri

i. Cherkasov(2006) Datası

Üç ana özellikte veriyi içermektedir: anti-mikrobik, ilaç ve ilaç gibi davranan moleküller. Bu veriler toplam 2684 adet bileşikten ve 345 adet ayırt edici fiziksel, kimyasal ve 2-3 boyutlu QSAR özelliklerinden oluşmaktadır (Çizelge 2.5) (Ayan vd., 2010c).

Çizelge 2.5 Cherkaov veri seti tablosu

Kategori	Bileşikler	Kaynak
Antimicrobials	523	Journal of Antibiotics(2006)
General drugs	959	Merck Index, 13.4th Edition
Druglike	1202	Assinex Gold (2004)

Antimikrobik veriler Antibiotik Dergisinden (2006), genel ilaçlar Merck Indeks veritabanından ve ilaç gibi davrananlar ise Assinex Altın kolleksiyonundan elde edilmiştir. İlk veriler Pehlivanli vd. tarafından oluşturulmuş olup bu veriler tekrar düzenlenerek MOE ve Adriana Code yardımı ile yeni öznitelikler hesaplanmış ve eklenmiştir.

Bu veri seti oluşturulurken aşağıdaki temel kriterler göz önünde bulundurulmuştur:

- H-Bağ alıcı sayısı 1 ile 10 arası
- H-bağ sağlayıcı sayısı 1 ile 5 arası
- Molekül Ağırlığı (MWT) 200 – 500 Dalton
- Dönen bağ sayısı ≤ 12
- Hydrophopicity 1 ile 7 arası
- Polar yüzey alanı 140 A^2

ii. Murcia-Soler(2003) Veri Seti

Orjinal veri seti aslında Merck Indeksin 12. sürümü tarafından onaylı olan 430 adet bileşiği içermektedir. Buna ek olarak 250 adet de herhengi bir ilaç özellik içermeyen bileşik eklenmiştir. 680 adet toplam bileşiğin Merck İndeks ve gerekli RO5 ve RO3 şartlarını sağlayan 641 bileşiği ve bunlardan çıkartılan 162 özellik bizim deneylerimizde kullanılarak test edilmiştir (Ayan vd., 2010c).

Çizelge 2.6 Murcia-Soler veri seti tablosu

	Terapatik Kategoriler	
#	***Drugs***	***416***
1	Analgesic (ağrı kesici)	66
2	Antibacterial (anti-bakteriyel)	94
3	Antidepressant (anti-depresan)	33
4	Antidiabetic (antidiabetik)	9
5	Antifungal (anti-mantar)	24
6	Antihistaminic (antihistaminik)	30
7	antihyperlipoproteinemic	18
8	Antihypertensive (tansiyon önleyici)	58
9	antiinflammatory	24
10	Diuretic (Üre artırıcı)	24
11	Sedative (Yatıştırıcı)	36
	İlaç Olmayan	***225***
	Toplam	***641***

QSAR özniteliklerinden bazılarını aşağıdaki Çizelge 2.7'de göstermekteyiz.

Çizelge 2.7 2D QSAR açıklayıcı nitelikleri

Kategoriler	**Nitelikler**
Fiziksel Özellikler	Weight, FCharge, logS, apol, bpol, mr, TPSA, density, vdw_area, vdw_vol, logP(o/w), SlogP, SMR
Alt Parçalara Bölünmüş Yüzey Alanları	SlogP_VSA0, SlogP_VSA1, SlogP_VSA2, SlogP_VSA3, SlogP_VSA4, SlogP_VSA5, SlogP_VSA6, SlogP_VSA7, SlogP_VSA8, SlogP_VSA9, SMR_VSA0, SMR_VSA1, SMR_VSA2, SMR_VSA3, SMR_VSA4, SMR_VSA5, SMR_VSA6, SMR_VSA7
Atom ve Bağ Sayıları	a_aro, a_count, a_IC, a_ICM, a_nH, b_1rotN, b_1rotR, b_ar, b_count, b_double, b_rotN, b_rotR, b_single, b_triple, chiral, chiral_u, reactive, rings, a_heavy, a_nBr, a_nC, a_nCl, a_nF, a_nI, a_nN, a_nO, a_nP, a_nS, b_heavy, VAdjEq, VAdjMa, lip_acc, lip_don, lip_druglike, lip_violation, opr_brigid, opr_leadlike, opr_nring, opr_nrot, opr_violation
Kier & Hall Bağlantıları ve Kappa Şekil İndexi	chi0v, chi0v_C, chi1v, chi1v_C, chi0, chi0_C, chi1, chi1_C, zagreb, Kier1, Kier2, Kier3, KierA1, KierA2, KierA3, KierFlex
Bitişiklik ve Uzaklık Matrisi	balabanJ, diameter, petitjean, petitjeanSC, radius, VDistEq, VDistMa, weinerPath, weinerPol, BCUT_PEOE_0, BCUT_PEOE_1, BCUT_PEOE_2, BCUT_PEOE_3, BCUT_SLOGP_0, BCUT_SLOGP_1, BCUT_SLOGP_2, BCUT_SLOGP_3, BCUT_SMR_0, BCUT_SMR_1, BCUT_SMR_2, BCUT_SMR_3, GCUT_PEOE_0, GCUT_PEOE_1, GCUT_PEOE_2, GCUT_PEOE_3, GCUT_SLOGP_0, GCUT_SLOGP_1, GCUT_SLOGP_2, GCUT_SLOGP_3, GCUT_SMR_0, GCUT_SMR_1, GCUT_SMR_2, GCUT_SMR_3
Pharmacophore Feature	a_acc, a_acid, a_base, a_don, a_hyd, vsa_acc, vsa_acid, vsa_base, vsa_don, vsa_hyd, vsa_other, vsa_pol
Kısmi Görevler	PEOE_PC+, PEOE_PC-, PEOE_RPC+, PEOE_RPC-, PEOE_VSA+0, PEOE_VSA+1, PEOE_VSA+2, PEOE_VSA+3, PEOE_VSA+4, PEOE_VSA+5, PEOE_VSA+6, PEOE_VSA-0, PEOE_VSA-1, PEOE_VSA-2, PEOE_VSA-3, PEOE_VSA-4, PEOE_VSA-5, PEOE_VSA-6, PEOE_VSA_FHYD, PEOE_VSA_FNEG, PEOE_VSA_FPNEG, PEOE_VSA_FPOL, PEOE_VSA_FPOS, PEOE_VSA_FPPOS, PEOE_VSA_HYD, PEOE_VSA_NEG, PEOE_VSA_PNEG, PEOE_VSA_POL, PEOE_VSA_POS, PEOE_VSA_PPOS, Q_VSA_HYD, Q_VSA_POS

iii. İlaç Veri Bankası

Bu veri bankasında kimyasal, farmakolojik veriler, ilaç yapıları ve kapsamlı ilaç hedef proteinleri ile QSAR özelliklerini elde etmek mükündür. Mayıs 2009 tarihi itibariyle 1382 adet FDA tarafından onaylanmış küçük yapıda ilaçlar, 123 adet FDA tarafından onaylı biotech ilaç yapıları, 172 adet yasalara aykırı ilaç, 71 adet nutraceutical yapı ve 3.200 adet de deneysel ilaç verisi bulunmaktadır. Biz bu verilerin RO5 ve RO3'e uygun olan yapıları arasından 4.000 adet ilacını kullanmaktayız. Bu ilaç verilerine ait yüzey ve şekil özelliği, 2 ve 3 boyutlu özdevimli ilinti tanımlayıcı özellikleri, 3 boyutlu özellik ağırlık RDF ve yüzey özdevimli ilinti özelliği olmak üzere toplam 350-400 adet öznitelik ele alınmıştır.

2.2 Protein Öngörüsü

"*Protein*" sözcüğünün kaynağı, Yunanca'nın "*birincil öneme sahip*" anlamını taşıyan πρώτα (prota) sözcüğüdür. Bu isim, proteinleri 1838'de ilk tanımlayan Jöns Jakob Berzelius tarafından verilmiştir. 1926'da James B. Sumner'in üreaz enziminin bir protein olduğunu göstermesine kadar, proteinlerin canlılar için ne derece önemli olduğu tam anlaşılmamıştır. Yapısı çözülen ilk proteinler arasında insülin ve miyoglobin bulunur ki, insülin için Sir Frederick Sanger 1958'de, miyoglobin için de Max Perutz ve Sir John Cowdery Kendrew 1962'de Nobel Kimya Ödülü kazanmıştır. Her iki protein de kırınım analizi ile üç boyutlu yapıları çözümlenen ilk proteinlerdendir[*].

Bilgisayar, elektronik ve kontrol bilim dallarındaki teknolojik gelişmeler moleküler biyoloji, genetik, eczacılık, biyoenformatik, kimya, kimyasal-biyoloji, fiziko-kimya, tıp vb... bilim dallarında hesaplamalı bilim dallarının kullanılmasına yol açmıştır. Böylece bu alanlarda hızlı gelişmeler gözlenmektedir. Özellikle aşağıda listelenen konularda hesaplamalı bilimlerin katkısı azımsanamayacak kadar çoktur. Çok kısa sürede veri üreten bilgileri işlemek için mutlaka hesaplamalı bilimlere ve mühendisliğe ihtiyaç duyulmaktadır[**].

i. Akıllı sistemlerle otomize edilmiş veri analizi ve iletimi
ii. İlaç keşfi ve ilaç geliştirme süreçleri
iii. Protein yapısı, biyolojik ve moleküler fonksiyonun belirlenmesi
iv. Küçük moleküllerin (potansiyel terapötik maddeler, aktif peptidler, ribozimler vs.)

[*] Nature 181(4610):662-6

[**] www.biyomuhendislik.com (02.11.2009)

ligandlarıyla(bağ yapabilecek bileşikleri ile) etkileşiminin araştırılması

v. DNA, RNA, protein birincil, ikincil sıra ve dizilimi araştırmaları

vi. Karmaşık genetik fonksiyon ya da regülasyon faaliyetlerinin tanımlanması

vii. Hastalık etkileşimlerinin hesaplanması ve hastalık oluşturacak protein yapılarınn önceden tahmini

viii. Herhangi bir biyolojik fonksiyonu arttıran ya da engelleyen küçük moleküllerin tasarlanması

ix. Genetik faktörlerin hastalık yatkınlığına etkilerini ortaya çıkarmak

x. Enzim sınıflandırılması, etkilerinin modellenmesi ve matematiksel model oluşturulması

xi. Heterojen biyolojik veritabanlarının entegrasyonu

xii. Etkileşimde bulunan gen ürünleri için bilgi ağları oluşturulması

xiii. Kimyasal reaksiyonlardan hücrelerarası iletişime kadar pek çok biyolojik faaliyet sürecinin simülasyonu

xiv. Büyük çaplı biyolojik deneylerden (GENOM projeleri gibi) çıkan sonuçların analizi

Yukarıda tanımlanan çalışmalardan özelikle protein fonksiyon öngörüsü, ikincil yapı tahmini, enzim sınıflandırılması ve motif çıkarımı gibi konulara odaklanacağız. Geliştirmiş olduğumuz modellerin bu problemler üzerindeki başarımlarını inceleyeceğiz.

2.2.1 Protein Veri Bankaları

i. Protein Data Bank (PDB Org.)

Protein Bilgi Bankası (PDB) proteinler ve nükleik asitler gibi büyük biyolojik moleküllerin, 3 boyutlu yapıları hakkında bilgileri içeren bir arşivdir. 1971 yılında Brookhaven Ulusal Laboratuarında kurulmuş olup bünyesinde 7 adet yapı bulundurur. 1998 yılında Yapısal Biyoinformatik Araştırma İşbirliği (Research Collaboratory for Structural Bioinformatics, RCSB) yönetim hakkını devralmıştır. Bu proje takımı Rutgers Üniversitesi (New Jersey, ABD) Kimya ve Kimyasal Biyoloji Bölümü ile San Diego California Üniversitesi Süper Bilgisayar Merkezi ve Skaggs Eczacılık Okulu ve Eczacılık Bilimi Bölümü öğretim üyelerinden oluşmaktadır. 2003 yılından itibaren wwPDB ile her proteinin makromoleküler yapısı tüm dünyaya sunulmaya başlanmıştır. wwPDB konsorsyumu bünyesine 2006 yılında PDB Japonya (PDBj), PDB Avrupa (PDBe) ve BMRB (Biyolojik Manyetik Rezonans

Bankası) Amerika'nın da katılımıyla çok hızlı bir şekilde genişlemiştir. PDB formatlı veriler genel olarak o molükülün yapısı hakkında bilginni yanında kimin bulduğu, hangi tarihte yayınlandığı, yapısal, kimyasal, fiziksel, 2 ve 3 boyutlu özellikleri, amino asit dizilimi, ikincil yapıları ve daha birçok bilgiyi barındırmaktadır. Bir molekülün şekli o molekülün işlevini anlamak için önemli bir kaynaktır. Kasım 2009 itibariyle bünyesinde 61.086 adet onaylı molekül yapısı içermektedir.

ii.Gen Bankası (Gene Ontology)

Gen Ontoloji 1998 yılında 3 model organizma (meyve sineği, fare ve bira ile maya) üzerinde araştırma yapan akademisyenler tarafından kurulmuştur. Günümüz itibariyle birçok model organizmanın veritabanlarını içermektedir. Ontoloji veritabanı üç etki alanı üzerinde çalışmalara devam etmektedir; hücresel bileşen (bir hücre yada ekstra selüler çevresel parçalar), moleküler fonksiyon (bağlayıcı ya da katalizör olarak bir genin elemental faaliyetleri) ve biyolojik süreç (moleküler olayların tanımlanmış başlangıç ve bitiş süreçleri, hücreler, dokular veya organizmaların işleyişi).

iii. SCOP

SCOP, proteinlerin yapısal sınıflandırılması üzerine kurulu bir veritabanıdır. Protein yapısal bölgelerinin amino asit dizileri ve üç boyutlu yapılarına dayanarak protein yapısal bölgelerinin (domain) elle yapılmış bir sınıflandırmasıdır Haziran 2009 itibariyle 1.75 nolu sürümü yayındadır. 38.221 adet PDB verisi, 110.800 adet fonksiyonu belirlenmiştir

Çizelge 2.8 SCOP 1. seviye sınıfları

Sınıf	Katlama (Fold)	ID
Alpha proteinleri	284	46456
Beta proteinleri	174	48724
Alpha ve Beta Proteinleri (a/b)	147	51349
Alpha ve Beta Proteinleri (a+b)	376	53931
Çoklu-fonksiyon Proteinler	66	56572
Membran ve Hücre Duvarı Proteinleri	58	56835
Küçük Proteinler	90	56992
Burgulu Proteinler	7	57942
Düşük Çözünürlüklü Proteinler	26	58117
Peptitler	121	58231
Suni Elde Edilmiş Proteinler	44	58788

İlk kez 1995'te yayımlanmış. 4 temel hiyerarşik sınıflandırma seviyesi vardır. Sınıf (bölgenin genel yapısal mimarisi), katlama (düzgün ikincil yapıların benzerlik durumları), süperaile(yapıal ve işlevsel benzerliklerden evrimsel ilişki çıkarımı), aile (dizgi benzerliği olanlar).

iv.PRINTS (Protein Tanısal Parmakizi Veritabanı)

PRINTS protein motifi parmakizi şeklinde yorumlanabilir. Proteinlerin bilinen işlevlerinin ortaya çıkarmış olduğu motiflerdir. Aslında bir işlev için aktif olarak çalıştığı düşünülen kısımlardır. Düzenli ifadeler şeklinde ifade edilebilir. Genelde proteinlerin tamamı aktif değildir. Eğer protein bir işlev yürütüyorsa örnek olarak bağlama veya taşıyıcı görevi gibi. Benzer görevli protein molekülleri alınır ve profilleri oluşturulur. Bu profillerden ortaya çıkan motifler yada diğer br tabirle parmak izleri kullanılarak oluşturulan bir bilgi kümesidir. 2000 civarında elde edilmiş düzenli ifade vardır.

v. PROSITE (Protein Aileleri için İşlevsel Bölgeler)

İsviçre BioEnformatik Enstitüsü tarafından oluşturulmakta olan bu veritabanı protein ailelerinin işlevsel bölgeleri üzerine çalışmalar yapmaktadır. Bünyesinde 2000'in üzerinde motif barındırmaktadır.

Örnek bir Prosite motifi P-x-[STA]-x-[LIV]-[IVT]-x-[GS]-G-Y-S-[QL]-G şeklindedir. Detaylı bir örnek Ek 4'de verilmiştir.

vi. PROFEAT (Protein Özellik Sunucusu)

PRINTS, PROSITE, PRODOM gibi proteinlerin aktif parçaları üzerinde düzenli ifadeler aranarak oluşturulan bir veritabanıdır. Yapısal, fizikokimyasal özellikleri, dipeptid amino asit bilgilerini, normalize edilmiş Moreau-Broto otokorelasyonu, Moran ve Geary otokorelasyonu, dağılım, birleşim ve bağlantıları, dizgi sırası, pseudo amino asit birleşim bilgilerini içeren bir veritabanıdır.

2.2.2 Proteinlerin Sınıflandırılması

Proteinler, yapıtaşları amino asit olan zincir halinde birbirlerine bağlanmasıyla oluşan polimerlerdir. Her proteinin kendisine has özelliklerinin olmasını sağlayan özel amino asit dizilimleri vardır. Proteinlerin işlevlerinin çoğu, kendisini oluşturan amino asitlerin özelliklerinin tayin edilmesiyle anlaşılabilir. Bunun yanında ortamın ortaya koyduğu özelliklerde proteinlerin işlevlerini etkilemektedir. Yüksek sıcaklık, basınç ve asidik bazlık

gibi özellikler amino asitlerin bağlarının yapısını değiştirmekte böylece işlevlerini de etkilemektedir. İnsandan virüse, proteinlerin oluşumunda kullanılan 22 çeşit amino asit vardır (Çizelge 2.9).

Her proteindeki amino asit dizisinin sırası bir gen tarafından tanımlanır ve genetik kod ile kodlanmıştır. Genetik kod 22 "standart" amino asit tanımlasa da proteinlerdeki amino asitler translasyon sonrası değişimle kimyasal olarak değişikliğe uğrar. Bu değişimler ya proteinin işlev görmeye başlamasından önce gerçekleşir ya da kontrol mekanizmalarının parçası olarak, proteinin işlevini değiştirmek için gerçekleşir. Bu zincirde bir amino asitin karboksil grubunun bir diğerinin amino grubuna bağlanmasıyla oluşan bağ peptit bağı olarak adlandırılır.

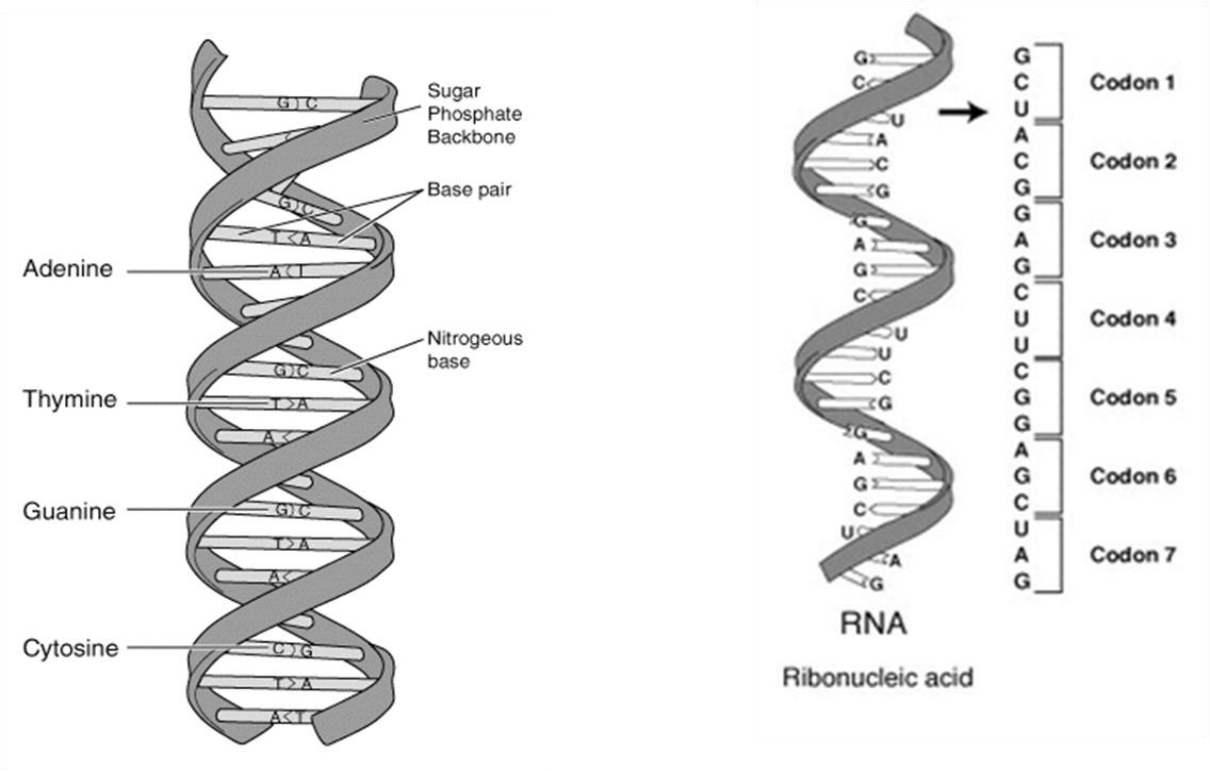
Çizelge 2.9 Amino asitler

Amino Asit	Tekli	Üçlü	R Grubu
Alanine	A	Ala	$-CH_3$
Arginine	R	Arg	$-CH_2CH_2CH_2NH-C(NH_2)_2$
Asparagine	N	Asn	$-CH_2NH_2CO$
Aspartic acid	D	Asp	$-CH_2COO^-$
Asparagine	B	Asx	$-CH_2NH_2CO$
Cysteine	C	Cys	$-CH_2SH$
Glutamine	Q	Gln	$-(CH_2)_3NO_2$
Glutamic acid	E	Glu	$-CH_2CH_2COO^-$
Glutamine	Z	Gln	$-CH_2CH_2CONH_2$
Glycine	G	Gly	-H
Histidine	H	His	$-CH_2$-imidiazol
Isoleucine	I	Ile	$-CH(CH_3)CH_2CH_3$
Leucine	L	Leu	$-CH_2CH(CH_3)_2$
Lysine	K	Lys	$-CH_2CH_2CH_2CH_2NH_3^+$
Methionine	M	Met	$-CH_2CH_2SCH_3$
Phenylalanine	F	Phe	$-CH_2-C_6H_5$
Proline	P	Pro	$-CH_2CH_2CH_2$-N
Serine	S	Ser	$-CH_2OH$
Threonine	T	Thr	$-CH(OH)CH_3$
Trytophan	W	Trp	$-CH_2$-indol
Tyrosine	Y	Tyr	$-CH_2-C_6H_4OH$
Valine	Z	Val	$-CH(CH_3)_2$

Çoğu protein, biyokimyasal tepkimelerde katalizör işlevi olan enzimlerdir ve metabolizma için yaşamsal bir role sahiptir. Başka proteinlerin ise yapısal veya mekanik işlevleri vardır: örneğin hücre iskeletindeki proteinler, hücrenin şeklini koruması için bir iskele görevi

yaparlar. Proteinler hücre haberleşmesi, bağışıklık yanıtı, hücre tutunması ve hücre bölünme döngüsünde yer alır.

Proteinlerin sınıflandırılması çalışmaları akademik yazımda genel olarak; biyolojik ve moleküler işlevlerin tahmini, enzimlerin sınıflandırılması, hücresel bileşiklerin bulunması, yapısal olarak aile ve süperaile grubunun belirlenmesi gibi geniş kapsamlı olarak birçok alanda yapılmaktadır. Bunları yaparken kimyasal ve fiziksel özelliklerin sayısallaştırılması, dizinsel yapıların üzerine oluşturalan modeller önemli bir yer işgal etmektedir.



Şekil 2.5 Örnek DNA (Çift sarmal) ve RNA (tek sarmal) yapısı[*]

2.2.3 Protein İkincil ve Üçüncül Yapılarının Tahmini

Protein molekülleri üzerinde eskiden günümüze gelen en temel sorulardan birisi herhangi iki molekül parçacığı birbirine ne kadar benzer yada ne kadar birbiriyle ilişkili olduğu sorusudur.

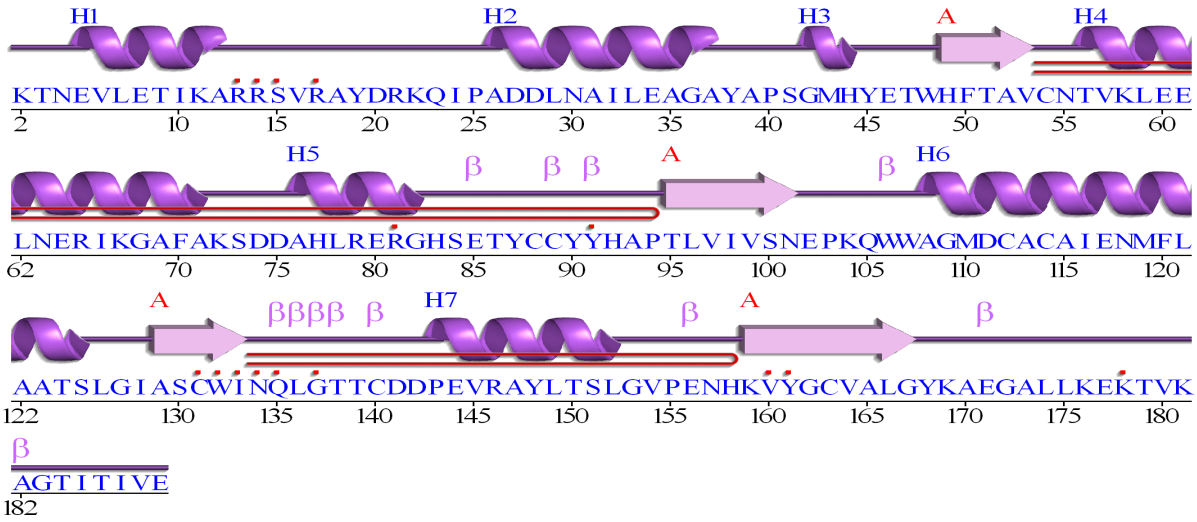
Protein dizgisel yapılarını dört ana başlıkta inceleyebiliriz.

- **Birincil Yapı** : Bir proteinin amino asit dizilimini belirten ifadelerdir. Fasta formatlı dosyalar buna en güzel örnektir.
- **İkincil Yapı** : Hidrojen bağları ile kararlılık oluşturan protein dizilimlerini belirtmek

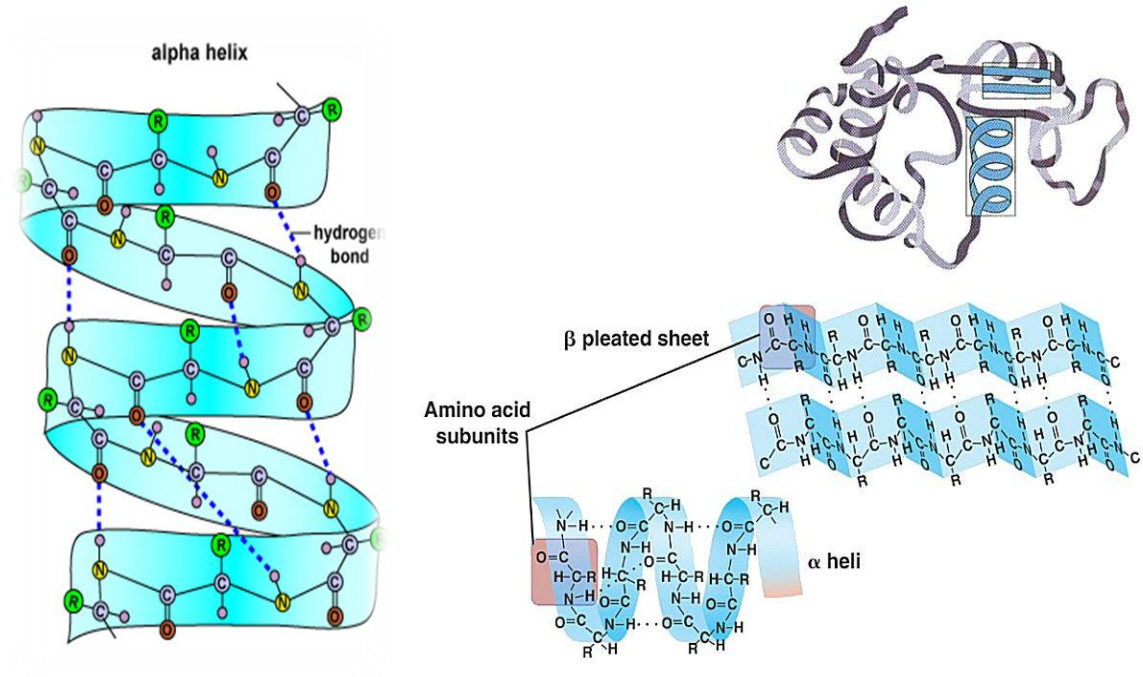
[*] Pearson Education, published as Benjamin Jummings

için gösterilir. İki farklı şekilde akademik yazımda ifade edilmektedir.

- **HSL** : Alpha Sarmal (H), Beta Sheet(β) ve Loop(L)
- **DSSP** : Alpha Sarmal(H), 3-Sarmal (G), 5-Sarmal(I), Izole Beta(B), Genişletilmiş Strand(E), Hidrojen Bağlı Dönüş(T) ve Bükülme(S)



Şekil 2.6 Birincil ve ikincil yapıların beraber gösterimi*

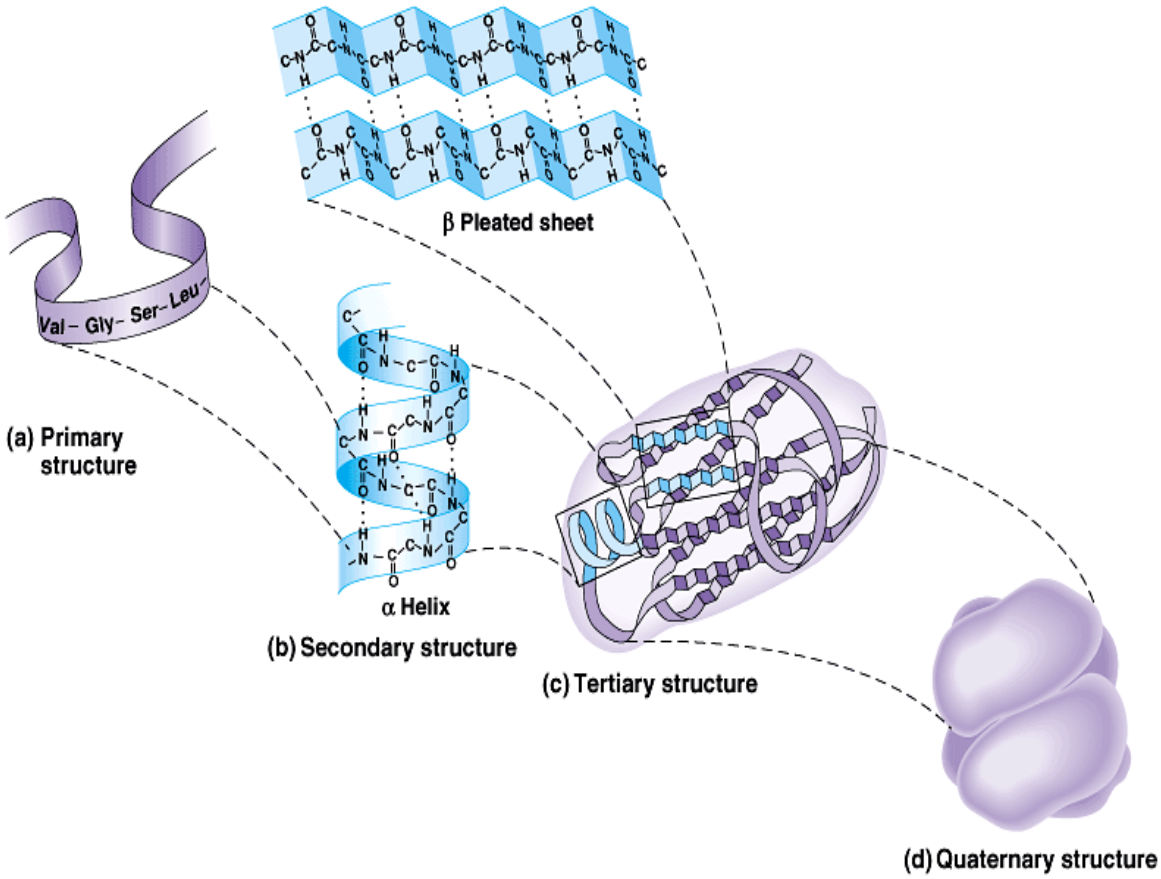


Şekil 2.7 Alpha helix ve β −sheet gösterimi**

* The Open Protein Structure Annotation Network

** Pearson Education, published as Benjamin Jummings

- **Üçüncül Yapı** : Özellikle tuz köprüleri, hidrojen bağları, disülfür bağları ve translasyon sonrası değişimler sayesinde kararlılık kazanan yapılardır. Proteinin yapısının şekli, ikincil yapıların birbiriyle olan uzaysal ilişkileri, yapısal fonksiyonları gösterir.



Şekil 2.8 Birincil, ikincil, üçüncül ve dördüncül yapıların gösterimi*

- **Dördüncül Yapı:** Proteinleirn katlanarak oluşturdukları doğal hallerden sonra proteinin birbirleriyle olan etkileşimi sonucu oluşan yapılardır.

2.2.4 Protein Moleküler İşlev Öngörüsü

En büyük protein veri bankalarından olan Pdb.org'dan 01.04.2009 tarihinde alınan toplam 51.820 adet farklı protein yapısı için ilk seviye moleküler işlevleri aşağıda verilmişitr. Bu işlevlerden bağlayıcı olan ve katalitik aktivite içeren protein yapıları için alt seviyelerede inilebilir.

* Pearson Education, published as Benjamin Jummings

Çizelge 2.10 İlk seviye moleküler fonksiyon işlevleri

GO ID	Moleküler Fonksiyon Adı (Türkçe)	Moleküler Fonksiyon Adı (İngilizce)	Protein Sayısı (Adet)
45.499	kimyasal uzaklaştırıcı aktivitesi	chemorepellent activity	3
10.860	proteazom düzenleyici aktivite	proteasome regulator activity	4
31.386	protein etiketi	protein tag	12
42.056	kimyasal çekim aktivitesi	chemoattractant activity	14
16.530	metal koruyucu aktivite	metallochaperone activity	44
45.735	besin deposu aktivitesi	nutrient reservoir activity	68
16.209	antioksidan aktivitesi	antioxidant activity	305
45.182	çeviri düzenleyici aktivite	translation regulator activity	316
15.457	destek protein ulaştırma aktivitesi	auxiliary transport protein activity	424
30.234	enzim regülatörü aktivitesi	enzyme regulator activity	2.377
9.055	elektron taşıma aktivitesi	electron carrier activity	3.388
30.528	kopyalama düzenleyici aktivite	transcription regulator activity	3.668
5.215	taşıyıcı aktivite	transporter activity	3.727
60.089	moleküler dönüştürücü etkinliği	molecular transducer activity	4.409
5.198	yapısal molekül aktivite	structural molecule activity	8.701
3.824	katalitik aktivite	catalytic activity	39.830
5.488	Bağlayıcı	binding	53.744

2.3 Koli basili (Escherichia coli) ve Mide Ülseri Bakterisi (Helicobacter pylori)

Koli basili memeli hayvanların kalın bağırsağında yaşayan faydalı bakteri türlerinden biridir. Normalde bağırsakta yaşadığı için, koli basili'nin çevresel sularda varlığı dışkı kirlenmesinin bir belirtisidir. Koli basili, pediyatrist ve bakteriyolog olan Theodor Escherich tarafından bebek dışkılarında keşfedilmiştir ve adını ondan alır; coli, "*kalın bağırsaktan*" demektir. E. coli, genel olarak bakteri biyolojisinin anlaşılması amacıyla üzerinde sıkça çalışılmış bir model organizma olmuştur. Canlılar arasında hakkında en fazla şey bilinen organizma olduğu söylenebilir.[*]

İnsanın bir günde dışkı yoluyla vücudundan geçen E. coli bakteri sayısı 100 milyar ila 10 trilyon arasındadır. Dışkıyı oluşturan bakteriler başlıca anerobik bakterilerdir, seçmeli

[*] Wikipedia

anerobik E. coli hücrelerinin sayısı diğer bakteri türlerinin binde biri dolayındadır. Başka hayvanlarda etkisiz olan bazı E. coli tipleri insana bulaştıklarında hastalık yapabilirler. Bunların en ünlüsü sayılan O157:H7 adlı serotip kanlı ishale ve ölüme yol açabilir. E. coli, normal bağırsak florasına aittir, biyolojik sınıflandırmada da bağırsaklarda yaşayan bakterilerden oluşan enterik bakteriler ailesinde yer alır. Bakteri çubuk şeklinde olup, boyutları 1-2 µm uzunluğunda ve 0.1-0.5 µm çapındadır. E. coli Gram-negatif bir bakteri olduğundan endospor oluşturmaz, pastörizasyon veya kaynatma ile ölür. Memeli hayvanların bağırsaklarında büyümeye adapte olmuş olduğu için en iyi vücut sıcaklığında çoğalır. E. coli tipleri içinde bulundukları hayvan için zararsız olmalarına rağmen insana geçtiklerinde hastalık yapabilirler. Bu hastalıklar arasında başlıca ishalli hastalıklar olmakla beraber idrar yolu enfeksiyonları, menenjit, peritonit, mastit, septisemi ve gram-negatif pnömoni de sayılabilir.

H. Pylori, mide ve duodenum'um çeşitli alanlarında yerleşen, gram (-), mikroaerofilik bir bakteridir. Yerleştiği yerlerde kronik enflamasyona neden olur. Bu kronik enflamasyon sonucunda duedenum ülseri, mide ülseri ve mide kanseri gelişebilir.

Tamamen, bizim oluşturmuş olduğumuz veritabanında gen veribankasından elde edilen 298 adet bağlayıcı ve 827 adet katalitik aktivite üyesi E.Coli bakterisi ile 195 adet bağlayıcı ve 280 adet katalitik aktivite üyesi H.Pylori bakterisini ele aldık. Daha sonra bu bakteri örneklerinden elde edilen PROFEAT (1447), PRINTS(1901) ve PROSITE (2023) motif veritabanlarını kullanılarak elde ettiğimiz bir frekans uzayına sahip olmuş olduk. Toplam 1600 örnek ve 5371 özellikten oluşan bir veri seti elde edilmiş oldu (Cataltepe vd., 2004).

2.4 Kanser ve Hastalık Verileri

Makine öğrenme veri havuzu barındırmış olduğu 187 adet farklı veri seti ile geliştirilmekte olan yeni algoritma yada modeller için test edilebilecek güzel bir platform olarak karşımıza çıkmaktadır. California Üniversitesi, Irvine, Makine Öğrenme ve Akıllı Sistemler Merkezi tarafından desteklenmekte olup bu proje kapsamında istenen bir veri bankasını ücretsiz olarak kullanabilmekteyiz. Bu veribankası havuzundan elde edilen hastalık ve kanser verilerini geliştirmiş olduğumuz öğrenme modellerine uygulayacağız.

2.4.1 Meme Kanseri Wisconsin Teşhis Veri Seti

1995 yılında, Wisconsin Üniversitesinden, Genel Cerrahi Bölümü öğretim üyesi Prof. Dr. William H. Wolberg ve aynı üniversitenin Bilgisayar Mühendisliği Bölümü öğretim üyelerinden Dr. W. Nick Street ve Dr. Olvi L. Mangasarian tarafından oluşturulmuş olan bu veri seti göğüs kitlesinden alınan iğne ucu büyüklüğündeki sayısal görüntülerden elde edilen özelliklerden oluşur. Veriler iyi huylu(zararsız) ve kötü huylu tümörler olarak iki ana grupta toplanır. Toplam 569 adet örnekten ve her bir veride 32 adet gerçek değerlikli özellikten oluşmaktadır. Bunlardan bazıları yarıçap, gri-skala değerleri, standart sapma, alanın çevresi, alanı, pürüzsüzlük, yoğunluk, içbükeylik (kontür içbükey kısımlarının şiddeti), simetrisi, fraktal boyut(öklid geometrisine alternatif bir model) vb... gibidir.

2.4.2 Şeker Hastalığı Verileri

Dr. Michael Kahm, Washington Üniversitesi, tarafından günün belirli saatlerinde elde edilerek hazırlanan bu veri setinde toplam 70 farklı kişiden 20 özellik elde edilerek hazırlanmıştır.

2.4.3 Akciğer Kanseri Verileri

Dr. Hong ZQ ve Dr. Yang JY tarafından oluşturulmuş olup 32 örnek ve 56 özellik içermektedir. KNN ile %77 gibi bir sonuç ile doğruluk hesaplanmıştır.

2.4.4 Kalp Rahatsızlığı Verileri

Dört ana veritabanının birleşiminden oluşmaktadır. Long Beach ve Cleveland Klinik Araştırma Merkezi, İsviçre Basel ve Zurich Üniversite Hastaneleri ile Budapeşte Macaristan Kardiyoloji Enstitüsü'nün hazırlamış oldukları verilerde. 303 hastadan alınmış olan 75 özellik içeren bir bilgi bankasıdır.

2.4.5 Hepatit Verisi

Carnegie-Mellon Üniversitesinden G. Gong ve Jozef Stefan Enstitüsünden Bojan Cestnik tarafından oluşturulmuş olup 155 örnek ve 19 özellik içermektedir.

2.4.6 Diğer kanserli Veriler

Yukarıda tanımlanan kanserli verilere ek olarak UCI'den elde edilen 32 örnek, 56 öznitelik ve 3 sınıflı Mide Kanseri verileri, 150 örnek , iris veri seti , 400 örnekli göğüs kanseri verisi,

kalın bağırsak veri kümesi, 5875 örnekli ve 26 öznitelikli Parkinson hastalığı gibi daha birçok veri kümesini deneylerimizde kullandık.

2.4.7 Bag of Words (Kelimeler Yığını)

2008 yılında California Üniversitesi, Irvine, öğretim üyelerinden Prof. Dr. David Newman tarafından oluşturulmuş olan bu veri seti çok yüksek boyutlu bir veri setidir. Detayları aşağıda verilmiştir.

Çizelge 2.11 Bag-of-words veri seti detayları

	Örnek Sayısı	Kelime Sayısı	Toplam Kelime Sayısı
Enron e-postaları	39.861	28.102	1.900.000
NIPS makaleleri	1.500	12.419	6.400.000
KOS blog bilgileri	3.430	6.906	467.714
NYTimes haberleri	300.000	102.660	100.000.000
PubMed makale özetleri	8.200.000	141.043	730.000.000

Toplam örnek sayısına bakıldığında 8.5 milyon civarı örnekten ve bu örneklere bağlı 140 bin civarında kelimeden oluşmaktadır. Verinin toplam boyutu ~10GB civarındadır. Bir bilgisayarda böyle bir verinin hesaplanması zor olacağından veri ön işleme ve boyut indirgeme gibi temel görevler yerine getirilerek veriler işlenmiştir.

3. ÖĞRENME YAKLAŞIMLARI ve UYGULAMA ALANLARI

Büyük boyutlardaki veri ve işaretlerin elle işlenmesi, analiz edilmesi oldukça güç bir olaydır. Öyleki bir protein molekülündeki bir aminoasit dizilişini elle işlemek ve üzerinde yeni modeller geliştirmek nerdeyse imkansızdır, ancak gözlemlenebilir. Vücut bünyesinde bulunan milyonlarca farklı boyutlarda ve dizilişlerde protein yapıları mevcuttur. Bu protein dizilişleri üzerinde oluşabilecek mutasyonlar (ekleme, silme, yer değiştirme vb...) protein işlevini değiştirebilir yada hiç etkilemeyebilir. Makine öğrenmesi bu tür büyük işaretler üzerinden uygun bilginin elde edilmesi var olan geçmişteki işaretleri kullanarak en uygun modeli oluşturmak üzerine kuruludur. Makine öğrenme metodlarının uygulama alanlarını Alpaydın'a (2004) göre sınıflandırma, kümeleme, eğri uydurma, özellik seçimi ve ilişki kurma şeklinde ifade edebiliriz.

3.1 Makine Öğrenme Yaklaşımları

Makine öğrenme kavramının gelişimi ile birlikte yeni ve farklı metodolojiler ortaya konmuştur. Makine öğrenmesinde öncelike öğrenme yapılacak veri kümesinin uygulanacak öğrenme metoduna uygun bir şekilde ön işleme tabi tutulması gerekmektedir. Bu yaklaşımları temelde iki(gözetimli, gözetimsiz) fakat genelde altı alt sınıfa ayırabiliriz:

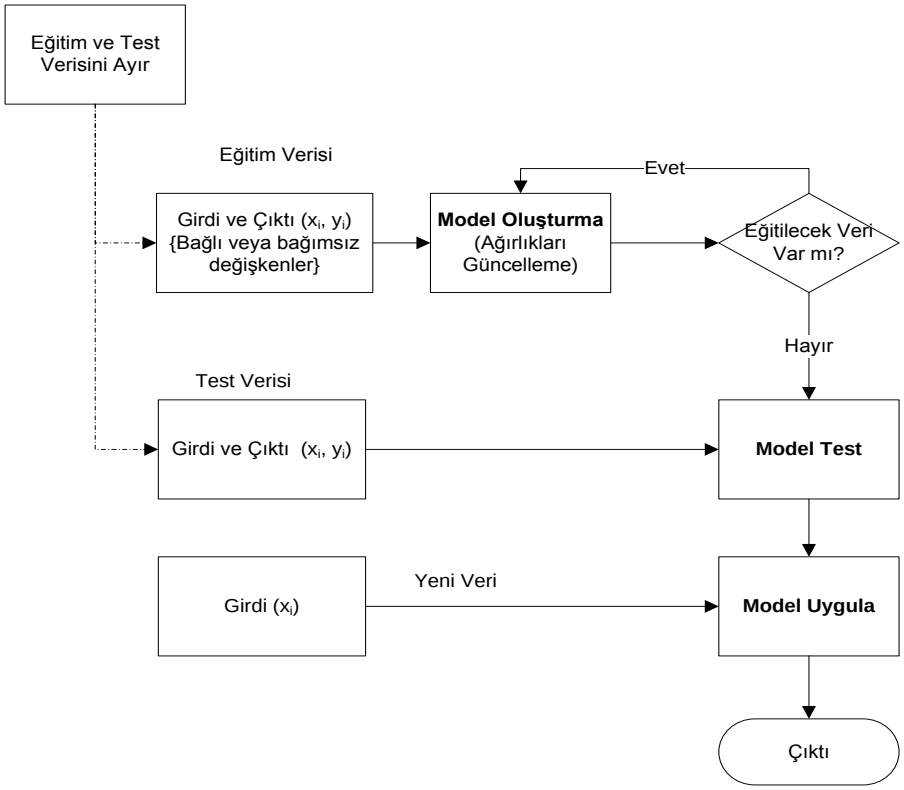
- Gözetimli öğrenme
- Gözetimsiz öğrenme
- Yarı-gözetimli öğrenme
- Ödüllü öğrenme

Bu sınınflardan özellikle gözetimli algoritmalar üzerine geliştirdiğimiz özellikle protein yapıları üzerinde ve metin sınıflandırma üzerine uygulanan "*Dizi Çekirdek*" (Bölüm 4.3) modelleri, "*Artımlı ve Azalımlı Çoklu Çekirdek Öğrenme*" modelleri (Bölüm 4.5), yarı gözetimli algoritmalardan Gauss Rastgele Alan Modeli üzerine kurulu yeni önerdiğimiz "*Etkin Alt Uzay Yarı Gözetimli Algoritma*" modeli detayları ile ele alınacaktır. Metodların performansları var olan diğer metodlarla karşılaştırmalı olarak verilecektir.

3.1.1 Gözetimli Öğrenme (Supervised Learning-SL)

Gözetimli öğrenme, eğitici ile öğrenme yada denetimli öğrenme şeklinde de ifade edilebilmektedir. Bu tip öğrenmede sınıfların sayısı ve hangi nesnelerin (x_i, y_i) hangi sınıfa olduğu (yani etiketi) bellidir. Çıktı etiketleri $\{y_i \in \mathbb{R}\}$ yada $\{y_i \in \mathbb{R}^d\}$ ayrık bir sınıf ise bu tip

öğrenmeye sınıflandırma, çıktı değerleri sürekli bir fonksiyon değeri ise bunada eğri uydurma denir. Bu öğrenme modelinde algoritmaya girdi olarak verilen etiketlenmemiş veri ile çıktı olarak düşünülen etiketli veri arasında bir eşleme yapılarak model geliştirilir ve başarım ödünleşmesi hesaplanır.



Şekil 3.1 Gözetimli öğrenme adımları

Modelin öğrenmesi için veriyi iki ana bölüme ayırmak gerekmektedir. Bu ayrılan parçalardan ilk bölümü modeli eğitmek yada öğretmek için geriye kalan kısmını ise modelin başarımını sınamak için kullanmak gerekir (Şekil 3.1). Eğitim verisini doğru bir girdi olarak verebilmek için veri madenciliğinde sıkça kullanılan işaretlerin sayısallaştırılması (resim yada videoların verktörel olarak ortama aktarılması, dökümanlardan gerekli olan kelime yada köklerin uzman sistemler tarafından elde edilerek modelin anlayacağı girdiye dönüştürülmesi, özellik kümesinden gereksiz özniteliklerin çıkarılması vb...), eğitim örneklerinin tiplerini belirleme, işaretlerden gürültüyü ayırma, işaretlerin birleştirilmesi, daha düşük boyuta indirgenmesi gibi işlemlerden geçirmek gerekmektedir.

Gözetimli öğrenmeye örnek olarak karar ağacı sınıflandırıcıları, yapay sinir ağları, destek vektör makinaları, naive bayes, en yakın komşuluk, C4.5, doğrusal ayrıştırma analizi verilebilir.

3.1.2 Gözetimsiz Öğrenme(Unsupervised Learning-USL)

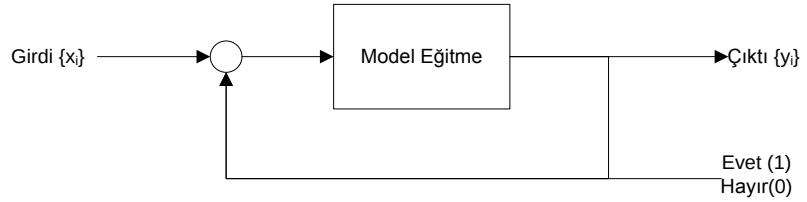
Sisteme sadece girdi değerleri verilir ve parametreler arasındaki ilişkiler ve ağırlıkları sistemin kendi kendine öğrenmesi beklenir. Gözetimsiz öğrenme $X = (x_1, x_2, \dots, x_n)$ n adet örnekten oluşan bir set olmak kaydıyla, tüm $x_i \in X$ için $i \in [n] \coloneqq \{1, \dots, n\}$. (Chapelle vd. , 2006) olmak üzere çıktı etiketlerini $\{y_i \in \mathbb{R}\}$ elde etmek şeklinde tanımlanabilir. Gözetimsiz öğrenmeye en güzel örnekler demetleme, yoğunluk tabanlı öbekleyiciler (dbscan, optics, denclue, clique, ...), veri sıkıştırma algoritmaları, resim analizi, örüntü tanıma şeklinde söylenebilir. Genel olarak örneklerin olasılıksal dağılımına yada birbirlerine olan uzaklık yada benzerlikleri baz alınarak yapılan yöntemlerdir.

3.1.3 Yarı-Gözetimli Öğrenme (Semi-Supervised Learning - SSL)

Yarı-gözetimli öğrenme, genel tanım olarak etiketli ve etiketsiz verilerin kullanıldığı bir öğrenme şeklidir. Kısıtlı olan etiketli veri seti ile büyük boyutlardaki etiketsiz veri setlerini kullanarak öğrenme işlemi gerçekleştirir. Birçok SSL algoritması temel bir sınıflandırma ve orijinal SL algoritması kullanılarak geliştirilmiştir. Farklı stratejiler veya sezgiler ile etiketsiz veriyi kullanarak model eğitme gerçekleştirilir (Abney, 2008).

3.1.4 Destekleyici/Ödüllü Öğrenme (Reinforcement Learning - RL)

Öğretmen, her girdi seti için olması (üretilmesi) gereken çıktı setini sisteme göstermek yerine, sistemin kendisine gösterilen girdilere karşılık çıktısını üretmesini bekler ve üretilen çıktının doğru veya yanlış olduğunu gösteren bir sinyal üretir (Littman ve Moore, 1996). Bu sinyal dikkate alınarak, eğitim süreci devam ettirilir. Ödüllü öğrenme, robot kontrolü, asansör sıralaması, telekomunikasyon, tavla ve satranç gibi çeşitli problem alanlarına uygulanabilir.



Şekil 3.2 Ödüllü öğrenme adımları

3.2 Veri Madenciliği

Veri madenciliği büyük miktarda veri içeren veri bankalarından gelecekle ilgili tahmin yapmamızı sağlayacak bağıntı, öznitelik, alt uzay ve kuralların bilgisayar programları

kullanarak oluşturulması ve aranmasıdır (Alpaydın, 2004). Başka bir deyişle veri madenciliği, büyük hacimli (protein veri bankaları, metinler, html sayfaları, banka ve muhasebe işlemleri gibi...) veri yığınları içerisinden karar alabilmek için potansiyel olarak faydalı olabilecek uygulanabilir ve anlamlı bilgilerin çıkarılmasına verilen isimdir. Veri madenciliği geniş anlamda veri analiz teknikleri bütünüdür. Mevcut problemleri çözmek, kritik kararları almak veya geleceğe yönelik tahminleri yapmak için gerekli olan bilgileri elde etmeye yarayan bir öğrenme modelidir. Ortaya çıkarılması hedeflenen bilgiler; üstü kapalı çok net olmayan önceden bilinmeyen, daha önce keşfedilmemiş ancak potansiyel olarak kullanılabilir ve keşfedilebilir bilgiler olmalıdırlar.

3.2.1 Veri Madenciliği Adımları

Veri madenciliği aslında bilgi keşfi sürecinin bir parçası şeklinde kabul görmektedir. Bilginin bazı yöntem ve modeller ile analiz edilmesi ve çıkan sonuçların bir uzman gözüyle yorumlanmasıyla geçmiş verilerden gelecek tahminleri yapma işlemi aslında bazı temel adımların oluşturulmasından sonra anlamlı hale gelmektedir. Bu adımlar:

- Gürültü temizleme ve tutarsız verilerin ayıklanması
- Verilerin birleştirilmesi ve ortak kaynak ulaşımı
- Öznitelik seçimi(Analiz için anlamlı özelliklerin kullanımı)
- Veri dönüşümü
- Örüntü üzerine uygulanacak akıllı yöntemlerin oluşturulması
- Örüntü değerlendirme ve örüntü oluşturma
- Bilgi ve sonuçların sunumu

3.3 Makine Öğrenme Alanları

Makine öğrenme ana olarak beş alan üzerine yoğunlaşmaktadır. Bunlar; etiketli bilgiler için sınıflandırma, etiketi belirsiz veriler üzerinde benzerlik yada uzaklık bağlantılı öbekleme, sürekli veri üzerinde yürüyecek bir eğri uydurma, model yada sınıflandırıcı için veriyi en iyi ifade edecek öznitelik kümesinin belirlenmesi ve ilişki belirleme şeklinde söylenebilir.

4. ÇEKİRDEK ÖĞRENME ve DESTEK VEKTÖR MAKİNALARI

Eldeki verilere en uygun modellerin kurulması, sınıfları belli(etiketli) veya sınıfları belirsiz(etiketsiz) girişlerin eğitim sonucunda karşı düştüğü sınıfların belirlenmesi makine öğrenememin en temel görevlerindendir. Örnek verilere en uygun modelin kararlaştırılması, seçilen bir uzaydaki, en uygun işaretin veya işlevin belirlenmesi; örüntü tanıma, işaret işleme ve kontrol gibi birçok alanın da en temel araştırma alanlarındandır. Çekirdek modelleri gözetimli öğrenmenin en dikkate değer yöntemlerinden biridir. Son on yılda akademik yazımda en çok bildiri ve makale yapılan yötemlerden birisi olarak dikkat çekmektedir. Yapısal olarak incelediğimizde doğrusal aryılamayan örnek uzayı daha yüksek boyutlu bir doğrusal uzaya taşıyarak doğrusal ayrılabilen bir matematiksel en iyileme dönüşüm işlemi olarak ele alınabilir. Önerdiğimiz yeni modelin alt yapısını oluşunu oluşturmak adına bu yapıyı detayları ile ele alalım.

4.1 Çekirdek Kuramı

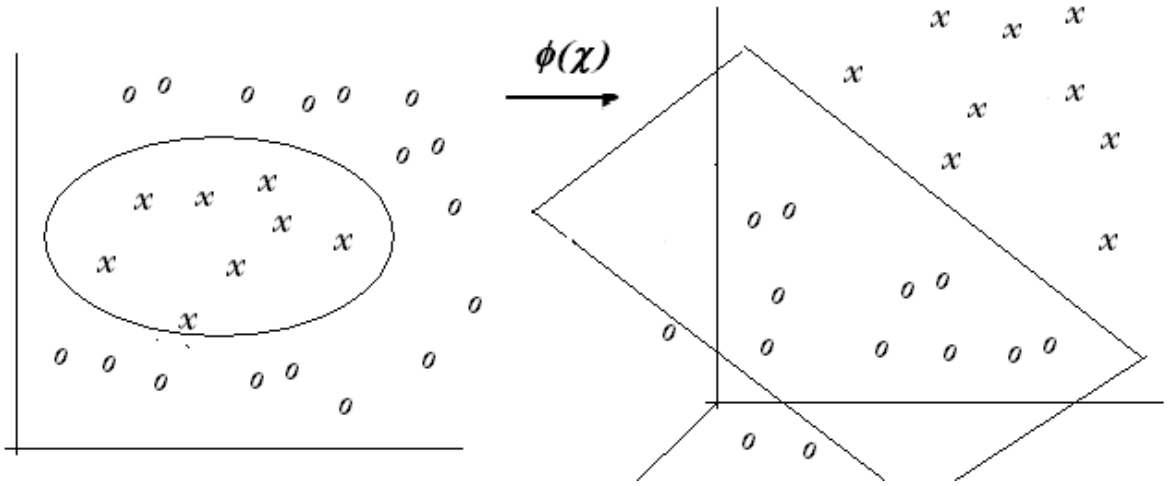
Giriş uzayındaki veriler her zaman doğrusal ayrılamayabilir. Çekirdek dönüşüm fonksiyonu yardımı ile daha yüksek boyutlu öznitelik uzayına örtük olarak taşındığı taktirde bir *aşırıdüzlem* tarafından doğrusal ayrılabilen bir problem haline dönüştürülebilir. Çekirdekler kullanılarak elde edilen yeni öznitelik uzayı büyüklükleri, doğrudan her bir eğitim verisinin işlevi olarak verilmeyip, eğitim verisinin iç çarpımlarının dönüşüm işlevi olarak verilebilir. İlk olarak 1964'te Aizerman tarafından çekirdek çarpımı şeklinde ortaya konmuş ve DVM'lerinde de günümüzde bu metodolojiyi sıkça kullanmaktayız. Şekil 4.1'de görüleceği üzere doğrusal olmayan düzlemin ϕ çekirdeğiyle doğrusal olarak ayrılabilen bir öznitelik uzayına dönüştürmemiz mümkündür. Yüksek boyutlu uzaya dönüşüm için daha önceden doğrusal uzay için elde edilen Hilbert uzayı ($\boldsymbol{H}$) matrisini oluşturmak gereklidir.

Örnek : $\phi: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ şeklinde ifade eden bir dönüşüm fonksiyonu olsun (Şekil 4.1). $(x_1, x_2) \rightarrow (x_1, x_2, x_3) \coloneqq \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right)$ gibi oluşturulacak yeni yeni uzayda veri kümelerinin düzgün bir aşırı düzlem ile ayrılabileceği varsayımı yapılır.

$$\Re^D \rightarrow \boldsymbol{\mathcal{H}}, \boldsymbol{x} \rightarrow \phi(\boldsymbol{x}) \tag{4.1}$$

H, Hilbert uzayını tekrar tanımlamak gerekirse

$$\boldsymbol{\mathcal{H}}_{ij} = y_i.y_j.\phi(\boldsymbol{x_i})^T.\phi(\boldsymbol{x_i}), \;\; \boldsymbol{\mathcal{K}}\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \phi(\boldsymbol{x_i})^T.\phi(\boldsymbol{x_i}) \tag{4.2}$$



Şekil 4.1 Giriş uzayının(*2D*), doğrusal ayrılabilen bir öznitelik uzayına(*3D*) çevrimi

şeklini alır ve $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$ doğrusal çekirdek şeklinde ifade edilir. Giriş uzayının yüksek boyutlu uzaya dönüşümü işlevinin belirlemek için iç çarpım çekirdeğinin hesaplanmasından başlanır ve elde edilen çekirdekten yüksek boyutlu dönüşüm işlevi çıkartılabilir. Çekirdekler üzerindeki şartlardan en önemlisi *Mercer* şartı olarak da bilinen $\boldsymbol{H}$ matrisinin simetrik olması, sürekli olması ve pozitif tanımlı olma şartlarını sağlaması gerekmektedir (Nilson, 1996; Vapnik, 1998; Baldi ve Barunak, 2001; Schölkopf ve Smola, 2002). Yukarıdaki şartları sağlayan bir dönüşüm fonksiyonu var ise buna Mercer çekirdek dönüşümü adı verilir.

4.1.1 Doğrusal Ayrılabilen İkili Sınıflandırma:

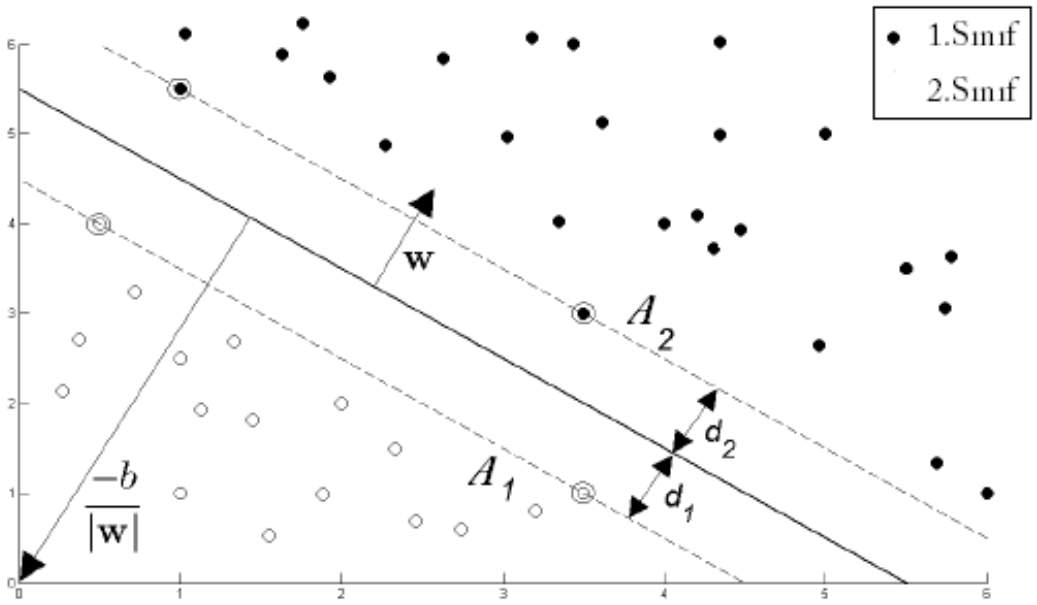
Doğrusal bir düzlem tarafından ayrılabilen pozitif ve negatif etiketli verilere ait ve *D* adet özelliği olan giriş uzayına ($\boldsymbol{x}_i$), karşılık düşen etiketleri (y_i) olmak üzere, bu iki uzayı ayıran uzaklığı en iyileyecek aşırıdüzlemi bulan yöntem olarak tasarlanmıştır. $\{(x_1, y_1), \ldots, (x_L, y_L)\}$ eğitim seti için $y_i \in \{-1, +1\}$ ve $\boldsymbol{x} \in R^{\boldsymbol{D}}$ olmak üzere bu iki sınıfa ait verileri doğrusal olarak ayırabilen düzlemin normali $\boldsymbol{w}$ ile gösterilirse bu aşırı düzlemi Şekil 4.2'deki gibi gösterebiliriz.

$$\boldsymbol{x}_i.\boldsymbol{w} + b \geq +1, \quad \forall y_i = +1 \tag{4.3}$$

$$\boldsymbol{x}_i.\boldsymbol{w} + b \leq -1, \quad \forall y_i = -1 \tag{4.4}$$

veya bu eşitsizlikleri kapalı biçimde yazdığımızda, *i=1,...L* için

$$y_i.(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1 \geq 0 \; \forall i \tag{4.5}$$



Şekil 4.2 Verileri doğrusal ayırabilen aşırıdüzlem

olarak gösterilebilir. Burda A_1 ve A_2 aşırıdüzlemini $y_i \in \{-1, +1\}$ sınıflarının sınır aşırıdüzlemleri şeklinde tanımlarız ve $y_i.(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1 = 0$ şeklinde ifade edebiliriz. Bu iki aşırıdüzlemler üzerindeki eğitim verilerine de destek vektörleri denir. Temel geometri bilgisinden yola çıkarak aşırıdüzlemler arasındaki ayrımı $y_i.l(x_i)/\|\boldsymbol{w}\|$ ile ifade edebiliriz (Vapnik, 1998; Suykens vd., 2002; Ayan ve Cansever, 2010c). Oluşturulacak modelin hata payını enazlamak için bu miktarı ençoklayan modeli oluşturmalıyız. Görüleceği gibi ayrımı ençoklamak için (4.5) denklemi kısıtını gözeterek$\|\boldsymbol{w}\|$ miktarını enazlamak gerekmektedir. Böyle bir problemin çözümü için sonsuz sayıda ayrıştırıcı düzleme böylecede bir o kadar $\boldsymbol{w}$ değeri elde edilebilir. İkinci dereceden denklem şekline ve problemi bir eniyileme modeline dönüştürdüğümüzde

$$min\ \frac{1}{2}\|\boldsymbol{w}\|^2 \quad y_i.(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1 \geq 0, \forall i \tag{4.6}$$

şeklini alır ve böylece kısıtlı enazlama modelini Lagrangian çarpanları $\alpha_i \geq 0, \forall i$ ile çoklanarak çözüm elde edilir.

$$L_1(\boldsymbol{w}, b, \alpha) \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 - \boldsymbol{\alpha}[y_i.(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1]\ \forall i \tag{4.7}$$

$$\equiv \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{L} \alpha_i . y_i . (\mathbf{x}_i . \mathbf{w} + b) + \sum_{i=1}^{L} \alpha_i \tag{4.8}$$

Denklem (4.8)'in, $\boldsymbol{w}$ ve b'ye göre türevi alındığında $\boldsymbol{w}$ ve b nin enazlama, Lagrangian çarpanlarının da ençoklama için gerekli ifade elde edilmiş olur. Oluşacak çözüm kümesinin *Karush-Kuhn-Tucker* teoremi olarak bilinen aşağıdaki şartları sağlaması gerekmektedir. Böylece oluşacak **w** aşırıdüzlemin formülü elde edilmiş olur.

$$\frac{\partial L_1}{\partial \mathbf{w}} = 0 \Longrightarrow w = \sum_{i=1}^{L} \alpha_i . y_i . \mathbf{x}_i \tag{4.9}$$

$$\frac{\partial L}{\partial b} = 0 \Longrightarrow \sum_{i=1}^{L} \alpha_i . y_i = 0 \,,\ \forall \alpha_i \geq 0 \tag{4.10}$$

Burda $\boldsymbol{w}$ değerini yerine (4.9)'daki eşitliği yerleştirdiğimizde ikincil Lagrangian denklemini elde etmiş oluruz.

$$L_2(\boldsymbol{\lambda}) \equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2} \sum_{i,j}^{L} \alpha_i \alpha_j \, y_i y_j (\boldsymbol{x}_i . \boldsymbol{x}_j) \; ve \; \alpha_i \geq 0 \; \forall i \,, \sum_{i=1}^{L} \alpha_i y_i = 0 \tag{4.11}$$

$$\equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{H} \boldsymbol{\lambda} \,, \qquad H_{ij} = y_i y_j (\boldsymbol{x}_i . \boldsymbol{x}_j) \tag{4.12}$$

Böylelikle ikincil Lagrangian modelini oluşturmuş oluruz. Oluşturulan ikincil modelde giriş uzayının iç çarpımı olarak ifade edilen bilgi aslında çekirdek metodu şeklinde ifade edilir. Çekirdek metodlara daha sonra detaylı değinilecektir. Enazlayan L_1'den ençoklayan L_2'ye geçtiğimizde bulunması gereken sadece $\boldsymbol{\alpha}$ vektörü olacaktır. Böyle bir dışbükey ikinci dereceden eniyileme problemin çözüm kümesine düşük boyutlu matrisler için kolaylıkla ulaşılabilir.

Destek Vektörleri :

Tüm eğitim seti yerine sadece destek vektörlerinin oluşturduğu bilgiler kullanılarak test edilen veri seti diğer yöntemlere göre daha hızlı çalışmaktadır. Destek vektörlerine karar verirken denklem (4.13)'deki $\boldsymbol{x}_s$ değerlerini sadece dikkate almamız terli olacaktır.

$$y_s . (\boldsymbol{x}_s . \boldsymbol{w} + b) = 1 \tag{4.13}$$

burda $\boldsymbol{w}$ yerine denklem (4.7)'deki ifadeyi koyduğumuzda

$$y_s.\left(\sum_{r\in S}\alpha_r.y_r.\boldsymbol{x}_r\,\boldsymbol{x}_s+b\right)=1\ ,\quad \alpha_r>0 \tag{4.14}$$

$y_s{}^2=1$ olduğundan (4.12) eşitliğini

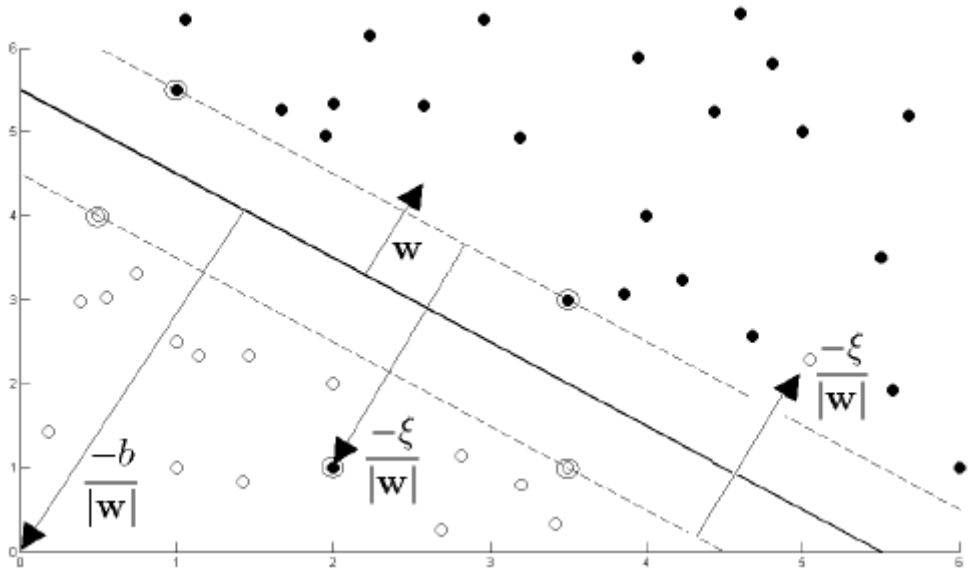
$$y_s{}^2.\left(\sum_{r\in S}\alpha_r.y_r.\boldsymbol{x}_r\,\boldsymbol{x}_s+b\right)=y_s \tag{4.15}$$

şekline çevirmiş oluruz burdan b yi destek vektörlerinden yola çıkarak elde edebiliriz. Herhangi bir destek vektör kullanarak aşırıdüzlemleri ayırma yerine destek vektörlerinin ortalamasını almak hata payını düzeltecektir (4.16).

$$b=\frac{1}{N_s}\sum_{s\in S}(y_s\ -\sum_{r\in S}\alpha_r.y_r.\boldsymbol{x}_r\,.\boldsymbol{x}_s) \tag{4.16}$$

4.1.2 Doğrusal Ayrılamayan İkili Sınıflandırma

Eğitim verilerin tümü doğrusal olarak ayrılamayabilir yada noktaların bazıları diğer düzlem içerisinde bulunursa çoğu zaman daha geniş bir pay ve bundan dolayı daha düşük bir yapısal risk elde edilebilir. Böyle durumlarda (Şekil 4.3) kısıtları gevşek olarak çözmek ve yapısal riski enazlayan verileride göz önünde bulundurmak gereklidir.



Şekil 4.3 Hata payı ile doğrusal ayırabilen aşırıdüzlem

İki aşırıdüzlem içerisinde düşen veya karar sınırının hatalı tarafında bulunan veri noktaları için kısıtlarını gevşeterek çözüme gidilirse, $\xi_1, \dots, \xi_L$ ödünleşme değişkenleri tanımlanır. Böylece denklemimiz (4.17)'deki gibi olacaktır.

$$y_i.(\boldsymbol{x}_i.\boldsymbol{w}+b)-1+\xi_i \geq 0\,, ve\ \xi_i \geq 0\ \forall i \tag{4.17}$$

Bu durumda en büyük pay ile çok boyutlu düzlemi bulmak için, (4.5) ve (4.6)'deki denklemlere yapay değişkenlerin toplamı eklenir. Kısıt üzerinden enazlanarak çözüme tekrar başlanır. (4.18)'de $\boldsymbol{C}$ parametresi öğrenme işlevlerinin karmaşıklığı ve izin verilen ayrıştırılamayan eğitim örneklerinin geveşeklik oranı arasındaki ödünleşimi niteler.

$$min\ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}\xi_i \quad y_i.(\boldsymbol{x}_i.\boldsymbol{w}+b)-1+\xi_i \geq 0\,, ve\ \xi_i \geq 0\ \forall i \tag{4.18}$$

Bu eniyileme problemi ikincil forma dönüştürülürse, aşağıdaki Lagrangian işlevi ile ençoklanarak çözülür,

$$L_1(\boldsymbol{w}, b, \boldsymbol{\lambda}) \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}\xi_i - \boldsymbol{\alpha}[y_i.(\boldsymbol{x}_i.\boldsymbol{w}+b)-1+\xi_i]\ ve\ \alpha_i \geq 0\ \ \forall i \tag{4.19}$$

$$\equiv \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}\xi_i - \sum_{i=1}^{L}\alpha_i.[y_i.(\boldsymbol{x}_i.\boldsymbol{w}+b)-1+\xi_i] - \sum_{i=1}^{L}\mu_i\xi_i \tag{4.20}$$

w, b ve ξ_i ye göre türevleri alındığında (4.9) ve (4.10)'e ek olarak

$$\frac{\partial L}{\partial \xi_i} = 0 \Longrightarrow C = \alpha_i + \mu_i, ve\ \mu_i \geq 0 \tag{4.21}$$

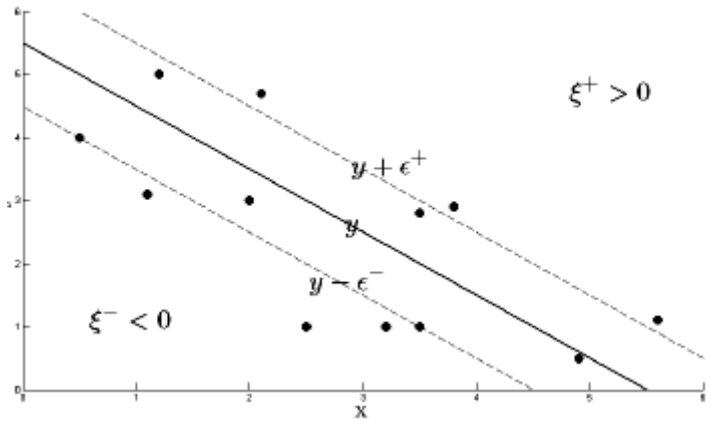
bulunur. Böylece elde edilecek α_i değerleri için $0 \leq \alpha_i \leq C$ kısıtı aranır.Böylece problem ikincil Lagrangian için ençoklama problemine dönüşür.

$$\max_{\lambda}\left[\sum_{i=1}^{L}\alpha_i - \frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{H}\boldsymbol{\alpha}\right] \quad ve\ 0 \leq \alpha_i \leq C\,, \sum_{i=1}^{L}\alpha_i.y_i = 0 \tag{4.22}$$

4.1.3 Eğri Uydurma

Eğri uydurma pozitif ve negatif sınıfların ayrımı yerine sürekli bilgiler üzerinde yapılmaktadır. $\{\boldsymbol{x}_i, y_i\}$ eğitim uzayı olmak üzere $i = 1...L$, $y_i \in R$ ve $\boldsymbol{x} \in R^{\boldsymbol{D}}$ için $y_i = \boldsymbol{w}.\boldsymbol{x}_i + b$ doğrusal düzleminin üstü ve altındaki duyarsız bölge içinde yer alması gerekmektedir. $|y_i' - y_i| < \epsilon$ ise eğer bu aralıktaki değerler için herhangi bir ceza

parametresi uygulanmaz. $y_i \pm \epsilon$ ile sınırlı alana $\epsilon -$ duyumsuz alan denir. Bu alanın dışındaki noktalar $\xi^+ > 0, \xi^\pm > 0$ için bir ceza parametresi uygulanır.



Şekil 4.4 $\epsilon -$ duyumsuz alan için eğri uydurma

$$y_i - \epsilon - \xi^- \leq y_i{}' \leq y_i + \epsilon + \xi^+, \forall i \tag{4.23}$$

Böylece eğri uydurma için hata fonksiyonu aşağıdaki gibi tanımlanır,

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}\left(\xi_i{}^+ + \xi_i{}^-\right) \tag{4.24}$$

4.23 ve 4.24'deki denklemler $\xi^+ > 0,\ \xi^- > 0, \forall i$ için enazlama yapılır. Lagrangian parametreleri $\alpha_i^+ \geq 0, \alpha_i^- \geq 0$ olmak üzere ve $\mu_i^+ \geq 0, \mu_i^- \geq 0, \forall i$ için

$$L_1(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\mu}, \epsilon, \boldsymbol{\xi}) \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}(\xi_i{}^+ + \xi_i{}^-) - \sum_{i=1}^{L}\left(\mu_i^+\xi_i{}^+ + \mu_i^-\xi_i{}^-\right)$$

$$-\sum_{i=1}^{L}\alpha_i^+(\epsilon + \xi_i^+ + y_i - y_i') - \sum_{i=1}^{L}\alpha_i^-(\epsilon + \xi_i^- - y_i + y_i') \quad \forall i \tag{4.25}$$

y_i yerin koyduğumuzda ve ayrı ayrı türevlerini aldığımızda,

$$\frac{\partial L_p}{\partial \boldsymbol{w}} = 0 \Longrightarrow w = \sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-).\boldsymbol{x}_i \tag{4.26}$$

$$\frac{\partial L_p}{\partial b} = 0 \Longrightarrow \sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-) = 0 \tag{4.27}$$

$$\frac{\partial L_p}{\partial \xi_i^+} = 0 \Longrightarrow C = \alpha_i^+ + \mu_i^+ \tag{4.28}$$

$$\frac{\partial L_p}{\partial \xi_i^-} = 0 \Longrightarrow C = \alpha_i^- + \mu_i^- \tag{4.29}$$

Lagrangian çarpanlarına ($\alpha_i^+ \geq 0,\ \alpha_i^- \geq 0, \forall i$) göre L_2ençoklama yapıldığında Lagrangian denklemini aşağıdaki ifadeye çevirebiliriz,

$$L_2 = \sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-)y_i' - \epsilon\sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-) - \frac{1}{2}\sum_{i,\mathrm{j}}^{L}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)(\boldsymbol{x}_i.\boldsymbol{x}_j) \tag{4.30}$$

burda $0 \leq \alpha_i^+ \leq C, 0 \leq \alpha_i^- \leq C$ ve $\mu_i^+, \mu_i^- \geq 0$ için $\sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-) = 0$, $\forall i$ Lagrangian'ı en çoklarsak ifade aşağıdaki gibi olur

$$\max_{\alpha_i^+, \alpha_i^-} \sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-)y_i' - \epsilon\sum_{i=1}^{L}(\alpha_i^+ - \alpha_i^-) - \frac{1}{2}\sum_{i,\mathrm{j}}^{L}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)(\boldsymbol{x}_i.\boldsymbol{x}_j) \tag{4.31}$$

Burda destek vektörleri indis i için $0 \leq \alpha_i^+, \alpha_i^- \leq C$ ve $\xi_i^- = 0$(yada $\xi_i^+ = 0$) denklemlerini sağlayan bilgilerden elde edilir. Böylece

$$b = y_s' - \epsilon - \sum_{m \in SV}^{L}(\alpha_i^+ - \alpha_i^-)(\boldsymbol{x}_m.\boldsymbol{x}_s) \tag{4.32}$$

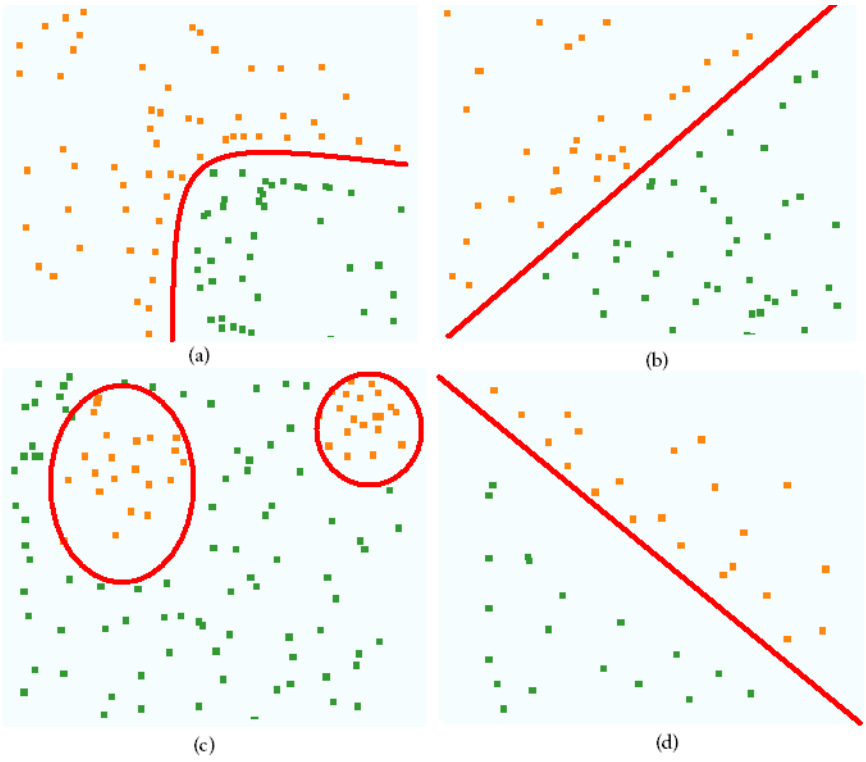
Destek vektörlerin ortalamasını aldığımızda

$$b = \frac{1}{Ns}\sum_{s \in SV}\left[y_s' - \epsilon - \sum_{m \in SV}^{L}(\alpha_i^+ - \alpha_i^-)(\boldsymbol{x}_m.\boldsymbol{x}_s)\right] \tag{4.33}$$

4.2 Çekirdek Modelleri ve Örnekler

Lagrangian (4.10) denklemlerindeki iç çarpımlar, istenilen doğrusal olmayan yüksek boyutlu çekirdek ile yer değiştirilerek yeni halini alırlar ve karar işlevlerinde de yeni çekirdek dönüşüm formülü kullanılır.

Burda farklı çekirdeklerin farklı parametreler altında nasıl davrandığını daha iyi anlayabilmek için dört adet iki boyutlu uzayda gösterilebilen yapay veri kümesi kullanılacaktır.



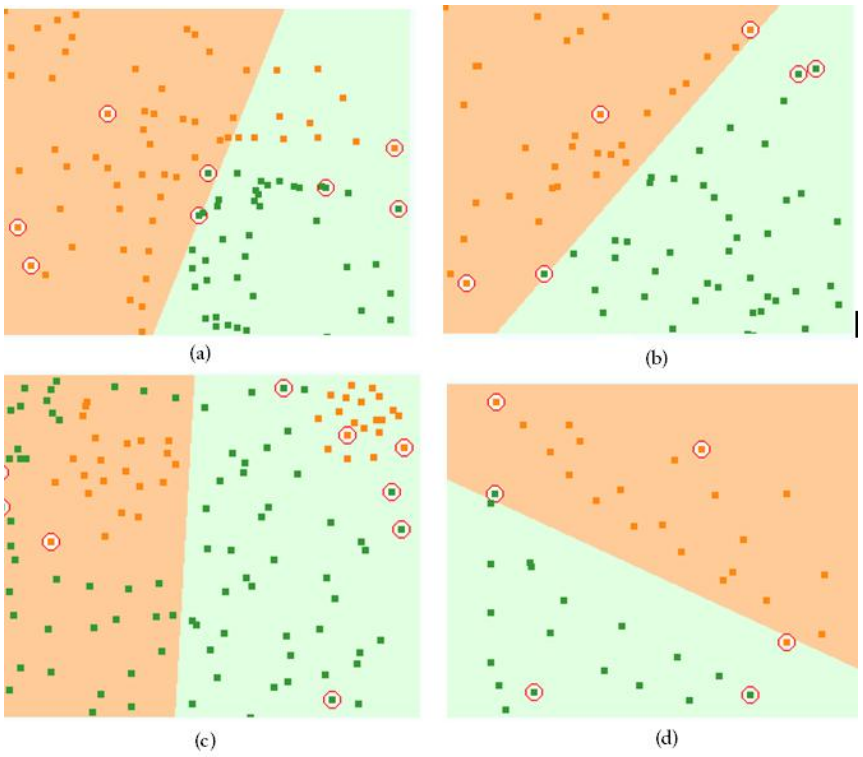
Şekil 4.5 Örnek veri kümeleri ve ayıran aşırı düzlemler

- Yapay Veri A : Polinomsal olarak (2.dereceden bir denklem ile) ayrılabilecek bir veri kümesi (Şekil 4.5a)
- Yapay Veri B, D: Doğrusal ayrılabilecek veri kümesi (Şekil 4.5b,d)
- Yapay Veri C : İçinde iki Gauss alanı olan bir veri kümesidir (Şekil 4.5c)

4.2.1 Doğrusal Çekirdek

Doğrusal dönüşüm işlemi, doğrusal bir aşırı düzlem oluşturularak çözüme gider. Burda asıl olan veri kümesini doğrusal olarak ayırabilecek bir doğru çizmeye çalışmaktır. Burda doğrusal çekirdek işlemini her hangi iki özellik için iç çarpım şeklinde tanımlayabiliriz.

$$\mathcal{K}(x_i, x_j) = x_i . x_j \tag{4.34}$$



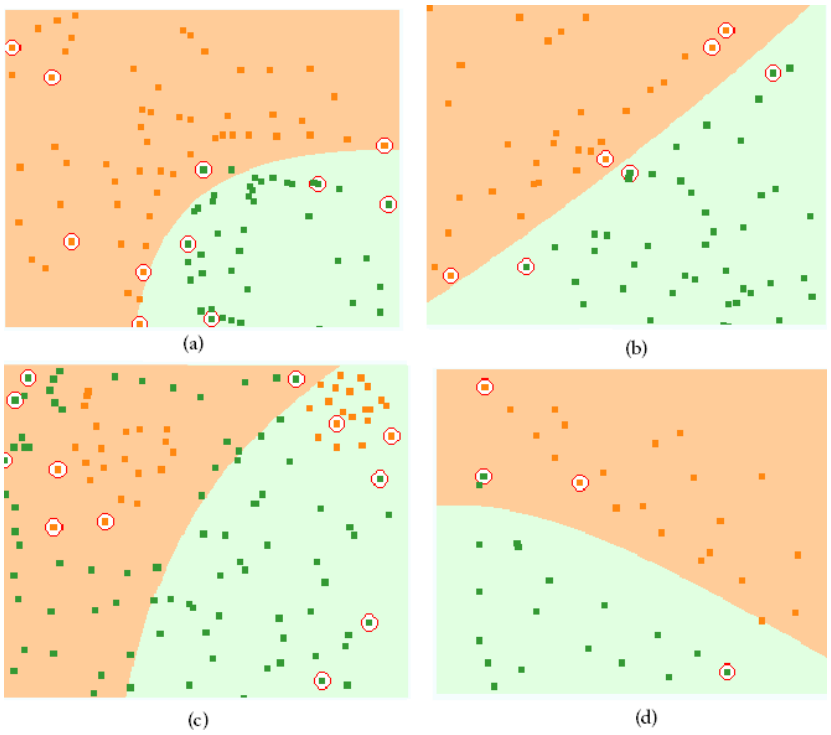
Şekil 4.6 Doğrusal çekirdek işlemi ile ayırma

Uygulamaya koyduğumuz yapay veri kümelerinin sonucunda Şekil 4.6'daki durum ortaya çıkmıştır. Şekil 4.6(b) ve (d)'de özellikler doğrusal bir düzlem tarafından hatasız olarak yada kabul edilebilir bir hata oranı ile ayrılabilirken, "Yapay Veri A" ve "Yapay Veri C" de ise gözle görülür hatalar gözlemlenmektedir. Destek vektörleri sınıfları nasıl ayıracağını öğrenememektedirler. Böyle bir durumda bu veri kümeleri için doğrusal çekirdek doğru bir çekirdek seçimi olmayacaktır.

4.2.2 Polinomsal Çekirdek

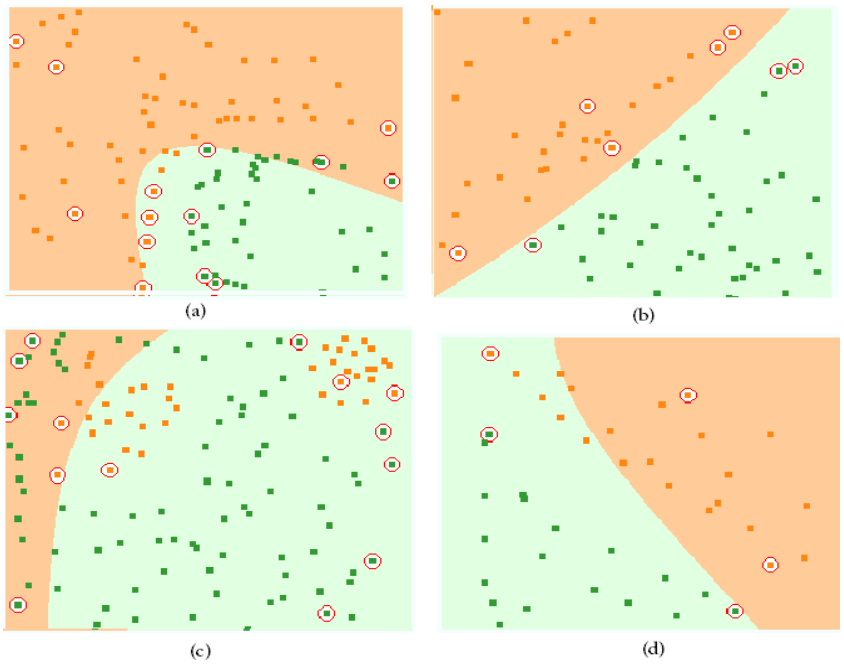
Sıkça kullanılan bir diğer çekirdek ise polinomsal çekirdektir (4.35). Oluşturduğumuz yapay veriler üzerinde polinomsal çekirdeğin başarımlarını gözlemlediğimizde diğer çekirdek fonksiyonları gibi parametrelere bağımlılık göstermektedir. Şekil 4.7'de de gözüktüğü üzere genel parametre seçimlerinde yada akademik yazımda sıkça kullanılan parametreler seçildiğinde Yapay Veri A, B, D'yi kabul edilebilir hata oranları ile ayırabilirken Yapay Veri C'yi düzgün ayıramadığı gözlemlenebilmektedir.

$$\mathcal{K}(x_i, x_j) = (a.x_i x_j + b)^d \qquad (4.35)$$



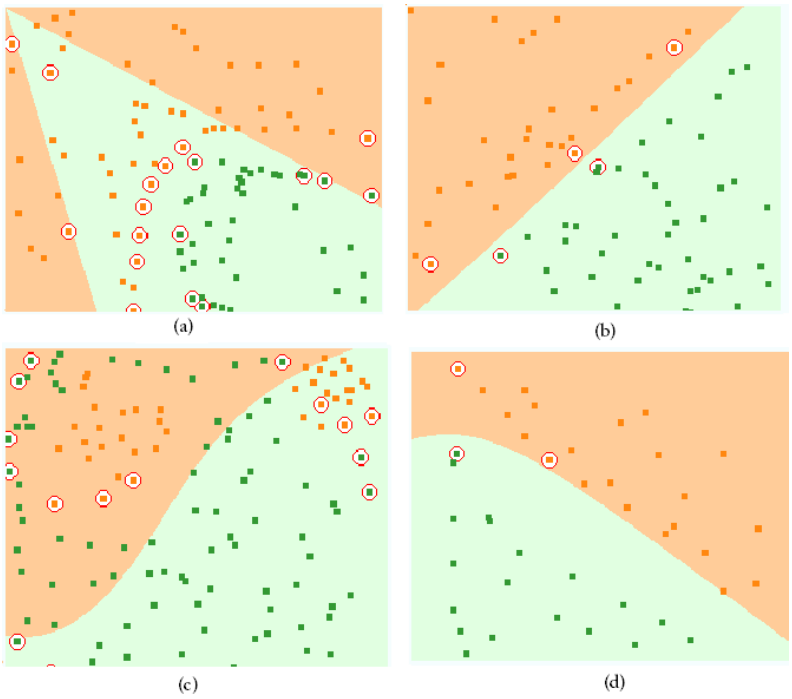
Şekil 4.7 Polinomsal çekirdek işlemi ile ayırma (*a=0.5, b=1.0, d=2*)

Polinomsal çekirdek fonksiyonunu biraz daha detaylı incelersek, özellikle var olan öznitelik uzayının nasıl daha yüksek boyutlu başka bir uzaya taşındığını daha iyi anlayacağız. Yapay Veri A kümesini sınıflandırırken derece parametresi *d*=2 için düzgün bir ayıraç olarak davranırken d>=3 için şekil anlamsız olmaktadır (Şekil 4.9a).

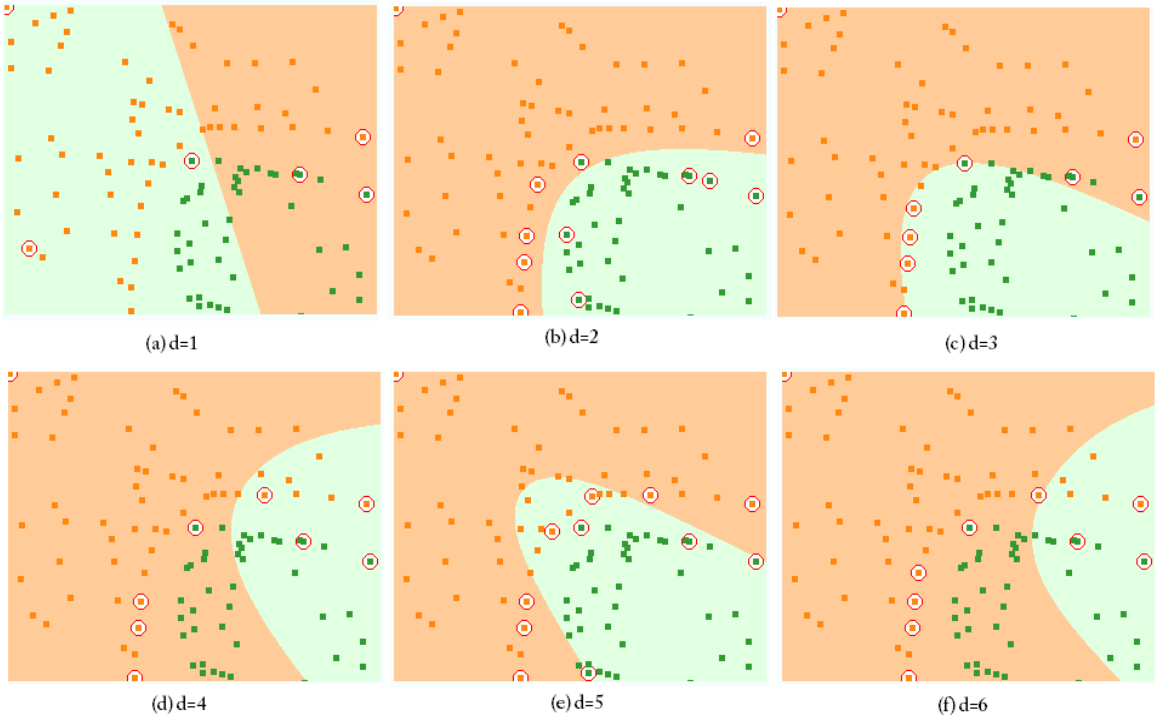


Şekil 4.8 Polinomsal çekirdek işlemi ile ayırma (*a=1.0, b=1.0, d=2*)

Yapay Veri B ve D için parametrelerin çok etikilemediği görülmektedir. Yapay Veri C yinede tam olarak ayrıştırılamamaktadır (Şekil 4.7c, Şekil 4.8c, Şekil 4.9c). Faklı polinomsal çekirdek değerleri için Şekil 4.10 da gözüktüğü üzere çok farklı davranabilmektedir.



Şekil 4.9 Polinomsal çekirdek işlemi ile ayırma (*a=2.0, b=3.0, d=3*)



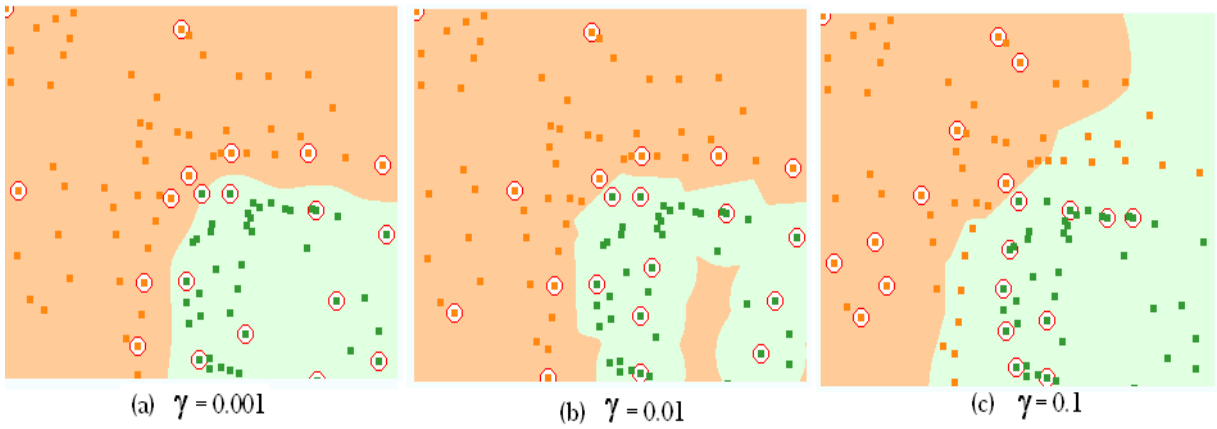
Şekil 4.10 Farklı polinomsal dereceler için aşırı düzlem (a=1, b=1)

4.2.3 Radyal Çekirdek

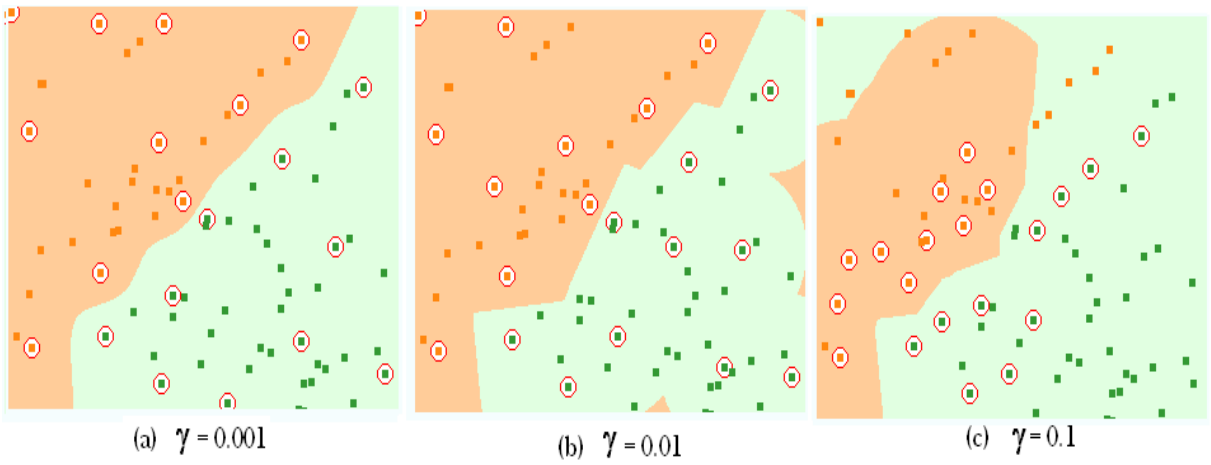
Destek vektör makinalarında en çok kullanılan çekirdeklerden bir başkası olan radyal tabanlı çekirdeğin başarımı uygun parametre seçimi durumunda diğer çekirdek fonksiyonlarına göre daha yüksektir. Radyal çekirdeği aşağıdaki gibi ifade edebiliriz (4.36).

$$\mathcal{K}(x_i, x_j) = e^{\left(-\frac{\|x_j - x_j\|^2}{\sigma^2}\right)} = e^{-\gamma\|x_j - x_j\|^2} \tag{4.36}$$

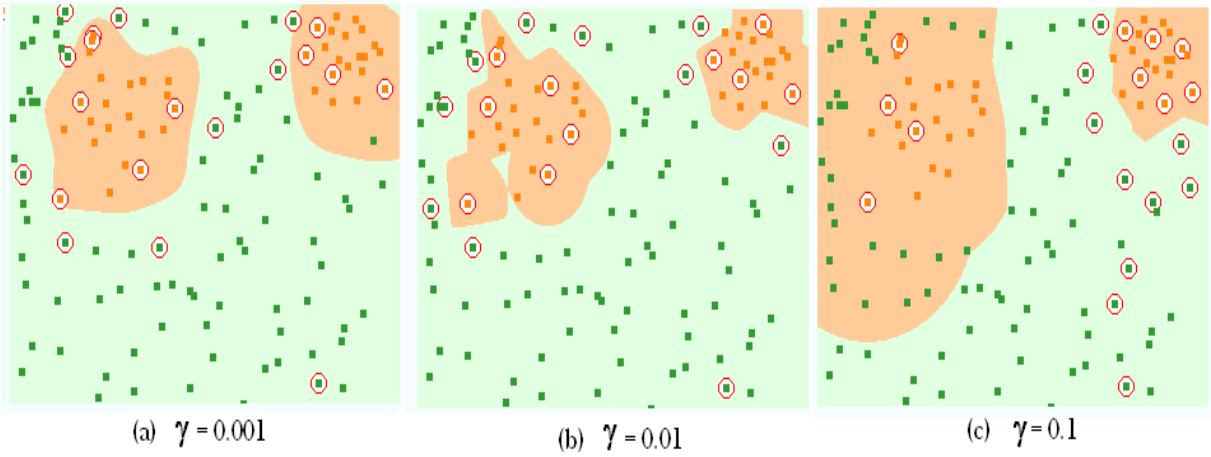
γ çarpanının çok büyük yada çok küçük olmasının algoritma üzerinde başarımı artıracağına dair olumlu bir örnek görülmemiştir. Aşağıdaki şekiller incelendiğinde



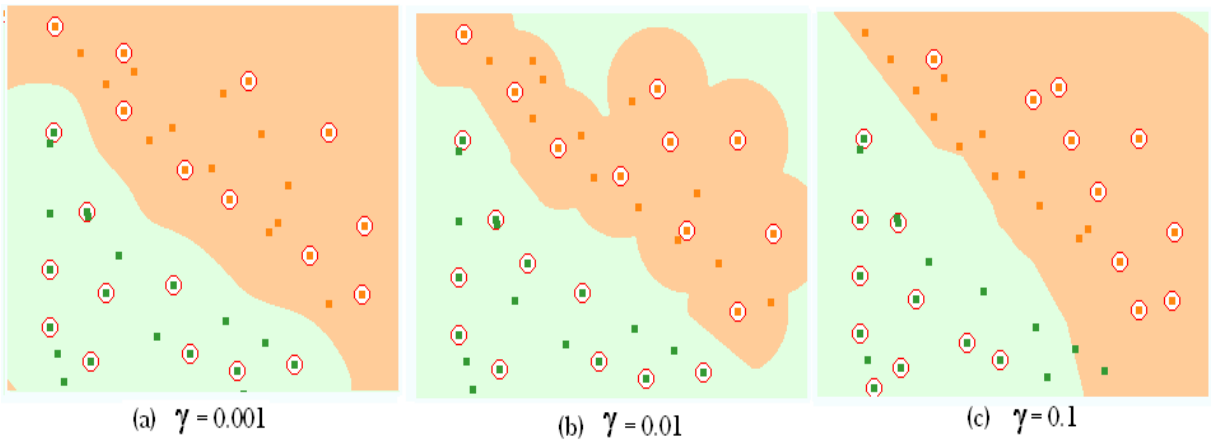
Şekil 4.11 Yapay Veri A üzerinde radyal tabanlı çekirdek işlemi ile ayırma



Şekil 4.12 Yapay Veri B üzerinde radyal tabanlı çekirdek işlemi ile ayırma



Şekil 4.13 Yapay Veri C üzerinde radyal tabanlı çekirdek işlemi ile ayırma



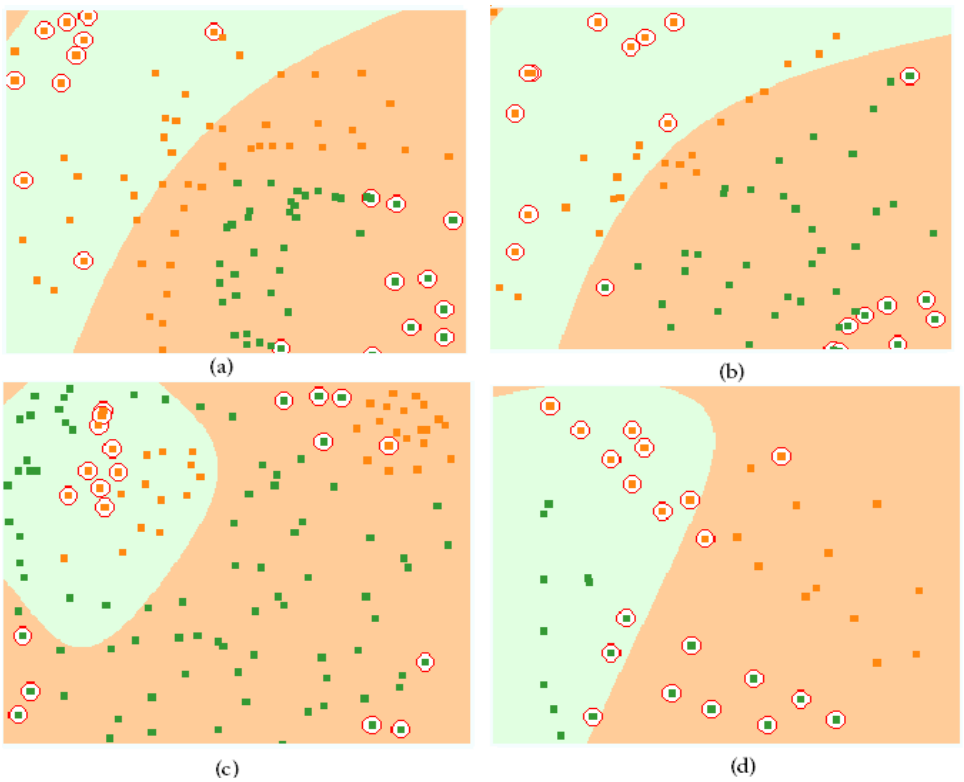
Şekil 4.14 Yapay Veri D üzerinde radyal tabanlı çekirdek işlemi ile ayırma

Radyal çekirdeğin sonuçlarını incelediğimizde görüleceği üzere küçük γ değerleri için fonksiyonun başarımı yüksek γ değerlerine göre daha iyi olduğu izlenmektedir. Bundan dolayı bir veri kümesini en iyi ayıran çekirdek aslında parametre seçimine doğrudan bağımlıdır.

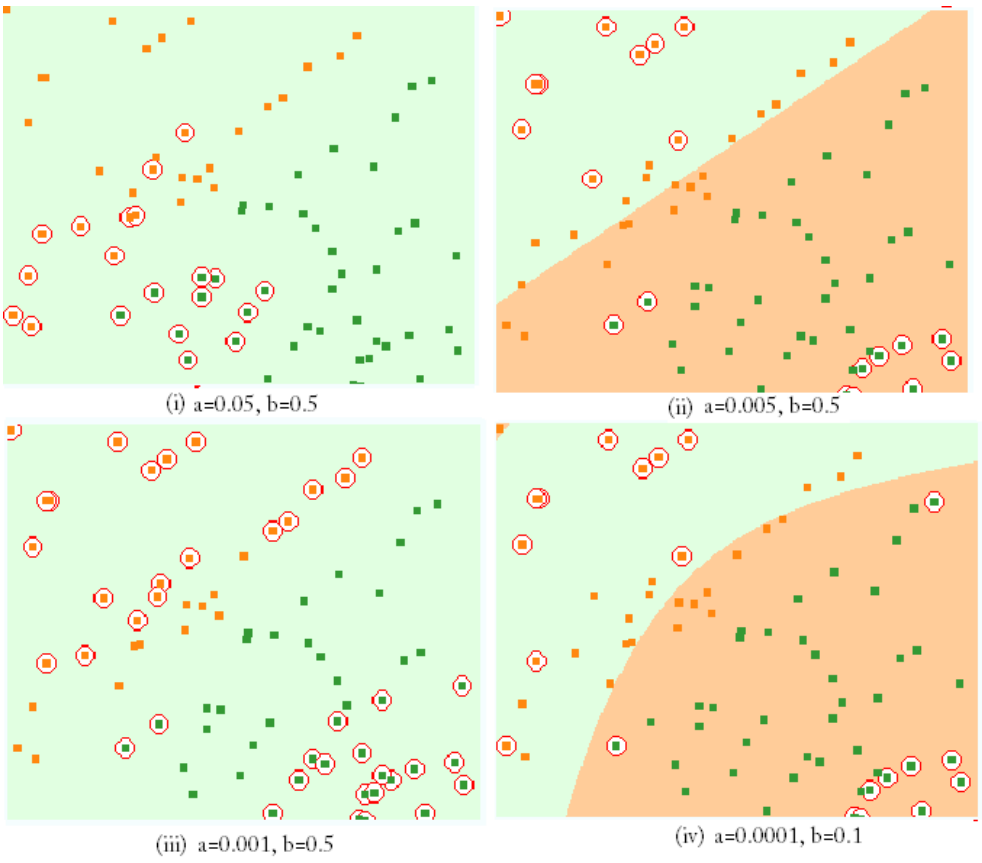
4.2.4 Sigmoid Çekirdek

Destek vektörlerin öğrenilmesi aşamasında sigmoid çekirdeğinin parametreleri çok etkin rol oynamaktadır. Çekirdek dönüşüm fonksiyonu tanh(...) şeklinde ifade edildiği için parametrelere daha da bağımlı haldedir.

$$\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = tanh(\gamma . \boldsymbol{x}_i{}^T \boldsymbol{x}_j + \mathrm{c}) \tag{4.37}$$



Şekil 4.15 Sigmoid çekirdek işlemi ile ayırma ($a = 1.0E-4, b = 0.01$)



Şekil 4.16 Sigmoid çekirdek işleminin "Veri B" üzerine sınıflandırma başarısı

Şekil 4.15'de standart bir parametre seçimi ile yapay ver kümeleri üzerindeki etkisini görmekteyiz. Daha detaylı olarak incelediğimizde parametrelerin veri kümesi üzerindeki eğitime etkisini görmek adına, Veri B (doğrusal ayrılabilen veri kümesi) üzerinde değişik parametreler uygulandı ve görüldü ki sınıflandırıcı çok farklı davranışlar sergileyebilmekte. Bundan dolayı en uygun parametre seçimi hala çekirdek metodların eksik kalan alanlarındandır. Bunun için akademk yazımda birçok makale yazılmakta ve yazılmaya da devam edecektir. Sonuçta teori bakımdan ele aldığınızda bile bu bir en iyileme problemidir.

4.2.5 Diğer Çekirdek İşlemleri

Akademik yazımda sıkça kullanılmayan fakat farklı çalışmalarda başarımlarını karşılaştırmak için kullanılan diğer çekirdek dönüşümlerini aşağıdaki gibi listeleyebiliriz :

- Üstsel Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|x_i - x_j\|}{2\sigma^2}\right)$
- Laplacian Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|x_i - x_j\|}{\sigma}\right)$
- ANOVA Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left(\sum_k \exp\left(-\sigma(\boldsymbol{x}_{i,k} - \boldsymbol{x}_{j,k})^2\right)\right)^d$
- Rasyonel İkinci Dereceden Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \frac{\|x_i - x_j\|^2}{\|x_i - x_j\|^2 + \theta}$
- İkinci Dereceden Çoklu Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sqrt{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 + \theta^2}$
- Ters İkinci Dereceden Çoklu Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{\sqrt{\|x_i - x_j\|^2 + \theta^2}}$
- Daire Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{2}{\pi}\cos^{-1}\left(-\frac{\|x_i - x_j\|}{\sigma}\right) - \frac{2}{\pi}\frac{\|x_i - x_j\|}{\sigma}\sqrt{1 - \left(\frac{\|x_i - x_j\|^2}{\sigma}\right)}$
- Küre Çekirdek: $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \frac{3}{2}\frac{\|x_i - x_j\|}{\sigma} + \frac{1}{2}\left(\frac{\|x_i - x_j\|}{\sigma}\right)^3$
- Dalga Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{\theta}{\|x_i - x_j\|}\sin\left(\frac{\|x_i - x_j\|}{\theta}\right)$
- Güç Çekirdeği : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = -\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^d$
- Log Çekirdek : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = -\log\left(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^d + 1\right)$
- Genelleştirilmiş T-Öğrenci Çekirdeği : $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{1 + \|x_i - x_j\|^d}$

- Spline Çekirdek :

$$\mathcal{K}(x_i, x_j) = 1 + x_i x_j + x_i x_j \min(x_i, x_j) - \frac{x_i + x_j}{2} \min(x_i, x_j)^2 + \frac{\min(x_i, x_j)^3}{3}$$

- Bayesian Çekirdek :

$$\mathcal{K}(x_i, x_j) = \prod_{m=1}^{N} \sum_{c \in \{0,1\}} P(Y = c | X^m = {x_i}^m) P\left(Y = c \middle| X^m = {x_j}^m\right)$$

- Histogram Kesişim Çekirdeği : $\mathcal{K}(x_i, x_j) = \sum_{i=1}^{n} min(x_i, x_j)$

- Chi-Square Çekirdek : $\mathcal{K}(x_i, x_j) = 1 - \sum_{i=1}^{n} \frac{(x_i - x_j)^2}{\frac{1}{2}(x_i - x_j)}$

- Cauchy Çekirdek : $\mathcal{K}(x_i, x_j) = \frac{1}{1 + \frac{\|x_i - x_j\|^2}{\sigma}}$

4.3 Dizgi Çekirdekleri

Herhangi iki karekter katarı yapısındaki, iki protein dizisi d_i ve d_j için benzerlik kriterini tek bir kesin sonucu olmamakla birlikte akademik yazımda günümüze kadar kullanılan bazı yöntem ve içerikler aşağıda vereceğiz. Bununla birlikte kendi geliştirmiş olduğumuz model ve yöntemleri de belirteceğiz.

Bir dizgi (dökümanlar, protein dizileri, DNA vb...) d ve $d \in \Sigma^*$ olmak üzere sonlu alfabe kümesi Σ için Kleene yakınsamasını $\Sigma^* := L_{k=0}^{\infty} \Sigma^k$ ile sembolize edebiliriz. Karakter katarı tipindeki bir yapının boyutunu da $l_x := |x|$ ile tanımlarız. Böyle bir durumda alfabe ve elemanlarını $\Sigma = \{A, .., Z, a, ..., a, 0, ..., 9, !, ?, ", ', \_, *, -, +, \#, \$, \%\}$, '_' boşluğu göstermesi kaydıyla tanımlamış oluruz. Buna göre aşağıdaki yapıları tanımlayabiliriz:

- Protein Birincil Yapıları, $\Sigma = \{A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V\}$
- Protein İkincil Yapıları, $\Sigma = \{A, H, L\}$ yada $\Sigma = \{G, H, I, T, E, B, S\}$ (DSSP formatında)
- DNA ve RNA, $\Sigma = \{A, C, G, T\}$
- Dökümanlar, $\Sigma = \{w_i\}, w_i \in W^N$, burda **W** : dökümanlarda bulunan kelimeler.

Yukarıdaki tanımlardan yola çıkarak, modellerimiz üzerinde iki farklı dizginin benzerliğini kullanan çekirdek makinalara, dizgi çekirdekleri denir. Bu tip çekirdek algoritmaları iki ana başlık altında inceleyebiliriz. İlki belirtilen dizgiler üzerine kurulu modellerden çıkarılan özellkleri kullanan yakınlık yada uzaklık çıkarımları. Diğeri ise doğrudan karakter katarları üzerine kurulu modellerdir. Bu modellerin bazılarını aşağıda detaylı inceleyeceğiz. Daha sonrada kendi geliştirmiş olduğumuz modelleri ve öğrenme yöntemlerinden bahsedeceğiz.

Standart doğrusal yada polinomsal çekirdekler direkt olarak dizi yapıları üzerinde kullanılamazlar. Ancak sayısal bir ifadeye dönüştürüldükten sonra yada ikili bir yapıda gösterilerek kullanılabilinir.

4.3.1 Kelime Torbası Çakirdeği (Bag-of-words Kernel)

Salton(1979) tarafından geliştirilen bu modele göre bir dökümanı sıralı olmayan kelimeler topluluğuyla belirtmek mümkün. Özellik uzayı $\boldsymbol{F}$ olası tüm kelimeler/özellik kümesini içermektedir ve bir döküman $\boldsymbol{x}$, seyrek vek törü $\boldsymbol{\phi(x)} \in F$ için dönüşümü yapılır.

$$\phi: d \rightarrow \phi(d) = \{tf(f_1, d), \dots, tf(f_N, d)\} \in R^N \tag{4.38}$$

yada

$$\phi: d \rightarrow \phi(d) = \{b(f_1, d), \dots, b(f_N, d)\} \in R^N \tag{4.39}$$

şeklinde tanımlanabilir. Burda $tf(f_i, d)$ $i.$ özellik için d dizgisindeki frekansı ve $b(f_i, d)$ de $i.$ özellik için d dizgisinin ikili dönüşüm fonksiyonudur. BOW çekirdeğini seyrek dönüşüm vektörlerinin $\boldsymbol{\phi(x)}$ iç çarpımı ile tanımlamak mümkündür (4.40)

$$\boldsymbol{\mathcal{K}}(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \sum_{w \in \boldsymbol{W}} \phi_w(x_i) . \phi_w(x_j) \tag{4.40}$$

Kullanılan bu çekirdek yapısı kelimelerin dizilişi ile ilgili herhangi bir bilgi içermemektedir. Özelliklerin yada kelimelerin bulunup bulunmaması bilgisini çekirdek bilgisi olarak sunmaktadır.

4.3.2 k-mer(k-gram) Çekirdek

Yukarıdaki modele çok benzer olan *k-mer* çekirdeğinde *k* adet ardı ardına gelen karakter katarının var olması temeline dayanır (Ching, 1979; Damashek, 1995). Bioinformatik bilim dalında *k-mer* şeklinde kullanılmasına rağmen bilgisayar bilimlerinde *k-gram* şeklinde ifade edilir. BOW algoritmasına benzer bir şekilde hesaplanır sadece olası tüm *k* uzunlukltaki karakter katarlarının bilgisini içerir. Sıralı olma durumu gözetilmeden sadece *k* uzunluklu katarın bulunup bulunmadığı yada frekansı yardımıyla çözüme gidilir. Kim ve Shawe-Taylor (1994) bu algoritmayı biraz daha geliştirip ağaç yapısında sorgulama yaparak hızlandırmışlardır.

Çizelge 4.2 d_1,d_2,ve d_3 için k-mer çekirdek bilgileri (küçük harf/ büyük harf duyarlı)

11	6	6
6	15	12
6	12	22

k=1

8	1	2
1	12	5
2	5	13

k=2

7	0	0
0	11	3
0	3	12

k=3

6	0	0
0	10	2
0	2	11

k=4

Çizelge 4.3 d_1,d_2,ve d_3 için k-mer çekirdek bilgileri (küçük harf/ büyük harf duyarsız)

13	12	11
12	21	17
11	17	28

k=1

8	4	5
4	12	5
5	5	13

k=2

7	3	3
3	11	3
3	3	12

k=3

6	2	2
2	10	2
2	2	11

k=4

Basit bir k-mer/n-gram örneği olarak d_1='Uğur AYAN', d_2 = 'Utku Efe Ayan' ve d_3= 'Ömer Eren Ayan' şeklinde üç dizi ele aldığımızda, k-mer'lik (*k=1,2,3,4*) çekirdek bilgileri aşağıdaki gibi oluşur (Çizelge 4.2 ve Çizelge 4.3). Burda 3x3 lük bir çekirdek elde edilmiş olur.

4.3.3 Spektrum Çekirdek

İlk olarak Leslie vd. (2002) tarafından biyolojik dizin analizi için tanımlanmıştır. İşleme sokulan herhangi iki dizginin k-mer'lerinin toplamı şeklinde ifade edilebilir. Σ alfabeyi, $d \in \Sigma^k$ bir k-merlik dizgiyi göstermek üzere ve #d(x) de k-mer'in frekansıdır. Buna göre çekirdek

$$\boldsymbol{\mathcal{K}}_k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \sum_{d \in \Sigma^k} \#d(x_i).\#d(x_j) \qquad (4.41)$$

şeklinde ifade edilir. $K_k(.)$ k-merlik bir spektrum dizgi çekirdek metodudur. Bu çekirdeği Leslie vd. (2002) düzgeleyerek $K_k{}^{Norm}(x_i, x_j) = K_k(x_i, x_j) / \left(\sqrt{K_k(x_i, x_i)} \sqrt{K_k(x_j, x_j)} \right)$ şekline getirmişlerdir. SCOP veritabanı için protein sınıflandırma problemlerinde k=3 ve k=4

değerleri en iyi sonuçları verdiği gözlemlenmiştir. Karşılaştırma amacıyla BLAST, PSI-Blast, HMM ve Fisher-SVM metodları kullanılmıştır. Spektrum çekirdek herhangi bir pozisyona ait bir bilgi yada özellik bünyesinde barındırmaz.

Yukarıda belirttiğimiz örneğe geri dönersek düzgelenmiş spectrum çekirdeği:

Çizelge 4.4 d_1,d_2,ve d_3 için spectrum *(k=1,2,3,4)* çekirdek bilgileri

1,0000	0,7263	0,5766
0,7263	1,0000	0,7011
0,5766	0,7011	1,0000

k=1

1,0000	0,4082	0,4564
0,4082	1,0000	0,3727
0,4564	0,3727	1,0000

k=2

1,0000	0,3419	0,3273
0,3419	1,0000	0,2611
0,3273	0,2611	1,0000

k=3

1,0000	0,2582	0,2462
0,2582	1,0000	0,1907
0,2462	0,1907	1,0000

k=4

gibi hesaplanır. Yukarıda tanımlanan spektrum çekirdek modelini *"Ağırlıklı Spektrum Çekirdek*" modeline genişletebiliriz. Leslie vd. (2002) bu işlemi doğrusal bir hesaplama ile yaparken hesap karmaşıklığı $O\left(d.l\left(d_i+d_j\right)\right)$ olur. Burda d k-mer'in boyutu ve $l\left(d_i+d_j\right)$ ise iki dizginin boyutları toplamıdır. Vishwanathan ve Smola (2003) ise sona ekleme ağaç yöntemi ile hesap karmaşıklığını $O\left(l\left(d_i+d_j\right)\right)$'na düşürmüşlerdir.

$$\boldsymbol{\mathcal{K}}\left(x_i,x_j\right)=\langle\phi(x_i),\phi\left(x_j\right)\rangle=\sum_i \alpha_i \boldsymbol{\mathcal{K}}_i^{Spektrum}\left(x_i,x_j\right) \tag{4.42}$$

burda $\alpha_i \geq 0$ olma şartı geçerlidir.

Leslie (2003b) gedikli spektrum çekirdek tanımını yapmış yani m kadarlık bir uzaklıkta eşleşen dizilerin bulunup bulunmamasına göre oluşturulan $K\left(x_i,x_j\right)=\langle\phi_m(\boldsymbol{x_i}),\phi_m\left(\boldsymbol{x_j}\right)\rangle$ için çekirdek tanımını $\phi_m(\boldsymbol{x_i})=\sum_{u\in x_i}\ \phi_m(\boldsymbol{u_i})$, ve

$$\phi_m(\boldsymbol{u})=\begin{cases}\phi_m(\boldsymbol{u})_{\sigma\in\Sigma^{\mathrm{d}}}\coloneqq 1 & \text{en çok m aralık için}\\ \phi_m(\boldsymbol{u})_{\sigma\in\Sigma^{\mathrm{d}}}\coloneqq 0 & \text{aksi durumda}\end{cases} \tag{4.43}$$

şeklinde yazılabilir. Bu çekirdek tanımını sadeleştirmiş olursak,

$$\boldsymbol{\mathcal{K}}\left(\boldsymbol{x_i},\boldsymbol{x_j}\right)=\sum_{\boldsymbol{u_i}\in\boldsymbol{x_i}}\sum_{\boldsymbol{u_j}\in\boldsymbol{x_j}}\Delta_{2\mathrm{m}}\left(\boldsymbol{u_i},\boldsymbol{u_j}\right) \tag{4.44}$$

biçiminde gösterebiliriz.

4.3.4 Altdizi Çekirdeği

Lodhi vd.(2002) tarafından ortaya konan bu kuram aslında ardı ardına gelen n-karakter (n-gram kernel) bilgisinin yanında herhangi altdizin bilgisinide çekirdek yapısına katan bir algoritmadır. $\pi \coloneqq (0,1)$ bozunum faktörü olmak üzere olası alt dizinlerin çarpımlarının toplamı şeklinde ifade edilebilir.

Çizelge 4.5 Örnek bir altdizi tablosu

	C-A	C-M	C-N	A-M	A-N	T-A	T-N	T-M
$\phi(CAM)$	π^2	π^3	0	π^2	0	0	0	0
$\phi(CAN)$	π^2	0	π^3	0	π^2	0	0	0
$\phi(TAN)$	0	0	0	0	π^2	π^2	π^3	0
$\phi(TAM)$	0	0	0	π^2	0	π^2	0	π^3

Örnek bir 4 adet 3 karakterlik bir dizin çekirdek hesaplamalarına bakıldığında (Çizelge 4.5) düzgeleme yapılmadan önceki çekirdek çarpımı $\boldsymbol{\mathcal{K}}(d_{CAM}, d_{CAN}) = \pi^4$, $\boldsymbol{\mathcal{K}}(d_{CAM}, d_{CAM}) = \boldsymbol{\mathcal{K}}(d_{CAN}, d_{CAN}) = 2\pi^4 + \pi^6$ olur, düzgeleme yapıldıktan sonra çekirdek çarpımımız $\boldsymbol{\mathcal{K}}(d_{CAM}, d_{CAN}) = \pi^4/(2\pi^4 + \pi^6) = 1/(2 + \pi^2)$ şeklinde bulunmuş olur. Görüleceği üzere tek bir 3 karakterli bir dizi için yapılan işlem karmaşıklığı bile yüksek boyuttadır. Bir kelimeler topluluğunu ele aldığımızda bu biraz daha karmaşık hale gelecektir. Bunun için tanım aşağıdaki gibi yapılmalıdır,

$$\begin{aligned}\boldsymbol{\mathcal{K}}_n(x_i, x_j) &= \sum_{m \in \Sigma^n} \langle \phi_m(\boldsymbol{x_i}), \phi_m(\boldsymbol{x_j}) \rangle = \sum_{m \in \Sigma^n} \sum_{p:m=\boldsymbol{x_i}[\boldsymbol{p}]} \pi^{l(p)} \sum_{r:m=\boldsymbol{x_j}[\boldsymbol{r}]} \pi^{l(r)} \\ &= \sum_{m \in \Sigma^n} \sum_{p:m=\boldsymbol{x_i}[\boldsymbol{p}]} \sum_{r:m=\boldsymbol{x_j}[\boldsymbol{r}]} \pi^{l(p)+l(r)}\end{aligned} \tag{4.45}$$

Bu çerkidek yapısının özellik uzayındaki etkisini düşürmek için düzgelemeyi,

$$\widehat{K}(x_i, x_j) = \langle \phi_m(\boldsymbol{x_i}), \phi_m(\boldsymbol{x_j}) \rangle = \langle \frac{\phi_m(\boldsymbol{x_i})}{\|\phi_m(\boldsymbol{x_i})\|} \cdot \frac{\phi_m(\boldsymbol{x_j})}{\|\phi_m(\boldsymbol{x_j})\|} \rangle = \frac{\boldsymbol{\mathcal{K}}(x_i, x_j)}{\sqrt{\boldsymbol{\mathcal{K}}(x_i, x_i)\boldsymbol{\mathcal{K}}(x_j, x_j)}} \tag{4.46}$$

şeklinde gösterebiliriz.

4.3.5 Ağırlıklandırılmış Dereceli Çekirdek

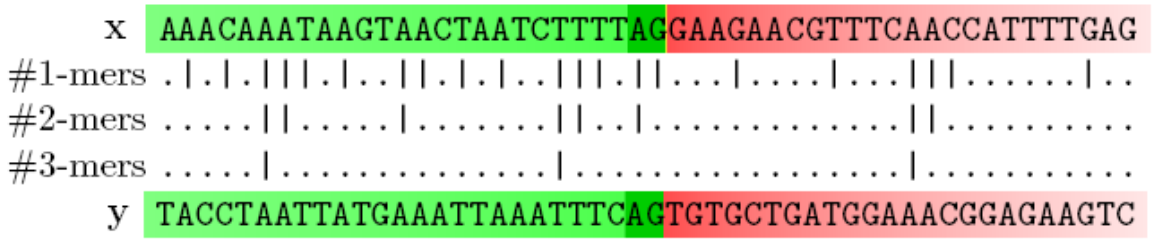
Ratsch ve Sonnenburg (2004) tarafından önerilen bu yapıda sıralı öğelerin pozisyon bilgisi de dahil edilerek daha etkili bir yöntem geliştirilmiştir. Herhangi iki dizi için *d*. dereceden *k-mer* bilgisi ile w_k ağırlık bilgisi $k = \{1, \dots, d\}$ için,

$$w_k = 2\frac{d-k+1}{d(d+1)} \tag{4.47}$$

şeklinde hesaplanır. Çekirdek çarpımını ise,

$$\mathcal{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{k=1}^{d} w_k \sum_{l=1}^{l-k+1} \Omega\left(s_{k,l}(\boldsymbol{x_i}) = s_{k,l}(\boldsymbol{x_j})\right) \tag{4.48}$$

ile gösterilir. Burda $s_{k,l}(.)$ l de başlayan k uzunluktaki bir sıralı katarı, $\Omega(.)$ ise durum doğru ise "1" yada değilse "0" bilgisini döndüren bir dönüşüm metodudur. Şekil 4.17'de gösterilen örnekte 3.dereceden bir çekirdeğin formulasyonu ifade edilmektedir { $k(\boldsymbol{x_i}, \boldsymbol{x_j}) = w_1.21 + w_2.8 + w_3.3$ }



Şekil 4.17 3. dereceden bir çekirdek örneği *

4.3.6 Gediklenmiş Dereceli Çekirdek

Önerdiğimiz bu modelde ise yukarıda tanımlanmış olan diğer metodlardan ayrı olarak normalde herhangi iki dizinin boyutlarının çok az farkla birbirine yakın olduğu düşünülmekte. Fakat bizim uygulayacağımız problemlerde herhangi iki protein dizisinin boyutları birbirinden çok farklı olabilir. Bundan dolayı önerilen ifadenin yerine kaydırılmış dereceli çekirdek metodu protein fonksiyon yada biolojik işlev öngörüsünde daha iyi çalışmaktadır.

$$\mathcal{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{k=1}^{d} w_k \sum_{l=1}^{l^*-k+1} \rho_l \sum_{\substack{s=0,\\ s+l\leq l^*}}^{S(l)} \mu_s \gamma(\boldsymbol{x_i}, \boldsymbol{x_j}) \tag{4.49}$$

* Ratsch ve Sonnenburg (2004)

$$\gamma(\boldsymbol{x_i}, \boldsymbol{x_j}) = \Omega\left(u_{k,l+s}(\boldsymbol{x_i}) = u_{k,l}(\boldsymbol{x_j})\right) + \Omega\left(u_{k,l}(\boldsymbol{x_i}) = u_{k,l+s}(\boldsymbol{x_j})\right) \quad (4.50)$$

4.3.7 Gedikli Ağırlıklandırılmış Dereceli Çekirdek

Burda ise önerilen modelde her bir dereceli sıralı alt dizin içinde bir gedik (gap) olma durumunun da eklenmesi ile oluşan çekirdek çarpımıdır.

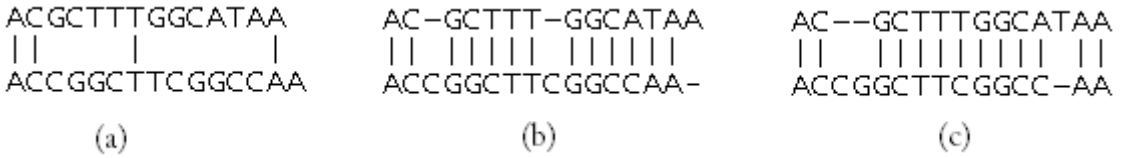
$$\mathcal{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{k=1}^{d} \sum_{g=0}^{G} w_{k,g} \sum_{l=1}^{l-k+1} \Omega\left(s_{k,l}(\boldsymbol{x_i}) \neq_g s_{k,l}(\boldsymbol{x_j})\right) \quad (4.51)$$

4.3.8 Gedikli Eniyileme Dereceli Çekirdek

Yukarıdakilerden farklı olarak öncelikle herhangi iki dizi üzerinde en uzun dizi bulan $\Omega(.)$ dönüşüm metodu oluşturulur ek olarak oluşturulan bu en uzun dizi üzerinde belirli bir hata katsayısı eklenerek olası elde edilebilecek en uzun sıralı dizinin boyutunu veren bir çekirdek çarpım metodudur.

$$\mathcal{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \max\left(\sum_{k=1}^{d} \sum_{l=1}^{L} \sum_{g=0}^{G} C(g)_{:=g} \Omega\left(s_{k,l}(\boldsymbol{x_i}) = s_{k,l}(\boldsymbol{x_j})\right)\right) \quad (4.52)$$

Burda *C(.)* gedik olan durumlar için (-) bir ceza parametresi aksi durumda ise bir genel hizalama algoritmasında olduğu gibi ağırlık parametresi bilgisini verir. Şekil 4.18'de gedikli olarak dizili iki dizinin eşleşme durumları gösterilmektedir.



(a) (b) (c)

Şekil 4.18 (a) İki dizinin kaydırılmamış durumdaki eşleşmesi, (b,c) gediklenmiş olan iki dizin eşleşme durumları

Şekil dikkatli incelendiğinde burdaki hizalama yapılan modele göre farklı k-mer skorları oluşabilmektedir. Bu gedikleme işlemi için akademik yazımdaki birkaç hizalama modelini (ClustarW, psi-BLAST, Genel Hizalama, Yerel Hizalama ve Hybrid Hizalama gibi...) kullanıp ortalama skor alma, yöntemin hata oranını düşürecektir.

4.3.9 Modellerin Başarımı

Kullandığımız veri kümesi olarak PDB'den alınan Biyolojik işlev ailesi, moleküler fonksiyon aile grubu ile Enzim aile grubuna ait en yüksek protein sayısına sahip üst sınıfları ele aldık. Biyolojik aktivite ailesinden hücresel işlev ile metabolik aktivite sınıfları, enzimler grubundan hidroliz ile taşıyıcılar ve son olarak da katalitik aktivite ve bağlayıcı moleküler fonksiyonlar sınıflandırmak için ele alındı. Sınıflandırıcı olarak kullanılan tüm çekirdek modellerinde %60 eğitim verisi %40 da test verisi olarak düşünüldü. Fakat protein yapılarında sıkça gözlemlenen benzer özellik içeren yapıların çokca bulunması eğitim verisinde doğruluğu yüksek olarak hesaplanmasına sebep olurken test aşamasında doğruluk oranının çok düşük olmasına sebep olmaktadır. Buna sebebiyet vermemek için benzer protein yapılarını eleme işlemi yapıldı.

Çizelge 4.6 Protein birincil yapısı üzerinde dizi çekirdeklerin başarımları

		Protein Miktarı	Çekirdek Modelleri						
			K-mer	AltDizi (Lodhi vd. 2002)	Ağırlıklı (Ratsch ve Sonn., 2004)	Gedikli Dereceli Çekirdek	Gedikli Ağılıklı Dereceli	Gedikli En İyileme Dereceli	BLAST
Biyolojik Aktiviteler	Hücresel İşlev	Tümü(21.022)	64,0%	66,3%	64,9%	67,5%	64,5%	71,9%	69,1%
		%90(10.938)	67,2%	74,4%	71,4%	68,7%	65,2%	75,8%	70,9%
		%70(9.346)	67,5%	69,2%	71,0%	69,4%	69,9%	75,4%	73,5%
		%40(9.078)	71,0%	77,6%	71,1%	78,6%	80,3%	78,0%	77,3%
	Metabolik Aktivite	Tümü (20.196)	66,1%	75,2%	67,2%	69,3%	67,5%	71,1%	73,6%
		%90 (6.478)	71,7%	79,2%	74,5%	75,4%	72,7%	75,6%	77,1%
		%70 (5.831)	68,7%	75,4%	71,9%	70,4%	70,9%	72,2%	81,1%
		%40 (4.311)	75,9%	78,2%	79,1%	77,8%	78,0%	76,9%	78,1%
Enzimler	Hidroliz Yapanlar	Tümü (11.692)	58,7%	62,9%	60,1%	63,9%	64,1%	60,7%	70,2%
		%90 (3.051)	61,3%	65,1%	64,4%	62,4%	61,2%	67,8%	67,1%
		%70 (2.806)	72,1%	79,8%	76,9%	74,9%	71,5%	78,1%	75,8%
		%40 (2.011)	78,4%	79,1%	79,8%	81,6%	80,3%	84,1%	88,9%
	Amine ve Fosfat Taşıyıcılar	Tümü (8.031)	65,1%	67,3%	69,1%	72,0%	68,2%	76,2%	66,6%
		%90 (3.081)	71,7%	76,1%	72,7%	77,6%	79,5%	83,4%	81,9%
		%70 (2.734)	68,7%	73,2%	71,4%	72,0%	72,1%	76,9%	71,3%
		%40 (2.518)	79,9%	81,3%	83,1%	81,2%	84,7%	83,4%	80,7%
Moleküler Fonksiyonlar	Bağlayıcılar	Tümü (32.107)	67,5%	59,3%	70,0%	72,3%	73,2%	76,9%	67,9%
		%90 (13.039)	74,3%	62,6%	77,5%	81,0%	78,5%	84,8%	79,7%
		%70(11.064)	71,5%	69,2%	73,2%	74,7%	71,8%	80,2%	73,8%
		%40(10.801)	83,6%	71,0%	86,2%	86,9%	89,7%	93,4%	87,0%
	Katalitik Aktivite	Tümü (27.150)	63,1%	66,2%	67,8%	70,1%	72,0%	75,0%	63,7%
		%90 (8.601)	76,4%	71,6%	78,8%	77,5%	75,9%	82,4%	77,8%
		%70 (7.470)	72,0%	78,1%	73,8%	72,5%	70,2%	81,1%	72,3%
		%40 (4.578)	81,7%	88,7%	84,6%	84,3%	84,5%	85,1%	87,8%

Başarımları karşılaştırmak ve gerçekten boyutu indirgenen veri kümesinin doğruluk oranı artıp artmadığını incelemek için; amino asit dizilişi %90, %70 ve %40 benzer olan proteinlerden sadece birisi örnek olarak alındı geri kalanlar ise elendi. Önerdiğimiz üç farklı dizi çekirdek modelini Çizelge 4.6'da detayları ile diğer modellerle karşılaştırma yaparak gösterdik. K-mer yönteminin genel olarak iyi bir sonuç vermediği altdizi çekirdeğinden diğer modellere göre başarımının biraz daha yüksek olduğu gözlemlendi. Çünkü bu modelde ele alınan tüm protein dizisi yerine tüm benzer alt yapılarının toplamı şeklinde bir öneri olmasından dolayı özellikle bizim yukarıda yapmaya çalıştığımız işlev tahmini gibi konularda sadece proteinin bu işle görevli aktif olan kısmının tespiti kolaylıkla yapılabilmektedir. Bizim önerdiğimiz modellerde ise gedikleme en iyileme modeli diğerlerine göre daha yüksek başarı oranı ortaya konmuştur. Bu modelde benzer yapılar arasında en yüksek olasılıklı kısmı almasından dolayı böyle bir sonucun çıkması beklenmekte idi. Yapısal bir sorgulama yapılsa idi o zaman bu modellerin başarısı diğerlerine göre yüksek çıkmayacaktır.

4.4 Çoklu Çekirdek Yöntemleri

Bir çok sınıflandırıcıda olduğu gibi çekirdek dönüşümlerinin de sınıflandırma başarısı verinin dağılımına bağımlılığı vardır. Cristianini ve Shawe-Taylor (2000), Pavlidis vd.(2001), Lanckriet vd. (2004b), Ben-Hur ve Noble(2005), Sonnenburg vd. (2006), Lin vd. (2009), Zhao vd.(2009), Kloft vd. (2010) yaptıkları çalışmalarda tek bir çekirdek dönüşüm fonksiyonu kullanmak yerine çoklu çekirdek fonksiyonu kullanmanın sınıflandırmanın başarısını artırdığını yaptıkları çalışmalarla ifade etmişlerdir. Özellikle Lanckriet vd.(2004a)'nin yapmış olduğu çalışmada protein yapılarını sınıflandırmak için kullanılan $\mathcal{K}_{sw}$(Smith-Waterman ölçüsü), $\mathcal{K}_B$(Blast), $\mathcal{K}_{Pfam}$(Pfam HMM ölçütü), $\mathcal{K}_{FFT}$(Hızlı Fourier Dönüşümü), $\mathcal{K}_{Ll}$ (Doğrusal Çekirdek), $\mathcal{K}_D$ (Difüzyon Çekirdeği), $\mathcal{K}_{RBF}$ (Radyal Çekirdek), $\mathcal{K}_{RND}$çekirdekleri kullanılarak dışbükey optimizasyon problemi şekline dönüştürerek çoklu çekirdek modeli tanımlanmış ve başarımı tekil çekirdek kullanımına göre arttığı hesaplanmıştır. Burda çoklu çekirdek kuramı

$$\mathcal{K}'(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{\mathrm{m}=1}^{\mathrm{M}} d_{\mathrm{m}}\, \mathcal{K}_m(\boldsymbol{x_i}, \boldsymbol{x_j}), d_{\mathrm{m}} \geq 0 \text{ ve } \sum_{\mathrm{m}=1}^{\mathrm{M}} d_{\mathrm{m}} = 1 \qquad (4.53)$$

şeklinde tanımlanmıştır. Sonnenburg vd. (2006) Lagrangian çözümünü oluştururken yarı-sonsuz doğrusal programlama yapmışlardır. Lanckriet vd. (2004a ve 2004b) ise yarı tanımlı

programlama ile çözüme ulaşırken Rakotomamonjy vd. (2007) yukarıdakilerden faklı olarak aşırı inişli problem şeklinde kullanmıştır. Burda türevlenebilir dönüşümü

$$J(d) = \max_{\alpha}\left\{-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j\sum_{\mathrm{m}=1}^{\mathrm{M}} d_{\mathrm{m}}\,\mathcal{K}_m(\boldsymbol{x_i},\boldsymbol{x_j}) + \sum_{\mathrm{i}}\alpha_i\right\} \tag{4.54}$$

şeklinde ifade edebiliriz. Bach vd. (2004) çoklu çekirdek problemini ardışık en az en iyileme problemine tekrar dönüştürmüştür. Diğerlerinden farklı olarak ayraç fonksiyonu ağırlıksız bir model ortaya koyarken eğitim aşamasında ağırlıklandırılmış bir öğrenim görme şansına sahiptir. Gönen ve Alpaydın (2008)'ın ortaya koyduğu kuramda ise Bach(2004)'ın ortaya koyduğu model üzerinde geçitleme fonskiyonu uygulayarak yerel uyarlamalı çoklu çekirdek yöntemi geliştirilmiştir. Bu geçitleme modeline göre geçitleme için kullanılacak ağırlık fonksiyonu aşağıdaki gibi belirtilmiştir,

$$\varsigma_m(\boldsymbol{x}) = \frac{exp(\langle v_m, x\rangle + v_{m0})}{\sum_{k=1}^{p} exp(\langle v_k, x\rangle + v_{k0})} \tag{4.55}$$

Kloft vd.(2010) çoklu çekirdek öğrenemeyi rastgele $lp(\ p \geq 1)$ normu için

$$maximize\ \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\left(\sum_{m=1}^{p}\left(\sum_{j=1}^{N}\sum_{k=1}^{N}\alpha_j\alpha_k y_j y_k k_m(x_j,x_k)\right)^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}} \tag{4.56}$$

optimizasyon problemine dönüştürmüştür. Çoklu çekirdekleri birleştirmek için ya sabit kurallar yada parametre tabanlı metodlar kullanılır. Sabit kurallar; Cristianini ve Shawe-Taylor (2000)'ın yapmış olduğu çalışmada olduğu gibi toplam yada çarpım şeklinde (4.57) veya Ben-Hur ve Noble (2005) de olduğu gibi "ikili çekirdeklerin" toplamı yada çarpımları şeklinde olabilir (4.58).

$$k_\mu(\boldsymbol{x_i},\boldsymbol{x_j}) = \sum_{n=1}^{N} k_n(\boldsymbol{x_i},\boldsymbol{x_j})\ \text{ yada } k_\mu(\boldsymbol{x_i},\boldsymbol{x_j}) = \prod_{n=1}^{N} k_n(\boldsymbol{x_i},\boldsymbol{x_j}) \tag{4.57}$$

$$k'_\mu\left(\left(x_i^a,x_j^a\right),\left(x_i^b,x_j^b\right)\right) = \sum_{n=1}^{N} k'_n\left(\left(x_i^a,x_i^b\right),\left(x_j^a,x_j^b\right)\right) + k'_n\left(\left(x_i^a,x_j^b\right),\left(x_j^a,x_i^b\right)\right) \tag{4.58}$$

Diğer bir çekirdek birleştirme yöntemi olarak parametre tabanlı fonksiyonlardır. Bunlarda doğrusal ve doğrusal olmayan fonksiyonlar olarak ayrılabilir. Lanckriet vd.(2002, 2004b) ve

Conforti ve Guido(2009)'nun yapmış olduğu çalışmalarda SMO ve konik eğri yöntemleri ile doğrusal birleştirmeye gidilmiştir. Kloft vd. (2010), Lin vd. (2010), Zhao vd.(2009), Qui ve Lane (2005), Tsuda vd. (2004), Lanckriet vd.(2004b), Fung vd. (2004)'ın çalışmalarında ise çekirdek dönüşümlerini doğrusal olarak birleştirirken kullanılan ağırlık parametrelerinin negatif olmaması temel motivsyon noktasıdır. Bunun haricinde KKT teoremi uygun olarak iç bükey doğrusal çekirdek birleşimini ise Xu vd. (2009a), Longworth ve Gales (2009, 2008), Zien ve Ong (2007), Kim vd. (2006), Bach vd. (2004) ve Bozquet ve Herrmann(2003) yaptıkları çalışmalarla ifade etmişlerdir. Bunun haricinde Xu vd. (2009b) ikili doğrusal birleştirme yöntemini kullanarak medikal veriler üzerinde başarımlarını göstermişlerdir. Doğrusal olmayan birleştirmeler ele alındığında ise benzerlik tabanlı, bayes tabanlı ve boosting yöntemi ile çekirdek birleşimi göze çarpmaktadır.

Lanckriet vd.(2002, 2004b) ortaya koyduğu kuramda kuralsal SVM çözümünden sonra birleştirilmiş çekirdek matrisi

$$\mathcal{K}_L = \left\{ \mathcal{K} : \mathcal{K} = \sum_{i=1}^{N} \mu_i \mathcal{K}_i \, , \mathcal{K} \geq 0, tr(\mathcal{K}) \leq c \right\} \tag{4.60}$$

şeklinde tanımlanarak seçilen çekirdeklerin pozitif yarı tanımlı olması zorunluluğu eklemiştir.

Chapelle vd. (2002) iç bükey birleştirme yöntemini baz alırken sınır ve kürenin yarıçaplarınn türevlerini θ paremetresine göre hesaplama yoluna gitmişlerdir (4.61).

$$\begin{aligned} \frac{\partial \|\boldsymbol{w}\|_2^2}{\partial \theta} &= -\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \frac{\partial k(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\partial \theta} \\ \frac{\partial R^2}{\partial \theta} &= \sum_{i=1}^{N} \beta_i \frac{\partial k(\boldsymbol{x}_i, \boldsymbol{x}_i)}{\partial \theta} - \sum_{i=1}^{N}\sum_{j=1}^{N} \beta_i \beta_j \frac{\partial k(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\partial \theta} \end{aligned} \tag{4.61}$$

α içbükey SVM en iyileme problemi ile ve β ise ikinci dereceden programlama ile çözülmektedir .

Bousquet ve Herrmann (2003) sınır için oluşturulan rampa fonksiyonunu tekrar düzenleyerek seçilen çekirdek fonskiyonunu çekirdek ağırlıklarına göre türevinin alarak oluşturmuşlardır.

$$\frac{\partial \|w_{\mu}\|_2^2}{\partial \mu_k} = -\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \frac{\partial k_{\mu}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\partial \mu_k} = -\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k_k(\boldsymbol{x}_i, \boldsymbol{x}_j) \, , \forall k \tag{4.62}$$

$\mathcal{K}_\mu$ çekirdeği eğitimi sırasında hesaplanan w_μ ağırlık vektörü olmak üzere standart SVM en iyileme probleminde rampa fonskiyonu hesaplanması sırasında negatif olmayan μ ler hesaplanmaktadır ve $\sum_{k=1}^{N} \mu_k = 1$ olacak şekilde düzgeleme yapılır.

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \; için \; \gamma \geq \underbrace{\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k_m(\boldsymbol{x}_i, \boldsymbol{x}_j) - \sum_{i=1}^{N} \alpha_i}_{S_m(\alpha)} \; , 0 \leq \alpha_i \leq C, \forall m \qquad (4.63)$$

Bach vd.(2004) ile Sonnenburg vd. (2006a, 2006b) çoklu çekirdek modelini γ'yı en azlayacak ve $\gamma \in \mathbb{R}$ ve $\alpha \in \mathbb{R}_+^N$ olacak şekilde QCQP üzerinde kurulu olarak geliştirmişlerdir (4.63). Problemi SILP en iyileme problemine dönüştürerek θ'yi ençoklayacak $\{\theta \in \mathbb{R}$ ve $\mu \in \mathbb{R}_+^p\}$ şekilde çözüme gitmişlerdir (4.64).

$$\sum_{k=1}^{p} \mu_k = 1 \; için \; \sum_{k=1}^{p} \mu_k S_m(\boldsymbol{\alpha}) \geq \theta \qquad (4.64)$$

Rakotomamonjy vd. (2008) ile Longworth ve Gales (2008, 2009) ise oluşturmuş oldukları SimpleMKL iskelet yapısında ek düzenleme sabiti (4.65) eklemişlerdir. Böylece eklenen bu terim çoklu çekirdekler için seyreklik düzengeci olarak işlev görmektedir.

$$\lambda \sum_{k=1}^{P} \left(\mu_m - \frac{1}{p}\right)^2 = \lambda \left(\sum_{k=1}^{P} {\mu_m}^2 - \frac{1}{p}\right) = {}^{+}\lambda \sum_{k=1}^{P} {\mu_m}^2 \qquad (4.65)$$

Sonuç olarak kendi oluşturduğumuz Yapay Veri C kümesi üzerinde başarımlarını incelediğimizde; çoklu çekirdek öğrenimi aynı parametreler kullanıldığı taktirde tekil çekirdeklere göre daha yüksek oranlarda başarım ve daha düşük oranlarda standart sapma oranları ortaya koymaktadır. Yukarıdaki tüm çoklu çekirdekleri ana yöntemler altında topladığımızda ortalama, çarpım, doğrusal, ikili, iç bükey, konik, oransal/ağırlıksal, yinelemeli şeklinde ele alabiliriz. SimpleMKL, SVM-MK ve LMKL araçlarının yardımıyla bu yöntemlerin başarımlarını incelediğimizde çok yüksek oranlarda fark olmamasına rağmen uygun parametreler seçildiğinde başarım oranları dahada arttığı görülecektir.

Çizelge 4.7 Yapay Veri C üzerinde tekil ve çoklu çekirdek başarımları

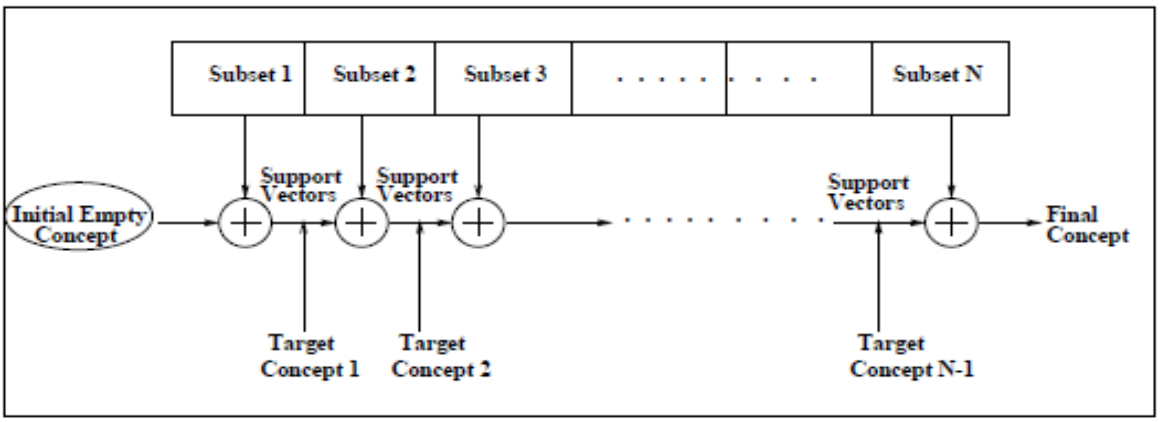
	Doğruluk Oranı	Destek Yöneyleri	Test Doğruluğu
Doğrusal Çekirdek	81.21±0.91	33.21±6.35	80.71±1.35
İkinci Dereceden Çekirdek	85.45±1.43	27.45±7.43	81.65±0.43
Polinomsal (d=3)	87.12±1.37	78.45±3.43	91.45±0.74

Polinomsal (d=4)	82.66±1.23	61.45±6.23	87.13±1.43
Radyal Çekirdek (g= 0.5)	90.24±1.78	95.45±1.01	94.41±1.43
Radyal Çekirdek (g= 1)	91.23±1.47	99.06±0.45	96.57±1.71
Radyal Çekirdek (g= 10)	92.34±1.63	97.89±0.89	96.76±0.11
ÇÇ(Ortalama)	92.23± 1.55	41.90± 3.32	94.43± 0.85
ÇÇ (Çarpım)	90.08± 1.15	99.28± 0.13	93.14± 1.11
ÇÇ (İç Bikey)	96.34± 1.43	41.45±2.43	96.35± 0.22
ÇÇ (Oransal)	89.13± 1.81	35.28± 4.45	90.97± 1.07
ÇÇ (Konik)	95.38± 1.01	43.28± 1.15	95.99± 0.33
ÇÇ (Yinelemeli)	94.11± 1.05	77.65± 6.15	95.09± 1.06
ÇÇ (Ağırlıksal)	91.45± 1.45	32.28± 4.15	93.12±0.99

Farklı Çoklu Çekirdek(ÇÇ) öğrenme modellerinin genel olarak tekil çekirdek modellere göre doğruluk oranlarının daha yüksek olduğu farkedilmektedir. Detaylı olarak ele alınan ilaç ve kanserli verilerde (Bölüm 4.8) doğruluk oranlarının düzenleştirme sabitine, polinomsal çekirdeklerin derecesine ve radyal tabanlı parametrelere göre farklılık göstereceği gözlemlenecektir.

4.5 Artımlı ve Azalımlı Çekirdek Öğrenme Modeli

Artımlı eğitim yada öğrenme ile ifade edilmek istenen çekirdek makinelerinde her yeni gelen veri kümesi üzerinde bir en iyileme sonucu olarak elde edilmesi beklenen destek vektörlerini elimizde tutup eğitime giren ve destek vektörleri olarak görülmeyen diğer verileri ise sonraki öğrenme adımları için ele almamak şeklinde söylenebilir (Syed vd., 1999).



Şekil 4.19 Artımlı eğitim süreci*

* (Syed vd., 1999)

Yukarda tanımlanan metodolojiye göre örnek tabanlı artımlı öğrenme iskeleti oluşturulmuş ve tüm örnekler yerine alt kümelere (ES_i) bölünerek sırasıyla yeni gelen örnek küme üzerinde bir öğrenme önerilmiştir. Böylece elde edilen destek vektörler saklanmıştır (DV_i). Geriye kalan örnekler gözardı edilmiştir çünkü DVM modeline göre bu tür sınıflandırıcılar için önemli olan Lagrangian çarpanı "0" olmayan destek vektörleridir, sınıflandırma işlevi bu vektörler üzerinden yürütülür. Bir sonraki veri kümesinde öğrenme adımı bir önceki eğitim kümesinin destek vektörleri ile birlikte ele alınır. Böyle bir iskelet yüksek boyutlu ortamlar yada hafıza yetersizliği problemi olan problemlerde uygun ve başarımı yüksek bir çalışma olarak ele alınmaktadır.

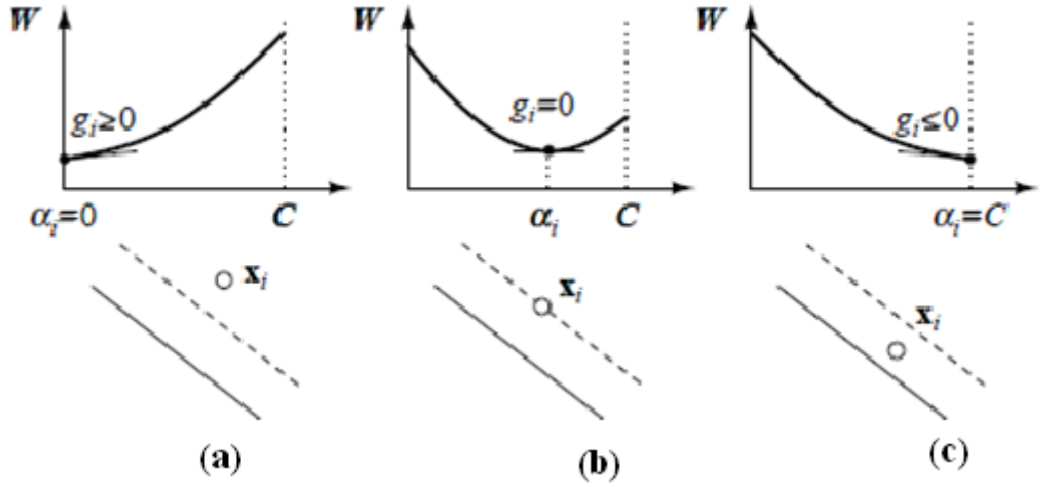
Galmeanu ve Andonie (2008)'in yapmış olduğu çalışmada artımlı ve azalımlı çekirdek öğrenmenin alt yapısını detaylı olarak ele almışlar ve başarımlarını diğer çekirdek öğrenme modelleriyle karşılaştırmışlardır. Syed vd. (1999) tarafından önerilen model için Kuhn-Tucker(KT) tüm eğitilen veri kümesi için sağlama şartı aranması gerektiğini vurgulanmış ve şöyle ifade edilmiştir;

$$\min_{0\le\alpha_i\le C} W = \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i Q_{ij}\alpha_j - \sum_{i=1}^{N}\alpha_i + b\sum_{i=1}^{N} y_i\alpha_i \tag{4.66}$$

burda α_i ler Lagrangian çarpanları olmak üzere, çekirdek $Q_{ij} = y_i y_j\langle\Phi(x_i),\Phi(x_j)\rangle$ şeklinde ifade edilir ve birinci dereceden (4.67)'yi sağlaması gerekir,

$$g_i = \frac{\partial W}{\partial\alpha_i} = \sum_{j=1}^{N} Q_{ij}\alpha_j + y_i b - 1 \begin{cases} < 0 & \alpha_i = C \\ = 0 & 0 \le \alpha_i \le C \\ > 0 & \alpha_i = 0 \end{cases}$$
$$h = \frac{\partial W}{\partial b} = \sum_{j=1}^{N} y_j\alpha_j \equiv 0 \tag{4.67}$$

Standart SVM eğitim sürecinde g_i ve α_i değerleri destek yöneylerini ve hata yöneylerini oluşmasında sebep olur. Şekil 4.20'de SVM eğitim sürecinde Langrangian çarpanlarının sonucuna göre oluşan destek yöneylerini (b) ve oluşan hata yöneylerini görmekteyiz(c).



Şekil 4.20 Artımlı eğitim süreci

Eğitim verilerini Şekil 4.20'de olduğu gibi temel olarak 3 ana kısıma ayırmamız mümkündür. Birinci grub eğitim kümesi $\{S\}$ Sınır destek vektörleri ($g_i = 0$), ikinci grub eğitim kümesi $\{\mathcal{E}\}$ Error (Hata) destek vektörleri ($g_i \leq 0$) ve üçüncü küme eğitim kümesine ise $\{R\}$ Rezerve destek vektörleri ($g_i > 0$) denir. Artımlı öğrenme sırasında yeni gelen tüm eğitim verilerinden $g_i > 0$ olanların hepsi bu gruba atanırlar. Diğer tüm yeni eğitim örnekleri {Ö} öğrenilmemiş yöneyler olarak eğitime sokulurlar.

Parçalı türevleri alınan denklemler,

$$\begin{aligned} g_i &= \sum_j Q_{ij}\alpha_j + \sum_{k\in S} Q_{ik}\Delta\alpha_k + \sum_{l\in Ö} Q_{il}\Delta\alpha_l + y_i(b+\Delta b) - 1 = 0, \forall i \in S \\ h &= \sum_j y_j\alpha_j + \sum_{k\in S} y_k\Delta\alpha_k + \sum_{l\in Ö} y_l\Delta\alpha_l = 0 \end{aligned} \tag{4.68}$$

şeklini alırlar. Daha da açık halde artımlar ifade edilirse,

$$\begin{aligned} \Delta g_i &= \sum_{k\in S} Q_{ik}\Delta\alpha_k + \sum_{l\in Ö} Q_{il}\Delta\alpha_l + y_i\Delta b = 0, \forall i \in S \\ \Delta h &= \sum_{k\in S} y_k\Delta\alpha_k + \sum_{l\in Ö} y_l\Delta\alpha_l = 0 \end{aligned} \tag{4.69}$$

burda $\{\Delta\alpha_l : \forall l \in Ö\}$ öğrenilmemiş yöneylerin Lagrangian katsayılarını, $\{\Delta\alpha_k : \forall k \in S\}$ sınır yöneylerin Lagrangian katsayılarını ve Δb ise KKT durumlarını sağlayacak dizgisel sapmayı belirtmektedir. Öğrenilmemiş yöney katsayılarını aslında sarsım katsayılarının çarpımları şeklinde yazabiliriz. Sarsım katsayısı $p: p \in [0,1]$. Başlangıçta p=0 için önceki durumu ifade eder ve yavaş Δp artımları ile sonuca gidilmeye çalışılır. p=1 olduğunda yukarıda belirtilen

üç temel kategoriden birine atanması beklenir. Katsayı duyarlılığı cinsinden sarsım artımlarını $\Delta\alpha_k = \beta_k \Delta p (k \in S),\ \Delta\alpha_l = \delta_l \Delta p (l \in \text{Ö}), \Delta b = \beta \Delta p$ şeklinde ifade edebiliriz. Tüm Lagrangian katsayıları sarsım artımlarıyla hesaplanabilir. Sınır yöneyler için $\alpha_k \rightarrow \alpha_k + \beta_k \Delta p, \forall k \in S$ ve öğrenilmemiş yöneyler için $\alpha_l \rightarrow \alpha_l + \delta_l \Delta p, \forall k \in \text{Ö}$ şeklindedir.Başlangıç adımı olarak $\alpha_l = 0\ ve\ b = 0$ olarak alınır. Sarsım artımlarına bağlı olarak katsayı duyarlılığını ,

$$\gamma_i = \frac{\Delta g_i}{\Delta p} = \sum_{k \in S} Q_{ik}\beta_k + \sum_{l \in \text{Ö}} Q_{il}\delta_l + y_i\beta = 0, \forall i \in S$$
$$\frac{\Delta h}{\Delta p} = \sum_{k \in S} y_k\beta_k + \sum_{l \in \text{Ö}} y_l\delta_l = 0 \qquad (4.70)$$

şeklinde yazabiliriz. Sarsım katsayılarından δ_l rastgele istenen bir sayı seçilecek şekilde tanımlanabilir. Yukarıdaki ifadeye sadeleştirip matris şeklinde çevirdiğimizde çözüm $\boldsymbol{Q}\beta = -\sum_{l \in \text{Ö}} \delta_l v_l$ şeklini alır böylece;

$$\boldsymbol{\beta} = \begin{bmatrix} \beta \\ \beta_{s1} \\ \vdots \\ \beta_{sn} \end{bmatrix}, \boldsymbol{v}_l = \begin{bmatrix} y_l \\ Q_{s_1 l} \\ \vdots \\ Q_{s_n l} \end{bmatrix} ve\ \boldsymbol{Q} = \begin{bmatrix} 0 & y_{s_1} & \cdots & y_{s_n} \\ y_{s_1} & Q_{s_1 s_1} & \cdots & Q_{s_1 s_n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{s_1} & Q_{s_n s_1} & \cdots & Q_{s_n s_n} \end{bmatrix} \qquad (4.71)$$

burda $ve\ \boldsymbol{R} = \boldsymbol{Q^{-1}}$ şeklinde tanımlarsak $\boldsymbol{\beta} = \sum_{l \in \text{Ö}} \delta_l \boldsymbol{R} v_l$ şeklinde çözüme gidilebilir. Sınır yöneylere ekleme yapıldığında $\boldsymbol{R}$ tekrar hesaplanarak,

$$\boldsymbol{R} \leftarrow \begin{bmatrix} \boldsymbol{R} & \begin{matrix} 0 \\ \vdots \end{matrix} \\ \begin{matrix} 0 & \cdots \end{matrix} & 0 \end{bmatrix} + \frac{1}{\gamma_{s_{n+1}}} \begin{bmatrix} \beta_{s_{n+1}} \\ 1 \end{bmatrix} \begin{bmatrix} \beta_{s_{n+1}} \\ 1 \end{bmatrix}^T, \beta_{s_{n+1}} = -\boldsymbol{R} v_{s_{n+1}} ve\ \gamma_{s_{n+1}} = Q_{s_{n+1}} + v_{s_{n+1}}^T \beta_{s_{n+1}} \qquad (4.72)$$

şeklinde düzenlenir. Çizelge 4.8'de artımlı öğrenme modeline göre hesaplanacak sarsım parametreleri ve kategorilendirme bilgileri detaylı olarak listelenmiştir.

Çizelge 4.8 Artımlı Öğrenme Model Parametreleri*

İlk Kategori	Yeni Kategori	Sarsım Artışı (Δp)	Kısıtlar ve Durumlar
Rezerve Yöneyler	Sınır Yöneyler	$-g_i/\gamma_i$	$\gamma_i < 0$
Sınır Yöneyler	Rezerve Yöneyler	$-\alpha_i/\beta_i$	$\beta_i < 0$
Hata Yöneyler	Sınır Yöneyler	$-g_i/\gamma_i$	$\gamma_i > 0$
	Artımlı Öğrenme		
Sınır Yöneyler	Hata Yöneyler	$(C - \alpha_i)/\beta_i$	$\beta_i < 0$
Öğrenilmemiş ($g_i < 0$)	Hata Yöneyler	$(C - \alpha_i)/\delta_i$	$\delta_i > 0$

* Cauwenberghs ve Poggio, 2001; Diehl ve Poggio, 2003; Romero vd., 2007; Galmeanu ve Andonie, 2008.

Öğrenilmemiş ($g_i < 0$)	Sınır Yöneyler	$-g_i/\gamma_i$	$\gamma_i > 0$
	Düzenleme Sarsım Parametresi		
Sınır Yöneyler	Hata Yöneyler	$(C-\alpha_i)/(\beta_i - \Delta C)$	$\beta_i > \Delta C$
	Kernel Sarsım Parametresi		
Sınır Yöneyler	Hata Yöneyler	$(C-\alpha_i)/\beta_i$	$\beta_i > 0$
Öğrenilmemiş ($g_i \neq 0$)	Sınır Yöneyler	$-g_i/\gamma_i$	$\gamma_i g_i < 0$
Öğrenilmemiş ($g_i > 0$)	Rezerve Yöneyler	$-\alpha_i/\delta_i$	$\delta_i < 0$
Öğrenilmemiş ($g_i < 0$)	Hata Yöneyler	$(C-\alpha_i)/\delta_i$	$\delta_i > 0$

Çizelge 4.9'da ise öğrenilen modelin sonucunda çekirdek fonksiyonu sonucunda göre kategorilendirme bilgisi verilmiştir.

Çizelge 4.9 Çekirdek parametrelerine göre örnek kategorilendirme

Önceki Örnek Kategorisi	$g_i > 0$	$g_i < 0$
Rezerve Yöneyler ($\alpha_i = 0$)	Rezerve	Öğrenilmemiş
Sınır Yöneyler ($0 \leq \alpha_i \leq C$)	Öğrenilmemiş	Öğrenilmemiş
Hata Yöneyler ($\alpha_i = C$)	Öğrenilmemiş	Hata

Bir başka öneri olarak *SV-L* Artımlı öğrenme modelinde (Rüping, 2001) ise Cauwenberghs ve Poggio(2001)'in tanımlamış olduğu yapıya ek olarak *L* düzenleme parametresi ekleyerek eski destek vektörler üzerindeki deneysel hatayı düzenlemek için tekrar yapılandırmıştır.

$$\Phi(w,\xi,\xi^*) = \frac{1}{2}(w^T.w) + C\left(\sum_{i\in I}\xi_i + L\sum_{j\in S}\xi_j^*\right) \quad (4.73)$$

Uygulama yapılan 10 farklı veri kümesi üzerindeki başarımları tek düze SVM ve artımlı SVM'e göre %1 oranında daha iyi sonuçlar verdiği gözlemlenmiştir.

4.6 Artımlı ve Azalımlı Çoklu Çekirdek Öğrenme Modeli

Bu yaklaşımda diğer çoklu çekirdek sınıflandırıcılarda olduğu gibi tek bir çekirdeğin yetersiz kaldığı yada istenen sonuca ulaşılamadığı varsayılarak tek çekirdek yerine çoklu çekirdek kullanılması önerilmiştir. Bu çoklu çekirdek modelini çok yüksek boyutlar üzerinde öğrenme yapan veri kümeleri için artımlı öğrenme modeli ile birleştirerek yeni bir yaklaşım ele alınmıştır. (4.53)'deki eşitlikten yola çıkılarak çoklu çekirdek öğrenim modelini içbükey en

iyileme problemine dönüştürerek artımlı ve azalımlı çoklu çekirdek öğrenimi modeline dönüştürdük.

$$\min \sum_k \frac{1}{d_k} \|w_k\| + C \sum_i \xi_i$$
$$y_i \sum_k \Phi_k(x_i) + y_i b \geq 1 - \xi_i \,, \forall i \tag{4.74}$$
$$\xi_i \geq 0 \quad \forall i, \quad d_k \geq 0 \quad \forall k, \qquad \sum_k d_k = 1$$

Artımlı ve azalımlı çekirdek modelinin temelinde yatan fikirlerden birisi olan öğrenme probleminin içbükey eniyileme problemine dönüştürülmesi için çoklu çekirdek katsayılarının 0 ve üzeri ve toplamlarının 1 olma şartı korunmuştur. Diğer bir dikkate alınması gereken kural ise KKT durumlarının kontrolüdür. Tekrar Lagrangian fonksiyonu olarak formule edilirse,

$$\frac{1}{2}\sum_k \frac{1}{d_k} \|w_k\| + C \sum_i \xi_i - \sum_k \mu_k d_k - \sum_i v_i \xi_i$$
$$- \sum_i \alpha_i (y_i w_k \Phi_k(x_i) + y_i b - 1 + \xi_i) - \gamma \sum_k (d_k - 1) \tag{4.75}$$

burda α_i (4.74)'deki denklemlerdeki Langrangian çarpanları, v_i ve μ_k ise negatif olmayan gevşeklik parametrelerini(ξ_i) ve çoklu çekirdeklerin ağırlıklarını(d_k) düzenleyen Lagrangian çarpanlarıdır ve γ ise *l1*-norm Lagrange eşitlik kısıtıdır. Yöney şekilde KKT durumlarını ifade edersek,

$$g_i = \sum_j \sum_k d_k Q_{ij} \alpha_j + y_i b - 1 \begin{cases} < 0 & \alpha_i = C \\ = 0 & 0 \leq \alpha_i \\ > 0 & \alpha_i = 0 \end{cases}$$
$$\frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j Q_{ij}^k + \mu_k - \gamma = 0 \tag{4.76}$$
$$h = \sum_j y_j \alpha_j = 0 \,, \qquad \sum_k d_k = 1$$

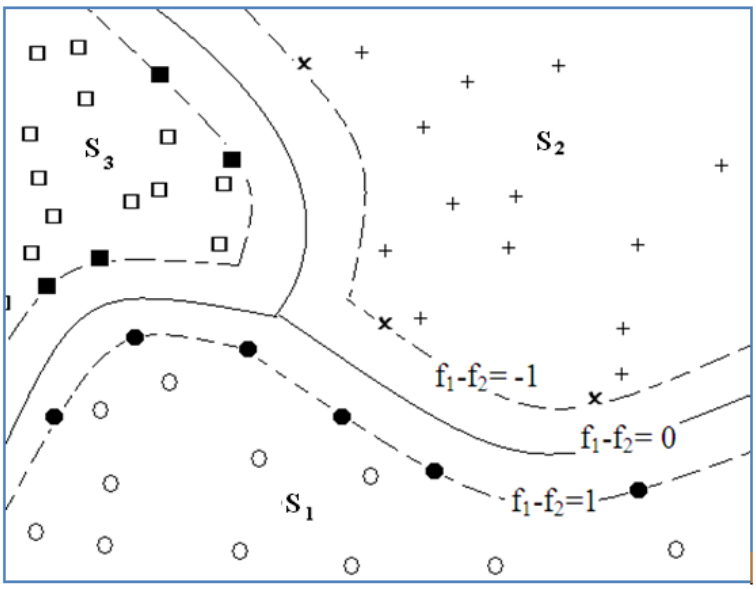
Her yeni örne eklendiğinde Lagrangian çarpanları ($0 \leq \alpha_i \leq C$) hesaplanarak türevsel dönüşüm bilgileri (4.77) hesaplanır ve KKT durumlarına göre sınıf ayrımına gidilir.

$$
\begin{aligned}
g_i &= \sum_j \alpha_j \sum_k \Delta d_k Q_{ij}^k + \sum_j \Delta\alpha_j \sum_{i\in S} Q_{ij}^k + \sum_{j\in \ddot{O}} \Delta\alpha_j \sum_k \Delta d_k Q_{ij}^k + y_i(b+\Delta b) - 1 = 0 \\
h &= \sum_j y_j \alpha_j + \sum_{k\in S} y_k \Delta\alpha_k + \sum_{l\in \ddot{O}} y_l \Delta\alpha_l = 0 \\
&\Delta\mu_k d_k + \mu_k \Delta d_k + \Delta\mu_k \Delta d_k = 0, \quad \forall k \in K \\
&\sum_i y_i \alpha_i = 0 \ , \qquad \sum_k \Delta d_k = 0
\end{aligned}
\tag{4.77}
$$

Doğrusal olmayan parametre çözümü ile $\Delta\mu_k, \Delta d_k, \Delta\alpha_k$ değerleri bulunarak çözüme gidilir. Artımlı ve azılımlı tek çekirdek öğrenme modelinde olduğu gibi g_i ve α_i değerleri baz alınarak artımlı eğitim modeli eğitme devam edilerek her yeri gelen veri kümesi yeni sınıfına atanır yada bir kenara kaldırılır.

4.7 Artımlı ve Azalımlı Çok Etiketli Çoklu Çekirdek Öğrenme Modeli

Yukarıdaki artımlı modellerin hepsi ikili sınıflar $y \in \{-1, +1\}$ için geçerli olmakla birlikte çoklu sınıflar için bu artımlı modeller *1'e karşı diğeri* yada *1'e karşı diğerleri* yöntemine dayalı olarak düşünülür ve uygulanmaktadır. Böyle bir yaklaşım yerine çoklu sınıflar için modeli geliştirdiğimizde problemimiz



Şekil 4.21 Çoklu sınıf için tek çekirdek sınır ve destek vektörleri

Çoklu sınıf ve çoklu çekirdek öğrenme modelini tanımlamadan önce artımlı çok sonıflı tek çekirdek modelini ele almak gerekir. Şekil 4.21'de tanımlandığı üzere sınır yöneylerini ifade edecek fonsksiyonları gösterdiğimizde herhangi iki sınır aşırı düzleminin genişliğini $2/\|w_1 - w_2\|$ şeklinde ifade edebiliriz. Herhangi bir yeni gelen veri kümesinin sınıfını karar

fonksiyonun $x_i \in S_k; k = arg\max_{j=1,..s} f_j(x_i)$ sonucu olarak atayabiliriz. İfadeyi formulüze edersek,

$$\min_{w_i,b_i} \frac{1}{2}\sum_{i=1}^{c}\sum_{j=i+1}^{c}\left\|w_i - w_j\right\|^2 + \frac{1}{2}\sum_{i=1}^{c}\sum_{k}\|w_i\|^2 + C\sum_{i=1}^{c}\sum_{j=i+1}^{c}\sum_{x_l\in S_{ij}}\xi_l^{ij} \quad , \quad \forall x_l \in S_{ij} \qquad (4.78)$$

hesaplanması gerekmektedir. Bunun için denklemin Lagrangian'ını oluşturduğumuzda

$$L = \frac{1}{2}\sum_{i=1}^{c}\sum_{j=i+1}^{c}\left\|w_i - w_j\right\|^2 + \frac{1}{2}\sum_{i=1}^{c}\sum_{k}\|w_i\|^2 + C\sum_{i=1}^{c}\sum_{j=i+1}^{c}\sum_{x_l\in C_{ij}}\xi_l^{ij} - \sum_{i=1}^{c}\sum_{j=i+1}^{c}\sum_{x_l\in C_{ij}}\mu_i^{ij}\xi_l^{ij}$$
$$-\sum_{i=1}^{c}\sum_{j=i+1}^{c}\sum_{x_l\in C_{ij}}\alpha_l^{ij}\left(y_l^{ij}\left[\left(v_i - v_j\right)^T\Phi(x_l) + \left(b_i - b_j\right)\right] - 1 + \xi_l^{ij}\right) \qquad (4.79)$$

diğer KKT kısıtları burdada geçerli olma koşulu ile w_i ve b_i ye göre türevlen aldığımızda,

$$\frac{\partial L}{\partial w_i} = cw_i - \sum_{j=1,j\neq i}^{c} w_j + \sum_{j=1,j\neq i}^{c}\left(\sum_{x_l\in C_i}\alpha_l^{ij}\Phi(x_l) - \sum_{x_l\in C_j}\alpha_l^{ij}\Phi(x_l)\right) = 0$$
$$\frac{\partial L}{\partial b_i} = \sum_{j=1,j\neq i}^{c}\left(\sum_{x_l\in C_i}\alpha_l^{ij} - \sum_{x_l\in C_j}\alpha_l^{ij}\right) = 0 \qquad (4.80)$$
$$\frac{\partial L}{\partial \xi_l^{ij}} = C - \alpha_l^{ij} - \mu_l^{ij} = 0$$

burdan

$$w_i = \frac{1}{c+1}\sum_{j=1,j\neq i}^{c}\left(\sum_{x_l\in C_i}\alpha_l^{ij} - \sum_{x_l\in C_j}\alpha_l^{ij}\right) \qquad (4.81)$$

yukarıdaki çözümlerden $2\frac{\partial L}{\partial w_i} + \sum_{j=\left(\sum_{x_l\in C_i}\alpha_l^{ij} - \sum_{x_l\in C_j}\alpha_l^{ij}\right)1,j\neq i}^{c}\frac{\partial L}{\partial w_i} = 0$ ile elde edilebilir. Çekirdek fonksiyonu $\hbar(x_1, x_2) = \phi(x_1)^T\phi(x_2)$ olmak üzere

$$f_i(x) = \frac{1}{c+1}\sum_{j=1,j\neq i}^{c}\left(\sum_{x_l\in C_i}\alpha_l^{ij}\hbar(x_l, x) - \sum_{x_l\in C_j}\alpha_l^{ij}\hbar(x_l, x)\right) + b_i \qquad (4.82)$$

Lagrangian çarpanları 0 olmaanlar için $\alpha_l^{ij}\left(w_i - w_j\right)^T\phi(x_l) + \left(b_i - b_j\right) - 1 = 0$ ve $\mu_l^{ij}\xi_l^{ij} = 0, i = 1, \dots, c \; ve \; j = i+1, \dots, c$ olmalı. En azlanacak fonksiyonu,

$$W = \frac{1}{2(c+1)}\left(\sum_{i=1}^{c}\sum_{j=i+1}^{c}\|\varpi^{*}\|^{2} + \sum_{i=1}^{c}\|\varpi^{**}\|^{2}\right) - \sum_{i=1}^{c}\sum_{j=i+1}^{c}\sum_{x_l \in C_{ij}} \alpha_l^{ij} + \sum_{i=1}^{c} b_i\, w_i$$

$$\varpi^{**} = \sum_{\substack{m=1,\\ m\neq i}}^{c}\left(\sum_{x_l \in C_i} \alpha_l^{im}\phi(x_l) - \sum_{x_l \in C_m} \alpha_l^{mj}\phi(x_l)\right) \tag{4.83}$$

$$\varpi^{*} = \sum_{x_l \in C_i}\left(2\alpha_l^{ij} + \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\alpha_l^{im}\right)\phi(x_l) - \sum_{x_l \in C_i}\left(2\alpha_l^{ij} + \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\alpha_l^{mj}\right)\phi(x_l) - \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\sum_{x_l \in C_m}\left(\alpha_l^{im} - \alpha_l^{mJ}\right)$$

burdan Lagrangian çarpanlarına göre ve etki parametresine göre türev alınırsa artımlı modele çekirdek modeline geçilir,

$$g_n^{ij} = \frac{\partial W}{\partial \alpha_n^{ij}} = \frac{y_n^{ij}}{c+1}\left[\sum_{x_l \in C_i}\left(2\alpha_l^{ij} + \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\alpha_l^{im}\right)\kappa_{ln} - \sum_{x_l \in C_i}\left(2\alpha_l^{ij} + \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\alpha_l^{mj}\right)\kappa_{ln} - \sum_{\substack{m=1,\\ m\neq i,j}}^{c}\sum_{x_l \in C_m}\left(\alpha_l^{im} - \alpha_l^{mJ}\right)\kappa_{ln} + \left(b_i - b_j\right)\right] - 1 \tag{4.84}$$

$$h = \frac{\partial W}{\partial b_i} = \sum_{\substack{m=1,\\ m\neq i}}^{c}\left(\sum_{x_l \in C_i} \alpha_l^{im} - \sum_{x_l \in C_m} \alpha_l^{im}\right) \tag{4.85}$$

yine aynı şekilde

$$g_n^{ij}\begin{cases} < 0 & \alpha_n^{ij} = C \\ = 0 & 0 \le \alpha_n^{ij} \le C \\ > 0 & \alpha_n^{ij} = 0 \end{cases} \tag{4.86}$$

yeni gelen veri kümesi grublar atanarak öğrenme işlevine devam edilir.

4.8 Modellerin Başarımları ve Sonuçlar

4.8.1 Kanser Verileri

Burda ele aldığımız kanser veri kümelerini ikinci bölümde detaylı ele almıştık. Önerdiğimiz modellerin başarımlarını farklı çekirdek paremetrelerine göre ele alıp detaylı inceleyeceğiz.

İlk olarak Göğüs kanseri veri kümesi üzerinde tekil çekirdeklerin başarımlarını inceleyelim;

Çizelge 4.10 Göğüs Kanseri Doğruluk Oranları

		Doğrusal	İkinci Derece	Polin. (d=3)	Polin. (d=4)	Polin. (d=5)	RBF (g=0.1)	RBF (g=0.5)	RBF (g=1)	RBF (g=2)	RBF (g=5)	RBF (g=10)	RBF (g=100)	MLP
C		En Küçük Kareler En İyilemesi												
2^{-10}	0,0009766	94,4%	95,1%	91,9%	82,4%	80,3%	62,7%	82,0%	85,9%	85,2%	93,7%	94,0%	93,7%	92,3%
2^{-9}	0,0019531	94,7%	95,1%	91,5%	81,5%	79,4%	62,7%	82,0%	85,9%	85,2%	93,7%	94,0%	93,7%	92,3%
2^{-8}	0,0039063	94,8%	95,4%	91,3%	80,8%	78,9%	62,7%	82,0%	86,0%	85,2%	93,7%	94,0%	93,7%	92,4%
2^{-7}	0,0078125	95,1%	95,8%	91,0%	80,1%	78,7%	62,7%	82,0%	86,1%	85,3%	93,8%	94,0%	93,7%	92,4%
2^{-6}	0,015625	95,3%	95,9%	90,6%	79,6%	78,3%	62,7%	82,0%	86,1%	85,5%	93,9%	94,1%	93,7%	91,9%
2^{-5}	0,03125	95,6%	96,0%	89,9%	79,2%	78,1%	62,7%	82,1%	86,2%	85,7%	94,1%	94,1%	93,7%	88,8%
2^{-4}	0,0625	95,8%	95,9%	89,3%	78,8%	78,0%	62,7%	82,3%	86,4%	85,9%	94,4%	94,3%	93,7%	83,5%
2^{-3}	0,125	95,9%	96,0%	88,8%	78,5%	77,8%	62,7%	82,6%	86,8%	86,1%	94,6%	94,4%	93,7%	82,8%
2^{-2}	0,25	95,9%	95,6%	88,3%	78,2%	77,7%	62,7%	83,0%	87,3%	86,6%	94,8%	94,7%	93,7%	79,2%
2^{-1}	0,5	95,9%	95,3%	87,9%	78,1%	77,6%	62,7%	83,5%	87,2%	87,2%	95,1%	95,0%	93,7%	77,3%
2^{0}	1	95,8%	94,7%	87,5%	77,9%	77,6%	62,7%	84,0%	86,6%	87,8%	95,3%	95,3%	93,7%	75,8%
2^{1}	2	95,7%	94,2%	87,1%	77,8%	77,5%	62,7%	84,6%	85,9%	88,4%	95,4%	95,6%	93,7%	74,6%
2^{2}	4	95,6%	93,5%	86,8%	77,7%	77,5%	62,7%	85,0%	85,3%	88,9%	95,6%	95,8%	93,8%	75,2%
2^{3}	8	95,6%	92,8%	86,6%	77,6%	77,5%	62,7%	85,5%	84,6%	89,4%	95,7%	96,0%	93,9%	74,7%
2^{4}	16	95,5%	92,0%	86,4%	77,5%	77,4%	62,7%	85,8%	84,1%	89,8%	95,8%	96,1%	94,0%	75,1%
2^{5}	32	95,5%	91,2%	86,2%	77,4%	77,4%	62,7%	86,2%	83,6%	90,2%	95,8%	96,2%	94,1%	74,4%
2^{6}	64	95,4%	90,4%	86,0%	77,4%	77,4%	62,7%	86,4%	83,1%	90,5%	95,9%	96,3%	94,2%	74,6%
2^{7}	128	95,4%	89,5%	85,8%	77,3%	77,4%	62,7%	86,7%	82,7%	90,7%	95,9%	96,4%	94,4%	74,8%
2^{8}	256	95,3%	88,7%	85,7%	77,3%	77,4%	62,7%	86,9%	82,3%	91,0%	95,8%	96,5%	94,5%	74,9%
2^{9}	512	95,3%	88,0%	85,6%	77,2%	77,4%	62,7%	87,1%	82,0%	91,2%	95,8%	96,5%	94,6%	74,9%
2^{10}	1024	95,3%	87,3%	85,4%	77,2%	77,3%	62,7%	87,3%	81,7%	91,4%	95,6%	96,4%	94,7%	75,0%
En yüksek		0,959	95,9%	96,0%	91,9%	82,4%	80,3%	62,7%	87,3%	87,3%	91,4%	95,9%	96,5%	94,7%
En düşük		0,944	94,4%	87,3%	85,4%	77,2%	77,3%	62,7%	82,0%	81,7%	85,2%	93,7%	94,0%	93,7%
Standart Sapma		0,003	0,3%	3,0%	2,1%	1,4%	0,7%	0,0%	2,0%	1,9%	2,3%	0,8%	1,0%	0,4%
Ortalama		0,955	95,5%	92,8%	87,6%	78,2%	77,8%	62,7%	84,6%	84,9%	88,4%	95,1%	95,4%	94,0%

Çizelge 4.10'da gözüktüğü gibi tekil çekirdek modelleri ele aldığımızda en yüksek başarımı(%96.5) radyal tabanlı çekirdek vermekte fakat dikkat edilirse en düşük başarım oranını(%62.7) da yine radyal tabanlı çekirdek veriyor. Bunun yanında farklı ***C*** düzenleştirme sabitlerinde (doğrusal olmayan çekirdek modeli için) farklı başarımlar gözlemlenmektedir. Doğrusal çekirdek ile ikincil dereceden çekirdekler tek başlarına incelendiğinde ikincil çekirdek en yüksek başarımı vermesine rağmen ortalamalara bakıldığında genelde doğrusal çekirdek modeli göğüs kanseri veri kümesinde başarımı daha yüksek olduğu fark edilecektir. Hatırlanacağı üzere DVM'ler aslında bir en iyileştirme problemidir yukarıda kullanılan

modellerin hepsinde en iyileştirmeyi "En Küçük Kareler En İyilemesi" yöntemi ile yapılmıştır. Çizelge 4.11'de ise farklı optimizasyon(en iyileme) modellerine göre aynı çekirdeklerin doğruluk oranları listelenmiştir. Bu en iyileme modelleride algoirtmanın başarımında büyük etkisi gözlemlenmektedir. Özellikle MLP çekirdeğindeki SMO ile en iyileme yapıldığında diğer en iyileme problemlerine göre çok düşük başarı yüzdesi oluştuğu gözlemlenmiştir.

Çizelge 4.11 Göğüs Kanseri QP ve SMO ile en iyileme başarımı

		Doğrusal	İkinci Derece	Polin. (d=3)	RBF (g=10)	MLP		Doğrusal	İkinci Derece	Polin. (d=3)	RBF (g=10)	MLP
C		QP						SMO				
2^{-10}	0,0009766	94,0%	91,2%	94,4%	92,3%	92,6%		95,8%	96,5%	94,0%	91,9%	37,3%
2^{-9}	0,0019531	94,7%	91,9%	94,4%	92,3%	92,8%		96,3%	97,4%	94,1%	91,9%	37,3%
2^{-8}	0,0039063	95,4%	92,5%	94,5%	92,3%	92,8%		96,6%	97,4%	94,2%	91,9%	37,3%
2^{-7}	0,0078125	96,3%	93,0%	94,6%	92,3%	92,8%		96,9%	97,4%	94,3%	91,9%	37,3%
2^{-6}	0,015625	96,8%	93,5%	94,7%	92,3%	92,7%		97,1%	97,0%	94,4%	92,0%	37,3%
2^{-5}	0,03125	97,2%	93,8%	94,8%	92,3%	92,7%		97,1%	96,7%	94,4%	92,3%	40,8%
2^{-4}	0,0625	97,4%	94,0%	94,8%	92,3%	92,8%		97,0%	96,3%	94,4%	92,6%	49,7%
2^{-3}	0,125	97,6%	94,1%	94,7%	92,3%	92,8%		96,9%	96,0%	94,8%	92,9%	55,7%
2^{-2}	0,25	97,8%	94,1%	94,7%	92,3%	92,8%		96,9%	95,9%	94,9%	93,3%	60,2%
2^{-1}	0,5	97,8%	94,0%	94,6%	92,3%	92,8%		96,9%	95,7%	94,9%	93,7%	63,5%
2^{0}	1	97,9%	93,9%	94,6%	92,3%	92,8%		96,9%	95,6%	95,0%	94,1%	66,3%
2^{1}	2	97,8%	93,8%	94,5%	92,4%	92,8%		96,8%	95,5%	95,0%	94,3%	68,5%
2^{2}	4	97,8%	93,7%	94,5%	92,4%	92,9%		96,8%	95,4%	95,0%	94,5%	70,3%
2^{3}	8	97,7%	93,6%	94,4%	92,5%	92,9%		96,8%	95,3%	95,0%	94,7%	71,7%
2^{4}	16	97,7%	93,5%	94,4%	92,7%	92,9%		96,8%	95,3%	95,0%	94,9%	72,9%
2^{5}	32	97,7%	93,5%	94,4%	92,9%	92,9%		96,7%	95,2%	95,0%	95,0%	74,0%
2^{6}	64	97,6%	93,4%	94,4%	93,2%	92,8%		96,7%	95,2%	95,0%	95,1%	75,0%
2^{7}	128	97,6%	93,4%	94,3%	93,4%	92,8%		96,7%	95,1%	95,0%	95,1%	75,8%
2^{8}	256	97,5%	93,3%	94,3%	93,7%	92,7%		96,7%	95,1%	95,0%	95,1%	76,6%
2^{9}	512	97,4%	93,3%	94,3%	93,9%	92,7%		96,9%	95,0%	95,0%	95,1%	77,2%
2^{10}	1024	97,4%	93,3%	94,3%	94,2%	92,6%		96,9%	94,6%	95,0%	95,9%	77,9%
En yüksek		0,979	97,9%	94,1%	94,8%	94,2%	92,9%		97,1%	97,4%	95,0%	95,9%
En düşük		0,940	94,0%	91,2%	94,3%	92,3%	92,6%		95,8%	94,6%	94,0%	91,9%
Standart Sapma		0,006	0,6%	0,4%	0,2%	0,7%	0,1%		0,1%	0,9%	0,3%	1,3%
Ortalama		0,973	97,3%	93,5%	94,5%	92,8%	92,8%		96,8%	95,7%	94,8%	94,0%

Başarımlara bakıldığında doğruluk oranı genel olarak verinin dağılım fonksiyonuna göre değişmekle birlikte çekirdek parametrelerinin iyi seçilmesi bu oranı artırmakta yada azalmaktadır. Çerkirdek parametrelerinin seçimi ile ilgili olarakda akademik yazımda bir çok makale bulunmaktadır. Alpaydin(2004) özellikle düzenleştirme sabitinin çok yüsek olması durumunda DV olarak kullanılacak örneklerin artacağını vurgulamıştır.

Çizelge 4.12 Mide Kanseri başarımları

C		Doğrusal	İkinci Derece	Polin. (d=3)	RBF (g=1)	RBF (g=10)	IKL (RBF)	MKL (İç Bükey)	MKL (P+G)	IMKL (P+G+ P)	IMKL (P+G+L)	IMKL (L+G)
2^{-10}	0,000977	65,1%	60,5%	58,7%	59,3%	59,3%	64,0%	63,9%	73,7%	75,1%	71,7%	71,9%
2^{-9}	0,001953	65,1%	59,6%	59,3%	59,3%	59,3%	61,6%	61,3%	62,9%	66,0%	59,9%	63,0%
2^{-8}	0,003906	65,7%	59,3%	59,7%	59,3%	59,3%	62,9%	66,3%	72,4%	73,3%	66,2%	72,9%
2^{-7}	0,007813	66,1%	59,4%	60,0%	59,3%	59,3%	61,5%	63,0%	72,6%	75,5%	67,6%	80,5%
2^{-6}	0,015625	66,4%	60,3%	60,5%	59,3%	59,5%	61,3%	66,9%	70,1%	72,2%	69,8%	77,4%
2^{-5}	0,03125	66,7%	60,8%	61,0%	59,3%	59,8%	62,9%	67,2%	69,8%	78,5%	79,8%	71,0%
2^{-4}	0,0625	66,9%	61,6%	61,5%	59,3%	60,2%	62,9%	68,5%	72,4%	79,5%	62,5%	71,8%
2^{-3}	0,125	66,9%	62,6%	61,7%	59,3%	60,8%	62,6%	63,2%	68,7%	71,9%	70,6%	72,8%
2^{-2}	0,25	66,9%	63,4%	61,6%	59,3%	61,5%	64,9%	64,7%	68,9%	77,8%	68,0%	78,7%
2^{-1}	0,5	66,9%	64,1%	61,5%	59,3%	62,0%	66,6%	69,7%	79,0%	85,2%	76,1%	80,7%
2^{0}	1	66,9%	64,9%	61,2%	59,3%	62,5%	66,0%	67,9%	71,2%	76,6%	75,5%	66,3%
2^{1}	2	66,9%	65,5%	61,0%	59,3%	63,1%	68,4%	64,4%	72,8%	76,3%	69,9%	86,3%
2^{2}	4	66,9%	66,0%	61,0%	59,3%	63,6%	69,8%	69,5%	70,0%	72,6%	61,4%	86,9%
2^{3}	8	66,9%	66,4%	61,0%	59,3%	63,9%	68,7%	66,2%	73,5%	75,3%	67,2%	73,0%
2^{4}	16	66,9%	66,8%	60,9%	59,3%	63,8%	70,0%	68,8%	70,9%	71,4%	61,0%	73,0%
2^{5}	32	66,9%	67,1%	60,9%	59,3%	63,8%	69,6%	67,4%	75,4%	84,2%	65,6%	72,9%
2^{6}	64	66,9%	67,3%	60,9%	59,3%	63,8%	68,8%	69,0%	78,8%	80,9%	71,8%	86,6%
2^{7}	128	66,9%	67,6%	60,9%	59,3%	63,7%	70,9%	63,5%	68,9%	76,4%	69,6%	76,3%
2^{8}	256	66,9%	67,8%	60,8%	59,3%	63,6%	68,5%	68,7%	69,2%	71,5%	68,6%	74,7%
2^{9}	512	66,9%	68,0%	60,8%	59,3%	63,5%	68,0%	68,2%	69,3%	70,9%	74,8%	75,5%
2^{10}	1024	66,9%	68,1%	60,8%	59,3%	63,5%	70,6%	64,0%	67,2%	72,0%	75,2%	76,1%

En yüksek	0,669	66,9%	68,1%	61,7%	59,3%	63,9%	70,9%	69,7%	79,0%	85,2%	79,8%
En düşük	0,651	65,1%	59,3%	58,7%	59,3%	59,3%	61,3%	61,3%	62,9%	66,0%	59,9%
Standart Sapma	0,003	0,3%	3,1%	0,5%	0,0%	1,8%	3,4%	2,3%	3,6%	4,7%	5,5%
Ortalama	0,667	66,7%	64,8%	60,9%	59,3%	62,3%	66,7%	66,6%	71,5%	76,0%	69,7%

Mide kanseri (345 örnek ve 7 öznitelik) veri seti üzerindeki çekirdek modellerinin doğruluk oranların incelediğimizde ise doğrusal çekirdeğin radyal tabanlı çekirdeklere ve polinomsal çekrideklere göre doğruluk oranlarının daha yüksek olduğu tespit edilmiştir. Bunun yanında yinelemeli artımlı Çekirdek Öğrenme modeli (IKL) ise doğrusal çakirdek hariç diğer tekil çekirdeklere göre başarımının daha yüksek olduğu incelendi. Bunun yanında çoğul çekirdeklerin(MKL, IMKL) genelinde başarı düzeyi diğer çekirdeklere göre daha yüksek olduğu incelenmektedir. Fakat buna rağmen başarı düzeyinin çok yüksek olmaması özniteliklerin düzensiz dağılımndan kaynaklandığı görüşünü ortaya koymaktadır (Çizelge 4.12) .

Cherkassky ve Mulier (1998) yine seçilecek düzenleştirme sabitinin karmaşıklık ile doğruluk arasında bir ödünleşme olduğunu belirtmiştir. Chalimourda vd.(2000,2004), Cherkassky ve Yunqian(2002, 2004), Ali ve Smith (2003), Debnath, R. ve Takahashi (2004), Frohlich ve Zell(2005), Gold ve Sollich (2005), Boardman ve Trappenberg(2006) ise en uygun çekirdek parametrelerin seçimi ve en iyilemesi ile ilgili önerilerde bulunmuşlardır. Çekirdek modellerin temelinde bir en iyileme problemi yattığından farklı en iyileme modelleri sonucunda da aynı parametreler için farklı sonuçlar verebilmektedir.

Bu örneklerin haricinde çoklu sınıflar için yada diğer diğer bir tabirle birden fazla etiketli veri kümesi için de çok sınıflı çekirdek modellerini ele eldığımızda bu konuda da farklı sonuçlar elde edebilmekteyiz. Bire-karşı-diğeri, bire-karşı-diğerleri, yada gibi farklı yöntemler karşımıza çıkmaktadır. Tüm verileri tek tek incelemek yerine resmin görünen yüzüne bakmak için Çizelge 4.13'de genel bir doğruluk tablosu veilmiştir. Tablodaki tüm veriler 10 katlı çarpraz geçerleme modeli ile ele alınarak eğitilmiş ve test edilmiştir. Bu tabloyu detaylı incelersek her veri seti için doğruluk oranı farklı parametre seçimleri için farklı oranlar oluşturduğunu farkedeceğiz. Ama yinelemeli olan IKL ve IMKL de toplam standart sapma diğerlerine göre biraz daha düşük olduğu ve doğruluk oranlarında ise çok az artış olduğu gözlemlenecektir. Bazı veri kümeleri için ise düşük doğruluk oranı verecektir. Düzenleştirme parametresinin etkisi ve verinin dağılımının bu sonuçlar üzerinde etkisi büyüktür. Çizelge 4.13' de her model için oluşturulmuş en yüksek doğruluk oranları belirtilmiştir.

Çizelge 4.13 Çoklu sınıfların en yüksek doğruluk oranları

	Örnek Sayısı	Öznitelik Sayısı	Sınıf Sayısı	Doğrusal (1-*)	İkinci Dereceden (1-*)	RBF (1-*)	MKL	IKL	IMKL
Akciğer Kanseri	32	56	2	73.4 ± 0.5	71.4 ± 3.5	70.1 ± 0.4	76.7 ± 3.5	71.4 ± 5.9	76.4 ± 3.1
İris	150	4	3	97.3 ± 1.1	96.6 ± 3.1	96.4± 3.8	97.2 ± 1.4	96.6 ± 3.1	95.1 ± 3.2
Yüz	640	15360	20	87.1 ± 2.2	86.1 ± 2.4	88.4 ± 1.7	89.1 ± 3.9	90.3 ± 5.7	91.0 ± 3.1
Parkinson	197	23	2	71.1 ± 1.3	72.1 ± 1.4	73.4 ± 0.5	71.4 ± 3.5	75.1 ± 0.4	70.2 ± 3.4
Göğüs Kanseri	569	30	2	97.9 ± 0.4	94.1 ± 0.9	94.2 ± 1.7	90.3 ± 6.4	93.0 ± 1.0	92.2 ± 0.9
Tiroid Hastalığı	7200	21	20	94.3 ± 2.4	95.1 ± 1.6	94.1 ± 1.8	97.1 ± 2.2	95.2 ± 0.3	98.5 ± 6.4
Kalp Rahatsızlığı	303	75	4	85.1 ± 2.4	84.1 ± 2.4	82.1 ± 2.2	80.1 ± 2.4	80.4 ± 1.7	83.1 ± 3.9
Şeker Hastalığı	70	20	2	70.2 ± 3.4	75.1 ± 1.3	77.1 ± 1.4	70.4 ± 0.5	76.4 ± 3.5	73.1 ± 1.2
Hepatit Hastalığı	155	19	2	80.1 ± 2.4	80.4 ± 1.7	83.1 ± 3.9	84.4 ± 1.7	80.1 ± 3.9	86.1 ± 2.2
Mide Kanseri	32	56	3	66.9 ± 2.0	68.1 ± 9.7	63.9 ± 6.1	75.2 ± 4.4	70.1 ± 3.1	86.9 ± 5.9
Şarap Verisi	178	4	3	99.1 ± 0.3	98.1 ± 1.2	98.4 ± 2.1	99.9 ± 0.4	99.4 ± 0.1	98.1 ± 0.7
DNA	2000	180	3	95.4 ± 1.1	95.4 ± 1.2	95.7 ± 3.3	95.0 ± 1.4	94.1 ± 1.1	95.4 ± 1.6
SatResimleri	4435	36	6	91.3 ± 5.7	91.0 ± 3.1	91.44 ± 2.9	91.9 ± 5.1	91.0 ± 1.11	91.3 ± 2.1
Letter	15000	16	26	97.4 ± 3.1	98.1 ± 1.1	96.7 ± 1.3	97.1 ± 4.0	97.3 ± 2.1	97.1 ± 3.1
Shuttle	43500	9	7	99.1 ± 0.2	98.7± 0.9	91.1 ± 6.4	99.1 ± 0.3	100.0 ± 0.0	100.0 ± 0.0
Glass	214	13	6	72.1 ± 1.4	73.4 ± 0.5	71.4 ± 3.5	75.1 ± 0.4	70.2 ± 3.4	71.1 ± 1.3
Vowel	528	10	11	97.1 ± 2.2	98.2 ± 0.3	98.5 ± 6.4	99.1 ± 0.3	98.1 ± 1.2	96.7 ± 1.3
Vehicle	846	18	4	82.1 ± 2.2	80.1 ± 2.4	83.1 ± 5.2	84.1 ± 2.4	82.2 ± 2.7	84.0 ± 3.1
Segment	2310	19	7	97.4 ± 3.1	94.1 ± 1.1	95.4 ± 1.6	98.1 ± 1.1	96.7 ± 1.3	95.0 ± 1.4
Enron e-posta	39861	28102	2	79.1 ± 5.4	81.1 ± 5.6	84.3 ± 5.4	81.2 ± 4.2	81.9 ± 2.6	84.7 ± 2.3
NIPS	1500	12419	4	55.1 ± 6.1	61.1 ± 2.4	64.1 ± 2.4	74.1 ± 2.4	68.1 ± 2.4	73.1 ± 1.4

4.8.2 İlaç Verileri

Elimizde bulun 4 farklı ilaç veri seti üzerinde önerdiğimiz modellerin başarımlarını detayları ile inceleyeceğiz. Burada detayları ile ele alacağımız ilaç veri kümeleri Cherkasov verisi (2684 örnek ve 164 öznitelik), Murcia-Soller (641 örnek ve 162 öznitelik), DrugDataBank (4800 orjinal+ 2000 rastgele örnek ve 345 öznitelik) ve Pharmeks* (251.158 örnek ve 312 öznitelik) içermektedir. Burda ele alınan son iki veri kümesi internet sitelerinden sdf formatında elde edilebilecek verilerdir. Bu veri kümelerinden QSAR ve QSPR özniteliklerini elde edebilmek için MOE ve Adriana.Code programlarından yararlandık. Böylece 300-400 arası öznitelik elde etmemize rağman veri ön işleme sırasında bazı verilerin bizim için gereksiz olduğuna karar verdik. Bazı öznitelikleri hesaplama sırasında problem oluşturduğu ve veri hesaplamasına olanak sağlamadığı için bazılarını eledik. Gerekli veriyi okumada güçlükler çıktı bundan dolayı bizim için gereksiz yada hesaplamada problemli olan öznitelikleri eledik. Daha öncede de belirtildiği gibi her veri kümesine uygulanan belli başlı çekirdek metodları yanında o veriye özel bazı farklı modeller de uygulanmıştır.

Genel olarak verilerin başarımlarını ele almak gerekirse,

Çizelge 4.14 Cherkasov ilaç veri seti doğruluk oranları

		Linear	Quadratic	Polinomial	RBF	MLP	IKL (K=Rbf)	IMKL (L + P+ RBF)	IMKL (L + P)
C		LSO							
2^{-20}	9,53674E-07	67,5%	73,4%	88,1%	65,6%	62,8%	71,0%	71,9%	80,3%
2^{-19}	1,90735E-06	67,8%	76,9%	81,6%	66,6%	64,7%	68,8%	72,1%	82,4%
2^{-18}	3,8147E-06	66,9%	80,3%	81,6%	65,6%	70,9%	69,3%	75,4%	87,5%
2^{-17}	7,62939E-06	71,3%	81,6%	81,9%	66,9%	60,9%	71,7%	73,4%	83,4%
2^{-16}	1,52588E-05	71,3%	80,0%	79,1%	65,6%	65,0%	70,9%	77,0%	86,4%
2^{-15}	3,05176E-05	64,7%	84,7%	80,9%	65,9%	65,3%	66,2%	67,7%	81,4%
2^{-14}	6,10352E-05	72,8%	82,8%	81,9%	66,3%	66,9%	72,8%	77,2%	86,8%
2^{-13}	0,00012207	78,8%	80,6%	80,0%	65,6%	66,6%	68,7%	76,6%	83,5%
2^{-12}	0,000244141	75,6%	84,4%	83,1%	66,6%	63,8%	73,2%	81,0%	85,4%
2^{-11}	0,000488281	79,1%	81,9%	81,3%	66,3%	65,6%	74,8%	81,8%	82,3%
2^{-10}	0,000976563	83,8%	82,5%	83,1%	66,3%	68,4%	75,6%	80,8%	84,6%
2^{-9}	0,001953125	84,4%	80,0%	81,9%	66,6%	66,3%	70,0%	71,8%	80,9%

* http://www.pharmeks.com

2^{-8}	0,00390625	83,1%	83,8%	80,3%	65,6%	67,8%	69,2%	72,2%	87,1%
2^{-7}	0,0078125	85,3%	84,4%	76,9%	65,9%	72,2%	75,6%	84,3%	91,7%
2^{-6}	0,015625	85,3%	81,6%	79,4%	65,9%	60,3%	70,0%	74,0%	82,0%
2^{-5}	0,03125	85,3%	81,9%	79,7%	66,9%	63,8%	69,3%	75,8%	90,8%
2^{-4}	0,0625	83,4%	82,8%	83,1%	66,9%	51,9%	71,4%	73,2%	90,8%
2^{-3}	0,125	82,8%	77,8%	79,7%	66,3%	48,1%	70,1%	79,0%	86,4%
2^{-2}	0,25	86,6%	83,1%	80,3%	66,9%	52,2%	70,6%	77,5%	92,7%
2^{-1}	0,5	84,1%	82,2%	80,0%	66,3%	58,8%	72,3%	80,2%	84,2%
2^{0}	1	80,9%	81,3%	84,4%	65,9%	50,3%	68,2%	70,5%	86,3%
2^{1}	2	83,4%	80,9%	82,5%	66,3%	50,9%	72,7%	74,5%	86,7%
2^{2}	4	84,4%	76,3%	82,5%	66,3%	51,3%	74,2%	83,7%	82,5%
2^{3}	8	81,3%	80,9%	77,8%	66,3%	51,9%	68,1%	78,0%	81,0%
2^{4}	16	82,5%	79,4%	83,4%	66,3%	54,7%	74,0%	74,3%	82,9%
2^{5}	32	80,0%	77,5%	82,8%	65,6%	48,4%	73,1%	78,4%	78,5%
2^{6}	64	83,1%	80,0%	80,9%	65,6%	46,3%	69,0%	72,3%	86,7%
2^{7}	128	84,4%	75,9%	81,3%	66,3%	51,3%	71,1%	78,2%	82,1%
2^{8}	256	84,1%	81,6%	83,4%	65,9%	53,4%	67,0%	69,3%	88,1%
2^{9}	512	77,5%	81,3%	79,4%	65,9%	58,1%	74,7%	78,3%	83,6%
2^{10}	1024	81,9%	79,4%	83,1%	66,3%	57,8%	73,9%	83,8%	82,2%
2^{11}	2048	80,9%	78,1%	83,1%	65,3%	54,7%	73,5%	82,1%	88,1%
2^{12}	4096	77,5%	81,3%	79,4%	66,9%	52,2%	73,8%	74,2%	82,8%
2^{13}	8192	79,4%	80,9%	77,8%	65,6%	55,0%	73,7%	80,9%	83,5%
2^{14}	16384	82,5%	78,4%	81,9%	65,3%	44,1%	74,5%	83,1%	84,1%
2^{15}	32768	80,9%	80,9%	83,1%	66,3%	58,1%	73,6%	78,1%	88,1%
2^{16}	65536	83,4%	79,4%	81,3%	65,6%	46,3%	71,6%	72,7%	88,8%
2^{17}	131072	81,3%	81,9%	81,6%	66,6%	55,0%	75,0%	79,1%	87,8%
2^{18}	262144	76,6%	79,4%	82,8%	65,6%	53,1%	69,8%	77,5%	87,1%
2^{19}	524288	80,6%	81,6%	80,6%	65,9%	47,8%	74,6%	83,3%	89,4%
2^{20}	1048576	77,8%	79,7%	81,3%	66,3%	50,6%	76,0%	83,7%	85,2%
En yüksek		0,866	86,6%	84,7%	88,1%	66,9%	72,2%	76,0%	84,3%
En düşük		0,647	64,7%	73,4%	76,9%	65,3%	44,1%	66,2%	67,7%
Ortalama		0,796	79,6%	80,5%	81,4%	66,1%	57,4%	71,8%	77,0%
Standart Sapma		0,057	5,7%	2,4%	2,0%	0,4%	7,6%	2,6%	4,4%

Çizelge 4.15 Murcia-Soller veri seti doğruluk oranları

		Linear	Quadratic	Polinomial	RBF	MLP	IKL (K=Rbf)	IMKL (L + P+ RBF)	IMKL (L + P)
C		LSO							
2^{-20}	9,53674E-07	67,5%	73,4%	88,1%	65,6%	62,8%	72,2%	77,6%	78,6%
2^{-19}	1,90735E-06	67,8%	76,9%	81,6%	66,6%	64,7%	66,6%	66,7%	79,3%
2^{-18}	3,8147E-06	66,9%	80,3%	81,6%	65,6%	70,9%	71,9%	77,9%	84,7%
2^{-17}	7,62939E-06	71,3%	81,6%	81,9%	66,9%	60,9%	76,1%	79,6%	82,3%
2^{-16}	1,52588E-05	71,3%	80,0%	79,1%	65,6%	65,0%	73,8%	80,9%	83,9%
2^{-15}	3,05176E-05	64,7%	84,7%	80,9%	65,9%	65,3%	73,8%	79,8%	92,6%
2^{-14}	6,10352E-05	72,8%	82,8%	81,9%	66,3%	66,9%	71,4%	74,8%	90,9%
2^{-13}	0,00012207	78,8%	80,6%	80,0%	65,6%	66,6%	71,9%	81,4%	83,4%
2^{-12}	0,000244141	75,6%	84,4%	83,1%	66,6%	63,8%	67,3%	75,5%	90,3%
2^{-11}	0,000488281	79,1%	81,9%	81,3%	66,3%	65,6%	72,9%	79,7%	86,1%
2^{-10}	0,000976563	83,8%	82,5%	83,1%	66,3%	68,4%	72,5%	81,6%	85,2%
2^{-9}	0,001953125	84,4%	80,0%	81,9%	66,6%	66,3%	76,4%	82,6%	88,3%
2^{-8}	0,00390625	83,1%	83,8%	80,3%	65,6%	67,8%	73,0%	82,6%	90,8%
2^{-7}	0,0078125	85,3%	84,4%	76,9%	65,9%	72,2%	67,6%	74,9%	94,3%
2^{-6}	0,015625	85,3%	81,6%	79,4%	65,9%	60,3%	70,4%	78,4%	81,6%
2^{-5}	0,03125	85,3%	81,9%	79,7%	66,9%	63,8%	71,2%	72,9%	90,1%
2^{-4}	0,0625	83,4%	82,8%	83,1%	66,9%	51,9%	72,1%	74,3%	88,3%
2^{-3}	0,125	82,8%	77,8%	79,7%	66,3%	48,1%	73,4%	75,0%	78,8%
2^{-2}	0,25	86,6%	83,1%	80,3%	66,9%	52,2%	70,0%	78,5%	92,2%
2^{-1}	0,5	84,1%	82,2%	80,0%	66,3%	58,8%	70,2%	70,7%	89,8%
2^{0}	1	80,9%	81,3%	84,4%	65,9%	50,3%	73,4%	81,7%	86,3%
2^{1}	2	83,4%	80,9%	82,5%	66,3%	50,9%	74,6%	78,6%	83,3%
2^{2}	4	84,4%	76,3%	82,5%	66,3%	51,3%	75,3%	81,3%	86,1%
2^{3}	8	81,3%	80,9%	77,8%	66,3%	51,9%	71,1%	76,3%	88,1%
2^{4}	16	82,5%	79,4%	83,4%	66,3%	54,7%	70,2%	76,9%	82,6%
2^{5}	32	80,0%	77,5%	82,8%	65,6%	48,4%	67,2%	71,0%	83,5%
2^{6}	64	83,1%	80,0%	80,9%	65,6%	46,3%	66,2%	66,9%	80,9%
2^{7}	128	84,4%	75,9%	81,3%	66,3%	51,3%	72,7%	78,6%	77,7%
2^{8}	256	84,1%	81,6%	83,4%	65,9%	53,4%	71,2%	73,8%	87,2%
2^{9}	512	77,5%	81,3%	79,4%	65,9%	58,1%	68,7%	78,4%	90,3%
2^{10}	1024	81,9%	79,4%	83,1%	66,3%	57,8%	72,5%	77,1%	81,8%
2^{11}	2048	80,9%	78,1%	83,1%	65,3%	54,7%	69,0%	69,1%	81,1%
2^{12}	4096	77,5%	81,3%	79,4%	66,9%	52,2%	76,8%	84,5%	88,0%

2^{13}	8192	79,4%	80,9%	77,8%	65,6%	55,0%	69,5%	73,5%	82,8%
2^{14}	16384	82,5%	78,4%	81,9%	65,3%	44,1%	65,6%	71,2%	82,2%
2^{15}	32768	80,9%	80,9%	83,1%	66,3%	58,1%	69,3%	73,6%	86,1%
2^{16}	65536	83,4%	79,4%	81,3%	65,6%	46,3%	70,4%	72,6%	89,1%
2^{17}	131072	81,3%	81,9%	81,6%	66,6%	55,0%	75,9%	85,6%	82,7%
2^{18}	262144	76,6%	79,4%	82,8%	65,6%	53,1%	67,1%	73,9%	83,2%
2^{19}	524288	80,6%	81,6%	80,6%	65,9%	47,8%	70,8%	71,7%	85,7%
2^{20}	1048576	77,8%	79,7%	81,3%	66,3%	50,6%	74,3%	74,5%	85,1%
En yüksek doğruluk oranı		0,866	86,6%	84,7%	88,1%	66,9%	72,2%	76,8%	85,6%

Daha yüksek boyutlu veriler üzerinde çalışıldığında hesap karmaşıklığı artmakta ve işlemler günlerce sürmektedir. Hesap karmaşıklığından kurtulmak için temelde iki yöntem bulunmaktadır. İlki veri kümesinin tamamını ifade edebilecek alt bir küme elde edilmesi yada veriyi daha düşük boyutlu bir veri yapısına dönüştürme. Bu temel probleme öznitelik indirgeme, gürültü indirgeme, alt uzay oluşturma gibi birçok isim verilebilir. En popüler boyut indirgeme modelleri PCA, ICA, vb. gibi özvektör tabanlı modellerdir. Bunlara ek olarak önerdiğimiz iki farklı istatiksel modelin (Kabartma ve mrMR) birleştirerek yeni bir model öne sürdüğümüz INISTA 2010'da ilaç veritabanları üzerinde uygulayarak başarımlarını karşılaştırdık.

Kabartma algoritması hedef örneğin komşuları arasındaki özellik ilişkisini ölçen en temel ve bilinen yöntemlerden birisidir (Marko ve Igor, 2003; Robnik-Šikonja ve Kononenko, 2003) Özellikler rastgele seçilen hedef örnek (*I*) için en yakın kendi sınıfından isabetli örneği (*Nhit*) ve en yakın başka sınıfa ait ıska örneğini (*Nmiss*) bularak öznitelik ağırlıklarını yinelemeli olarak günceller.

$$w_i := w_i + \frac{\Delta(I^i, Nmiss(I^i) - \Delta(I^i,\ Nhit(I^i)}{Range(I^i)} \tag{4.87}$$

Bununla birlikte kabartma algoritmasınn yetersiz kaldığı problemden ilki en yakınlık mesafesi ağırlıklandırılmış uzayda değil orjinal öznitelik uzayında çalışmakta olmasıdır ki bunun sonucunda çok yüksek değerlerde negatif sonuc elde edebilmekteyiz. Yüksek gürültülü verilerde yada sapkın özniteliklerde yüksek negatif değerliklerden dolayı yanlış etiketlendirme yapabilmektedir. Bunun için marjin optimizasyonu yaparak düzenlemeye gidilir.

Giriş : Veri Kümesi

1. Ağırlık vektörünü, $\boldsymbol{W}$:= 0.0
2. ***for*** c:=1 ***to*** *C* ***do***
3. Rastgele bir örnek($\boldsymbol{I_c}$) seç
4. En yakın *k* adet isabet örneği seç (Nhit)
5. ***for each*** $sınıf\ C \neq sınıf(\boldsymbol{I_c})$ ***do***
6. En yakın *k* adet ıska örneği seç (Nmiss)
7. ***for*** i:=1 **to** *f* ***do***
8. $$w_i \coloneqq w_i - \sum_{j:=1}^{k} \frac{\Delta_i(\boldsymbol{I_c}, Nhit_j)}{m.k} + \sum_{c \neq sınıf(\boldsymbol{I_c})} \left(\frac{P(C)}{1 - P(sınıf(\boldsymbol{I_c}))} \right) \sum_{j:=1}^{k} \left(\frac{\Delta_i(\boldsymbol{I_c}, Nmiss_j)}{m.k} \right)$$
9. ***end***

Şekil 4.22 Kabartma algoritması

Bildirimizdeki kullandığımız diğer bir yöntem ise mRMR yöntemidir(Ding ve Peng, 2003). mRMR algoritması sınıf etiketleriyle en ilişkili (relevant) öznitelikleri seçmeye çalışırken eş zamanlı olarak seçilen öznitelikler arasındaki artıklığı, fazlalığı minimize etmeye çalışan bir filtreleme yöntemidir. Seçilmek istenen öznitelik kümesinin seçilebilecek en iyi küme olmasını garanti etmek için iki koşulun karşılanması gerekir. Bunlardan ilki en ilişkili özellik:

$$\max D(S,c), D = \frac{1}{|S|} \sum_{x_i \in S} I(x_i; c) \tag{4.88}$$

diğeri ise minimum artıklık (minimum redundancy) koşuludur:

$$\min R(S) = \frac{1}{|S|^2} \sum_{\{x_i, x_i\} \in S} I(x_i, x_j) \tag{4.89}$$

tabi bu iki özellik tek başlarına yeterli olmayacağından dolayı ikisi arasında ilişki kuracağımız ortak bilgi farkı

$$\max_{x_i \in \{\boldsymbol{X} - \boldsymbol{X_S}\}} I(x_i; c) - \frac{1}{|\boldsymbol{X_S}|} \sum_{x_j \in \boldsymbol{X_S}} I(x_i, x_j) \tag{4.90}$$

ve ortak bilgi oranlarını

$$\max_{x_i \in \{\boldsymbol{X} - \boldsymbol{X_S}\}} I(x_i; c) / \frac{1}{|\boldsymbol{X_S}|} \sum_{x_j \in \boldsymbol{X_S}} I(x_i, x_j) \tag{4.91}$$

bulmalıyız. Bu iki yöntemi birleştirdiğimizde, özniteliğin kalitesini gösteren ve ağırlıklandıran kabartma yöntemi ile mRMR modeli, kabartma algoritmasının ağırlıklandırdığı özniteliklerin sonucunda aşağıdaki şekli alır;

$$\max_{S} \frac{1}{|S|} \sum_{x_i \in S} W_i \tag{4.92}$$

mRMR ise bu ağırlıklandırılmış öznitelikler üzerinden hesaplanır ,

$$\max_{x_i \in \{X - X_S\}} W_i - \frac{1}{|X_S|} \sum_{x_j \in X_S} |P(x_i, x_j)| \tag{4.93}$$

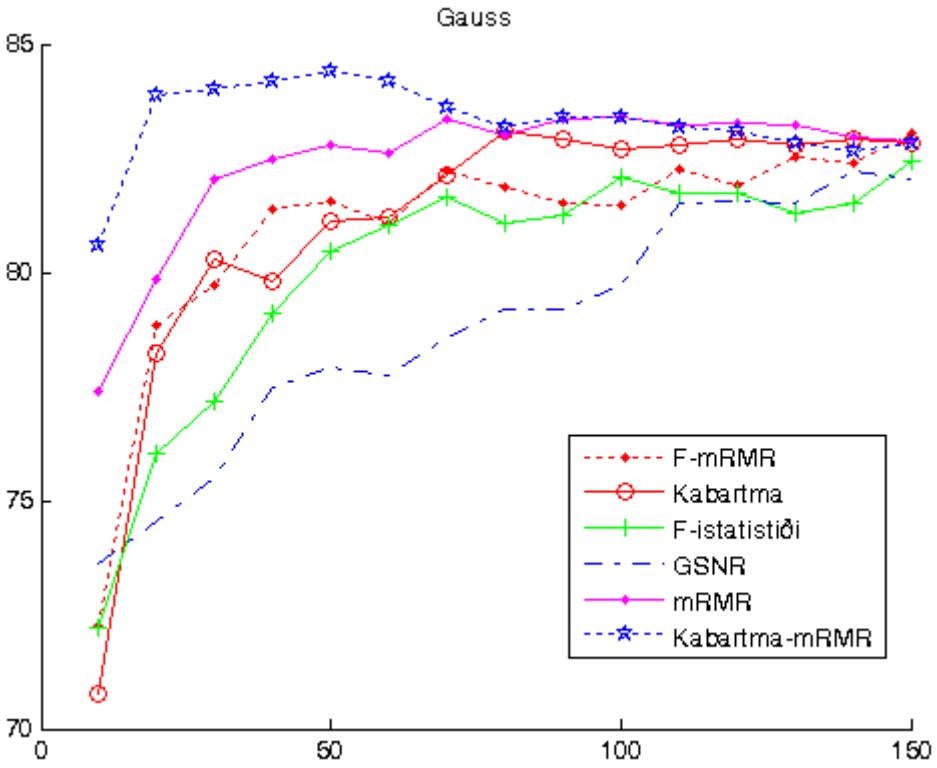
yada

$$\max_{x_i \in \{X - X_S\}} W_i / \frac{1}{|X_S|} \sum_{x_j \in X_S} |P(x_i, x_j)| \tag{4.94}$$

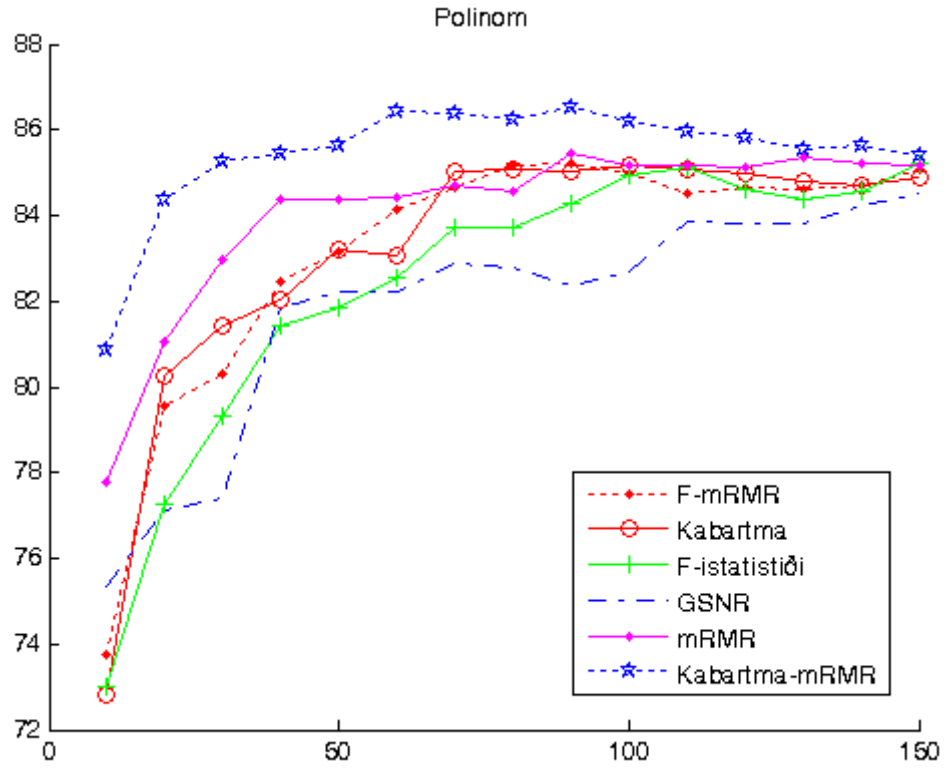
$P(x_i, x_j)$ burda Pearson korelasyon katsayısı olarak tanımlanır.

Kullandığımız bazı öznitelik indirgeme metodlarından F-statistics, kabartma, mRMR, GSNR (Kerr ve Churchill, 2001) ve önerdiğimiz kabartma-mRMR algoritmasını başarımlarını ela almak gerekirse. Özellkle düşük boyutlu özellik seçimlerinde uygulamış olduğumuz yöntemin diğer tek başına kullanılan yöntemlere göre çok daha fazla başarımının yüksek olduğu gözlemlendi.

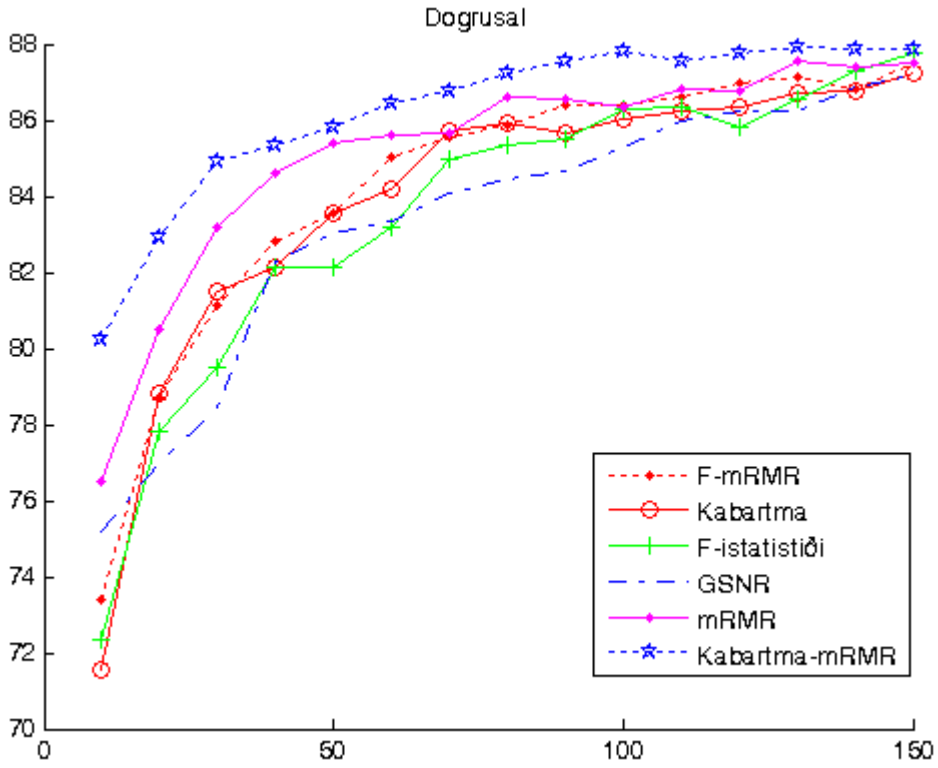
Şekil 4.23'de Cherkasov verisi üzerindeki ilaç verileri üzerinde yeni önerdiğimiz öznitelik indirgeme modelinin başarısı diğer modellere göre daha yüksektir. Çünkü önerilen modelde hem özniteliklerin ağırlıklandırılması sözkonusu hemde ağırlıklandırılan öznitelikler üzerinden özniteliklerin birbirlerine bağımlılıkları ve benzerlikleri ile en az artıklığı bularak ve bunları oranlayarak sonuca gidilmeştir. Böylece asıl gerekli olan öznitelikler ağılıklandırılmış bir önem gözetilerek elde edilir. Polinomsal, Sigmoid ve doğrusal çekirdek modellerinde dikkate değer bir yüksek başarım gözlemlenmektedir. Her türlü veri boyutunda tüm sonuçlara göre daha başarılıdır. Gauss'da ise düşük boyutlu veri kümesinde yada diğer bir tabirle az seçilen özellikler ile yapılan işlemlerde doğruluk oranı diğerlerine nazaran daha yüksek çıkmıştır.


Gauss
85
80
75
70
0
50
100
150
F-mRMR
Kabartma
F-istatistiði
GSNR
mRMR
Kabartma-mRMR

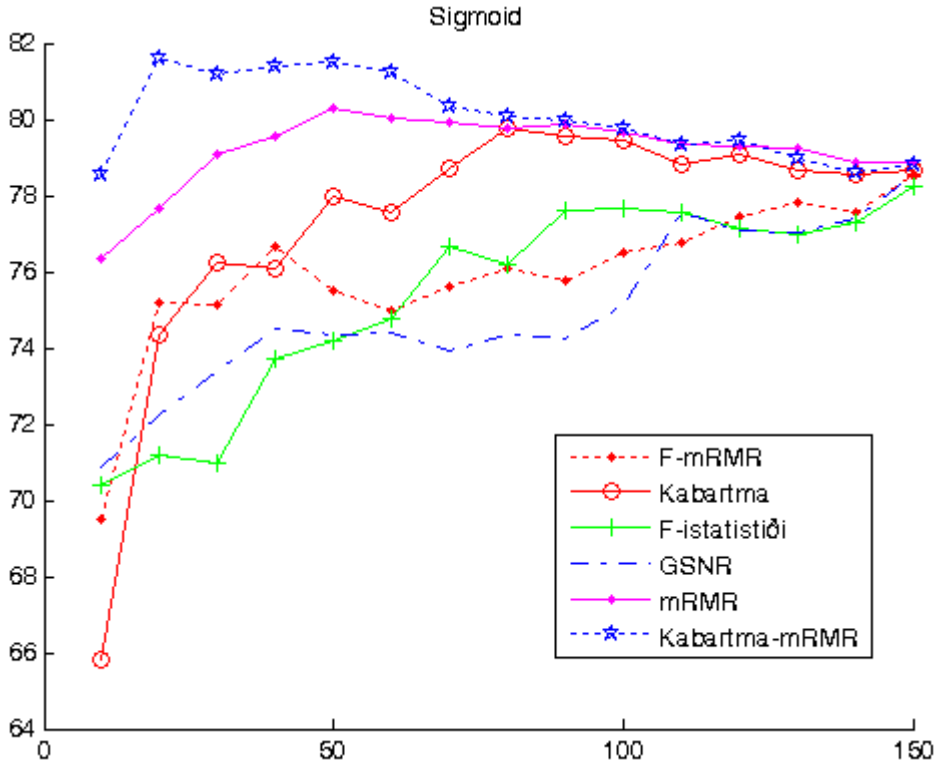
(a)


Polinom
88
86
84
82
80
78
76
74
72
0
50
100
150
F-mRMR
Kabartma
F-istatistiði
GSNR
mRMR
Kabartma-mRMR

(b)



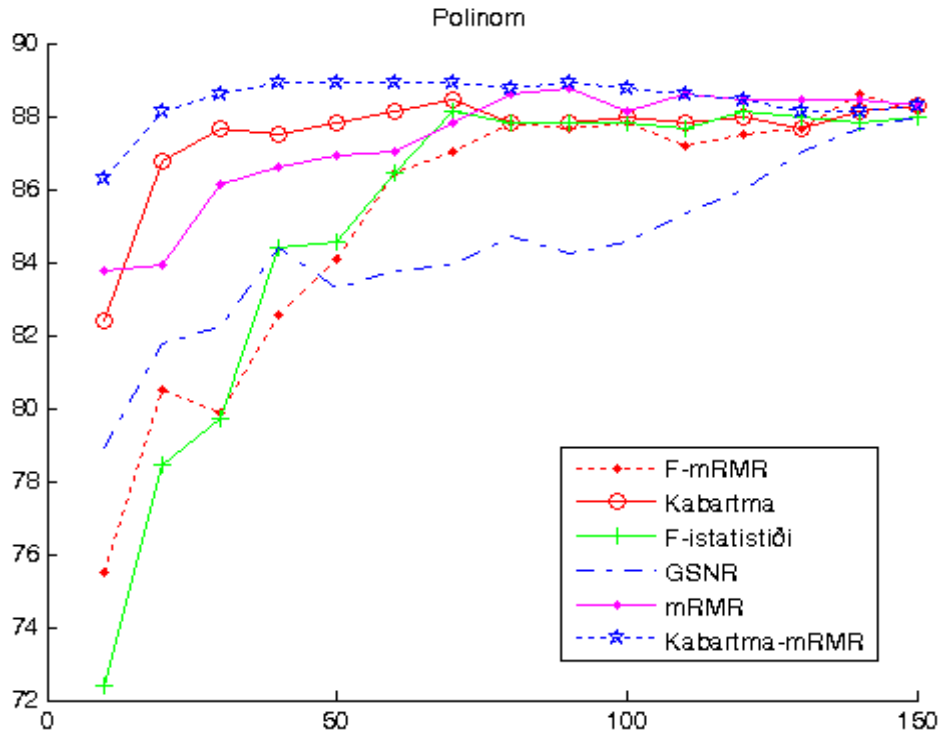
(c)



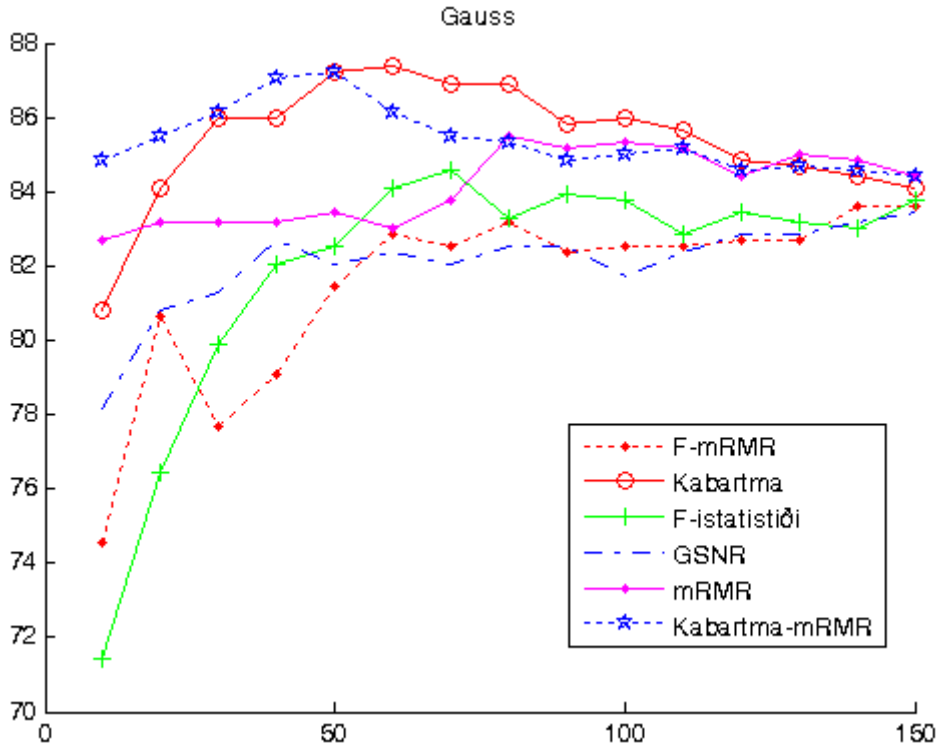
(d)

Şekil 4.23 Öznitelik indirgemesi yöntemi ile Cherkasov verisi başarımları

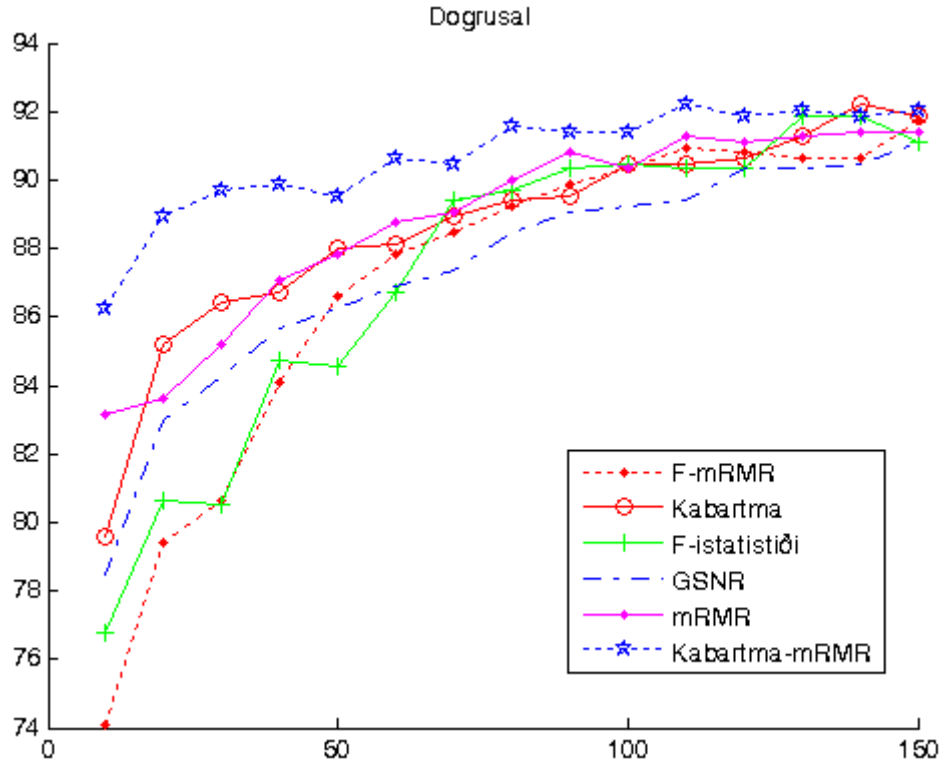
Burda *C=1*, d=3 olarak seçilmiştir. Farklı katsayılar ve farklı modeller için başarımlarını aşağıdaki gibi gösterebiliriz,



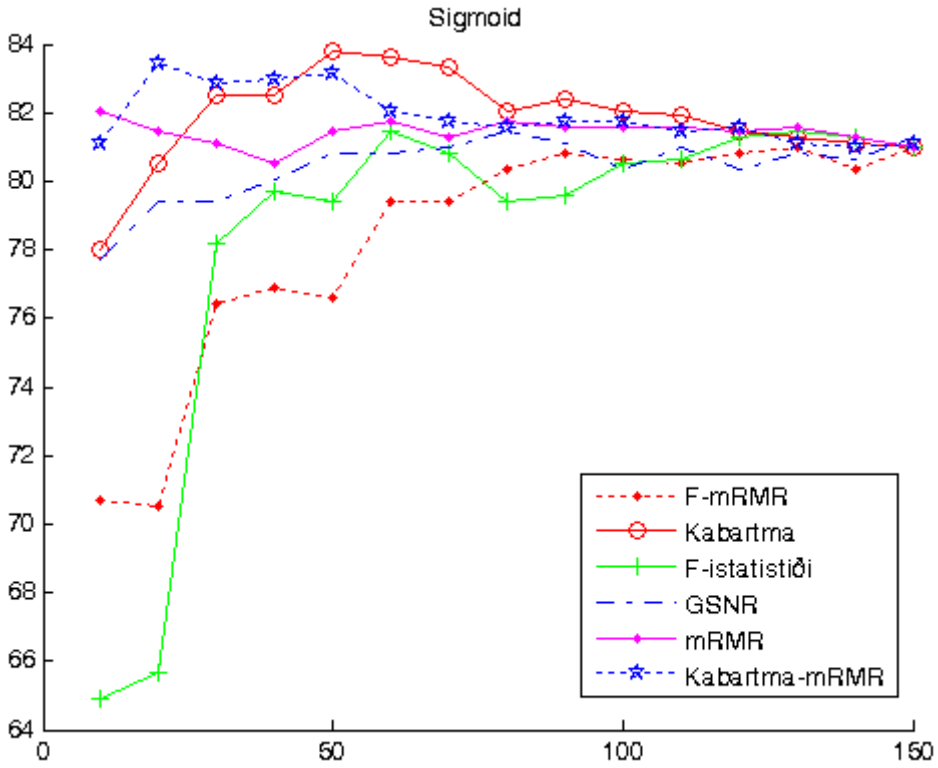
(a)



(b)



(c)



(d)

Şekil 4.24 Öznitelik indirgemesi yöntemi ile Murcia verisi başarımları

Şekil 4.24'deki Murcia-Soler verisinde de benzer özellkler gösteremesine rağmen sigmoid ve Gauss çekirdek modelinde düşük boyut seçimlerinde başarısı yüksek olmasına rağmen öznitelik boyutunu artırdığınızda başarı oranı diğer modellere eşit yada düşük olarak hesaplanmıştır. Bunun sebebi olarak seçiln öznitelikler arasında eşit dağılım olmasından kaynaklandığı düşünülmektedir. Böyle olması durumunda kabartma algoritmasının ağılıklandırması yetersiz kalacak ve model burda kabartma modeli gibi davranacaktır.

Çizelge 4.16 DrugBank veri seti doğruluk oranları

C		Linear	Quadratic	Polinomial	RBF	MLP	IKL (K=Rbf)	IMKL (L + P+ RBF)	IMKL (L + P)
2^{-10}	0,000976563	78,9%	76,6%	73,3%	86,2%	67,8%	66,6%	84,0%	82,1%
2^{-9}	0,001953125	77,2%	72,1%	80,8%	77,6%	58,1%	69,0%	78,4%	80,9%
2^{-8}	0,00390625	74,3%	74,0%	77,3%	79,3%	62,1%	62,2%	80,4%	81,4%
2^{-7}	0,0078125	83,0%	78,5%	76,0%	84,5%	63,9%	75,2%	82,1%	76,7%
2^{-6}	0,015625	83,3%	71,8%	70,7%	75,7%	53,7%	67,9%	75,8%	82,7%
2^{-5}	0,03125	77,0%	77,9%	73,8%	78,7%	55,9%	61,5%	87,3%	76,9%
2^{-4}	0,0625	77,9%	74,5%	74,7%	81,9%	42,6%	63,2%	88,0%	72,8%
2^{-3}	0,125	80,2%	76,5%	72,3%	77,9%	46,7%	63,5%	78,6%	78,3%
2^{-2}	0,25	83,1%	80,8%	72,7%	88,1%	45,3%	62,0%	87,5%	81,5%
2^{-1}	0,5	83,4%	77,7%	71,2%	79,7%	51,5%	64,6%	77,8%	75,9%
2^{0}	1	80,0%	77,0%	75,8%	79,2%	40,8%	61,7%	82,6%	74,2%
2^{1}	2	76,9%	72,1%	76,7%	77,3%	48,4%	68,0%	80,3%	72,3%
2^{2}	4	76,3%	67,7%	75,7%	69,8%	44,5%	70,9%	77,5%	80,0%
2^{3}	8	73,0%	71,9%	75,5%	77,7%	49,3%	66,0%	78,6%	73,2%
2^{4}	16	73,8%	70,2%	78,6%	70,6%	45,6%	67,1%	78,1%	76,2%
2^{5}	32	71,8%	74,2%	79,3%	79,9%	38,9%	71,6%	73,3%	79,3%
2^{6}	64	77,9%	75,7%	79,2%	81,5%	38,4%	60,9%	84,6%	82,7%
2^{7}	128	77,4%	71,3%	78,7%	71,4%	43,3%	65,9%	81,8%	78,7%
2^{8}	256	77,9%	75,9%	73,7%	77,2%	49,1%	58,8%	83,1%	74,8%
2^{9}	512	76,4%	73,3%	71,1%	79,0%	50,6%	69,9%	74,9%	56,8%
2^{10}	1024	73,3%	69,5%	77,2%	75,3%	52,0%	73,5%	79,5%	60,4%
2^{11}	2048	72,2%	76,2%	76,1%	82,8%	52,6%	72,8%	83,5%	62,5%
2^{12}	4096	68,0%	79,2%	73,4%	79,4%	47,9%	73,7%	79,2%	65,1%
2^{13}	8192	70,0%	72,3%	70,9%	72,6%	47,9%	68,1%	80,5%	61,4%
2^{14}	16384	81,7%	72,0%	75,6%	78,5%	35,4%	66,4%	82,7%	63,5%
2^{15}	32768	78,5%	78,3%	75,9%	87,0%	57,0%	72,7%	88,3%	58,4%

En yüksek		83,4%	80,8%	82,1%	88,1%	70,5%	75,2%	88,3%	85,6%
En düşük		54,9%	67,7%	70,7%	69,8%	35,4%	58,8%	73,3%	56,8%
Ortalama		74,4%	74,5%	75,8%	79,0%	51,3%	66,4%	80,9%	73,0%
Standart Sapma		6,5%	3,2%	2,9%	4,3%	8,8%	4,3%	3,9%	8,3%

Çizelge 4.17 Pharmeks veri seti doğruluk oranları

C		Linear	Quadratic	Polinomial	RBF	MLP	IKL (K=Rbf)	IMKL (L + P+ RBF)	IMKL (L + P)
2^{-10}	0,000976563	64,4%	62,1%	65,7%	78,8%	63,0%	66,1%	80,7%	75,3%
2^{-9}	0,001953125	65,4%	65,4%	68,5%	75,4%	55,5%	62,4%	77,7%	79,9%
2^{-8}	0,00390625	68,5%	69,1%	70,4%	70,0%	59,9%	56,1%	76,1%	72,7%
2^{-7}	0,0078125	71,5%	74,6%	69,0%	76,6%	58,5%	67,1%	81,3%	69,1%
2^{-6}	0,015625	80,6%	70,6%	65,0%	74,0%	44,7%	64,7%	75,6%	73,5%
2^{-5}	0,03125	73,0%	70,6%	71,6%	72,7%	49,2%	57,9%	80,2%	74,2%
2^{-4}	0,0625	77,8%	67,6%	72,8%	80,1%	40,8%	60,7%	86,2%	68,9%
2^{-3}	0,125	71,0%	74,4%	68,6%	76,4%	42,2%	55,4%	70,6%	72,9%
2^{-2}	0,25	77,8%	71,0%	64,6%	79,6%	44,1%	53,8%	86,9%	79,3%
2^{-1}	0,5	80,5%	70,6%	61,4%	75,3%	45,7%	63,2%	76,3%	71,7%
2^{0}	1	75,2%	76,8%	68,3%	71,7%	39,7%	57,4%	81,8%	67,3%
2^{1}	2	70,9%	72,0%	70,3%	70,7%	39,2%	60,5%	76,6%	62,6%
2^{2}	4	72,2%	65,3%	71,9%	69,0%	37,7%	65,6%	73,2%	77,5%
2^{3}	8	64,9%	66,9%	74,3%	75,1%	45,6%	62,0%	74,9%	71,6%
2^{4}	16	73,6%	63,1%	69,6%	67,8%	41,7%	57,8%	73,4%	69,7%
2^{5}	32	68,3%	67,2%	73,3%	76,5%	38,4%	68,7%	70,9%	77,6%
2^{6}	64	72,9%	66,0%	70,4%	80,9%	30,4%	57,2%	82,2%	81,2%
2^{7}	128	73,5%	65,1%	75,9%	70,4%	42,3%	63,8%	72,0%	69,2%
2^{8}	256	73,1%	73,4%	68,1%	71,8%	48,1%	48,9%	80,6%	67,5%
2^{9}	512	75,3%	67,4%	66,3%	70,2%	50,1%	69,2%	72,2%	53,8%
2^{10}	1024	63,7%	63,8%	68,3%	71,7%	45,1%	63,6%	70,1%	59,5%
En yüksek		80,6%	76,8%	75,9%	80,9%	63,0%	69,2%	86,9%	81,2%
En düşük		63,7%	63,1%	61,4%	67,8%	30,4%	48,9%	70,1%	53,8%
Ortalama		73,7%	69,2%	69,5%	74,0%	45,8%	61,0%	77,1%	71,2%

Çizelge 4.16 ve 4.17'den da anlaşılacağı üzere genel olarak ifade edilen *C* parametresne bağımlılık burda da gözlemlenmektedir. Yine tekil çekirdek seçimi yerine çoklu çekirdek

modellerin seçimi başarımı artırmıştır. Hatta yeni önerdiğimiz Artımlı çoklu çekirdek modeli burda en yüksek doğruluk oranı sonucuna erişmiştir. Bunun sebebi olarak çoklu çekirdeklerden uygun çekirdeklerin seçimi oluşturulan aşırıdüzlem için gerekli destek yöneylerin seçimini etkilemesidir. MKL'de sadece iki çekirdek yani doğrusal ve polinomsal çekirdek kullanılırken, IMKL (Artımlı) de ise üç farklı çekirdek (doğrusal, polinomsal ve Gauss) kullanılmıştır.

Pharmeks veri kümesini ele aldığımızda 250.000'den fazla ilaç bileşiği içermektedir. Önerdiğimiz çekirdek modellerinde özellikle artımlı modellerin başarımı daha yüksek çıkmaktadır ama yinede genel olarak incelendiğinde diğer tüm veri kümelerine nazaran daha düşük bir doğruluk oranı hesaplanmaktadır.

5. YARI GÖZETİMLİ ÖĞRENME

Uygun işlev yada kümeleyici oluşturmak için etiketli ve etiketsiz verileri birlikte kullanan bir öğrenme şeklidir. Özellikle yüksek boyutlarda işlemler gerektiren problemlerde (metin sınıflandırma, protein yapılarının tahmini, mikrodizin tanımı, web sayfalarının sınınflandırılması, ilaç dizayn, vb...) gözetimli öğrenmeye göre tüm bilgilerin etiketinin bilinmesine gerek olmadığından çok daha az işlem karmaşıklığı ortaya çıkaraktadır. Bunun yanında sadece etiketli verileri kullanarak eğitilen klasik sınıflandırıcıların aksine yarı gözetimli öğrenmede etiketli verilerin yanında etiketsiz verilerde öğrenme ve sınıflandırma işlevinde kullanılırlar. Diğer gözetimli öğrenme yaklaşımlarında tüm verilerin hangi sınıfa ait olduğu ve özelliklerinin bilinmesi zorunluluğu bulunmaktadır.

Yarı gözetimli öğrenmenin iskeleti, verilen $\chi = \{x_1, \dots, x_e, x_{e+1}, \dots, x_n\} \in \mathbb{R}^n$ veri seti için x_i $(i \leq e)$ etketli e adet veri içinden bir giriş olmak üzere ve x_b $(e+1 \leq b \leq n)$ etiketi bilinmeyen b adet veri içinden bir giriş olmak üzere toplam n adetlik veri setinin öğrenilerek etiketsiz verilerin etiketlerinin çıkış olarak belirlenmesi kuralına dayanır. Etiketleri $y_i \in \{-1, +1\}$ şeklinde ve $e << b$ olarak tanımlamak mümkündür.

5.1 Giriş

Yarı gözetimli öğrenmede ilk fikir etiketsiz verilerin kendi kendine öğrenmesi yada etiketlemesi (self-training) modelidir. Aslıdna bu modelde yinelemeli olarak gözetimli bir algoritma ile eğitme modeline dayanmaktadır. Akademik yazımda Scudder(1965), Fralick(1967), Agrawala(1970)'ın yapmış olduklaı çalışmalar ilk olarak yarı gözetimli öğrenme benzeri modellerdir. Gerçek manada yarı gözetimli öğrenme modeli ile tam uyumlu yöntemlerden "Uyarlanabilir Destek Vektör Makinaları(Transductive-SVM)" modeli Vapnik ve Chervonenkis(1974) ile Vapnik ve Sterin(1977) tarafından önerilmiştir. 1970'li yıllarda Fisher doğrusal ayırtaç fonksiyonu ile etiketsiz verilerin etiketlenmesi önerilmiştir(Hosmer, 1973; McLAchlan, 1977; O'neill, 1978; McLachlan ve Ganesalingam, 1982). Olasılıksal modellerden Gauss tabanlı beklenen ençokluk fonksiyonu(Dempster vd., 1977), katlıterimler dağılımı karışımı (Cooper ve Freeman, 1970) ve iki Gauss dağılımının karışımı (Ratsaby ve Venkatesh, 1995) temeline dayanan modeller de etiketli ve etiketsiz verileri kullanarak öğrenme yapan algoritmalardır. 1990'lı yılların başında yüksek boyutlu veriler ve gerçek zamanlı işlemler üzerinde (doğal dil işleme, ses ve konuşmacı tanıma, metin sınıflandırma, resim işleme vb...) çalışmalar hızlanmıştır (Board ve Pitt, 1989; Yarowsky, 1995; Castelli ve

Cover, 1996; Nigam vd.,1998; Blum ve Mitchell, 1998; Collins ve Singer, 1999; Joachims, 1999; Seeger, 2001).

Biz burda tanımlanan tüm yarı gözetimli algoritmalar yerine yazmış olduğumuz bildirilerde karşılaştırmak için ele aldığımız çizge tabanlı yarı-gözetimli öğrenme metodlarından "Rastgele Gauss Alanları" (Zhu vd., 2003), "Yerel ve Genel Tutarlılık Modeli" (Zhou vd., 2004) ve "Düşük Yoğunluk Ayrımı" (Chapelle ve Zien, 2005) modelleri ile geliştirmiş olduğumuz "Etkin Yarı Gözetimli Öğrenme" modelini anlatacağız.

5.2 Algoritmalar

5.2.1 Rastgele Gauss Alanları Modeli

V düğümler ve ***E*** düğümler arası yolları göstermek üzere ***G* = (*V*, *E*)** çizgesi üzerinde tanımlı *X* veri seti için **G** çizgesinde tanımlı reel-değerli bir fonksiyon olan $f: \mathcal{V} \rightarrow \mathbb{R}$ için $\mathcal{V} = \mathcal{L} \cup \mathcal{U}$ şeklinde tanımlıdır. Burda $\mathcal{L}$ etiketli verileri, $\mathcal{U}$ etiketsiz verileri göstermektedir. İkinci dereceden maliyet fonksiyonunu

$$E(f) = \frac{1}{2}\sum_{i,j} w_{ij}\left(f_i - f_j\right)^2 \qquad (5.1)$$

ile bulabiliriz. $f_j = y_i$ ve $i = 1, \ldots, e$ ile çıkış bilgisini göstermektedir. Algoritmanın adımları aşağıdaki tanımlanabilir (Zhu vd., 2003) :

- Benzerlik matrisi $\boldsymbol{\mathcal{W}}$'yi oluşturulur $(Not: \mathcal{W}_{ii} = 0)$
- Köşegen matris $\boldsymbol{\mathcal{D}}$'yi $\mathcal{D}_{ii} = \sum_{j=1}^{l} w_{ij}$ kurulur.
- $\boldsymbol{\mathcal{W}}$ ve $\boldsymbol{\mathcal{D}}$ matrislerini $\boldsymbol{\mathcal{W}} = \begin{bmatrix} \mathcal{W}_{ee} & \mathcal{W}_{eb} \\ \mathcal{W}_{be} & \mathcal{W}_{bb} \end{bmatrix}$, $\boldsymbol{\mathcal{D}} = \begin{bmatrix} \mathcal{D}_{\mathrm{ee}} & 0 \\ 0 & \mathcal{D}_{\mathrm{bb}} \end{bmatrix}$ şeklinde tekrar düzenlenir ve $\boldsymbol{f} = [f_e \quad f_b]^T$ çıkış verileri oluşturulur.
- Etiketsiz verileri $\boldsymbol{f_b} = (\mathcal{D}_{\mathrm{bb}} - \mathcal{W}_{bb})^{-1}\mathcal{W}_{be}f_e$ ile çözülür.
- $i = b$ için$f_i > 0$ için $y_i = +1$ ve $y_i = -1$ şeklinde etiketlenir.

Görüldüğü üzere algoritmanın ilk etiketli değerlere bağımlılığı söz konusudur. Buna ek olarak dengeli etiketli sınıfların bulunması da modelin başarımını çok büyük ölçüde etkilemektedir. Akademik yazımda bu konularda bazı çalışmalar yapılmaktadır.

5.2.2 Yerel ve Genel Tutarlılık Modeli

Zhou vd.(2004) tarafından oluşturulan bu modelde,

- Benzerlik matrisi $\boldsymbol{\mathcal{W}}$ oluşturulur (Not: $\mathcal{W}_{ii} = 0$)
- Köşegen matris $\boldsymbol{\mathcal{D}}$'yi $\mathcal{D}_{ii} = \sum_{j=1}^{l} \mathcal{w}_{ij}$ kurulur.
- $\boldsymbol{\mathcal{S}}$ matrisini oluşturulur ($\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{D}}^{-1/2}\boldsymbol{\mathcal{W}}\boldsymbol{\mathcal{D}}^{-1/2}$)
- Bir değere yakınsayana kadar özyinelemeli olarak $\boldsymbol{F}(t+1) = \gamma \boldsymbol{\mathcal{S}}\boldsymbol{F}(t) + (1-\gamma)\boldsymbol{\Upsilon}$, fonksiyonu çözülür. Burda $\gamma \in (0,1)$ ve t anlık adımı belirtmektedir.
- Elde edilen $\boldsymbol{F}^*$ için her x_i için eğer $F_{ij} > F_{ik}\ (k \neq j)$ sağlıyorsa $y_i = j$ şeklinde etiketlenir.

Özyinelemeli çözüm aslında $\boldsymbol{F}^*$ en iyileme sonucun maliyet fonksiyonunu (5.2) en azlamak için kullanılır.

$$\min Cost(\boldsymbol{F}) = \frac{1}{2}\left(\sum_{i,j=1}^{n} W_{ij}\left\|\frac{1}{\sqrt{D_{ii}}}F_i - \frac{1}{\sqrt{D_{jj}}}F_j\right\|^2 + \tau\sum_{i=1}^{n}\|F_i - Y_i\|^2\right) \tag{5.2}$$

Denklemin ilk bölümü yumaşaklık (smothness) kısıtı, ikinci kısım ise uygunluk(fitting) kısıtıdır. Bu maliyet en azlama fonksiyonunu düşük boyutlu veriler için matris formuna dönüştürürsek,

$$\min Cost(\boldsymbol{F}) = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{\mathcal{L}}\boldsymbol{F} - \boldsymbol{Y}^T\boldsymbol{F}) \tag{5.3}$$

şeklini alır. Burda $\boldsymbol{\mathcal{L}}$ normalize edilmiş Laplacian matrisidir ve $\boldsymbol{\mathcal{L}} = \boldsymbol{I} - \gamma\,\boldsymbol{\mathcal{S}}$ şeklinde tanımlanır ve $\tau > 0$ düzenleme parametresi ile γ arasında $\gamma = \frac{1}{1+\tau}$ şeklinde bir ilişki vardır çünkü ;

$$\left.\frac{\partial Cost(\boldsymbol{F})}{\partial \boldsymbol{F}}\right|_{F=F^*} = \boldsymbol{F}^* - \boldsymbol{\mathcal{S}}\boldsymbol{F}^* + \tau(\boldsymbol{F}^* - \boldsymbol{Y}) = 0 \tag{5.4}$$

ve

$$(I - \gamma\boldsymbol{\mathcal{S}})\boldsymbol{F}^* = \boldsymbol{Y} \tag{5.5}$$

olduğundan (5.4) ve (5.5) ortak çözümünden $\gamma = \frac{1}{1+\tau}$ elde edilir.

5.2.3 Düşük Yoğunluk Ayrımı

Yoğunluk duyarlı uzaklık ölçümü tanımını Chapelle ve Zien (2005), parzen pencereleri yöntemi genelleştirmesi ve uyum sağlayan DVM'leri üzerinde yapmıştır. p yolu $\mathcal{G} = (\mathcal{V}, E)$. grafiği üzerindeki, $(p_k, p_{k+1}) \in E$ ve $1 \leq k \leq |p|$ olmak üzere l boyutlu bir $\mathcal{V}$ noktalar kümesi üzerindeki bir yol için, $p \in \mathcal{V}^l$, $l =: |p|$ şeklinde tanımlıdır. $P_{i,j}$, x_i ve x_j arasındaki tüm yolları göstermek üzere,

$$\max_{p \in P_{i,j}} \min_{k < |p|} \hat{p}\left(\frac{1}{2}\left(x_{p_k} + x_{p_{k+1}}\right)\right) \tag{5.6}$$

Yaklışık bir sonuç için,

$$\begin{aligned} &\approx c.exp\left[-\frac{1}{2\sigma^2}\left(\min_{p \in P_{i,j}} \max_{k < |p|} d(p_k, p_{k+1})\right)^2\right] \\ &\equiv k(x_i, x_j) \end{aligned} \tag{5.7}$$

şeklinde pozitif yarı tanımlı çekiredek metodu elde edilmiş olur. Bu çekirdek metoda "*bağlantısal çekirdek*" denir. Çekirdekler yolların boyutuna bağlı değildir. Bunu sağlamak ve denklemdeki en çoklama terimini yumuşatmak için aşağıdaki ifade kullanılabilir

$$max^{\rho}(p) \coloneqq \frac{1}{\rho} ln\left(1 + \sum_{k=1}^{|p|-1}\left(e^{\rho d(p_k, p_{k+1})} - 1\right)\right) \tag{5.8}$$

$\rho \to \infty$ için denklem elde edilmiş olur. Eğer $\rho \to 0$ olur ise, $max^{\rho}(p)$ değeri yol $p \in P_{i,j}$ üzerindeki noktalar arasındaki uzaklıkların toplamına eşit olur. Üçgen eşitsizliğinden herhangi iki nokta arasındaki uzaklık hiç bir zaman $d(i,j)$'den yani öklit uzaklığından küçük olamaz, $\|x_i - x_j\|_2$ şeklinde ifade edilebilir. Böylece Gauss RBF çekirdeği kendiliğinden oluşmuş olur. Seyrek çizelge tabanlı en küçük yol uzaklığı $\rho \to 0$ için Isomap'e eşit olur (Chapelle ve Zien, 2005).

Algoritma 4 temel adımdan oluşmaktadır. Bunlar sırasıyla :

- Etiketli ve etiketsiz veriler için *G* matrisini oluşturmak
- Etiketli noktalardan tüm noktalara olan uzaklık matrisi $D^{\rho}{}_{[lx(l+u)]}$, $\rho - yol$ enazlama ile bulunur(5.9).

$$D_{i,j}^{\rho} = \frac{1}{\rho^2} ln\left(1 + \min_{p \in P_{i,j}} \sum_{k=1}^{|p|-1} \left(e^{\rho d(p_k, p_{k+1})} - 1\right)\right)^2 \quad (5.9)$$

- Doğrusal olmayan çekirdek K dönüşümünü $K_{i,j} = e^{\left(-D_{i,j}^{\rho}/2\sigma^2\right)}$ ile elde edilir. Doğrusal dönüşüm için $\sigma \to \infty$ olur ise, $K = -\frac{1}{2} H^l D^{\rho} H^{(l+u)}$ şeklini alır, burda *mxm* lik bir merkezcil matris olan $\boldsymbol{H}^m = \boldsymbol{I}_m - \frac{1}{m}\mathbf{1}.\mathbf{1}^T$ ile elde edilir. $\mathbf{1}$ bir sütun vektördür.
- Modeli elde edilen K dönüşüm çekirdeğini DVM ile eğitip tahmin gerçekleştirilmiş olur.

5.3 Önerilen Etkin Yarı Gözetimli Öğrenme Modeli

Bu modelimizde etkin öğrenme ile yarı gözetimli öğrenme modelleri birleştirilerek, sorgu bazlı uyarlanabilir çoklu etiketleme yapan bir etkin öğrenme modeli önereceğiz. Roy ve McCallum (2001)'ın çalışmasında etken öğrenme modellerinden, rastgele seçim modeli, yoğunluk tabanlı sorgulama (McCallum ve Nigam, 1998), öğrenicinin en az şüpheli örneği seçmesi modeli (Lewis ve Gale, 1994), komite tabanlı sorgulama (Seung vd., 1992; Freund vd., 1997) modeli ve Bayes tabanlı (Cauwenberghs ve Poggio, 2001) etken öğrenme modelleri kıyaslamalı olarak ele almıştır. Bizim etken öğrenme modelimiz ise Zhu vd.(2003a, 2003b)'in sorgulamasında kullandığı Bayes tabanlı bir etken öğrenme modeli olmakla birlikte farklı olarak k-adet sorgusal örneğin etken olarak etiketlenmesi durumu söz konusudur.

$\boldsymbol{\mathcal{V}}$ düğümler ve E düğümler arası yolları göstermek üzere G = ($\boldsymbol{\mathcal{V}}$, E) çizgesi üzerinde tanımlı $(x_1, y_1), \ldots, (x_e, y_e)$ etiketli ve $x_{e+1}, \ldots, x_{e+b}$ etiketi belirsiz $e \ll b$ olmak üzere X veri seti olsun. G çizgesinde tanımlı reel-değerli bir fonksiyon olan $f: \mathcal{V} \to \mathbb{R}$ için etiketli ve etiketi belirsiz örnekleri içeren düğümler $\mathcal{V} = E \cup \mathcal{B}$ şeklinde tanımlıdır. Burda E etiketli verileri, $\mathcal{B}$ etiketi belirsiz verileri göstermektedir. GRAM modelinde f ikinci dereceden enerji fonksiyonu (5.1) aynı zamada maliyet fonksiyonu olarak kullanılmaktadır. Olasılık dağılımını Gauss rastgele alan fonksiyonu ile gösterebiliriz ;

$$p_{\beta}(f) = \frac{e^{-\beta E(f)}}{Z_{\beta}} \quad (5.10)$$

ile belirtebiliriz. β evrik sıcaklık parametresini, Z_{β} ise tüm etiketli veriler üzerindeki ifadeyi normalize edecek bölümleme fonksiyonu olan $Z_{\beta} = \int_{f|L=f_e} e^{-\beta E(f)} df$ bilgisini verir. Burda

enerjiyi enazlayacak *E(f)* fonksiyonu harmonikdir. Yani etiketsiz veriler için $(D-W)f=0$ ve etiketli veriler için ise $f_j = y_i$ ve $i = 1, \dots, e$ ile çıkış bilgisini göstermelidir. Oluşan $\Delta = (D-W)$ matrisi tümleşik Laplacian matrisi ile ifade edilir. Harmonik özelliği komşuluk etiketi belirsiz noktaların ortalaması $f(j) = (1/d_j)\sum_{i\sim j} w_{ij} f(i), \forall i \in \mathcal{B}$ şeklinde ifade edebiliriz. İlginlik matrisini $W_{ij} = e^{(1-x_i{}^T x_j/\|x_i\|\|x_j\|)}$ ile hesaplayabiliriz. Harmonik çözüm için $h_e = Y_e$ şartına uygun olarak $\Delta h = 0$ çözümüne bakılır $h_b = (D_{bb} - W_{bb})^{-1} W_{be} Y_e$.

Etken öğrenme modellerinden riski en azlayacak sorguyu seçme modeli yöntemi ile etken öğrenme gerçekleştirilmesi hedeflenmişir. Diğer rastgelelik yada komite tabanlı sorgulamalar yerine böyle bir yöntemle k-adet sorgu örneğinin etiketlenmesi başarımı artırdığı gözlemlenmişir. Harmonik enerjiyi en azlayacak gerçek risk fonksiyonunu Bayesian sınıflandırıcı hatasını genişleterek bulabiliriz.

$$Risk(f) = \sum_{i=1}^{n} \sum_{y_i=0,1} [sgn(f_i) \neq y_i] p^*(y_i|E) \tag{5.11}$$

Burda gerçek risk fonksiyonunda $p^*(y_i|E)$ bilinemediğinden dolayı bazı varsayımlarda bulunacağız. Etiketli örnekler için $p^*(y_i|E) \approx f_i$ kabul edilerek yinelemeli olarak yeni bir risk hesaplanır.

$$Risk'\left(f^{+(v_k)}\right) = \sum_{i=1}^{n} \min\left(f_i{}^{+(v_k)}, 1 - f_i{}^{+(v_k)}\right) \tag{5.12}$$

Böylece *k* adet $Risk'\left(f^{+(v_k)}\right)$ en azlayan örnek sorguya katılarak son etiket bilgisi elde edilir. Etiket bilgisini çizge üzerinde güncellemek için Sherman-Morrison-Woodbury formulünden yararlanılır. Algoritmanın genel olarak adımları,

---

Giriş : Etiketli $\{E\}$ ve Belirsiz $\{B\}$ veriler, risk fonk. $Risk'(.)$
Çıkış : Etiketsiz verilerin etiket vektörü
1. $\mathcal{W}$ ve $\mathcal{D}$ matrisleri oluşturulur
2. while $s\{E\} \neq 0$
3. $Risk'(.)$ hesapla
4. En iyi *k* adet $x_i \in \{B\}, i := 1, \dots, k$ bul
5. $\forall x_i$ için GRAM ile sorgula ve y_i sınıfını belirle
6. $\{E\} = \{E\} \cup \langle x, y \rangle_i$
7. $\{B\} = \{B\} - \langle x, y \rangle_i$
8. Çizgeyi güncelle
9. end

Şekil 5.1 Etkin k-sorgulamalı Yarı Gözetimli Öğrenme

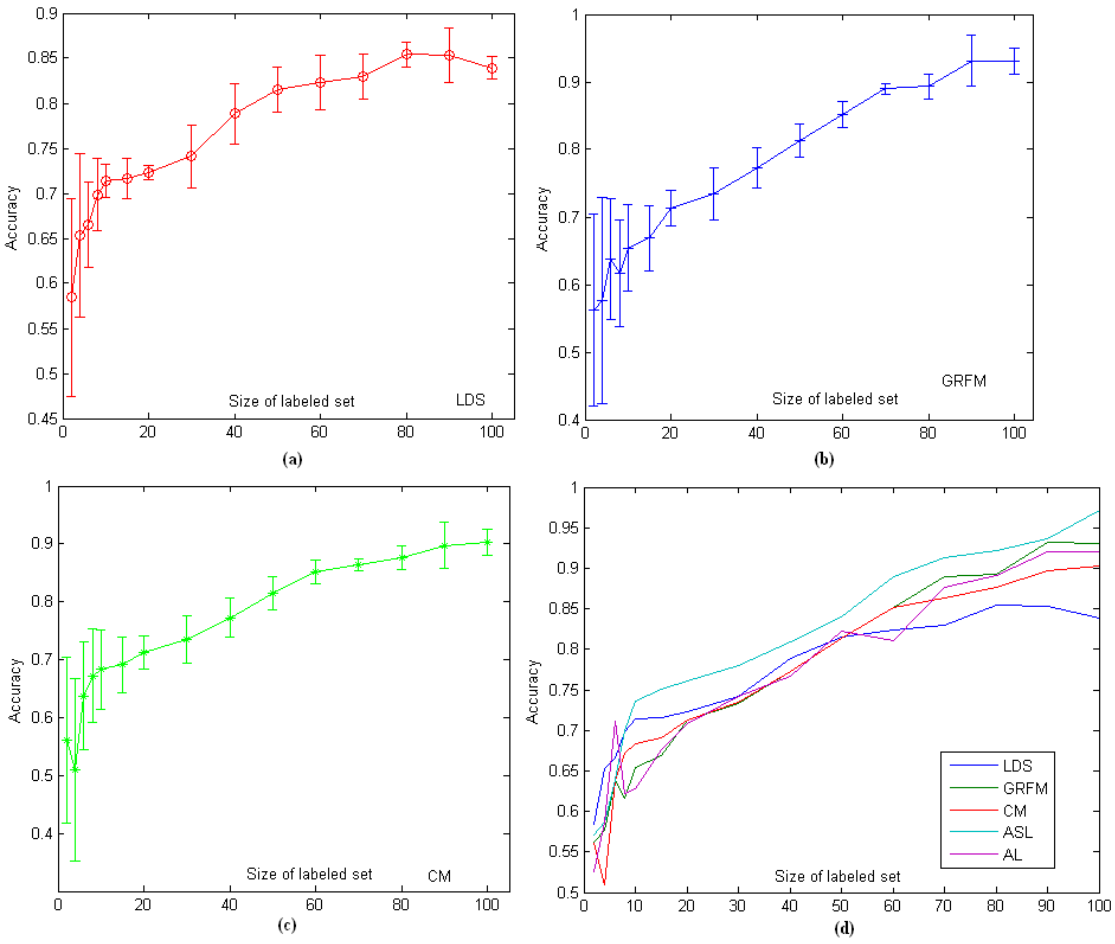
şeklinde ifade edebiliriz. Dengesiz dağılıma sahip etiketli verilerin etkisini azaltmak için sütun vector F_j^* yi ortalaması 0 ve standart sapması 1 olan $\hat{F}_j^*$ ile güncellemek gerekir.

$$\hat{F}_j^* = \frac{F_{ij}^* - \bar{F}_j^*}{s_j} \; ve \; s_j = \sqrt{\frac{\sum_{i=1}^{n}\left(F_{ij} - \bar{F}_j\right)^2}{n}} \tag{5.13}$$

Normalizasyon adımı uygulındıktan sonra standart çıkış matrisi $\hat{\boldsymbol{F}}^*$ sonunda satılardaki en iyileme değerini etiketi belirsiz *i.* veriyi sınıf *j* ye atanabilmesi için $\hat{\boldsymbol{F}}_{ij}^* > \hat{\boldsymbol{F}}_{ik}^*$, k=1,..., c, $k \neq j$ şartını sağlaması gerekmektedir.

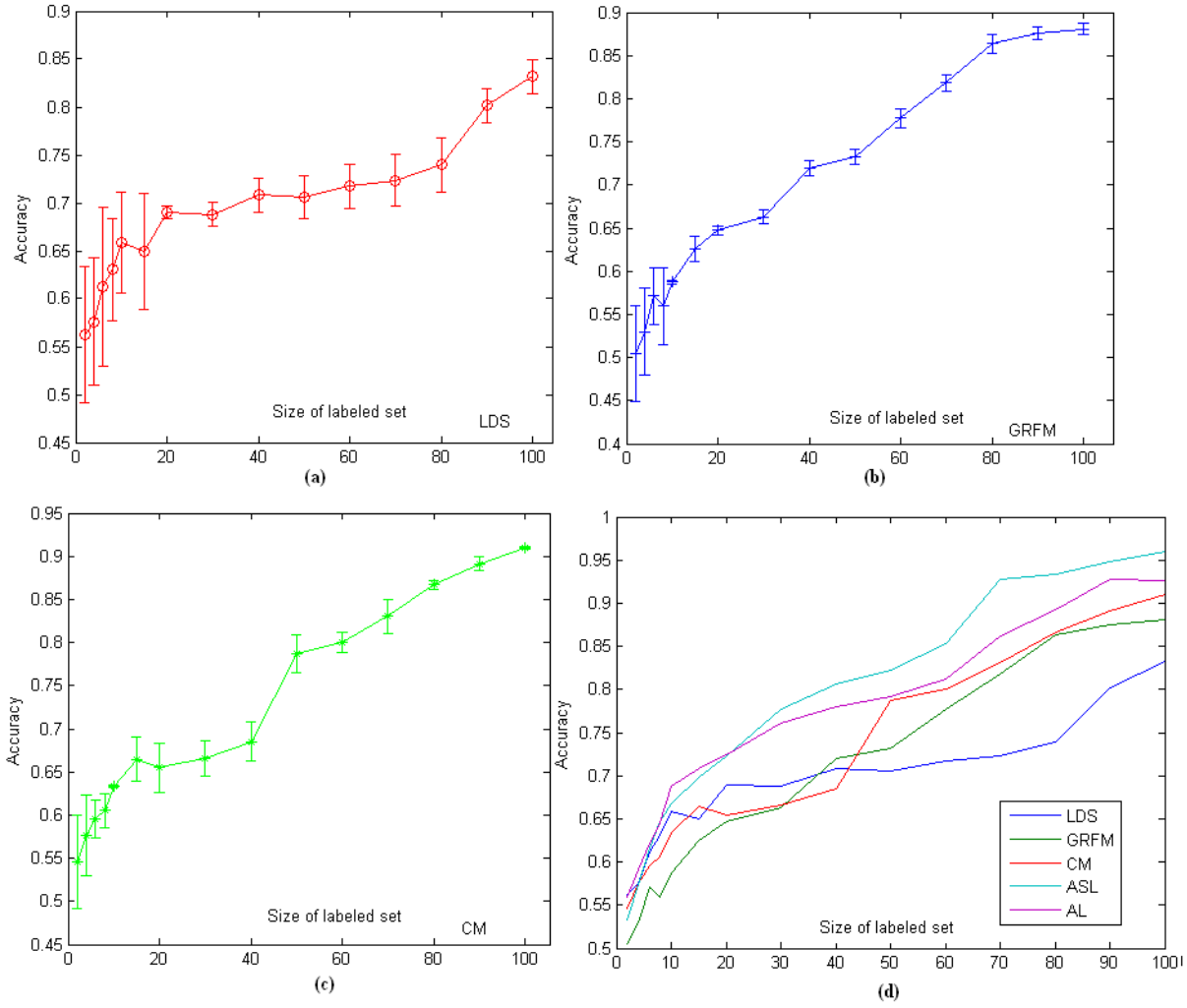
5.4 Modelin Başarımı ve Sonuçlar

Burda ele aldığımız Düşük Yoğunlu Ayrımı (LDS), Tutarlılık model (CM), Gauss Rastgele Alan Modeli(GRFM), Etkin Öğrenme(AL) ve önerdiğimiz Etkin Yarı Gözetimli Öğrenme modelinin(ASL) sonuçları gösterilmiştir.



Şekil 5.2 Cherkasov verisi etiketli veri - doğruluk oranı

LDS(düşük yoğunluk ayrımı), ASL(etken yarı gözetmli öğrenme), AL(etken öğrenme, en yakın komşuluk), Gauss modelleri ele alınan Şekil 5.2'de etiketini bildiğimiz verilerin sayısına göre yine 10 katlı çarpraz geçerleme yöntemi kullanılarak eğitilen verimizin doğruluk şemaları ile standart sapma oranları gösterilmiş olup. Çok yüksek oranda olmasan genel olarak önerdiğimiz etkin yarı gözetimli öğrenme modelinin diğer modellere göre daha iyi sonuçlar verdiği görülmüştür. Özellikle daha düşük sayıda bilinen etiketli veriler kümesindeki başarı oranı diğerlerine göre daha yüksektir. Bunun sebebi her öğrenme adımında en az hatasal oranı olan veri kümesi alındığından tekrar eğitilen bu verileri modelle eğittimizde aralarında ayrık olan verileri daha iyi öğrenebilmektedir. Hata payını da azaltmış olmaktayız.



Şekil 5.3 Drugbank verisi etiketli veri - doğruluk oranı

Benzer şekilde yukarıdaki şekil incelendiğinde önerdiğimiz model diğer modellere göre çok daha iyi başarı göstermektedir. Diğer modellerin teorisinde verilerin dağılımı sözkonusu bizimkinde ise her öğrenme adımında k adet en az hata oranı olan veri kümesini tekrar ele alıp

eğitme sözkonusu olduğundan dağılımın etkisini azaltmış böylece hata oranını da düşürmüş olmaktayız.

6. PROTEİN DİZİLERİ ve İŞARETLERİN SAYISALLAŞTIRILMASI

Bu bölümde protein yapılarının tespiti için önediğimiz iki model ele alınacaktır. ilkinde motif tabanlı bir öğrenme, diğerinde ise Saklı Markov Modeller üzerinden durum tabanlı bir skorlama modelidir. Bu öğrenme modellerini anlatmadan önce öncelikle protein nedir? Ne iş yapar? Niçin yapılarını ve işlevlerini tahmin etmeye ihtiyacamız var? sorularının yanıtını verelim ve proteinlerle ilgili gerekli bazı temel kavram ve bilgileri ele alalım.

6.1 Proteinler ve Temel Kavramlar

Tüm yaşayan canlıların temelinde proteinler vardır. Proteinler, amino asitlere birleşmesiyle oluşan daha büyük boyutlu moleküllerdir. Proteinlerin,

- kimyasal reaksiyonları katalize etmek (enzimler),
- gen tanımınınu düzenlenmesi,
- hücre, doku ve virüslerin birçok yapısal özelliklerini katalize etmek yada engellemek,
- hücreler arası karbondioksit, oksijen ve gerekli diğer maddeler için vasküler yolla taşıyıcılık yapmak,
- bağlama görevini yerine getirmek,
- canlıların yapı taşlarını oluşturmak,
- vücut kan şekerini düzenlemek,
- hastalık yapmak(bakteriler), iyileştirmek (hastalık yapan toksinlere karşı vücudun ürettiği interferon),
- kanın pıhtılaşmasını sağlamak,
- su ve elektrolit dengesinin korunmasında doğrudan yada dolaylı olarak görevl almak,
- sinirsel uyarıların iletiminde rol almak,
- yeni dokuların yapılması,
- enzim ve hormonların yapımında görev almak ve yapılarında bulunmak

gibi çok geniş kapsamlı görevleri vardır. Vücudun yapı taşlarının temelinde yine proteinler vardır. Örneğin saç, tırnak ve tüylerde bulunan sert yapıyı olusturan keratin isimli madde bir proteindir. Bazı proteinler, kasları kemiğe bağlayan tendonlarda bulunan dayanıklı naylon benzeri bir maddeyi olustururlar. Hayvan vücudunda organların ve yumuşak dokuların yapı unsurudur. Kaslardaki kasılmayı sağlayan aktin ve miyozin proteinden yapılmış ipliklerdir.

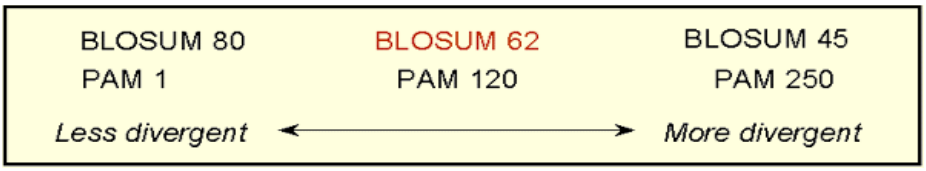
Ayrıca hücreye gelen mesajları getirenler de, mesajları alan ve degerlendirenler de yine proteinlerdir. Hücrenin içine giris çıkısları kontrol eden kapılar ve pompa sistemleri de proteinlerden oluşmuşturlar. Kimyasal reaksiyonları hızlandıranlar yine proteinlerdir..

Proteinlerin önemini ve işlevini anlamak için bazı proteinleri ve işlevleri şöyle sıralayabiliriz :

- "*Hemoglobin*" kandaki oksijeni dokulara taşır,
- "*kolajen*" derinin pürüzsüz elastikiyetini ve kemiklerin dayanıklılığını sağlar,
- "*insülin ve glukagon*" vücuttaki seker metabolizmasını düzenler,
- "*immunoglobülinler*" bakteri ve virüslere karsı vücudu savunur,
- "*fibrinojen ve trombin*" ise kanın pıhtılaşmasını sağlar,
- "*rodopsin*" retinaya ısık çarptıgında görme etkisini baslatır,
- "*transferin*" kandaki demiri taşır

Proteinler kimyasal olarak başka moleküllerle eşleştiklerinde proteindeki elektromanyetik yüklerin dağılımı ve böylece proteinin şekli değişir. Protein şekil değiştirirken aynı zamanda da bir hareket üretir. Bir proteinin hareketi iş oluşturur. Canlılar tarafından gerçeklestirilen sindirim, soluma, üreme gibi tüm fizyolojik fonksiyonları temin eden *patikalar*, birarada çalışan protein grupları tarafından oluşturulmuş olur.

Akademik yazımda protein yapıları iki şekilde incelenir. İlki protein yapılarından direkt elde edilen verilerden özellik (skorlama yöntemli benzerlik kriterleri, kimyasal yada fiziksel özelliklerin sayısallaştırılması) çıkarılarak karşılaştırılma yapan yöntemler. İkincisi ise var olan yapı üzerine kurulu yeni bir model üzerinden elde edilmiş sayısallaştırılmış verilerdir. HMM-HHM karşılaştırılması, çekirdek makinaları buna birkaç örnek olarak gösterilebilir. Yaygın olarak kullanılan skorlama yöntemi herhangi iki protein dizisinin birbirlerine olan benzerliklerini belirli bir yerine koyma matrisi ile skorlama yapmakdıt. Bu yerine koyma matrislerinden en bilinenleri PAM ve BLOSUM matrisleridir. BLOSUM matrisinin yüksek, PAM matrisinin düşük numaralı olanları yakın protein yapısına sahip diziler için, tersi durum ise uzak homoloji yapısına sahip protein dizileri için oluşturulmuştur (Şekil 6.1).



Şekil 6.1 Blosum ve Pam matrisleri[*]

6.1.1 Dizi Hizalama Algoritmaları

Herhangi iki dizinin birbirlerine homolojik benzerliği yada işlevsel benzerliklerini bulan ilk yöntemlerdendir ve hala günümüzde en çok kullanılan yöntemlerdir. Bir DNA yada protein dizisi nesilden nesile geçerken bazı değişikliklere uğrar bunlara genel olarak mutasyon denir. Birçok mutasyon sebebi olmasına rağmen bunlar üzerinde durmayacağız. Mutasyon çeşitlerine 3 ana başlıkta incelenebilir :

- **Silinme (Deletion) :** Var olan bir amino asitin silinmesi **{ AAGC → AAC }**
- **Ekleme (Insertion) :** Protein dizisi üzerinde bir noktaya yeni bir amino asit eklenmesi **{ AAG → AATG }**
- **Yerine Koyma (Substitution) :** Var olan bir amino asitin başka biriyle yer değiştiilmesi **{ AACG → AATG }**

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y
A	4	–1	–2	–2	0	–1	–1	0	–2	–1	–1	–1	–1	–2	–1	1	0	–3	–2
R	–1	5	0	–2	–3	1	0	–2	0	–3	–2	2	–1	–3	–2	–1	–1	–3	–2
N	–2	0	6	1	–3	0	0	0	1	–3	–3	0	–2	–3	–2	1	0	–4	–2
D	–2	–2	1	6	–3	0	2	–1	–1	–3	–4	–1	–3	–3	–1	0	–1	–4	–3
C	0	–3	–3	–3	9	–3	–4	–3	–3	–1	–1	–3	–1	–2	–3	–1	–1	–2	–2
Q	–1	1	0	0	–3	5	2	–2	0	–3	–2	1	0	–3	–1	0	–1	–2	–1
E	–1	0	0	2	–4	2	5	–2	0	–3	–3	1	–2	–3	–1	0	–1	–3	–2
G	0	–2	0	–1	–3	–2	–2	6	–2	–4	–4	–2	–3	–3	–2	0	–2	–2	–3
H	–2	0	1	–1	–3	0	0	–2	8	–3	–3	–1	–2	–1	–2	–1	–2	–2	2
I	–1	–3	–3	–3	–1	–3	–3	–4	–3	4	2	–3	1	0	–3	–2	–1	–3	–1
L	–1	–2	–3	–4	–1	–2	–3	–4	–3	2	4	–2	2	0	–3	–2	–1	–2	–1
K	–1	2	0	–1	–3	1	1	–2	–1	–3	–2	5	–1	–3	–1	0	–1	–3	–2
M	–1	–1	–2	–3	–1	0	–2	–3	–2	1	2	–1	5	0	–2	–1	–1	–1	–1
F	–2	–3	–3	–3	–2	–3	–3	–3	–1	0	0	–3	0	6	–4	–2	–2	1	3
P	–1	–2	–2	–1	–3	–1	–1	–2	–2	–3	–3	–1	–2	–4	7	–1	–1	–4	–3
S	1	–1	1	0	–1	0	0	0	–1	–2	–2	0	–1	–2	–1	4	1	–3	–2
T	0	–1	0	–1	–1	–1	–1	–2	–2	–1	–1	–1	–1	–2	–1	1	5	–2	–2
W	–3	–3	–4	–4	–2	–2	–3	–2	–2	–3	–2	–3	–1	1	–4	–3	–2	11	2
Y	–2	–2	–2	–3	–2	–1	–2	–3	2	–1	–1	–2	–1	3	–3	–2	–2	2	7
V	0	–3	–3	–3	–1	–2	–2	–3	–3	3	1	–2	1	–1	–2	–2	0	–3	–1

Şekil 6.2 Blosum 60 matrisi

[*] Bioinformatics and functional genomics, by Jonathan Pevsner.

Genel olarak iki dizinin benzerlik skorunu hesaplamak için dinamik programlama algoritmaları kullanılmaktadır. Bu algoritmalar için elde edilmiş farklı skor tabloları bulunmaktadır. Bunlardan en önemlileri PAM ve BLOSUM'dur (Şekil 6.2)

6.1.2 Genel Hizalama (Needleman-Wunsch Modeli)

İki dizgi arasındaki tüm sıralı amino asit dizisine bakarak skor tayin eder. 1970 yılında Saul Needleman ve Christian Wunsch tarafından ortaya atılan bir algoritmadır.

$$F(i,j) = max\begin{cases} F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - g \\ F(i,j-1) - g \end{cases} \quad (6.1)$$

Burda "g" gap cezası olarak kullanılmaktadır. Mutasyon sonucu oluşacak cezalandırma parametresidir. Şekil 6.3'te basit olarak bir skor matrisi kullanmadan uyuşuyorsa "+1" uyuşma yok ise "-2" koyduğumuzda oluşan bir genel hizalamanın sonucu görülmektedir.

		G	T	G	A	C	T
	0	−2	−4	−6	−8	−10	−12
A	−2	−1	−3	−5	−5	−7	−9
G	−4	−1	−2	−2	−4	−6	−8
T	−6	−3	0	−2	−3	−5	−5
G	−8	−5	−2	1	−1	−3	−5
T	−10	−7	−4	−1	0	−2	−2
G	−12	−9	−6	−3	0	−1	−3
C	−14	−11	−8	−5	−2	1	−1
G	−16	−13	−10	−7	−4	−1	0

	−	G	T	G	−	A	C	T
	A	G	T	G	T	A	C	G
	−2	+1	+1	+1	−2	+1	+1	−1

$\Sigma = 0$

Şekil 6.3 GTGACT ile AGTGTGCG dizilerinin genel hizalaması

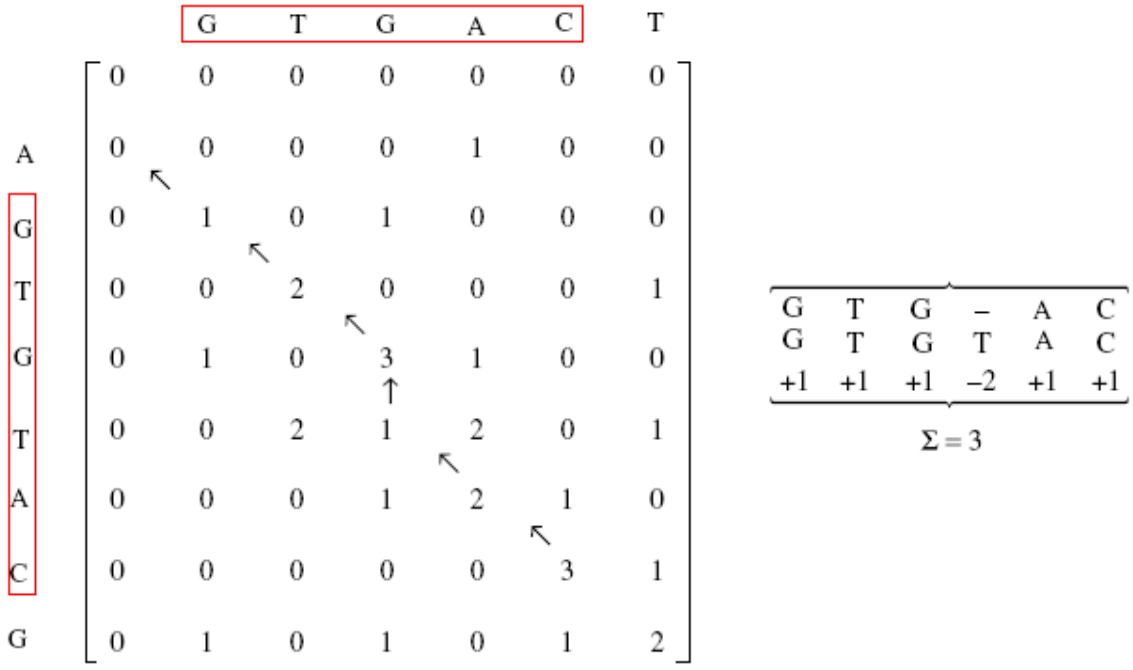
6.1.3 Yerel Hizalama (Smith-Waterman Modeli)

1981 yılında Temple Smith ve Michael Waterman tarafında önerilen bir metoddur. Önemli olan tüm dizi değil dizi içindeki en yüksek skoru olan alt parçacıktır. Bu model proteinlerin işlevlerini skorlandırma için önemli bir algoritmadır.

$$F(i,j) = max \begin{cases} 0 \\ F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - g \quad (Silme) \\ F(i,j-1) - g \quad (Ekleme) \end{cases} \qquad (6.2)$$

$$Durumları : F(0,0) = 0, F(0,i) = 0, F(j,0) = 0$$

Akademik yazımda bu model üzerine inşa edilmiş bir çok algoritma vardır. Özellikle blast (Altschul vd., 1990), primer-blast, lgBlast, snp, blastp, psi-blast (Altschul vd., 1997), phi-blast, blastx, tblastn, tblastx, PowerBlast(Zhang ve Madden, 1997), bl2seq, blastn, megablast, discontiguous megablast gibi ...



Şekil 6.4 GTGACT ile AGTGTGCG dizilerinin yerel hizalaması

6.1.4 Tekrarlanan Uyuşmalar

Genel olarak protein dizileri çok büyük uzunluktaki verilerdir. Bunun için tek bir yerel yada genel hizalamadan daha ziyade, olası oluşabilecek birden fazla yerel hizalamalarının toplamı alınır. 1987 yılında Waternan ve Eggert tarafından ortaya sürülen bir metotodolojidir.

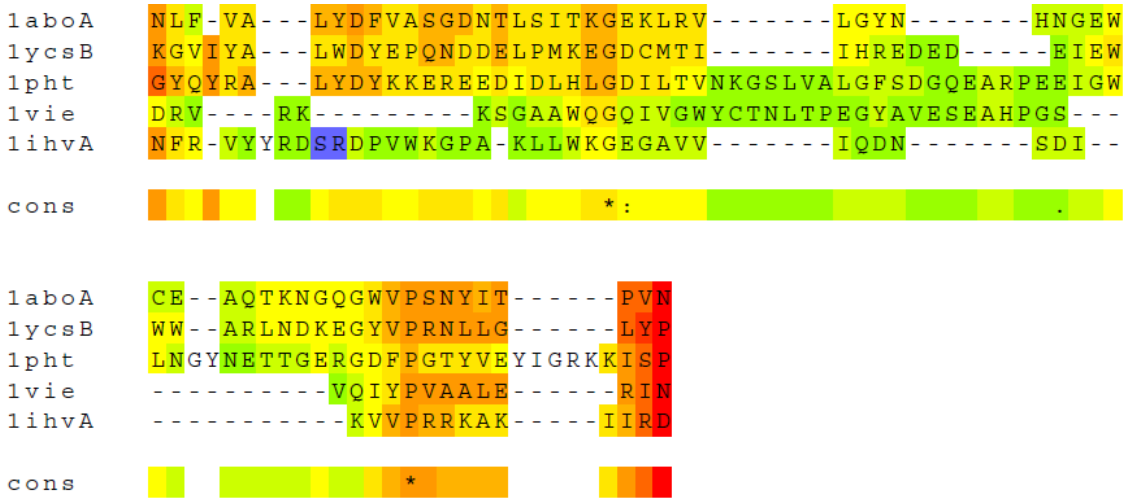
$$F(i,j) = max \begin{cases} 0 \\ F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - g \quad (Silme) \\ F(i,j-1) - g \quad (Ekleme) \end{cases} \qquad (6.3)$$

$$F(i,0) = max \begin{cases} F(i-1,0) \\ F(i-1,j) - T \end{cases}$$

$$F(0,0) = 0$$

6.1.5 Çoklu Hizlama (Multiple Sequence Alignment)

Çoklu hizalama yöntemleri 3 yada daha fazla protein yapısının üst üste koyularak en uygun parçalarının benzerliği üzerine kurulu metodlardır (Barton, 1990). Bu yöntemlerden bazılarını Musca (Parida vd, 1998), Tcoffeee(Notredame, 2000), DIALIGN (Morgenstern, 2004), Kalign(Lassmann ve Sonnhammer, 2006), ClustalW (Larkin vd., 2007) şeklinde sırlayabiliriz.



Şekil 6.5 Çoklu hizalama örneği

Dizilerin çoklu hizalanmasın sonucu skorlama yöntemi ile protein aileri ve alt aileleri bulunarak sınıflandırma yapılabildiği gibi bune ek olarak aynı aile gruplarının oluşturduğu motifler de elde edilebilmektedir.

6.2 Önerilen Motif Tabanlı Yöntem

Motifler benzer bioyolojik işlevleri yada benzer yapısal özellik gösteren protein yapılarından elde edilmiş desenlerden oluşan yapılardır. Özellikle proteinlerin fonksiyonel sınınflandırmasında önemli bir rolü olan motifler için hazırlanmış PROSITE(Falquet vd., 2002) , ProDom(Corpet vd., 1999), BLOCKS(Henikoff vd., 2000), PRINTS(Attwood vd., 2002), Pfam(Bateman vd., 2002) gibi birçok yapısal desen içeren veritabanları bulunmaktadır. 2007'de RECOMB konferansı için yapmış olduğumuz çalışmada PROSITE ve PRINTS veritabanlarından elde edilen desenlerin oluşturduğu genel bir veritabanı ile elde ettiğimiz Koli Basili ve Mide Ülseri baktarilerini sınıflandırmaya çalıştık. Belli başlı makine öğrenme modelleri ile başarımlarını gözlemledik. Fakat dikkate değer bir doğruluk belirlenemedi. Sebep olarak var olan desenlerin verilerimiz için özel oluşturulmadığı, var olan hazır desenler olmasından kaynaklandığı sonucuna varıldı. Bir başka çalışmamızda

oluşturulan koli basili ve mide ülseri verilerinin kendi aralarında gerçekleştirdiğimiz çoklu hizalama yöntemlerinden elde edilen desenlere göre ele almanın başarımı artırdığı gözlemlendi. PROSITE motiflerinin bazı örnekleri EK 7'de gösterildi.

Çizelge 6.1 Koli Basili ve Mide Ülseri Motif tabanlı sınıflandırma sonuçları

	Koli Basili				**Mide Ülseri Bakterisi**			
	ClustalW	Profeat	Prosite	Prints	ClustalW	Profeat	Prosite	Prints
3NN	69	69	67	73	62	60	58	58
5NN	70	69	68	73	62	60	60	58
ANN	74	74	73	69	63	55	57	59
Doğrusal	81	79	77	70	77	69	65	68
RBF	81	80	74	77	78	75	74	72
IKL	79	79	77	71	76	75	73	70
MKL (RBF+ Pol)	84	81	82	79	74	77	76	71

Gene Ontology veritabanından elde edilen Koli Basili (298 adet bağlama yapan, 287 adet Katalitik aktivite) ve Mide Ülseri (195 bağlama yapan, 280 adet katalitik aktivite) bakterilerinin sınıflandırılmasında elde edilen motiflerin aynı zamanda biyolojik işlevler içinde geçerli olacağından dolayı başarımı düşük olmuştur. Yukarıdaki sınıflandırıcılara uygulamadan önce gereksiz öznitelikler temizlendiği taktirde başarım artacaktır (Ayan ve Cansever, 2010b).

6.3 PHA-Kernel (Pairwise HMM Alignment Kernel)

Burda yeni önerdiğimiz model ise ikili Saklı Markov Modeller ile çekirdek metodların birleştirilmesi temeline dayanmaktadır. Akademik yazımda sıralamadan, boyuttan bağımsız ve protein yapılarının sınıflandırılmasına elverişli olmasından dolayı SMM tabanlı biçok model geliştirilmiş olmakla birlikte, ilk olarak Krogh(1994, 1998) tarafından ortaya atılmıştır. Daha sonraları birçok SMM tabanlı yöntemler geliştirilmiştir, özellikle HMMER(Eddy, 1998), HMMSTR (Bystroff vd., 2000), HM-SVM(Altun vd., 2003), HHPred (Söding vd., 2006) ön plana çıkan modellerdir. Bu türetilmiş modellerin temelinde Durbin vd. tanımlamış olduğu log-odd skorları baz alınarak yüksek skor sahibini protein yapısına en uygun eş olarak bulma yatar.

Tipik bir saklı Markov Modelinde $X = \{x_1, \dots, x_N\}$ N faklı durumları, $Y = \{y_1, \dots, y_M\}$ olası gözlemler, $a_{pq}: P(\pi_i = q|\pi_{i-1} = p)$ durum geçiş olasılıkları ve $b_p(m): P(x_i = m|\pi_i = p)$ çıkış olasılıkları olmak üzere tanımlanır. Durum geçiş olasılıklarını

$$a_{pq} = \frac{\sum_k \sum_t 1\left(y_t^k = x_p \; ve \; y_{t+1}^k = x_q\right)}{\sum_k \sum_t 1\left(y_t^k = x_p\right)}, \quad b_p(m) \geq 0, \quad \sum_{m=1}^{M} b_p(m) = 1 \tag{6.4}$$

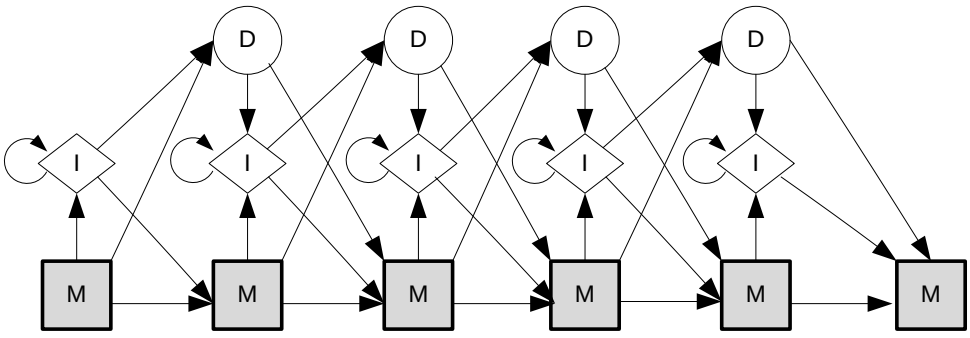
İlk durum olasılıklarını $\pi_i \equiv P(y_1 = x_i)$ şeklinde tanımlanır. Saklı Markov modellerinde 3 temel problem ele alınmaktadır.

i. Model(γ) verildiğinde, bu modelin gözlem dizisi G=$y_1 y_2 \dots y_T$'i üretme olasılığı $P(G|\gamma)$.

ii. Model(γ) ve G=$y_1 y_2 \dots y_T$ verildiğinde, bu gözlem dizisinin oluşturma ihtimali en yükek olan durum dizisini $D = x_1, \dots, x_T$ bulma olasılığı $P(D|G, \gamma)$.

iii. Gözlem dizilerinden oluşan eğitim seti verildğinde bu eğitim setini en yüksek oranda gerçekleyen model parametrelerinin hesaplanması.

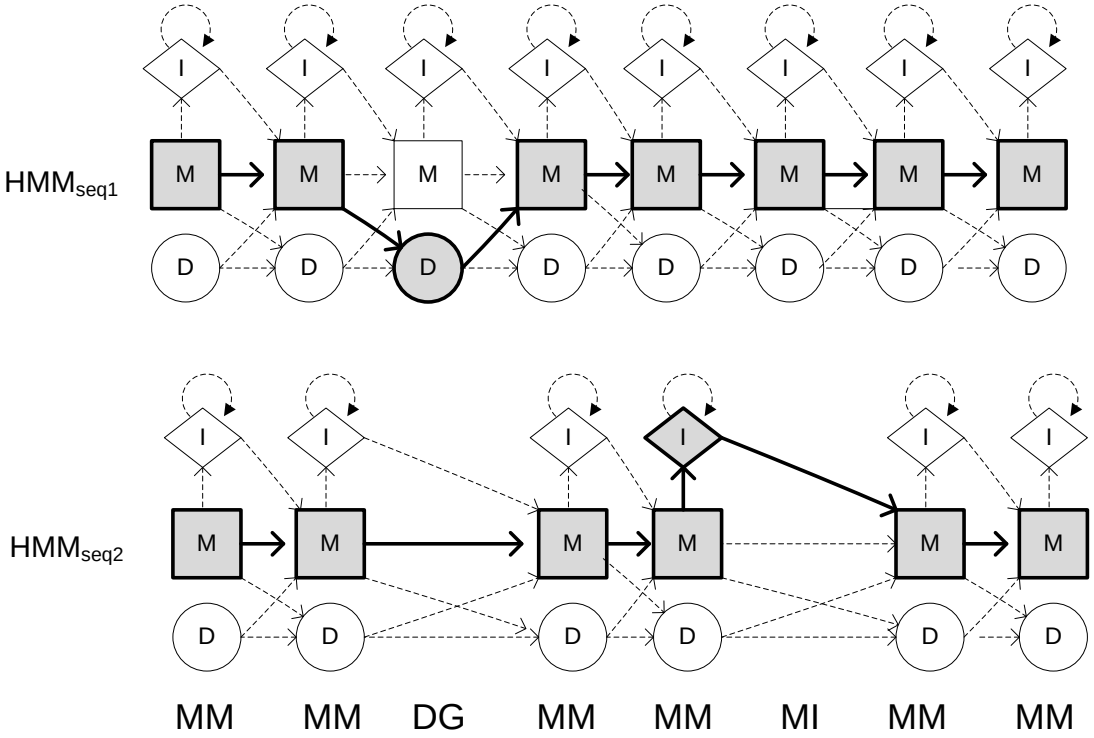
Bizim problemimizde elimizde gözlemler sonucu elde edilen bir modelimiz bulunmakta ve SMM modeli ile birlikte yeni gelecek herhangi bir durum dizisinin SMM modelleri arasında Uyuşan(M-Match), Ekleme(I-Insertion), Silme(D-Deletion), Boşluk(G-Gap) oluşma olasılıklarını bularak durum ataması yapılır. Bu işlemin sonucunda elde edilen log-olasılıksal toplamlarına göre profillerden (protein dizi ailelerinden) elde edilen SMM'ler arasında skorlama işlemi yapılmaktadır. Şekil 6.6'da protein dizilerinden elde edilen SMM'lerin topolojisi verilmiştir.



Şekil 6.6 Protein yapıları için Saklı Markov Modeli topolojisi

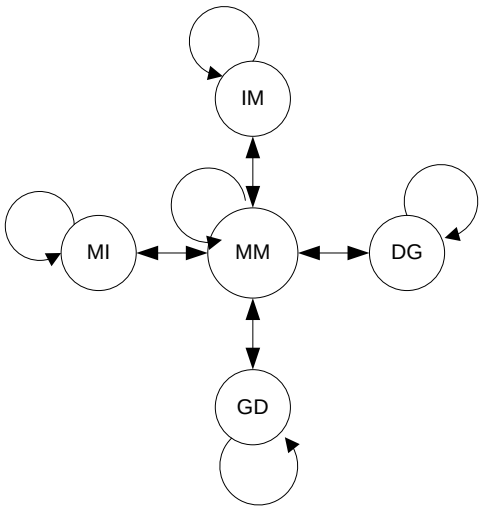
Herhangi iki dizinin, yada dizi ailelerinin SMMlerini karşılıklı olarak dinamik programlama ile skorlama yapılarak bir pozitif yarı tanımlı çekirdek yapısı oluşturulur(Şekil 6.7). Böylece

çekirdek modelleri ile halı hazırdaki profile tabanlı SMM'lerin skorları kullanılarak yeni bir çekirdek modeli oluşturulmuş olur(6.5).



Şekil 6.7 HMM dizilerinin skorlama (MM-MI-DG-IM-GD)

Yukarıdaki eşleşen durumlardan istenmeyenler çıkarılabilir yada yeni ara durumlar eklenerek başka farklı modeller de geliştirilebilir. Biz burda MM, MI, DG, IM, GD durumlarını baz alarak PHA-Kernel yapısı oluşturulmuş oldu (Şekil 6.8).



Şekil 6.8 Kabul edilebilir durumlar (MM-MI-DG-IM-GD)

Görüldüğü üzere tipik bir SMM ile protein yapı ve ilev tahmininde durum olarak 20 amino asit yada 4 nucleotid alınırdı. Bizim burda önerdiğimiz modelde ise tipik bir gediklenmiş bir protein sınıfına ait yapıların tamamı için bir SMM modeli oluşturup oluşturulan bu model üzerinden her bir proteininin yukarıda belirtilen kabul edilebilir durumlarını bulup var olan amino asit dizilişleri yerine bu durumları koyarak tipik bir hizilama algoritması yöntemi gibi skorlama yapmak şeklinde tanımlanabilir. Böyle bir durumda tipik oluşacak mutasyonların önüne geçilerek hata oranı düşürülmüş olacaktır. Her bir mutasyon aslında skorlama yapılırken eksi bir katsayıya sahiptir. Fakat bu modelde var olan sınınfların protein dizilerinin tamamı üzerinden model oluşturulduğundan modelin başarımı daha yüksek çıkmakta ve yerel mustasyonlar göz ardı edilmektedir. Skorlama için aşağıdaki dinamik çözüme gidilir,.

$$S_{i,j}^{MM} = S^{aa}(q_i, p_j) + \ max \begin{cases} S_{i-1,j-1}^{MM} + log\left(q_{i-1}^{MM}.p_{j-1}^{MM}\right) \\ S_{i-1,j-1}^{IM} + log\left(q_{i-1}^{IM}.p_{j-1}^{MM}\right) \\ S_{i-1,j-1}^{MI} + log\left(q_{i-1}^{MM}.p_{j-1}^{MI}\right) \\ S_{i-1,j-1}^{DG} + log\left(q_{i-1}^{DM}.p_{j-1}^{MM}\right) \\ S_{i-1,j-1}^{GD} + log\left(q_{i-1}^{MM}.p_{j-1}^{DM}\right) \end{cases}$$

$$S_{i,j}^{MI} = max \begin{cases} S_{i-1,j}^{MM} + log\left(q_{i-1}^{MM}.p_{j}^{MI}\right) \\ S_{i-1,j}^{MI} + log\left(q_{i-1}^{MM}.p_{j}^{II}\right) \end{cases} \tag{6.5}$$

$$S_{i,j}^{DG} = max \begin{cases} S_{i-1,j}^{MM} + log(q_{i-1}^{MD}) \\ S_{i-1,j}^{DG} + log(q_{i-1}^{DD}) \end{cases}$$

benzer şekilde $S_{i,j}^{IM}$ ve $S_{i,j}^{GD}$ de hesaplanır. Hesaplamalarımızda HMMER, HHPRED yöntemlerinden elde edilen SMM skorları ile log-odd skorlarını ele alarak çekirdek yapısı oluşturulur.Durbin vd. log-odd skorlama için,

$$\tau_{lo} = \log\frac{P(s_1, \dots, s_L|\xi_{path})}{P(s_1, \dots, s_L|Null)} = \log\frac{P(s_1, \dots, s_L|\xi_{path})}{\prod_{l=1}^{L} f^{x_l}} \tag{6.6}$$

uyuşan durumların olasılıksal toplamları şeklinde ifade etmiştir. Böylece çekirdek yapısı için gerekli Hessian matrisini elde etmiş oluruz.

Daha önce ele aldığımız Biyolojik Aktivite ve Enzimler veri kümesine uyguladığımızda Çizelge 6.2'den de gözükeceği üzere diğer dizi çekirdek tabanlı yöntemlere göre daha yüksek oranda başarılı olmuştur.

Çizelge 6.2 PHA Çekirdek sonuçları

Biyolojik Aktiviteler							
Hücresel İşlev				Metabolik Aktivite			
Tümü	90%	70%	40%	Tümü	90%	70%	40%
81,40%	85,50%	86,20%	86,60%	84,72%	87,57%	88,95%	89,24%

Enzimler							
Hidroliz Yapanlar				Amine ve Fosfat Taşıyıcılar			
Tümü	90%	70%	40%	Tümü	90%	70%	40%
85,45%	88,62%	90,05%	88,76%	89,94%	94,12%	89,60%	92,05%

Bu modelin başarımı tek tek her bir proteindeki amino aist dizilişi yerine çünkü proteinlerde çok farklı mutasyon çeşitleri olabilmekte, haritaya yukarıdan bakarak sınıfın genel modeli üzerinden tekrar protein dizilerini modele ait yeni bir yapı ortaya koyar. Daha sonra çekirdek hesabı için dinamik programlama ile skorlama yapılır. Daha önceki akademik yazımlardaki SMM tabanlı modellerle arasındaki fark olarak diğer modellerde HMMER, HHPREd, HMM-SVM vb.. tek te k protienlerin SMM skorlarını alıp en yakın olan proteini seçme fikrine dayanmaktadır. Bizim modelimizde ise bu skorları bir çekirdek dönüşümü şekline getirip çekirdek modelleri ile eğitmek şekilindedir.

7. SONUÇLAR ve ÖNERİLER

Makine öğrenme yöntemleri, uygulancak veri kümeleri için en iyi modeli ortaya koymak ve bu öne sürülen modelin parametrelerini öğrenmek üzerine kuruludur. Özellikle yüksek boyutlu veri yapılarında (protein ve dna dizleri, web sayfaları, ilaç veri kümeleri, hastalaık verileri, 2 ve 3 boyutlu imgelerin bulunduğu görüntü ve videolar gibi) başarımı artırmak ve işlem karmaşıklığını azaltmak için öznitelik indirgeme, boyut indirgeme, alt uzay seçimi gibi yöntemler geliştirilmektedir. Biz bu tezde önerdiğimiz gözetimli öğrenme, gözetimsiz öğrenme ve öznitelik indirgeme yöntemlerini fizyolojik verilerden ilaç tasarımı ve hastalık veri kümelerine uyguladık. Fakat oluşturduğumuz ve savunduğumuz tüm yukarıdaki öğrenme modelleri, özellikle vektör uzayı şekline getirilmiş farklı veri kümeleri üzerine uygulanabilir.

Her bölümde yapılan deneysel çalışmalardan elde edilen sonuçların özetlerini aşağıdaki gibi sıralayabiliriz:

- İlaçların milyon dolarlar mertebesinde çok yüksek maliyet gerektiren işlemlerinden yola çıkılarak ilaç tasarımını bilgisayar ortamına taşımak ve ilaç adayı moekülleri yeni önereceğimiz modellerle tespit etmek hedeflenmişti ve bu şu an itibariyle %80-95 doğruluk oranı aralığında gerçeklenmiş durumdadır. Kullandığmız düşük boyulardaki Cherkasov ve Murcia-Soler veri kümesi, orta boyutlardaki Drug Veri bankası ve 250.000 civarında ilaç verisi içeren çok yüksek boyutlardaki Pharmeks veri kümeleri üzerinde önerdiğimiz algoritmalarımızı eğittik ve sonuçları detayları ile belirttik. İlaç adayı olabilecek veri kümeleri için daha önceden akademik yazımda belirtilen 5 kuralı ile ek 2 kuralı da gözeterek MOE ve Adriana.Code programları yardımıyla bu verilerde 350-400 adet arasında QSAR öznitelik elde edildi. Özellikle Pharmeks veri kümesinde bazı verileri işleme sırasında verilerin tutarsızlığı ve eksikliği, bilgisayarların yetersizliği, yetersiz bellek ve günlerce süren hesaplamalardan (bu veri kümesi için 1.5 haftadan fazla QSAR öznitelik hesaplaması sürdü) dolayı bazı öznitelikleri elemek zorunda yada o niteliğin ağırlık çarpanını sıfırlamak zorunda kaldık.
- Özellikle etiketsiz verinin çok olması ve etiketli yani sınıfı belli olan verinin az olması durumunda gözetimli öğrenme modellerinin başarısı düşük olmaktadır. Bunun için yarı-gözetimli öğrenme modelleri günden güne daha hızlı gelişmektedir. Bizim önerdiğimiz bu model de var olan tüm etiketli veri kümesi üzerinde etiketsiz veriler için Harmonik enerjiyi en azlayacak gerçek risk fonksiyonunu Bayesian yöntemi ile elde edip en az risk taşıyan k adet veriyi sorgu tabanlı etkin öğrenme modeli ile

eğiterek hatayı genele yayma amacı güdülmektedir. Daha sonra elde edilen sonuçların bir düzgeleyici ile tekrar düzenlenerek etiketini belirlemek hedeflenmiştir. Sonuçlar incelendiğinde dikkate değer bir başarı elde edildiği gözlemlenmektedir. Bu yapısal teoriyi diğer var olan yarı gözetimli öğrenme modellerine de uygulamak zor değildir. Yapılacak işler arasında bu yöntemi gözetimsiz öğrenme ve diğer yarı gözetimli öğrenme yöntemlerine geliştirmek hedeflenmektedir.

- UCI veri kümelerinden elde edilen hastalık ve kanser veri kümelerindeki başarı oranları ilaç veri kümelerindekine göre çok daha yüksek olmuştur hatta bazı çekirdek parametreleri için bu oran %100'ü bulmuştur.

Genel olarak gözlemlenen bazı çıkarımları da aşağıda belirtmek gerekirse:

- Cristianini ve Shawe-Taylor (2000), Pavlidis vd.(2001), Lanckriet vd. (2004b), Ben-Hur ve Noble(2005), Sonnenburg vd. (2006), Lin vd. (2009), Zhao vd.(2009), Kloft vd. (2010) çalışmlarında belirttiği gibi bizim sonuçlarımızda da gözlemlediğimiz üzere tekil çekirdek yerine çoklu çekirdek kullanıldığında, çoğul çekirdeklerin birleştirme modeline bağlı olarak kullanılan aynı parametreler için doğruluk oranı daha yüksek olmaktadır.
- Çoklu çekirdekler konusu ele alındığında çoklu çekirdeklerin birleştirilmesi konusunda akademik yazımda çok faklı modeller öner sürülmüştür. İç bikey, dış bükey, SMO, çarpım, toplamlar, ortalama, yerel toplam, yinelemeli vb.. bu modellerin tamamını karşılaştıran ve gerçekte hangi modelin neye göre daha iyi sonuçlar verdiği tam olarak bir araştırma konusu olarak sunulmadığında özellikle bu konuda daha birçok farklı çalışmalar yapılabileceği düşünülmektedir.
- Genel olarak sonuçlar incelendiğinde ve akademik yazımdaki yapılan diğer çoklu çekirdek modelleri detayları ile ele alındığında, çoklu çekirdek modellerinde 2 ile 5 çekirdek modelini kullanmak bir veri kümesinin doğruluk oranını artırmak adına yeterli olmaktadır. Bundan daha fazla kullanılan çoklu çekirdek yapılarında işlem karmaşıklığının artmasının yanında doğruluk oranını gözle görülür derecede artırdığı görülmemektedir. Belki bir başka araştırma konusu olarak kullanılacak çekirdek modeli sayısı ve bu modellerin neler olması gerektiği ile parametre kestirimi düşünülebilir. Aslında bu konuda yeni bir kavram olarak karşıma çıkan Infinite Kernel Learning konusu da ele alınabilir. Burda da sonsuz çekirdek öğrenme modeli olarak yeni kavramlar gelmektedir.

- İkili sınıflarda iyi çalışan çekirdek modelleri çoklu sınıflar için de benzer durum pek söz konusu değil yada düşünülmemiştir. İncelenen makalelerin sonucunda çoklu sınıflar için önerilen yöntem sayısı kısıtlıdır. Teoremin temelinde yatan ikili sınıflar için en iyi ayırabilecek aşırı düzlem seçiminden dolayı bu konuda eğitim yada sınıflandırma sırasında bir sınıfa karşı diğerleri yada birine karşı diğeri gibi basit yöntemler yerine bizimde ortaya koymaya çalıştığımız gibi matematiksel teorisi daha iyi olabilecek yeni ve daha orjinal fikirler ileri sürülebilir. Özellikle çok çekirdekli çoklu sınıflar için akademik yazımda çok çok az sayıda akademik makale bulunmaktadır. Bizim ileri sürdüğümüz yöntemde de artımlı çoklu sınıflar için çoklu çekirdek modeli konusu ele alınmaktaydı. Bu yeni model ve yöntemler araştırmaya açık konular olarak gelecek çalışmalara ışık tutacağı umulmaktadır.
- Çekirdek yöntemleri üzerinde araştırma yapılabilecek belli başlı bazı konular bulunmaktadır. Bunlardan en önemlisi ve hala çözüm tam olarak belirli olmayan çekirdek modelleri için hangi parametrelerin seçileceği ki bizim çalışmalarımızda çok yüksek ve çok küçük *C* parametresi seçimi başarıyı düşürdüğü yada artırmadığı gözlemlenmektedir. En azından başarı oranının yüksek olabileceği aralık belirlenebilir. Buna ek olarak her bir çekirdeğin kullandığı diğer parametreler içinde böyle bir çalışma yapmakta fayda vardır.
- Çekirdek modellerinin temelinde yatan eniyileme problemini farklı yöntemlerle iyileştirme yoluna gitmek de bir başka araştırma konusu olabilir. Özellikle KKT durumları ele elındığında Lagrangian çarpanlarının 0 ile *C* çekirdek düzgeleme paremetresi arasında seçilmesi, eğer böyle bir durum söz konusu değil ise bu çarpanın "0" şeklinde ifade edilmesi yerine bir ceza katsayısı eklenerek modele eklenip tekrar eğitilmesi düşünülebilir.
- Yarı gözetimli öğrenme modellerinde, eğitim kümesindeki verilerin dağılımı modelini başarımını çok yüksek oranda etikilemektedir. Bu tip modellerde eğer eğitim sırasında böyle bir düzensiz bir dağılım söz konusu ise bu dağılımı engellemek için daha önce detayları ile vermiş olduğumuz şekilde bir normalizasyon uygulanmalıdır. Aksi taktirde öğrenme entropy değeri yüksek olan veri kümesi üzerinde yoğunlaşacağından hep aynı etiketli kümeyi öğrenecektir.
- Diğer bir konu ise etkin yarı gözetimli öğrenme modelinde sorgu tabanlı modeli uyguladığımızda tek bir sorgu örneği modeli seçildiği taktirde eğitim çok uzun

sürmekte ve başarımı etiketli küme sayısı fazla olan tarafa doğru kaymaktadır. Bunu engellemek için yeter düzeyde seçilecek uygun bir k paremetresi ile etken öğrenemeye katılacak bu k adet örneği için bu yanlış etiketleme oranını azaltılabildiği ve hesap karaşıklığı diğer modele göre azaldığı gözlemlenmiştir.

Tüm yukarıdaki çıkarımlar ve gözlemler sonucunda fizyolojik verilen üzerinde önerilen tüm model ve yöntemlerin başarısı ve sonuçları bu konularda çalışan başka araştırmacılar ve son kullanıcılar için faydalı olacağını düşünmekte ve özellikle sonuçlar kısmında ele aldığmız araştıma konusu olabilecek çalışmalar kısımlarının da fikir planında yeni ufuklar açmasını temenni etmekteyiz.

KAYNAKLAR

Abney, S. (2008), Semisupervised Learning for Computational Linguistics, Chapman ve Hall/CRC, Florida.

Aha, D. ve Kibler, D., (1991), "Instance-based Learning Algorithms", Machine Learning, 6:37-66.

Agatonovıc-Kustrın, S. ve Beresford, R., (2000), "Basic Concepts of Artificial Neural Networks Modelling and Its Application In Pharmaceutical Research", J. Pharmaceutcal and Biomedical Analysis, 22: 717-727.

Agrawala, A. K., (1970), "Learning with a probabilistic teacher", IEEE Transactions on Information Theory, 16:373–379.

Ajay, W., Walters, P. ve Murcko, M. A., (1998), "Can We Learn To Distinguish Between "Drug-Like" and "Nondrug-Like" Molecules?", The Journal of Medicinal Chemistry., 41: 3314-3324.

Ali, S. ve Smith, K.A., (2006), "On Learning Algorithm Selection for Classification", Applied Soft Computing, Elsevier Science, 6(2): 119-138.

Alpaydın, E. (2004), Introduction to Machine Learning, The MIT Press, Cambridge.

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. ve Lipman, D.J., (1990), "Basic local alignment search tool", J Mol Biol 215 (3): 403–410

Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z., Miller, W. ve Lipman, D.J., (1997), "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", Nucleic Acids Res., 25:3389-3402

Altun,Y., Tsochantaridis, I. ve Hofmann, T., (2003) "Hidden Markov Support Vector Machines", Proc. of 20th International Conference on Machine Learning (ICML).

Amasyalı, F. M., (2007), Yeni Makine Öğrenme Metodları ve Ilaç Tasarımına Uygulamaları. Doktora Tezi, YTÜ Fen Bilimleri Enstitüsü (yayımlanmamış).

Anzalı, S., Barnıckel, G., Cezanne, B., Krug, M., Fılımonov, D. ve Poroıkov, V., (2001), "Discriminating Between Drugs and Nondrugs By Prediction of Activity Spectra For Substances (Pass)", The Journal of Medicinal Chemistry., 44(15), 2432-2437.

Arciniegas, F., Bennett, K., Breneman, C. ve Embrechts, M.J., (2000), "Molecular Database Mining Using Self-Organizing Maps For The Design of Novel Pharmaceuticals", Smart Engineering System Design, 10:477 – 482.

Attwood, T.K., Blythe, M., Flower, D.R., Gaulton, A., Mabey, J.E., Maudling, N., McGregor, L., Mitchell, A., Moulton, G., Paine, K. and Scordis, P., (2002), "PRINTS and PRINTS-S shed light on protein ancestry", Nucl. Acids Res. 30(1):239-241.

Ayan, U. ve Cansever, G., (2010a) "New Learning Approach for Drug Design", IEEE Signal Processing & Communication Applications , SIU 2010, 22-24 April, Diyarbakır, Turkey, 2010.

Ayan, U. ve Cansever, G., (2010b), "Drug/NonDrug Clustering Using Semi-Supervised Learning Approaches", RECOMB 2010, 12-15 August, Lisbon, Portugal, 2010.

Ayan, U. , Pehlivanli, A.Ç. ve Cansever, G., (2010c), "Feature Selection Criteria for Drug/Nondrug Compounds Based on Kernel Approaches", International Symposium on Innovations in Intelligent Systems and Applications, INISTA 2010, 21-24 June 2010, Kayseri, Turkey, 2010.

Bach, F. R., Lanckriet, G. R. G. ve Jordan, M. I., (2004), "Multiple kernel learning, conic duality, and the SMO algorithm", Proceedings of the 21st International Conference on Machine Learning pp: 41-48.

Baldi, P., ve Brunak, S., (2001), Bioinformatics: The Machine Learning Approach, The MIT Press, Cambridge.

Barton, G. J., (1990), "Protein multiple sequence alignment and flexible pattern matching", Meth. Enzymol., 183:403–427.

Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S.R., Griffiths-Jones, S., Howe, K.L., Marshall, M. ve Sonnhammer, E.L., (2002), "The Pfam Protein Families Database", Nucl. Acids Res. 30(1):276-280.

Baykara, T., Çaylı, H., Çelik, H., Tokat, M. ve Ünalan, T., (2004), "Türkiye'de İlaçta Veri Koruması ve Uygulanmasının Mali Etkileri" adlı rapor, (http://www.aifd.org.tr/detay.asp?ID=75&db=sektorun_sorunlari.htm)

Bayram, E., Santago Ii, P., Harris, R., Xiao, Y., Clauset, A., ve Schmitt, J.D., (2004), "Genetic Algorithms and Self-Organizing Maps: A Powerful Combination For Modeling Complex Qsar and Qspr Problems", Computer-Aided Molecular Design, 18: 483-493.

Bemis, G.W. ve Murcko, M.A., (1996), "Properties of Known Drugs 1: Molecular Frameworks", Journal of Medicinal Chemistry, 39: 2887-2893.

Bemis, G.W. ve Murcko, M.A., (1999), "Properties of Known Drugs 2: Side Chains", Journal of Medicinal Chemistry, 42: 5095-5099.

Ben-Hur, A. ve Noble, W.S., (2005), "Kernel methods for predicting protein-protein interactions", Bioinformatics, 21(Suppl 1):i38-46.

Blum, A. ve Mitchell, T., (1998), "Combining labeled and unlabeled data with co-training", COLT: Proceedings of the Workshop on Computational Learning Theory.

Board, R. ve Pitt, L., (1989), "Semi-supervised learning", Machine Learning, 4(1):41–65.

Boardman, M. ve Trappenberg,T., (2006), "A Heuristic for Free Parameter Optimization with Support Vector Machines", Proceedings of the 2006 IEEE International Joint Conference on Neural Networks (IJCNN 2006), pp. 1337-1344.

Boukharouba, K., Bako, L. ve Lecoeuche, S., (2009), "Incremental and Decremental Multi-category Classification by Support Vector Machines", Proceeding of the International Conference on Machine Learning and Applications, 2009.

Bousquet, O. ve Herrmann, D. J. L., (2003), "On the complexity of learning the kernel matrix", In Advances in Neural Information Processing Systems 15.

Brown, F.K., (1998), "Chemoinformatics: What Is It and How Does It Impact Drug Discovery", Annual Reports in Medicinal Chemistry, 33: 375-384.

Brüstle, M, Beck, B., Schindler, T., King, W., Mitchell, T. ve Clark, T., (2002), "Descriptors, Physical Properties, and Drug-Likeness", Journal of Medical Chemistry, 45(16): 3345-3355.

Bystroff, C., Thorsson, V. ve Baker, D., (2000), "HMMSTR: a Hidden Markov Model for Local Sequence-Structure Correlations in Proteins", J.Mol. Biolog. vol.301:173-190.

Byvatov, E., Fechner, U., Sadowski, J. ve Schneider, G., (2003), "Comparison of Support Vector Machine and Artificial Neural Networks Systems For Drug/Nondrug Classification", Journal of Chemical Information and Modeling, 43**:**1882-1889.

Cassabaum, M. L., Waagen, D. E., Rodriguez, J. J. ve Schmitt, H. A., (2004), "Unsupervised optimization of support vector machine parameters", Automatic Target Recognition XIV. Proceedings of the SPIE, Volume 5426 edited by Firooz A. Sadjadi, pages 316-325.

Castelli, V. ve Cover, T., (1996), "The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter", IEEE Transactions on Information Theory, 42, 2101–2117.

Cataltepe, Z., Ayan, U. ve Aygun, E., (2004), "Protein Function Prediction Using Motifs, Sequence Features, Alignment Scores", Eighth Annual International Conference on Computational Molecular Biology, RECOMB 2004, March 27-31, San Diego, California, USA.

Cauwenberghs, G. ve Poggio, T., (2001), "Incremental and decremental support vector machine learning", In Advances in Neural Information Processing Systems 13. MIT Press.

Cedeno, W. ve Agrafiotis, D., (2005), "Particle Swarms For Drug Design", IEEE Congress on Evolutionary Computation, CEC 2005, 2-4 September 2005, Edinburgh, UK

Chalimourda, A., Scholkopf, B. ve Smola, A. J., (2000), "Choosing 'nu' in support vector regression with different noise models—theory and experiments", IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, Volume 5, edited by Shun-Ichi Amari et al., Pages 199-204.

Chalimourda, A., Scholkopf, B. ve Smola, A. J., (2004), "Experimentally optimal 'nu' in support vector regression for different noise models and parameter settings", Neural Networks, Volume 17, Issue 1, January 2004, Pages 127-141.

Chapelle, O., Vapnik, V., Bousquet, O. ve Mukherjee, S., (2002), "Choosing multiple parameters for support vector machines", Machine Learning, 46(1-3):131-159.

Chapelle, O. ve Zien, A., (2005), "Semi-supervised classification by low density separation", In Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, pages 57–64, 2005.

Chapelle, O., Scholkopf, B., ve Zien, A., (2006), Semi-Supervised Learning, The MIT Press, Cambridge.

Cherkasov, A., (2006), "Can Bacterial-Metabolite-Likeness' Model Improve Odds of In-Slico' Antibiotic Discovery?", Journal of Chemical Information and Modeling, 46: 1214-1222.

Cherkassky, V. ve Yunqian, M.A., (2002), "Selection of meta-parameters for support vector regression", Artificial Neural Networks -- ICANN 2002: International Conference, Madrid, Spain, August 2002, Proceedings, edited by José R. Dorronsoro, pages 687-693.

Cherkassky, V. ve Yunqian, M.A., (2004), "Practical selection of SVM parameters and noise estimation for SVM regression", Neural Networks, Volume 17, Issue 1, January 2004, 113-126.

Ching, Y. S., (1979), "N-gram statistics for natural language understanding and text processing", IEEE Trans. Pattern Analysis and Machine Intelligence, 1(2):164–172, Apr. 1979.

Chipperfield, A. J., Fleming, P. J., ve Fonseca, C. M., (1994), "Genetic Algorithm Tools For Control Systems Engineering", Proc. Adaptive Computing In Engineering Design and Control, 128-133.

Cristianini, N. ve Shawe-Taylor, J., (2000), An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press.

Clardy, J. ve Walsh, C., (2004), "Lessons From Natural Molecules", Nature, 432: 829-837.

Clark, D.E. ve Pickett, S.D., (2000), "Computational Methods For The Prediction of Drug-Likeness", Drug Discovery Today, 5(2): 49-58.

Coley, A.D., (1999), An Introduction to Genetic Algorithms For Scientists and Engineers, World Scientific, Singapore.

Collins, M. ve Singer, Y., (1999), "Unsupervised models for named entity classification", EMNLP/VLC-99.

Conforti, D. ve Guido, R., (2009), "Kernel based support vector machine via semidefinite programming: Application to medical diagnosis", Computers and Operations Research, 2009.

Corpet, F., Gouzy, J., Kahn, D., (1999), "Recent improvements of the ProDom database of protein domain families", Nucleic Acids Res. 27:263-267.

Cover, T.T., ve Hart, P.E., (1967), "Nearest Neighbour Pattern Classification", IEEE Transaction on Information Theory, 11:21-27.

Damashek, M., (1995), “Gauging similarity with n-grams: Language-independent categorization of text”, Science, 267(5199):843–848.

Daview, D.L. ve Bouldin, D.W., (1979), “A Cluster Seperation Measure”, IEEE Transaction on Pattern Analysis Machine Intelligence, 224:227.

Debnath, R. ve Takahashı, H., (2004), “An efficient method for tuning kernel parameter of the support vector machine”, Proceeding of the IEEE International Symposium on Communications and Information Technology, 2004 (ISCIT 2004), Volume 2, pp. 1023-1028.

Diehl, C.P. ve Cauwenberghs, G., (2003), “SVM Incremental Learning, Adaptation and Optiization”, In Proceedings of the 2003 International Joint Conference on Neural Networks.

Dietterich, T. G., (1999), “An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, RAndomization”, Machine Learning, 1-22.

Ding, C. ve Peng, H., (2003), “Minimum redundancy feature selection from microarray gene expression data”, CSB03.

Eddy, S.R., (1998), “Profile hidden Markov models”, Bioinformatics. vol.14: 755-763.

Embrechts, M. J., Arciniegas, F., Özdemir, M., Momma, M., Breneman, C.M., Lockwood, L., Bennett, K. P. ve Kewley, R.H., (2002), “Stripmining For Molecules”, International Joint Conference on Neural Networks, IJCNN '02, 12-17 May, 2002, Hawaii, USA.

Engel, T., (2006), “Basic Overview of Chemoinformatics”, Journal of Chemical Information and Modeling, 46: 2267-2277.

Ester, M., Kriegel, H., Sander, J., ve Xu, X., (1996), “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”, KDD 96, 2-4 August 1996, Portland, Oregon.

Falquet, L., Pagni, M., Bucher, P., Hulo, N., Sigrist, C.J, Hofmann, K. ve Bairoch, A., (2002), "The PROSITE database, its status in 2002", Nucl. Acids Res. 30(1):235-238.

Henikoff, J.G., Greene, E.A., Pietrokovski, S. ve Henikoff, S., (2000), "Increased coverage of protein families with the blocks database servers", Nucl. Acids Res. 28(1):228-230.

Fralick, S. C., (1967), “Learning to recognize patterns wothout a teacher”, IEEE Transactions on Information Theory, 13:57–64.

Frank, E., Hall., M., ve Pfahringer, B., (2003), “Locally Weighted Naive Bayes”, 19th Conference in Uncertainty in Artificial Intelligence, UAI’03, 7-10 August 2003, Acapulco, Mexico.

Frohlich, H. ve Zell, A. (2005), “Efficient parameter selection for support vector machines in classification and regression via model-based global optimization”, Proceedings of the 2005 IEEE International Joint Conference on Neural Networks (IJCNN '05), Volume 3, pages 1431-1436

Freund, Y., Seung, H., Shamir, E., ve Tishby, N., (1997), “Selective sampling using the Query By Committee algorithm”, Machine Learning, 28, 133–168.

Frimurer, T. M., Bywater, R., Nærum, L., Nørskov Lauritsen L., ve Brunak, S., (2000), “Improving The Odds In Discriminating “Drug-Like” From “Non Drug-Like” Compounds”, Journal of Chemical Information and Modeling, 40: 1315-1324.

Galmeanu, H. ve Andonie, R., (2008), “Implementation Issues of an Incremental and Decremental SVM”, In Proceedings of Artificial Neural Networks - ICANN 2008.

Gasteiger, J. ve Engel, T., (2003), Chemoinformatics: A Textbook, Wiley-Vch, Weinheim.

Gasteiger, J., Teckentrup, A., Terfloth, L. ve Spycher, S., (2003), “Neural Networks As Data Mining Tools In Drug Design”, J. Phy. Org. Chem, 16: 232- 245.

Gates, G.W., (1972), ”The Reduced Nearest Neighbour Rule”, IEEE Transaction on Information Theory, 18:431-433.

Geneste, N.M., Watson, K.A., Alsberg., B.K. ve King, R.D., (2002), “New Approach Pharmacophore Maping and QSAR Analysis Using Inductive Logic Programming” Application to Thermolysin Inhibitors and Glycogen Phosphorylase b Inhibitors”, Journal of Medical Chemistr, 45:399-409.

Gold, C. ve Sollich, P., (2005), “Fast Bayesian Support Vector Machine Parameter Tuning with the Nystrom Method”, Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN '05), Volume 5, pages 2820-2825

Gönen, M. ve Alpaydın, E., (2008), "Localized Multiple Kernel Learning", In Proceedings of the 25th International Conference on Machine Learning (ICML 2008), Helsinki, Finland, 352-359.

Holland, J.H., (1973), “Genetic Algorithms and the Optimal Allocation of Trials”, SIAM J. Comput. Volume 2, Issue 2, pp. 88-105.

Holte, R.C., (1993), “Very Simple Classification Rules Perform Well on Most Commonly Used Datasets”, Machine Learning, vol. 11, pp. 63-91.

Hutter, M. C., (2007), “Separating Drugs From Nondrugs: A Statistical Approach Using Atom Pair Distribution”, The Journal of Chemical Information and Modeling, 47: 186-194.

Jaynes, E.T., (1957), “Information Theory and Statistical Mechanics”, Physical Review, 106(4):620-630.

Joachims, T., (1999), “Transductive inference for text classification using support vector machines”, Proc. 16th International Conf. on Machine Learning (pp. 200-209, Morgan Kaufmann, San Francisco, CA.

Karakoc, E., Sahinalp, C. ve Cherkasov, A., (2006), “Comparative Qsar and Fragments Distribution Analysis of Drugs, Drug Likes, Metabolic Substances and Antimicrobial Compounds”, Journal of Chemical Information and Modeling, 46(5): 2167-2182.

Karna, K. N., ve Breen, D. M., (1998), “An Artificial Neural Networks Tutorial: Part 1 Basics”, Neural Networks, 1:4-23.

Keller, H.T., Pichota, A. ve Yin, Z., (2007), “A Practical View of Druggability”, Current Opinion In Chemical Biology, 10: 357-361.

Kerr, M. K. ve Churchill, G. A., (2001), “Experimental design for gene expression microarrays”, Biostatistics 2:183-201

Kewley, R., Embrecht, M.J. ve Breneman, C., (1999), “A Soft Computing Approach for the Design of Novel Pharmaceuticals”, Transaction of to 1999 IEEE Midnight-Sun Workshop on Soft Computing Methods in Industrial Applications, Kuusamo, Finland.

Kim, J. Y. ve Shawe-Taylor, J.,(1994), “Fast String Matching using n-gram Algorithm”, Software-Practice and Experience, vol 24(1), 79-88.

Kohonen, T., (1990), “The Self-Organizing Map”, Proc. of IEEE, 78(9): 1464-1480

Krogh, A., (1998), “An introduction to hidden Markov models for biological sequences”, In Salzberg, S., Searls, D., and Kasif, S., editors, Computational Methods in Molecular Biology, pages 45–63. Elsevier.

Krogh, A., Brown, M., Mian, I. S., Sjolander, K., ve Haussler, D., (1994), "Hidden Markov models in computational biology: Applications to protein modeling", J. Mol. Biol., 235:1501–1531.

Lanckriet, G. R. G., Bie, T. D., Cristianini, N., Jordan, M. I. ve Noble, W. S. (2004a), "A statistical framework for genomic data fusion", Bioinformatics, 20, 2626-2635.

Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E. ve Jordan, M. I. (2004b), "Learning the kernel matrix with semidefinite programming", Journal of Machine Learning Research, 5, 27-72.

Larkin, M.A., Blackshields, G., Brown, N.P., Chenna, R., McGettigan, P.A., McWilliam, H., Valentin, F., Wallace, I.M., Wilm, A., Lopez, R., Thompson, J.D., Gibson, T.J. ve Higgins, D.G., (2007), "ClustalW and ClustalX version 2", Bioinformatics 2007 23(21): 2947-2948.

Lassmann, T. ve Sonnhammer, E.L.L., (2006), "Kalign, Kalignvu and Mumsa: web servers for multiple sequence alignment", Nucleic Acids Research, 34:W596-W599

Leslie, C., E. Eskin, ve W. S. Noble, (2002) "The spectrum kernel: A string kernel for SVM protein classification", In R. B. Altman, A. K. Dunker, L. Hunter, K. Lauderdale, and T. E. Klein, editors, Proceedings of the Pacific Symposium on Biocomputing, pages 564–575, Kaua'i, Hawaii, 2002.

Leslie, C., Eskin, E., Weston, J. ve Noble, W., (2003a), "Mismatch string kernels for discriminative protein classification", Bioinformatics, 20(4).

Leslie, C., Kuang, R. ve Eskin, E.,(2003b), "Inexact matching string kernels for protein classification", In Kernel Methods in Computational Biology, MIT Press series on Computational Molecular Biology, pages 95–112. MIT Press.

Leslie, C. ve Kuang, R., (2004), "Fast string kernels using inexact matching for protein sequences", Journal of Machine Learning Research, 5:1435–1455.

Lewis, D., ve Gale,W., (1994), "A sequential algorithm for training text classifiers", Proceedings of the International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp. 3–12.

Li, A.P., (2005), "Preclinical In Vitro Screening Assays For Drug-Like Properties", Drug Discovery Today: Technologies, 2(2): 179-185.

Li, Q., Bender, A., Pei, J. ve Lai, L., (2007), "A Large Descriptor Set and A Probabilistic Kernel-Based Classifier Significantly Improve Druglikeness Classification", The Journal of Chemical Information and Modeling, 47 (5): 1776–1786.

Lim, T., Loh, W. ve Shih, Y., (2000), "A Comparison of Prediction Accuracy, Complexity and Training Time of ThirtyThree Old and New Classification Algorithms", Machine Learning, 40:203-229.

Lin, Y., Liu, T. ve Fuh, C., (2009), "Dimensionality reduction for data in multiple feature representations", In Advances in Neural Processing Systems 21.

Lipinski, C. A., Lombardo, F., Dominy, B. W., ve Feeney, P. J., (1997), "Experimental and Computational Approaches to Estimate Solubility and Permeability in Drug Discovery and Development Settings", Advanced Drug Delivery Reviews, 23: 3–25.

Lipinski, C. A., (2000), "Drug-like properties and the causes of poor solubility and poor permeability", Journal of Pharmacological and Toxicological Methods, 44:235-249.

Lipinski, C. A., (2003), "Physicochemical Properties and The Discovery of Orally Active Drugs: Technical and People Issues", Molecular Informatics Confronting Complexity, Proceedings of The Beilstein International Workshop, Logos Verlag, Berlin.

Lipinski, C. A., (2004), "Lead- and Drug-Like Compounds: The Rule-of-Five Revolution", Drug Discovery Today: Technologies, 1(4), 337-341.

Littman M. L. ve Moore A. W., (1996), "Reinforcement Learning: A Survey", Journal of Artificial Intelligence Research 4, 237–285

Liu, Y., (2005), "Drug Design By Machine Learning: Ensemble Learning For Qsar Modeling", The 4th Int. Conference On Machine Learning and Applications, ICMLA 05, 15-17 Aralık, 2005, Los Angeles, California, USA.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N. ve Watkins, C.(2002), "Text Classification Using String Kernels", Journal of Machine Learning Research, 2:419–444.

Longworth, C. ve Gales, M. J. F., (2009), "Combining derivative and parametric kernels for speaker verification", IEEE Transactions on Audio, Speech, and Language Processing, 17(4):748-757.

Marko, R.S. ve Igor, K., (2003), "Theoritical and empirical analysis of ReliefF and rReliefF", Machine Learning Journal, 53:23-69.

McCallum, A. K., ve Nigam, K., (1998), "Employing EM in pool-based active learning for text classification", Proceedings of ICML-98, 15th International Conference on Machine Learning (pp. 350–358). Madison, US: Morgan Kaufmann Publishers, San Francisco, US.

Morgenstern, B., (2004), "DIALIGN: Multiple DNA and Protein Sequence Alignment at BiBiServ", Nucleic Acids Research 32, W33-W36.

Mitchell,T., (1997), Machine Learning, McGraw Hill.

Muegge, I., Heald, S.L. ve Britelli, D., (2001), "Simple Selection Criteria For Drug-Like Chemical Matter", The Journal of Medicinal Chemistry., 44(12): 1841-1846.

Murcia-Soler, M., Pe´Rez-Gimenez, F., Garcia-M., J., Salabertsalvador, M. T., Diaz-Villanueva, W. ve Castro-Bleda, M. J., (2003), "Drugs and Nondrugs: An Effective Discrimination With Topological Methods and Artificial Neural Networks", The Journal of Chemical Information and Modeling, 43: 1688-1702.

Needleman, S. B. ve Wunsch, C. D., (1970), "A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol., 48:443–453.

Nigam, K., McCallum, A. K., Thrun, S. ve Mitchell, T., (2000), "Text classification from labeled and unlabeled documents using EM", Machine Learning, 39, 103–134.

Nilson, J. N., (1996), Introduction to Machine Learning, Standford University, California.

Notredame, C., Higgins,D. ve Heringa, J., (2000), "T-Coffee: A novel method for multiple sequence alignments", Journal of Molecular Biology, 302, 205-217.

Ozdemir, M., Embrechts, M.J., Arciniegas, F., Breneman, C., Lockwood, L. ve Bennett, K. (2001), "Feature Selection for In-Silico Drug Design Using Genetic Algorithms and Neural Networks", Proceedings 2001 SMCia Mountain Workshop on Soft Computing in Industrial Applications, IEEE Press, Blacksburg, Virginia, 53-57.

Parida, L., Floratos, A. ve Rigoutsos, I., (1998), "MUSCA: An Algorithm for Constrained Alignment of Multiple Data Sequences", Proceedings 9th Workshop on Genome Informatics, Tokyo, Japan.

Parker, D.B., (1982), "Learning logic", Invention report S81-64, File 1, Office of Technology Licensing, Stanford University.

Parker, D.B., (1985), Massachusetts Institute of TechnoIogy, Center for Computational Research in Economics and Management Science, TR-47, Cambridge, MA.

Parzen, E., (1962), "On Estimation of a Probability Density Function and Mode", The Annals of Mathematical Statistics, 33: 1065-1076.

Pavlidis, P., Weston, J., Cai, J. ve Grundy, W.N., (2001), "Gene functional classification from heterogeneous data", In Proceedings of the 5th Annual International Conference on Computational Molecular Biology.

Pehlivanli, A.C., (2008), "Consensual Classification of Drug/Nondrug Compounds for Drug Design", Phd Thesis, Çukurova University, Adana, Turkey.

Pehlivanli, A.C., Ersoy, O.K. ve Ibrikci, T., (2008), "Drug/nondrug classification with consensual Self-Organising Map and Self-Organising Global Ranking algorithms", Int. J. Computational Biology and Drug Design, Vol. 1, No. 4, pp.434–445.

Rakotomamonjy, A., Bach, F., Canu, S. ve Grandvalet, Y., (2007), "More efficiency in multiple kernel learning", Proceedings of the 24th International Conference on Machine Learning (pp. 775-782).

Rakotomamonjy, A., Bach, F., Canu, S. ve Grandvalet, Y., (2008), "SimpleMKL", Journal of Machine Learning Research, 9:2491-2521.

Robnik-Šikonja, M., Kononenko, I., (2003), "Theoretical and Empirical Analysis of ReliefF and RreliefF", Machine Learning 53, 23–69.

Romero, E., Barrio, I., ve Belanche, L., (2007), "Incremental and Decremental Learning for Linear Support Vector Machines", Proceedings of the International Conference of Artificial Neural Networks, ICANN 2007.

Rosenblatt, F. (1958), "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain", Psychological Review , 65 (6): 386-408.

Roy, N. ve McCallum, A., (2001),"Toward optimal active learning through sampling estimation of error reduction", Proc. 18th International Conf. on Machine Learning, pp. 441–448, Morgan Kaufmann, San Francisco, CA

Rumelhart, D. E., Durbin, R., Golden, R., ve Chauvin, Y., (1995), Backpropagation: Theory, Architectures and Applications, Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Rüping, S., (2001), "Incremental Learning with Support Vector Machines", Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01)

Salton, G., (1979), "Mahematics and Informational Retieval", Journal of Documentation, 35(1): 1-29.

Samuel, A.L., (1959), "Some Studies in Machine Learning Using the Game of Checkers," IBM. Journal 3, 211-229.

Schölkopf, B. ve Smola, A. J., 2002, Learning With Kernels: Support Vector Machines, Regularization and Beyond. The MIT Press, Cambridge.

Scudder, H. J., (1965), "Probability of error of some adaptive pattern-recognition machines", IEEE Transactions on Information Theory, 11:363–371.

Seeger, M., (2001), Learning with labeled and unlabeled data (Technical Report), University of Edinburgh.

Seung, H. S., Opper, M., ve Sompolinsky, H., (1992), "Query by committee", Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory , pp. 287–294.

Simon, H. A., (1983), "A comparison of game theory and learning theory", In H. A. Simon, Models of bounded rationality. Vol.2: Behavioral economics and business organization, pp. 269-274.

Smith, T. F. ve Waterman, M. S.,(1981), "Identification of common molecular subsequences", J. Mol. Biol., 147:195–197.

Sonnenburg, S., Rätsch, G., Schäfer, C. ve Schölkopf, B., (2006a), "Large scale multiple kernel learning", Journal of Machine Learning Research, 7, 1531-1565.

Sonnenburg, S., Rätsch, G., Schäfer, C. ve Schölkopf, B., (2006b), "A general and efficient multiple kernel learning algorithm", In Advances in Neural Information Processing Systems 18.

Söding, J., Biegert, A., ve Lupas, A.N., (2006), "The HHpred interactive server for protein homology detection and structure prediction", Nucleic Acids Research, vol.33, 244-248.

Sun, Y., (2007),"Iterative RELIEF for Feature Weighting: Algorithms, Theories, and Applications", IEEE Transactions on Pattern Analysis and Machine Intelligent 29, 1035–1051

Suykens, J., Gestel, T. V., Brabenter, J. D., Moor, B. D. ve Vandewalle, J., (2002), Least Squares Support Vector Machines, World Scientific, New Jersey.

Syed, N.A., Liu, H. ve Sung, K.K., (1999), "Incremental Learning with Support Vector Machines," in Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI-99), July 31 - August 6, 1999, Stockholm, Sweden.

Turing, A., (1950), "Computing Machinery and Intelligence", Mind , 59: 433-460.

(2000), "T-Coffee: A novel method for multiple sequence alignments", Notredame, Higgins, Heringa, JMB, 302 pp:205-217.

Vapnik, V., (1998), Statistical Learning Theory, John Wiley & Sons, New York.

Veber, D.F, Johnson, S.R., Cheng, H.Y., Smith, B.R, Ward, K.W. ve Kopple, K.D., (2002), "Molecular Properties that Influence The Oral Bioavailability or Drugs", The Journal of Medicinal Chemistry, 45: 2615-2623

Vishwanathan, S. V. N. ve Smola, A. J., (2003), "Fast kernels string and tree matching", In NIPS, pages 569-576.

Wagener, M. ve Van Geerestein, V.J., (2000), "Potential Drug and Nondrugs: Prediction and Identification of Important Structural Features", J. Chem. Inf. Comput. Science, 40: 280-292.

Walters, W.P. ve Murcko, M.A., (2002), "Prediction of Drug-Likeness", Advance Drug Delivery Reviews, 54: 255-271.

Wang, Z., Durst, G.L., Eberhart, R.C., Boyd, D.B. ve Miled, Z.B., (2004), "Particle Swarm Optimization and Neural Network Application for QSAR", 18th International Parallel and Distributed Processing Symposium, IPDPS'04, Sante Fe, New Mexico.

Werbos, P.J. (1974/1994), The Roots of Backpropagation, NY: John Wiley & Sons. Includes Werbos's 1974 Harvard Ph.D. thesis, Beyond Regression.

Weston, J., Cruz, F.P., Bousquet, O., Chapelle, O., Elisseeff, A. ve Schölkopf, B., (2003), "Feature Selection and Transduction for Prediction of Molecular Bioactivity for Drug Design", Bioinfomatics, 19:764-771.

Widrow, B., ve Hoff, M. E. (1960), "Adaptive Switching Circuits", IRE WESCON Convention Record, 4:96-104, August 1960.

Widrow, B., (1992), “Generalization and Information Storage In Networks of Adaline 'Neurons'”, Self-Organizing Systems , 435-461.

Wishart, D., (2005), The Bioinformatics of Small Molecules, University of Alberta, Canada.

Xu, Z., Jin, R., King, I. ve Lyu, M.R., (2009a), “An extended level method for efficient multiple kernel learning”, In Advances in Neural Information Processing System 21.

Xu, Z., Jin, R., Ye, J., Lyu, M.R., ve King, I., (2009b), “Non-monotonic feature selection”, In Proceedings of the 26th International Conference on Machine Learning.

Yarowsky, D., (1995), “Unsupervised word sense disambiguation rivaling supervised methods”, Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, pp. 189–196.

Zien, A. ve Ong, C.S., (2007), “Multiclass Multiple Kernel Learning”, in Proceedings of the 24th International Conference on Machine Learning, Corvallis, OR, 2007.

Zhang, J. ve Madden, T.L., (1997), "PowerBLAST: A new network BLAST application for interactive or automated sequence analysis and annotation", Genome Res., 7:649-656.

Zhao, B., Kwok, J.T. ve Zhang,C., (2009), “Multiple kernel clustering”, In Proceedings of the 9th SIAM International Conference on Data Mining.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., Schölkopf, B., “Learning with Local and Global Consistency”, Advances in Neural Information Processing Systems 16, (Eds.) Thrun, S., L. Saul and B. Schölkopf, MIT Press, Cambridge, Mass. (2004a) 321-328

Zhou, D. ve Scholkopf. B., (2004b) “Learning from labeled and unlabeled data usingrandom walks”, In DAMG’04: Proc. of the 26th Pattern Recognition Symposium,2004.

Zhu, X. J., Ghahramani, Z., Lafferty, J., (2003a), “Semi-supervised Learning Using Gaussian Fields and Harmonic Functions”, Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC.

Zhu, X. J., Lafferty, J., Ghahramani, Z., (2003b), “Combining Active Learning and Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions”, In Proc. of the ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining. pp. 58-65

EKLER

Ek 1 Teknik terimler sözlüğü

active	etken/aktif
artificial intelligence	yapay us
backpropagation	geriye yayılım
clustering	öbekleme
decremental	azalımlı
domain	bir proteinin fonksiyona sahip bölümü
empirical error	deneysel hata
feature space	öznitelik/özellik uzayı
fold	katlama
global	tümel
hyperplane	çok boyutlu düzlem/aşırıdüzlem
incrementel	artımlı
kernel	çekirdek
least mean square	en küçük ortalamalı kareler
literature	bilimsel yazın
margin	pay
maximization	ençoklama
minimization	enazlama
optimal	en uygun
optimization	eniyileme
perturbation	sarsım
regularization	düzenlileştirme
robust	gürbüz
string	dizgi / karakter dizisi
support vector machine	destek vektör

Ek 2 Kullanılan bazı teknik terimlerin açıklamaları

Aşırı Eğitme : Bir sınıflandırıcının eğitim kümesi üzerindeki başarımının yüksek olması fakat eğitim sürecinde görmediği örnekler üzerindeki performansının düşük olmasını aşırı eğitme şeklinde ifade edebiliriz.

Çarpraz Geçerleme : Algoritmalarda rastgeleliği azaltmak için kullanılan en önemli yöntemlerden birisidir. K adet alt kümeye ayrılan veri kümesinden her defasında farklı K-1 adet kümeyi eğitmek ve geriye kalan 1 adet kümeyi de algoritmanın başarımını test etmek için kullanılmaktadır. Böylece algoritma K defa tekrarlanmış olur. Başarım hesabı K defa elde edilen sonuçların performansının ortalaması hesaplanarak elde edilir.

Eğitim Kümesi : Bir algoritmanın modelini oluşturmak için gerekli alt veri kümesidir.

Öbekleme : Çeşitli nitelikteki parçaları birbirine olan uzaklarına, benzerliklerine veya özelliklerine göre ayırma.

Rastgele Başarı : Bir algoritmanın başarılı olduğunu belirten en küçük performanstır.

Sınıflandırma : Veri setinin sınırlı sayıdaki tanımlı sınıflara kestirimidir.

Test Kümesi : Eğitim kümesi sonucunda oluşturulmuş olan bir modelin başarımını ölçmek için eğitim kümesinde olmayan diğer bir alt kümedir.

Ek 3 PDB Formatlı bir bileşik örneği

```
HEADER    DNA-RNA HYBRID                          05-DEC-94   100D
TITLE     CRYSTAL STRUCTURE OF THE HIGHLY DISTORTED CHIMERIC DECAMER
TITLE    2 R(C)D(CGGCGCCG)R(G)-SPERMINE COMPLEX-SPERMINE BINDING TO
TITLE    3 PHOSPHATE ONLY AND MINOR GROOVE TERTIARY BASE-PAIRING
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: DNA/RNA (5'-R(*CP*)-D(*CP*GP*GP*CP*GP*CP*CP*GP*)-
COMPND   3 R(*G)-3');
COMPND   4 CHAIN: A, B;
COMPND   5 ENGINEERED: YES
SOURCE    MOL_ID: 1;
SOURCE   2 SYNTHETIC: YES
KEYWDS    A-DNA/RNA, DOUBLE HELIX, DNA-RNA HYBRID
EXPDTA    X-RAY DIFFRACTION
AUTHOR    C.BAN,B.RAMAKRISHNAN,M.SUNDARALINGAM
REVDAT   3   24-FEB-09 100D    1       VERSN
REVDAT   2   01-APR-03 100D    1       JRNL
REVDAT   1   31-MAR-95 100D    0
JRNL        AUTH   C.BAN,B.RAMAKRISHNAN,M.SUNDARALINGAM
JRNL        TITL   CRYSTAL STRUCTURE OF THE HIGHLY DISTORTED CHIMERIC
JRNL        TITL 2 DECAMER R(C)D(CGGCGCCG)R(G).SPERMINE
JRNL        TITL 3 COMPLEX--SPERMINE BINDING TO PHOSPHATE ONLY AND
JRNL        TITL 4 MINOR GROOVE TERTIARY BASE-PAIRING.
JRNL        REF    NUCLEIC ACIDS RES.           V.  22  5466 1994
JRNL        REFN                ISSN 0305-1048
JRNL        PMID   7816639
JRNL        DOI    10.1093/NAR/22.24.5466
REMARK   1
REMARK   2
REMARK   2 RESOLUTION.    1.90 ANGSTROMS.
REMARK   3
REMARK   3 REFINEMENT.
REMARK   3   PROGRAM     : X-PLOR
REMARK   3   AUTHORS     : BRUNGER
REMARK   3
REMARK   3  DATA USED IN REFINEMENT.
REMARK   3   RESOLUTION RANGE HIGH (ANGSTROMS) : 1.90
REMARK   3   RESOLUTION RANGE LOW  (ANGSTROMS) : 8.00
REMARK   3   DATA CUTOFF           (SIGMA(F)) : 1.000
REMARK   3   DATA CUTOFF HIGH         (ABS(F)) : NULL
REMARK   3   DATA CUTOFF LOW          (ABS(F)) : NULL
REMARK   3   COMPLETENESS (WORKING+TEST)   (%) : NULL
REMARK   3   NUMBER OF REFLECTIONS             : 2314
REMARK   3  NUMBER OF NON-HYDROGEN ATOMS USED IN REFINEMENT.
REMARK   3   PROTEIN ATOMS            : 0
REMARK   3   NUCLEIC ACID ATOMS       : 408
REMARK   3   HETEROGEN ATOMS          : 14
REMARK   3   SOLVENT ATOMS            : 67
REMARK   3
REMARK   3  B VALUES.
REMARK   3   FROM WILSON PLOT           (A**2) : NULL
REMARK   3   MEAN B VALUE      (OVERALL, A**2) : NULL
REMARK   3   OVERALL ANISOTROPIC B VALUE.
REMARK   3    B11 (A**2) : NULL
REMARK   3    B22 (A**2) : NULL
REMARK   3    B33 (A**2) : NULL
REMARK   3    B12 (A**2) : NULL
```

```
REMARK   3    B13 (A**2) : NULL
REMARK   3    B23 (A**2) : NULL
REMARK   3
REMARK   3  ESTIMATED COORDINATE ERROR.
REMARK   3   ESD FROM LUZZATI PLOT        (A) : NULL
REMARK   3   ESD FROM SIGMAA              (A) : NULL
REMARK   3   LOW RESOLUTION CUTOFF        (A) : NULL
REMARK   3
REMARK   3  ISOTROPIC THERMAL MODEL : NULL
REMARK   3
REMARK   3  ISOTROPIC THERMAL FACTOR RESTRAINTS.    RMS    SIGMA
REMARK   3   MAIN-CHAIN BOND              (A**2) : NULL  ; NULL
REMARK   3   MAIN-CHAIN ANGLE             (A**2) : NULL  ; NULL
REMARK   3   SIDE-CHAIN BOND              (A**2) : NULL  ; NULL
REMARK   3   SIDE-CHAIN ANGLE             (A**2) : NULL  ; NULL
REMARK   3
REMARK   3  PARAMETER FILE  1  : NULL
REMARK   3  TOPOLOGY FILE  1   : NULL
REMARK   3
REMARK   3  OTHER REFINEMENT REMARKS: NULL
REMARK   4
REMARK   4 100D COMPLIES WITH FORMAT V. 3.15, 01-DEC-08
REMARK 100
REMARK 100 THIS ENTRY HAS BEEN PROCESSED BY BNL.
REMARK 200
REMARK 200 EXPERIMENTAL DETAILS
REMARK 200  EXPERIMENT TYPE               : X-RAY DIFFRACTION
REMARK 200  DATE OF DATA COLLECTION       : NULL
REMARK 200  TEMPERATURE          (KELVIN) : 263.00
REMARK 200  PH                            : 7.00
REMARK 200  NUMBER OF CRYSTALS USED       : NULL
REMARK 290
REMARK 290      SYMOP   SYMMETRY
REMARK 290     NNNMMM   OPERATOR
REMARK 290       1555   X,Y,Z
REMARK 290       2555   -X+1/2,-Y,Z+1/2
REMARK 290       3555   -X,Y+1/2,-Z+1/2
REMARK 290       4555   X+1/2,-Y+1/2,-Z
REMARK 290
REMARK 290     WHERE NNN -> OPERATOR NUMBER
REMARK 290           MMM -> TRANSLATION VECTOR
REMARK 290
REMARK 290 CRYSTALLOGRAPHIC SYMMETRY TRANSFORMATIONS
REMARK 290 THE FOLLOWING TRANSFORMATIONS OPERATE ON THE ATOM/HETATM
REMARK 290 RECORDS IN THIS ENTRY TO PRODUCE CRYSTALLOGRAPHICALLY
REMARK 290 RELATED MOLECULES.
REMARK 290   SMTRY1   1  1.000000  0.000000  0.000000        0.00000
REMARK 290   SMTRY2   1  0.000000  1.000000  0.000000        0.00000
REMARK 290   SMTRY3   1  0.000000  0.000000  1.000000        0.00000
REMARK 290   SMTRY1   2 -1.000000  0.000000  0.000000       11.99000
REMARK 290   SMTRY2   2  0.000000 -1.000000  0.000000        0.00000
REMARK 290   SMTRY3   2  0.000000  0.000000  1.000000       22.42000
REMARK 290   SMTRY1   3 -1.000000  0.000000  0.000000        0.00000
REMARK 290   SMTRY2   3  0.000000  1.000000  0.000000       20.38500
REMARK 290   SMTRY3   3  0.000000  0.000000 -1.000000       22.42000
REMARK 290
REMARK 500 THAN 6*RMSD (M=MODEL NUMBER; RES=RESIDUE NAME; C=CHAIN
REMARK 500 IDENTIFIER; SSEQ=SEQUENCE NUMBER; I=INSERTION CODE).
REMARK 500
```

```
REMARK 500 STANDARD TABLE:
REMARK 500 FORMAT: (10X,I3,1X,A3,1X,A1,I4,A1,3(1X,A4,2X),12X,F5.1)
REMARK 500
REMARK 500 EXPECTED VALUES PROTEIN: ENGH AND HUBER, 1999
REMARK 500 EXPECTED VALUES NUCLEIC ACID: CLOWNEY ET AL 1996
REMARK 500
REMARK 500  M RES CSSEQI ATM1   ATM2   ATM3
REMARK 500     DC A   2   C4' -  C3' -  C2' ANGL. DEV. =  -4.2 DEGREES
REMARK 500     DC A   2   O4' -  C1' -  N1  ANGL. DEV. =   3.2 DEGREES
REMARK 500     DC A   2   N1  -  C2  -  O2  ANGL. DEV. =   4.0 DEGREES
REMARK 500     DC A   2   O3' -  P   -  O5' ANGL. DEV. =  17.0 DEGREES
REMARK 500     DG A   3   C4' -  C3' -  C2' ANGL. DEV. =  -5.0 DEGREES
REMARK 500     DG A   3   O4' -  C1' -  N9  ANGL. DEV. =   6.9 DEGREES
REMARK 500     DG A   3   O3' -  P   -  O5' ANGL. DEV. =  25.6 DEGREES
REMARK 500     DC A   5   O4' -  C1' -  N1  ANGL. DEV. =   2.2 DEGREES
REMARK 500     DC A   5   O3' -  P   -  O5' ANGL. DEV. =  26.2 DEGREES
REMARK 500     DC A   5   O3' -  P   -  OP1 ANGL. DEV. = -16.4 DEGREES
REMARK 500 REMARK: NULL
REMARK 525  M RES CSSEQI
ORIGX1      1.000000  0.000000  0.000000        0.00000
ORIGX2      0.000000  1.000000  0.000000        0.00000
ORIGX3      0.000000  0.000000  1.000000        0.00000
SCALE1      0.041701  0.000000  0.000000        0.00000
SCALE2      0.000000  0.024528  0.000000        0.00000
SCALE3      0.000000  0.000000  0.022302        0.00000
ATOM      1  O5'   C A   1      -4.549   5.095   4.262  1.00 28.71           O
ATOM      2  C5'   C A   1      -4.176   6.323   3.646  1.00 27.35           C
ATOM      3  C4'   C A   1      -3.853   7.410   4.672  1.00 24.41           C
ATOM      4  O4'   C A   1      -4.992   7.650   5.512  1.00 22.53           O
ATOM      5  C3'   C A   1      -2.713   7.010   5.605  1.00 23.56           C
ATOM      6  O3'   C A   1      -1.379   7.127   5.060  1.00 21.02           O
ATOM      7  C2'   C A   1      -2.950   7.949   6.756  1.00 23.73           C
ATOM      8  O2'   C A   1      -2.407   9.267   6.554  1.00 23.93           O
ATOM      9  C1'   C A   1      -4.489   7.917   6.825  1.00 20.60           C
HETATM  488  O   HOH B  78      -8.873  -3.504  22.680  1.00 79.41           O
HETATM  489  O   HOH B  79      -9.310  -1.446  19.391  1.00 71.79           O
HETATM  490  O   HOH B  85      13.277  -1.574   6.340  1.00 80.86           O
HETATM  491  O   HOH B  88      -2.029 -10.316  17.858  1.00 83.02           O
CONECT  411  412
CONECT  412  411  413
CONECT  413  412  414
CONECT  414  413  415
CONECT  424  423
MASTER      323    0    1    0    0    0    4    6  489    2   14    2
END
```

Ek 4 Örnek Prosite Motifi

ID CUTINASE_1; PATTERN.

AC PS00155;

DT APR-1990 (CREATED); NOV-1997 (DATA UPDATE); MAR-2005 (INFO UPDATE).

DE Cutinase, serine active site.

PA P-x-[STA]-x-[LIV]-[IVT]-x-[GS]-G-Y-S-[QL]-G.

NR /RELEASE=46.4,178022;

NR /TOTAL=20(20); /POSITIVE=20(20); /UNKNOWN=0(0); /FALSE_POS=0(0);

NR /FALSE_NEG=0; /PARTIAL=0;

CC /TAXO-RANGE=??EP?; /MAX-REPEAT=1;

CC /SITE=11,active_site;

DR P63880, CUT1_MYCBO , T; P63879, CUT1_MYCTU , T; P63882, CUT2_MYCBO , T;

DR P63881, CUT2_MYCTU , T; P0A537, CUT3_MYCBO , T; P0A536, CUT3_MYCTU , T;

DR P00590, CUTI1_FUSSO, T; Q96UT0, CUTI2_FUSSO, T; Q96US9, CUTI3_FUSSO, T;

DR P41744, CUTI_ALTBR , T; P29292, CUTI_ASCRA , T; P52956, CUTI_ASPOR , T;

DR Q00298, CUTI_BOTCI , T; P10951, CUTI_COLCA , T; P11373, CUTI_COLGL , T;

DR Q8X1P1, CUTI_ERYGR , T; Q99174, CUTI_FUSSC , T; P30272, CUTI_MAGGR , T;

DR Q8TGB8, CUTI_MONFR , T; Q9Y7G8, CUTI_PYRBR , T;

3D 1AGY; 1CEX; 1CUA; 1CUB; 1CUC; 1CUD; 1CUE; 1CUF; 1CUG; 1CUH; 1CUS; 1CUU;

3D 1CUV; 1CUW; 1CUY; 1CUZ; 1FFA; 1FFB; 1FFC; 1FFD; 1FFE; 1OXM; 1XZA; 1XZB;

3D 1XZC; 1XZD; 1XZE; 1XZF; 1XZG; 1XZH; 1XZJ; 1XZK; 1XZL; 1XZM; 2CUT;

DO PDOC00140;

Prosite motiflerleri için PDOC ile başlayan bir isimlendirme yapılmaktadır. İlk motifi ele alıdığımızda desenin nerden elde edildiği, hangi sınıfa ait olduğu bilgi ile içeriğine bakıldığında desenin kendisini görmekteyiz :

PDOC00284 1 11-S plant seed storage proteins signature
Desen : N - G - x - [DE](2) - x - [LIVMF] - C - [ST] - x(11,12) - [PAG] - D

Birkaç örnek vermek gerekirse :

PDOC00284 1 11-S plant seed storage proteins signature
PDOC00633 2 14-3-3 proteins signatures
PDOC00653 3 2'-5'-oligoadenylate synthases signatures and profile
PDOC00168 1 2-oxo acid dehydrogenases acyltransferase component lipoyl binding site
PDOC01048 1 2C-methyl-D-erythritol 2,4-cyclodiphosphate synthase signature
PDOC00175 2 2Fe-2S ferredoxin-type iron-sulfur binding domain signature and profile
PDOC00065 1 3-hydroxyacyl-CoA dehydrogenase signature
PDOC00697 1 3-hydroxyisobutyrate dehydrogenase signature
PDOC00116 1 3'5'-cyclic nucleotide phosphodiesterases signature
PDOC00997 1 4-diphosphocytidyl-2C-methyl-D-erythritol synthase signature
PDOC01031 1 4-hydroxybenzoyl-CoA thioesterase family active site
PDOC00339 1 43 Kd postsynaptic protein signature
PDOC00176 2 4Fe-4S ferredoxin-type iron-sulfur binding domain signature and profile
PDOC00627 2 5'-nucleotidase signatures
PDOC00390 1 6-phosphogluconate dehydrogenase signature
PDOC00759 2 6-pyruvoyl tetrahydropterin synthase signatures
PDOC00631 1 7,8-dihydro-6-hydroxymethylpterin-pyrophosphokinase signature
PDOC00572 1 AAA-protein family signature
PDOC51237 28 ABC transporter family profiles
PDOC00364 2 ABC transporter integral membrane type-1 domain profiles
PDOC00692 1 ABC transporter integral membrane type-2 domain profile
PDOC51426 1 ABL domain profile
PDOC00826 2 Acetate and butyrate kinases family signatures
PDOC50980 2 Acetyl-coenzyme A carboxyltransferase domain profiles
PDOC00423 2 Aconitase family signatures
PDOC00019 2 Actinin-type actin-binding domain signatures
PDOC00340 3 Actins signatures
PDOC00070 2 Acyl-CoA dehydrogenases signatures
PDOC00686 2 Acyl-CoA-binding (ACB) domain signature and profile

ÖZGEÇMİŞ

Doğum tarihi	13.02.1978	
Doğum yeri	İstanbul	
Lise	1993-1996	Samsun Bafra Anadolu Lisesi
Lisans	1996-2001	Boğaziçi Üniversitesi Mühendislik Fakültesi İnşaat Mühendisliği Bölümü
Yüksek Lisans	2001-2004	Boğaziçi Üniversitesi Fen Bilimleri Enstitüsü Sistem ve Kontrol Mühendisliği Bölümü
Doktora	2004-2007	İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Bölümü
Doktora	2007-2010	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik Müh. Anabilim Dalı, Kontrol ve Oto. Prog.

Çalıştığı kurum(lar)

2000-2001	Boğaziçi Üniversitesi Polymer Araştırma Merkezi Öğrenci Araştırma Görevlisi
2001-2002	Haliç Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Böl. Araştırma Görevlisi
2002-Devam	T.C. Kültür Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Böl. Araştırma Görevlisi
2005-2007	İstanbul Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü Yarı Zamanlı Araştırmacı - Tübitak 105E164 Project