SON RAPOR

APACHE FLİNK İLE FİYAT DURUM TAHMİNİ ANALİZİ

Danışman:

Doç. Dr. Suat ÖZDEMİR

Grup Üyeleri:

141180035 SİMGE NUR KABATAŞ

131180031 ZEYNEP FENERCİ

BM495 BİLGİSAYAR PROJESİ I

Ocak 2018

İÇİNDEKİLER

1.Giriş 4
2.Projenin Gerçekleştirilmesi 4
2.1.Proje Özet 4
2.2.Proje Hedefi 5
2.3.Projede Mevcut Aşama 5
Veri Seçimi ve Önişleme 5
Veri Çözümleme 5
Ölçekleme (scaling) 5
2.3.1.Tasarım Kararları 6
2.3.1.1.Yazılım Modüller 6
2.3.1.1.1.Kullanılan Teknolojiler 6
2.3.1.2.Projeye Dair İstisnai Durumların İdaresi 7
2.3.2.Test 7
2.3.2.1.Test Edilen Özellikler 7
2.3.2.1.1. Performans Testi 7
2.3.2.1.2. Güvenlik Testi 8
2.3.2.1.3.Başarı ve Başarısızlık Kriterleri 8
2.3.2.2.Test Sonuçları 8
2.3.2.3.Teste Yönelik Belirtilmemiş Değişiklikler 9
3. Apache Flink 9
3.1. Apache Flink Tarihçesi 10
3.2. Ekosistemi 10
3.2.1. Depolama 11
3.2.2. Dağıtım 11
3.2.3. Flink çekirdeği 12
3.2.4. API'ler ve Kütüphane 12
3.2.4.1. DataSet API'sı 12
3.2.4.2. DataStream API'sı 12
3.2.4.3. Tablo API'sı 12
3.2.4.4. Gelly 13
3.2.4.5. FlinkML - Flink için Makine Öğrenimi 13
3.2.4.6. FlinkCEP - Flink için kompleks olay işleme 14

3.3. Mimarisi 14
3.4. Özellikleri 15
3.5.Apache Flink'in Kullanildiği Yerler 16
3.5.1.Bouygues Telecom - Fransa'da üçüncü büyük mobil sağlayıcısı. 16
3.5.2.King - Candy Crush Saga'nın yaratıcısı 16
3.5.3.Zalando - Avrupa'da E-Ticaret Şirketinin Liderliği 17
3.5.4.Otto Group - Dünyanın en büyük ikinci online perakendecisi 17
3.5.5.ResearchGate - En büyük akademik sosyal ağ 17
3.5.6.Alibaba Group - Dünyanın en büyük perakendecisi 17
3.5.7.Capital One - Fortune 500 finansal hizmetler şirketi 17
4.Durum Değerlendirmesi 18
4.1.Test Durumları 18
4.2.Gereksinim Değişiklikleri 18
4.3.Projenin Gerçekleştirilmesi 18
4.4.Projenin Sonuçlandırılması 19
5.Projenin Katkıları 20
5.1.Ekip Üyelerine Katkıları 20
5.2.Bölüme Katkıları 21
5.3.Ülkemize Katkıları 21
6.Projeye Yönelik Gelecek Çalışmaları 21
KAYNAKLAR 21

1.Giriş

Bitcoin, herhangi bir yerden herhangi bir kişiye online ödeme yapmayı sağlayan merkezi olmayan bir para sistemi ve bu sistemin para birimidir. Bire bir elektronik nakit sistem olarak dizayn edilen açık kaynaklı bir yazılımdır. Bitcoin'in finansal politikalara tepki olarak doğması ilgi uyandırmış ve kısa sürede yayılmasına sebep olan olsa da merkezi bir yapıya sahip değildir. Özellikle bankaların her para transferinde veya alışverişlerde aldıkları komisyonlar ve işlem ücretleri Bitcoin'e geçişi hızlandırmıştır. Bitcoin kullanımındaki artış ve yatırım aracı olarak görülmeye başlaması ile merkezi olmayan bu sistemin çalışma prensibi ve geleneksel döviz piyasalarına etkileri çalışmanın amacını oluşturmaktadır. Bunun için 2009-2015 yılları arasındaki Bitcoin günlük kur fiyatları ile dünyadaki en çok kullanılan çapraz kur fiyatları arasındaki etkileşim, Granger nedensellik analizi ile test edilmiştir. Bitcoin, yeniçağın sanal para birimi. 10 yıl kadar önce Satoshi Nakamato tarafından ortaya çıkarılan; ancak o günlerde değeri pek anlaşılmayan Bitcoin, şu an hemen hemen tüm dünyadan kabul gören bir sanal para birimi olma özelliği taşıyor. Bitcoin'nin hem sahibinin belirsizliği hem de finans politikalarına tepki olarak doğması ilgi uyandırmış ve kısa sürede yayılmasına sebep olmuştur. Bitcoin ile sunulan finansal özgürlük anlayışı, dünyada çok hızlı bir şekilde kabul görmeye başlaması değerini gün geçtikçe yükselmeye başlamıştır. Biz de projemizde bitcoin değerinin artıp artmama durumunun analizini apache flink gerçekleştirdi.

2.Projenin Gerçekleştirilmesi

2.1.Proje Özet

Bu projede Bitcoin değerinin artıp artmama durum analizini tahmin etmek için makine öğrenme algoritmalarını apache flink ile uyguladık. Araştırmamızın ilk aşaması apache flinkin ne olduğu, kullanım alanları, makine öğrenmesi algoritmaları için kullanıldığı kütüphaneler vs. Bunlara dair çalışmlar yapıldı.

Araştırmamızın ikinci aşamasında, bitcoin kavramının ne olduğu, nerelerde kullanıldığı, bizim amacımız olan durum analizi için gerekli verisetleri vs. araştırmalar yapıldı. Araştırmalara göre elimizdeki verisetinde günlük açılış-kapanış değerleri ve buna karşılık çıktı olarak artıp artmadığını belirten etiketimiz bulunmaktadır. Bu sınıflandırma problemi olduğu içi apache flinkin flinkml kütüphanesini bu tip problemlerde desteklediği algoritmalardan SVM algoritması üzerinde gerçekleştirme yapıldı.

2.2.Proje Hedefi

Bitcoin değerinin artıp artmayacağının tahminini yapmaktır. Bu tahmini de apache flink ile gerçekleştirmektir.

2.3.Projede Mevcut Aşama

Bitcoin ve euro, dolar vb. para birimleriyle ilgili güvenilir veri setleri araştırılıp analiz edilme aşamasındadır. Projeyle alakalı makaleler incelenerek literatür taraması yapılmaktadır.Bitcoin ve euro, dolar vb. para birimleriyle ilgili güvenilir veri setleri araştırılıp analiz edilme aşamasındadır. Projeyle alakalı makaleler incelenerek literatür taraması yapılmaktadır.

Veri Seçimi ve Önişleme

Verilerin kalitesi, miktarı, hazırlanması ve seçimi, bir makine öğrenme çözümünün başarısı için kritik önem taşır. Başarı sağlamak için atılacak ilk adım, seçim yanlılığından kaçınmaktır. Seçim eğilimi, modeli üretmek için kullanılan örneklerin, özellikle de yeni ve görünmeyen verilerle modelin gelecekte kullanılabileceği durumları tam olarak temsil etmediği durumlarda ortaya çıkar.

Veri Çözümleme

Veri setleri genellikle dağınıktır. Bu setlerde eksik değerler (ör. NA), aykırı değerler ve benzeri öğelerden oluşur. Modelleme ve analiz öncesinde ham verilerin kullanılabilir hale getirilmesi gerekmektedir. Bunun için veri setinin ayrıştırılması, temizlenmesi, dönüştürülmesi ve ön işleme tabi tutulması gerekir. Bu süreç çoğunlukla veri karıştırma veya veri çözme olarak adlandırılır. Eksik olan veriler için, genellikle eksik değerlerin doldurulması veya değiştirilmesi/atılması için kullanılan bir tekniktir ve enterpolasyona kavramsal olarak çok benzerdir. Verilerin organize edilerek, makine öğrenmesi algoritmalarında kullanılabilir hale getirilmesi için R bazı paketler önermektedir.

Ölçekleme (scaling)

Buna ek olarak, bazen veri setleri dengesiz değer ve metrikler de içerebilir. Böyle durumlarda veri setlerinin ölçeklemek (scaling) gerekecektir. Genelde iki yaklaşım vardır: Özellik ölçekleme) ve Standart (normalize edilmiş) forma dönüştürme. Özellik ölçeklendirme, belirli özelliklerin modellere ve tahminlere hâkim olmasını önlemek için farklı özelliklerin değer aralıklarını benzerlik içine getirerek uygulanır. Aynı zamanda makine öğrenme optimizasyon algoritmalarını (hız, yakınsama, vb.) çalıştırırken hesaplama problemlerini önlemek için kullanılır.

2.3.1.Tasarım Kararları

2.3.1.1.Yazılım Modüller

2.3.1.1.1.Kullanılan Teknolojiler

Projeyi oluşturacak alt parçalar aşağıda belirtilmiştir.

- Virtualbox
- Linux Mint
- Apache Flink
- Bitcoin ve para birimi verisetleri

Kaynak kod Java programlama diliyle yazılacak bir uygulamadır. Kodlamada Eclipse kullanılacak olup Linux Mint işletim sistemi üzerinde Apache Flink kullanılarak proje yapılacaktır.

Eclipse: Projenin mobil uygulama kısmının hazırlanmasında kullanılacak Android IDE'sidir. **Apache Flink:** Apache Flink, veri akışları üzerinden dağıtılmış hesaplamalar için iletişim, hata toleransı ve veri dağıtımı sağlayan bir veri akış motoru olan açık kaynaklı bir platformdur. Ayrıca Flink, Hadoop ile tamamen uyumlu ölçeklenebilir bir veri analizi çerçevesi ve çok yüksek hızda üretilen verileri işleyebilen büyük ölçekli veri işleme çerçevesidir. Apache Flink, yeni nesil Büyük Veri aracıdır. Flink, Apache' nin üst düzey bir projesidir. Flink hem akış işlemeyi hem de toplu işlemeyi kolayca çalıştırabilir. Apache Flink'in pipeline mimarisi, akışlı verilerin mikro toplu mimarilere (Spark) göre daha düşük gecikme ile daha hızlı işlenmesini sağlar.

Apache Flink, aşağıdaki gereksinim türlerini etkin bir şekilde çözebilen güçlü açık kaynaklı bir platformdur:

- Toplu İşleme
- İnteraktif işlem
- Gerçek zamanlı akış işlemi
- Grafik İşleme
- Yinelemeli İşleme
- Bellek içi işleme

Flink, MapReduce'un alternatifi olup, verileri MapReduce'dan 100 kat daha hızlı işler. Flink kendi veri depolama sistemini sağlamıyor. Dağıtılmış depolama alanından veri alır. Sorgu optimizasyonunu, veritabanı sistemlerinden gelen kavramları ve verimli paralel bellek içi ve çekirdek dışı algoritmaları MapReduce çerçevesi ilebütünleştirerek gerçekleştirilir. Apache

Flink esas olarak akış modeli üzerine kurulu olduğundan, akışlı mimari kullanarak verileri yineler.

Linux Mint: Ubuntu ve Debian temel alınarak geliştirilen, masaüstü kullanıcılarına yönelik bir Linux dağıtımıdır. Ubuntu ve Debian işletim sistemlerinin çekirdek (Kernel) mimarisi üzerine temellendirilmiş topluluk tabanlı ve açık kaynaklı ücretsiz bir Linux dağıtımıdır. İlk defa 2006 yılında yayınlanan Linux Mint kısa sürede büyük bir topluluğun dikkatini kendi üzerine çekmeyi başarmış ve en çok kullanılan Linux dağıtımlarından biri haline gelmiştir. **MS Word:** Projenin dokümanlarını yazmak için kullanılacak olan programdır.

Uygulama yaparken kullanılacak olan araç ve teknikler aşağıdaki kriterler doğrultusunda kararlaştırıldı;

- Görünüm ve etkileşim
- Anlaşılabilirdik ve kullanılabilirlik
- İleri Geliştirme ve Bakım Süreçleri
- Kullanacağımız platform, araçların eklenti desteği
- Kişisel bilgisayarımızın destekleyip desteklemediği

2.3.1.2.Projeye Dair İstisnai Durumların İdaresi

Bitcoinin fiyatının veya artışının bir sınırının olmaması, etkili olan siyasi kararlar vs. gibi durumların ortaya çıkması proje için istisnai durumdur.

2.3.2.Test

2.3.2.1.Test Edilen Özellikler

Test tasarım özellikleri numarası	Tanım
TST_1	Veri setlerinin doğruluğu
TST_2	Makine öğrenemsi algoritmlarının kodları
TST_3	Kodların çalıştırılacağı derleyicinin güvenirliği
TST_4	Doğru tahmin edilip edilmediği
TST_5	Yüksek başarı elde edildi mi?

Tüm aşamalar, nesneler ve fonksiyon kodları bir birim olarak ayrı ayrı teste tabi tutulacaktır. Her birim derleyicinin bulduğu hataları düzeltip raporlayacaktır.

2.3.2.1.1. Performans Testi

Bu projede aynı anda birden fazla kullanıcı sisteme giriş yapabileceği için sunucu kapasitesi ve cevap verme süresinin optimum düzeyde olması proje için çok önemlidir. Çok kullanıcılı girişler simule edilerek projenin performansı test edilecektir.

2.3.2.1.2. Güvenlik Testi

Bu projede her kayıtlı kullanıcının kendine ait bir kullanıcı adı ve şifresi olacaktır. Bu sayede kullanıcılara ait kişisel bilgilere yetkisi olmayan kişiler erişim sağlayamayacaktır. Ayrıca veritabanına dışarıdan yetkisiz girişleri engellemek için v
eritabanı yöneticisine ait kullanıcı ve şifre belirlenerek veritabanı güvenliği sağlanacaktır.

2.3.2.1.3.Başarı ve Başarısızlık Kriterleri

Projenin test aşaması sonucunda;

- Güvenlik açığı olmaması
- Algoritmaların doğru çalışması

Kriterlerinin sağlanması durumunda projenin başarılı olduğu kabul edilecektir. Kriterlerin sağlanmaması durumunda test başarısız sayılacak ve proje üzerinde tekrar gerekli hata düzeltmeleri yapılacaktır.

2.3.2.1.4.Test Ortamı ve Şekli

Test tanımları ve test araçları olgunlaştırılır. Testler gerçekleştirilmeye başlamadan önce ortamın testlere hazır olup olmadığının kontrolünün yapılması gereklidir. Hazırlanan test yazılımlarının testler öncesinde doğrulanmış olması önem taşımaktadır. Resmi testler öncesi test edilecek test ortamına entegrasyonunun gerçekleştirildiği ve yazılımın testlere hazır olup olmadığının kontrolü Apache Flink'in desteklediği makine öğrenmesi algoritmaları üzerine başarı yüzdesi ve tahminleri ölçen fonksiyonlar uygulanarak yapılır. Bu testler, test edilecek yazılımın en temel yeteneklerini doğrulayan test tanımları arasından seçilmelidir. Ancak bu testler başarılı olarak gerçekleştirilirse, resmi olarak testlere başlanmalıdır.

2.3.2.2.Test Sonuçları

Projenin test aşaması sonucunda;

- Güvenlik açığı olmaması
- Algoritmaların doğru çalışması

Kriterlerinin sağlanması durumunda projenin başarılı olduğu kabul edilecektir. Kriterlerin sağlanmaması durumunda test başarısız sayılacak ve proje üzerinde tekrar gerekli hata düzeltmeleri yapılacaktır.

Verinin Önemi: Veri madenciliğinde amaç çok büyük miktardaki ham veriden değerli bilginin çıkarılmasıdır. Çok miktarda güvenilir (hata ve eksiklerin olmadığı) veri önşarttır çünkü

çözümün, yani çıkarılan kuralların kalitesi öncelikle verinin kalitesine bağılıdır. Veri madenciliği simya değildir; taşı altına çeviremeyiz.

Uzmanın Önemi: Veri madenciliği çalışması bilgisayarcıların ve uygulama konusundaki uzmanların ortak çalışmasıdır. Her ne kadar olabildiğince otomatik olmasını istesek de uzmanların yardımı ve desteği olmadan başarılı olmak söz konusu değildir. Uzmanlar amacı tanımlar. Uygulama ile ilgili sonuca yararlı olabilecek her tür bilginin sisteme verilmesi

gerekir ve bunları da ancak uzmanlar bilir. Ayrıca çalışma ile alınan sonuçların yorumlanması ve geçerlemesi uzmanlar tarafından yapılır.

Sabrın Önemi: Veri madenciliği tek aşamalı bir çalışma değildir, tekrarlıdır. Sistem ayarlanana dek birçok deneme gerektirebilir. Çalışma uzun olabilir. Buna çalışan ekibin ve yönetimin hazırlıklı olması, kısa vadede çok büyük beklentilere sahip olunmaması gerekir. Bir veri madenciliği çalışması için öncelikle çok miktarda kaliteli veri gerekir. Amaç bu veri içinde saklı, gelecekle ilgili tahmin yapmakta kullanılabilecek kural ve bağıntıların çıkarılmasıdır. Böyle bir çalışmanın başarılı olması için uygulama konusundaki uzmanların veri tabanları ve veri madenciliği konusundaki uzmanlarla beraber çalışması gerekir. Çalışma uzun sürebilir; zaman ve sabır gerekir.

2.3.2.3.Teste Yönelik Belirtilmemiş Değişiklikler

Knn algoritması çalıştırılmaya çalışılıyor. Fakat hata verdiği için başarıyı ölçemiyoruz.

3. Apache Flink

Apache Flink, veri akışları üzerinden dağıtılmış hesaplamalar için iletişim, hata toleransı ve veri dağıtımı sağlayan bir veri akış motoru olan açık kaynaklı bir platformdur.

Ayrıca Flink, Hadoop ile tamamen uyumlu ölçeklenebilir bir veri analizi çerçevesi ve çok yüksek hızda üretilen verileri işleyebilen büyük ölçekli veri işleme çerçevesidir. Apache Flink, yeni nesil Büyük Veri aracıdır. Flink, Apache' nin üst düzey bir projesidir. Flink hem akış işlemeyi hem de toplu işlemeyi kolayca çalıştırabilir. Apache Flink'in pipeline mimarisi, akışlı verilerin mikro toplu mimarilere (Spark) göre daha düşük gecikme ile daha hızlı işlenmesini sağlar.

Apache Flink, aşağıdaki gereksinim türlerini etkin bir şekilde çözebilen güçlü açık kaynaklı bir platformdur:

- Toplu İşleme
- İnteraktif işlem
- Gerçek zamanlı akış işlemi

- Grafik İşleme
- Yinelemeli İşleme
- Bellek içi işleme

Flink, MapReduce'un alternatifi olup, verileri MapReduce'dan 100 kat daha hızlı işler. Flink kendi veri depolama sistemini sağlamıyor. Dağıtılmış depolama alanından veri alır. Sorgu optimizasyonunu, veritabanı sistemlerinden gelen kavramları ve verimli paralel bellek içi ve çekirdek dışı algoritmaları MapReduce çerçevesi ile bütünleştirerek gerçekleştirilir. Apache Flink esas olarak akış modeli üzerine kurulu olduğundan, akışlı mimari kullanarak verileri yineler [10].

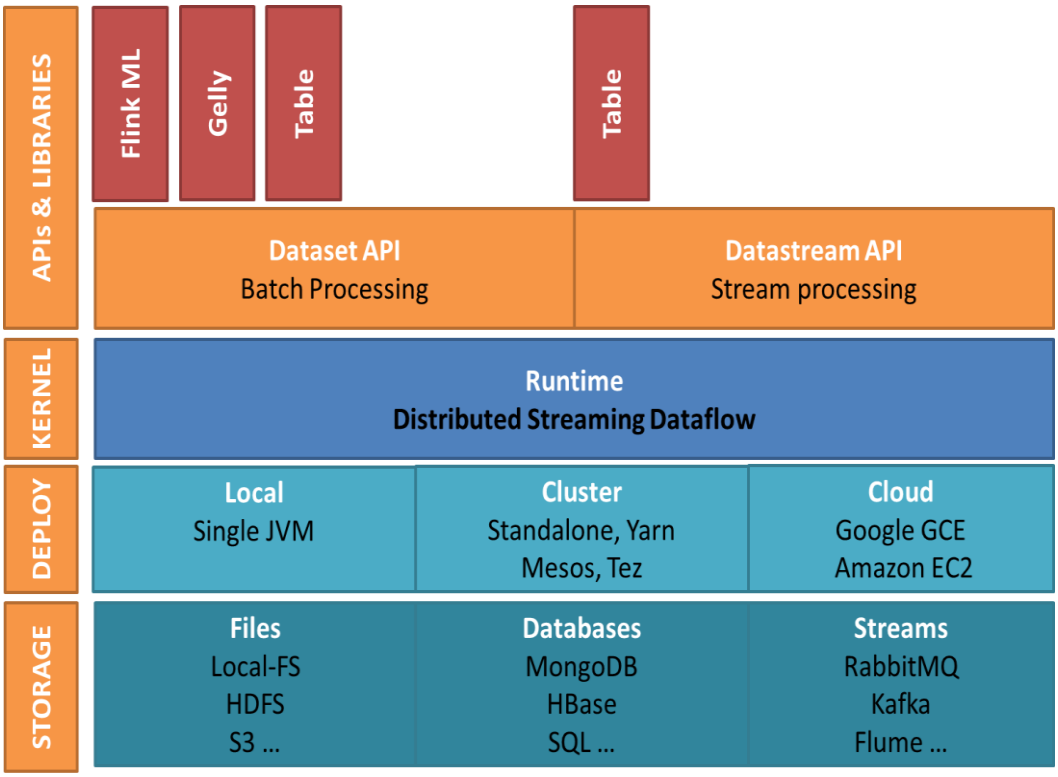
3.1. Apache Flink Tarihçesi

Apache Flink, 2008'de Volker Markl, The Stratosphere adlı proje fikrinin oluşturmasıyla başlatıldı. Nisan 2014'te Apache'de kuluçkaya bırakılmış ve Aralık 2014'te en üst düzey bir proje haline gelmiştir. Flink, hızlı / çevik anlamına gelen Almanca bir kelimedir. Flink logosu şekil 2.1'de görüldüğü gibi Hadoop ekosistemiyle uyumlu bir sincaptır.[10]

Şekil 2.1. Apache Flink logosu

3.2. Ekosistemi

Şekil 2.2' deki diyagram Apache Flink'in ekosistemini göstermektedir[11].



Şekil 2.2. Apache Flink ekosistemi

3.2.1. Depolama

Flink, yalnızca herhangi bir depolama sistemi olmayan bir hesaplama motorudur. Farklı depolama sistemlerinden veri okur ve yazar, aynı zamanda akışlı sistemlerdeki verileri tüketebilir. Flink'in yazma verilerini okuyabileceği depolama sistemlerinden bazıları aşağıda belirtilmiştir[11]:

- HDFS - Hadoop Dağıtılmış Dosya Sistemi
- Local-FS - Yerel Dosya Sistemi
- HBase - Hadoop ekosisteminde NoSQL Veritabanı
- MongoDB - NoSQL Veritabanı
- RDBMS - Herhangi bir ilişkisel veritabanı
- S3 - Amazon'dan Basit Depolama Hizmeti
- Kafka - Dağıtık Mesajlaşma Sırası
- Flume - Veri Toplama ve Birleştirme Aracı
- RabbitMQ - Mesajlaşma Sırası

3.2.2. Dağıtım

Dağıtım / kaynak yönetimi, Flink ekosisteminin ikinci katmanıdır. Flink'i şu modlarda dağıtabilirsiniz:

- Yerel mod - Tekli düğümde, tek JVM'de
- Küme - Çok düğümlü kümede, aşağıdaki kaynak yöneticisi ile
- Standalone - Flink'in varsayılan kaynak yöneticisi.
- YARN - Bu Hadoop'un bir parçası, Hadoop 2.x'te tanıtılan, çok popüler kaynak yöneticisidir
- Mesos - Bu genelleştirilmiş bir kaynak yöneticisidir
- Bulut - Amazon veya Google bulutunda[11].

3.2.3. Flink çekirdeği

Üçüncü katman ise Çalışma Zamanı - Dağıtılmış Akışlı Veri Akışı, ki bu da Apache Flink'in çekirdeği olarak da bilinir. Bu katman, dağıtılmış işlem, hata toleransı, güvenilirlik, yerel yinelemeli işleme yeteneği vb. Sağlar[11].

3.2.4. API'ler ve Kütüphane

Bu Flink API'leri ve Kütüphaneleri aşağıda listelenen Flink için çeşitli yetenekler sunar:

3.2.4.1. DataSet API'sı

Bu API, çeşitli kaynaklardan üretilen, örneğin metin veya CSV dosyaları okumak veya yerel koleksiyonlardan elde edilen verileri işler ve kullanıcıların haritalama, filtreleme, katılma, gruplandırma gibi dönüşümleri uygulamasına izin verir. Metin ya da CSV dosyalarına veri yazmak ya da çıktıyı istemciye geri döndürmek için batar. Esas olarak dağıtılmış işlemler içindir. Ayrıca, akışlı çalışma zamanında toplu uygulamayı yürütür.

3.2.4.2. DataStream API'sı

Dönüşümleri, verilerin sürekli bir akışında işler. Canlı veri akışını işlemek için, filtreleme, durumları güncelleme, pencereleri tanımlama, bir araya getirme, vb. Gibi çeşitli işlemleri sağlar. Bu, mesaj kuyrukları, soket akışları, dosyalar vb. Gibi çeşitli akışlı kaynaklardan gelen verileri tüketebilir ve veriyi yazma için Lavabolar yoluyla veriye geri gönderebilir veri dosyaları veya komut satırı terminali gibi standart çıktı. Java ve Scala'yı desteklemektedir.

3.2.4.3. Tablo API'sı

Tablo API, kullanıcıların ilişkisel akış ve toplu işleme için SQL gibi diller üzerinden geçici analiz yapmalarını sağlar. Bu gömülebilir DataSet ve DataStream API'ler hem Flink arasında Java ve Scala . Aslında, kullanıcıların Flink'in üst kısmında, veri işleme için karmaşık kod yazmalarından kurtaran SQL sorguları çalıştırmasına olanak tanır.

Tablo API'sinin temel kavramı, ilişkisel şema içeren bir tabloyu temsil eden bir tablodur. DataSet veya DataStream'i kullanarak Tablolar oluşturabilirsiniz. Bir DataSet veya DataStream'e dönüştürür veya bir TableEnvironment kullanarak bir tablo kütüğüne kaydeder. Kayıtlı tablolar, düzenli SQL sorguları ile sorgulanabilir.

Tablo her zaman belirli bir TableEnvironment'a bağlıdır. Farklı TableEnvironments tablolarını birleştirmek mümkün değildir.

Farklı Tablo API operatörleri seçilir, filter / where, groupBy, join, LeftOuterJoin, RightOuterJoin, Union, Intertersect vb.[11]

3.2.4.4. Gelly

Kullanıcıların grafik oluşturmak, dönüştürmek ve işlemek için işlemler yürütmesine olanak tanıyan grafik işleme motorudur. Ayrıca, grafik uygulamalarının geliştirilmesini basitleştirmek için algoritma kütüphanesini de sağlar. Gelly, grafiği toplu işlem API'leri tarafından sağlananlara benzer üst düzey işlevler kullanarak değiştirebilir ve değiştirebilir. API'leri Java ve Scala'da mevcuttur. Scala yöntemleri, Java işlemlerinin üstünde sarıcılar olarak uygulanmaktadır.

Noktalar ve kenarlardaki bir veri kümesi, Gelly'deki bir grafiği temsil eder. DataSet'in benzersiz kimliği bir köşe ve bir değer tanımlarken kaynak kimliği bir kenarı, hedef kimliğini ve değeri tanımlar. Dönüşümler ve Yardımcı Programlar, Dönüşümler, grafik metrikleri, mutasyonlar ve mahalle toplamaları içeren Graph sınıfının yöntemleridir.

Tek bir sistem üzerinde Gelly aracılığıyla uçtan uca veri analizi gerçekleştirmek mümkündür. Gelly API, hem kayıt temelli hem de grafik tabanlı analiz kullanan uygulamaların uygulanması için DataSet Flink API'si ile sorunsuz bir şekilde karıştırılabilir[10].

3.2.4.5. FlinkML - Flink için Makine Öğrenimi

Scala'daki makine öğrenme kitaplığı, sezgisel API'ler ve Apache Flink'teki makine öğrenme uygulamalarını idare edebilecek etkili bir algoritma sağlar. Bildiğimiz gibi makine öğrenme algoritmaları doğasında iteratif olmaktadır, Flink, yinelemeli algoritmanın aynısını oldukça etkin ve verimli bir şekilde ele alması için doğal destek sağlamaktadır.

FlinkML, Veri Önişleme ile birlikte Denetimli öğrenmeyi (Optimizasyon çerçevesi, Çoklu doğrusal regresyon ve SVM algoritmaları) ve Denetimsiz öğrenmeyi (k-Nearest Neighbors katılmaktadır) destekler.

FlinkML'in temel kavramlarından biri, karmaşık veri analizi pipelinelarının oluşturulmasına olanak tanıyan scikit-learn esin kaynağı pipeline mekanizmasıdır[10].

3.2.4.6. FlinkCEP - Flink için kompleks olay işleme

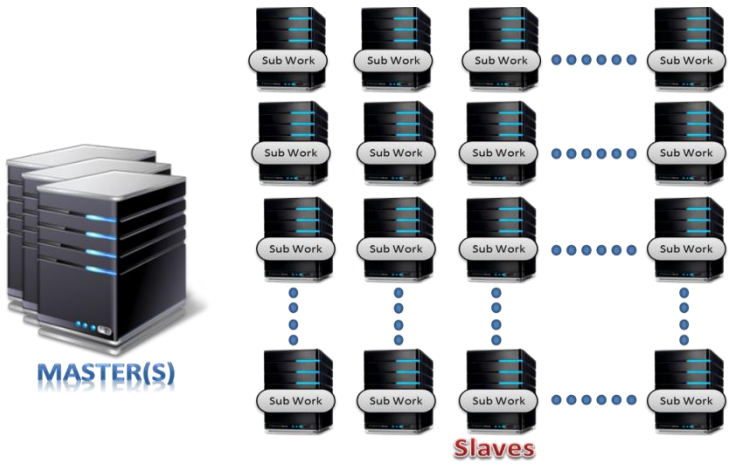
FlinkCEP, veri akışı verilerinde karmaşık olay modellerinin kolaylıkla algılanmasına olanak tanır; bu, verilerin analizini elde etmek için eşleşen dizilerin bulunmasında yararlı olur. Eşleşen olayları karşılaştırmak ve bulmak için, veri akışının equals () ve hashcode () yöntemlerini uygulamak gereklidir.

Desen API'sı karmaşık olay modellerini kolayca tanımlar. Desenler, kullanıcıların olaylar için koşulları tanımlayabileceği farklı durumlara sahiptir. Çeşitli desen işlemleri, Java, Scala'da bulunabilen begin, next, followingBy, vb. Içerir.

Günümüzde, borsa trendleri ve kredi kartı sahtekârlığı algılama gibi finansal uygulamalar CEP'den faydalanıyor. Ayrıca, RFID tabanlı izleme ve izlemede de yararlıdır. Örneğin, öğelerin düzgün bir şekilde kontrol edilmediği bir depoda hırsızlığı tespit eder. Ayrıca, şüpheli kullanıcı davranışı modelleri belirleyerek ağ saldırılarının tespitinde kullanımını bulabilir[10].

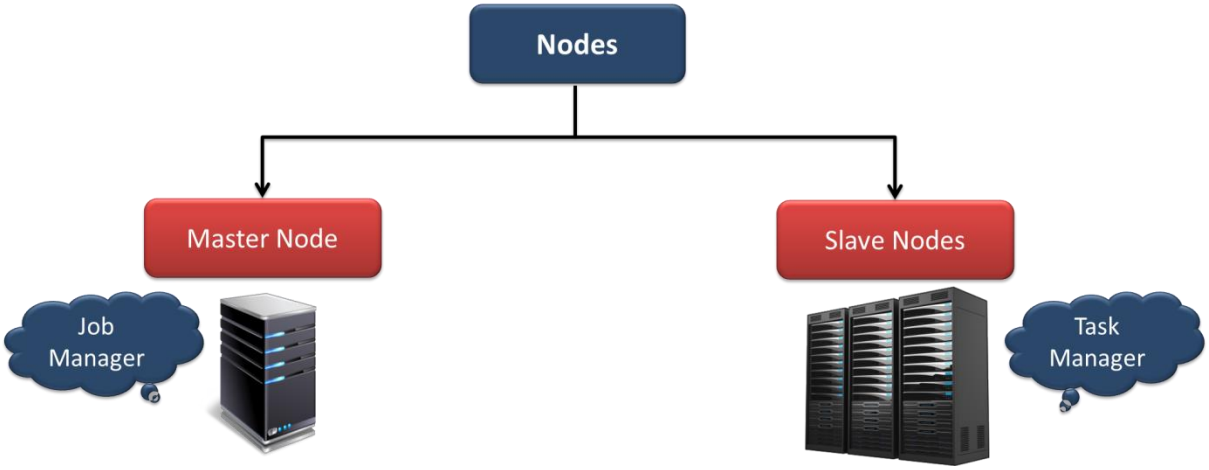
3.3. Mimarisi

Flink, şekil 2.3' deki gibi master-slaves yani usta-köle tarzında çalışır. Master, kölelerin işçi düğümleri olduğu kümenin yönetici düğümüdür. Şekilde gösterildiği gibi master, kullanıcının iş ya da uygulamayı gönderebileceği kümenin merkezidir. Ardından master işi bölecek ve kümedeki slavelere teslim edecektir. Bu şekilde, Flink veriyi yıldırım hızlı bir hızda işleyebilmesini sağlayan dağıtılmış bilgi işlem gücüne sahip olur[9].



Şekil 2.3. Master-slave yapısı

İki çeşit düğüm vardır: master yani ana ve slave yani köle düğüm vardır. Ana düğümde, "Job Manager" adı verilen Flink ana sunucusu çalışır ve tüm köle düğümlerde Flink'in "Task Manager" adı verilen köle yapılandırılır[9].



Şekil 2.4. Flink node yapısı

3.4. Özellikleri

Aşağıda Apache Flink' e ait özellikler sıralandırılımıştır.

- Flink gerçek bir akış işleme motorudur.
- Flink'in veri akışlı çalışma zamanı çok yüksek verim sağlar.
- Flink, herhangi bir gecikme olmaksızın veriyi saniyenin altındaki aralıkta işleyebilir.
- Flink, olayların gecikmeli olarak veya arızalı olarak ulaştığı akış işleme ve pencerelemeyi destekler.
- Flink, veriyi yıldırım hızında işler (dolayısıyla 4G Büyük Veri olarak da bilinir).
- Donanım, düğüm, yazılım veya bir işlemin başarısızlığı kümeyi etkilemez.
- Flink yönetilen hafızada çalışır ve asla bellek yetersizliği yaşanmaz.
- Flink, verilerini işlemek için çeşitli depolama sistemi ile entegre edilebilir; çeşitli kaynak yönetimi araçlarıyla dağıtılabilir.
- Flink, bir programın yürütülmesinden önce optimize ediciyle birlikte gönderilir; optimize edilmiştir.
- Flink son derece ölçeklenebilir. Artan gereksinimlerle, flink kümesini ölçeklendirebiliriz.
- Flink, verileri işlemek için önceden tanımlanmış birçok operatör içerir. Tüm ortak işlemler bu operatörler kullanılarak yapılabilir.
- Hesaplama sırasında özel durumunu koruyabilir.
- Toplu işlem ve veri akışının her ikisinde de ortak çalışma zamanı vardır.
- Flink'in API'leri, tüm ortak işlemleri kapsayacak şekilde geliştirildiğinden, programcılar verimli bir şekilde kullanabilirler.

3.5.Apache Flink’in Kullanildiği Yerler

3.5.1.Bouygues Telecom - Fransa'da üçüncü büyük mobil sağlayıcısı.

Bouygues Group, Fortune'un "Global 500" te yer alıyor. Bouygues, Flink'i gerçek zamanlı olay işleme ve günün 24 saati çalışan bir sistemde günde milyarlarca mesaj analitiği için kullanıyor. Bouygues Apache Flink'i API ve gerçek zamanlı akışta desteklediği için seçti, böylece şirketin aradığı düşük gecikme süresini sağladı. Buna ek olarak, sistemleri, sistemdeki iş mantığını genişletmelerinde onlara yardımcı olan diğer çözümlerle karşılaştırıldığında Flink ile en kısa sürede çalışıyordu. Bouygues, müşteri deneyimi, ağ üzerinde genel olarak neler olduğunu ve ağ gelişimi ve operasyonları açısından neler olduğunu gerçek zamanlı anlayışlar elde etmek istedi. Bunun için ekibi, kullanıcı deneyiminin kalitesine ilişkin göstergeleri belirlemek için ağ donanım günlüklerini analiz edecek bir sistem kurdu. Sistem günde 2 milyar olayı (saniyede 500.000 olay) ve 200 milisaniyeden daha kısa bir gecikmeye (nakil katmanı tarafından mesaj yayını ve Flink'teki veri işleme de dahil olmak üzere) ihtiyaç duyar. Bu, her biri 1 gigabayt bellekte sadece 10 düğüm olduğu bildirilen küçük bir kümede elde edildi. Planlama, Flink'in akış işlemesini veriyi dönüştürmek ve zenginleştirmek ve türetilen akış verilerini, birden çok tüketicinin analitik için mesaj aktarma sistemine geri itmek için kullanmaktı. Bu yaklaşım açıkça seçildi. Flink'in akış işleme kapasitesi, gecikme gereksinimini karşılarken, yüksek güvenilirlik, yüksek erişilebilirlik ve kullanım kolaylığıyla birlikte, veri işleme ve hareket hattını tamamlamak için Bouygues ekibine izin verdi. Örneğin, Apache Flink çerçevesi hata ayıklama için idealdir ve yerel yürütmeye geçiş yapılabilir. Flink, programların nasıl çalıştıklarını anlamanıza yardımcı olmak için program görüntülemeyi de desteklemektedir. Ayrıca, Flink API'leri hem geliştiricilere hem de veri bilimcilerine caziptir.

3.5.2.King - Candy Crush Saga'nın yaratıcısı

200'den fazla ülkede ve bölgede oynanan, önde gelen çevrimiçi eğlence şirketi King, 200'den fazla oyun geliştirmiştir. 300 milyondan fazla aylık kullanıcı her gün farklı oyunlardan ve sistemlerden 30 milyondan fazla etkinlik ürettiğinde, herhangi bir akış analizi kullanımı vakası gerçek bir teknik zorluk haline gelir. Maksimum esneklik muhafaza ederken veri analizlerini kullanarak bu muazzam veri akışlarını işlemek, Apache Flink tarafından üstesinden gelinen büyük bir zorluktu. Flink, King'deki veri bilimcilerinin bu muazzam veri akışlarına gerçek zamanda erişmesini sağlar. Böyle kompleks bir oyun uygulamasıyla bile Flink kutudan çıkan çözümü sağlayabilir.

3.5.3.Zalando - Avrupa'da E-Ticaret Şirketinin Liderliği

Zalando'nun dünya çapında 16 milyondan fazla müşterisi vardır ve gerçek zamanlı süreç izlemesi için Apache Flink kullanmaktadır. Akış tabanlı bir mimari, Zolando tarafından kullanılan mikro hizmet yaklaşımını güzelce desteklemektedir ve Flink, iş süreci izleme ve Sürekli Ayıkla, Dönüştür ve Yükle (ETL)

3.5.4.Otto Group - Dünyanın en büyük ikinci online perakendecisi

Otto Group BI Department, açık kaynak seçeneklerinden hiçbirinin gereksinimlerini karşılamadığı için, büyük verilerini işlemek için kendi akış motorunu geliştirmeyi planlıyordu. Flink'ü test ettikten sonra departman, kullanıcı aracısı kimlik belgesine sahip olmaya ve akış işlemi yoluyla bir arama oturumunun tanımlanmasına uygun buldu.

3.5.5.ResearchGate - En büyük akademik sosyal ağ

ResearchGate, hem toplu hem de akış işleme için veri altyapısında birincil araçlarından biri olarak 2014'ten beri Flink kullanıyor. Flink'i ağ analizleri için kullanıyor ve üyelerine kusursuz bir deneyim sağlamak için yinelenen algılama yakınıyor.

3.5.6.Alibaba Group - Dünyanın en büyük perakendecisi

Alibaba, web portalını kullanarak alıcılar ve tedarikçilerle çalışır. Flink'in varyasyonu (Blink olarak anılmaktadır) şirket tarafından çevrimiçi öneriler için kullanılmaktadır. Apache Flink, kullanıcılara ürün önerirken gündüzleri yapılan satın almaları göz önüne almaya olanak tanır. Bu, etkinlik olağandışı yüksek olduğunda özel günlerde (tatiller) önemli bir rol oynar. Bu, verimli akış işlemesinin toplu işleme üzerinde oynadığı bir örnek.

3.5.7.Capital One - Fortune 500 finansal hizmetler şirketi

Önde gelen bir tüketici ve ticari bankacılık kuruluşu olan şirket, müşteri etkinliği verilerini gerçek zamanlı olarak izlemek konusunda zorlandı. Bunu müşterinin saptamasını ve çözmesini istiyorlardı.

E-ticaret arama sonuçlarının gerçek zamanlı olarak optimize edilmesi: Alibaba'nın arama altyapısı ekibi, gerçek zamanlı olarak ürün ayrıntılarını ve envanter bilgilerini güncellemek için Flink'i kullanır ve kullanıcılar için alaka düzeyi artırır.

Veri bilim ekipleri için akış olarak işleme hizmeti: King (Candy Crush Saga'nın yaratıcıları), gerçek zamanlı analizleri, Flink ile çalışan bir dahili platform aracılığıyla veri bilimcileri tarafından kullanılabilir hale getirir ve böylece oyun verilerinden edindiğiniz bilgilere vakit ayırır.

Ağ / sensör izleme ve hata algılama: Fransa'nın en büyük telekom sağlayıcılarından biri olan Bouygues Telecom Flink'i kablolu ve kablosuz ağlarını izlemek için kullanıyor ve ülke genelindeki kesintilere hızlı yanıt vermeyi mümkün kılıyor.

İş zekası altyapısı için ETL: Zalando, Flink'i kullanarak verileri daha kolay yükleme için veri ambarına dönüştürür; karmaşık yükleri nispeten basit olanlara dönüştürür ve analitik son kullanıcıların veriye daha hızlı erişmesini sağlar.

4.Durum Değerlendirmesi

4.1.Test Durumları

Çalıştırdığımız SVM algortimasına göre test sonucumuz %51 başarı elde etmiştir. Bu sonucumuz gösteriyorki algoritma yeterince öğrenme gerçekleştirememiştir. Bunun nedeni olarak featureler arsasındaki korelasyonun yüksekliğinden olduğu düşünülmektedir.

4.2.Gereksinim Değişiklikleri

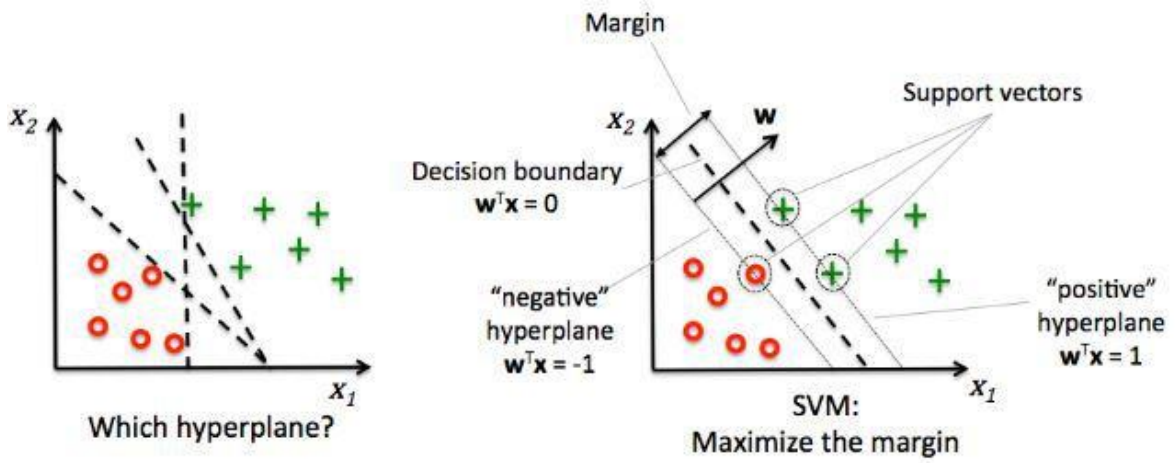
Algoritmanın daha iyi başarı elde etmesini sağlamak için veri ön işleme aşaması üzerinde featureler üzerinde gerekli değişiklik yapılabilir.

4.3.Projenin Gerçekleştirilmesi

4.3.1. SVM Algoritması

SVM çok güçlü ve çok yönlü olan makine öğrenme modelidir. Doğrusal ya da doğrusal olmayan sınıflandırma, regresyon ve hatta aykırı değer tespitini gerçekleştirmek için kullanılır. Aşağıdaki figur Irıs veri setinden alınmış 2 sınıfa aittir. Soldaki grafikte , olası 3 doğrusal karar sınırını göstermektedir. Kesikli çizgi ile temsil edilen modeli öylesine kötü ayırmıştırki sınıfları düzgün şekilde ayrılamaz. Diğer iki model bu eğitim setinde mükemmel bir şekilde çalışır. Ancak karar sınırları bu modellerin yeni örneklerde de muhtemelen çok iyi çalışmamasına sebep olabilir. İşte bu noktada SVM imdadamıza yetişiyor.

Sağdaki grafikte SVM modeli tarafından oluşturulmuş karar sınırı iki sınıfı birbirinden ayırmakla kalmayıp birbirine en yakın 2 sınıf üyesinden, mümkün olan en uzak nokta üzerinden geçmektedir [13].



Şekil 4.1. Support Vectors Machine Algoritması Grafiği

4.4.Projenin Sonuçlandırılması

Bu lisans tezinde Apache Flink üzerinde makine öğrenmesi algoritmalarıyla bicoin durum tahmini yapılıyor. Uygulanan SVM ve Knn algoritmalarının başarıları karşılaştırıldı. Bu makine öğrenmesi algoritmalarının performansını incelemek için uyguladığımız değerlendirme bitcoin verilerinin verimliliği, çeşitli paralel akışlı ML algoritmalarının uygulanması için iyi bir başlangıç olabileceğini kanıtlamaktadır.

Verisetimizde o güne ait açılış ve kapanış değerleri arasındaki fark değeri baz alınarak artıp azaldığı belirtilmiştir. Bu veri seti şekil 4.2.'de görülmektedir.

Date	Open	Close	State
2017-10-19	5590.0	5718.5	artmamis
2017-10-18	5606.5	5579.4	artmis
2017-10-17	5738.7	5606.5	artmis
2017-10-16	5673.2	5738.8	artmamis
2017-10-15	5800.0	5680.0	artmis
2017-10-14	5626.7	5800.0	artmamis
2017-10-13	5429.8	5640.0	artmamis
2017-10-12	4824.2	5429.7	artmamis
2017-10-11	4745.4	4822.6	artmamis
2017-10-10	4774.9	4748.5	artmis
2017-10-09	4594.4	4772.1	artmamis
2017-10-08	4423.2	4592.0	artmamis
2017-10-07	4363.1	4432.0	artmamis
2017-10-06	4319.6	4363.1	artmamis

Şekil 4.2. Ön İşlemden Geçirilmiş Bitcoin Veriseti

Başarı sonucuna göre SVM algoritmamızın eğitiminin yetersiz olduğu görülmektedir. Yani öğrenme tam anlamıyla gerçekleştirilememiştir. Verdiğimiz eğitim seti üzerinde uyguladığımız

SVM algoritması feature olarak bitcoinin gün içindeki açılış ve kapanış değerlerini alıp etiket olarak da artıp artmadığını göstermektedir. Çıktıların bir kısmı şekil 4.2.' de görülmektedir.

```
(DenseVector(1.8103912536534611E-4, 1.8517916007014122E-4),1.0)
(DenseVector(1.8191055316493863E-4, 1.8103908592472304E-4),1.0)
(DenseVector(1.8459488193024742E-4, 1.81910513534469E-4),1.0)
(DenseVector(1.9606278898724405E-4, 1.8735693609538124E-4),1.0)
(DenseVector(1.8696483440257527E-4, 1.9603658383819856E-4),1.0)
(DenseVector(1.829976593449303E-4, 1.8677538016574117E-4),1.0)
(DenseVector(1.7985473516437742E-4, 1.8340704666865206E-4),1.0)
(DenseVector(1.7646412941748536E-4, 1.7972821262136538E-4),1.0)
(DenseVector(1.905530255101774E-4, 1.7646409097355674E-4),1.0)
```

Şekil 4.3. SVM Algoritması Tahminlerin Bir Kısmı

Test setimizde 292 adet veri bulunmaktadır. Bunlar arasında 150 tanesi başarılı tahmin yapmıştır. Başarı oranı hesaplandığında %51 olduğu görülmektedir. Bu da algoritmanın yeteri kadar eğitilmediğini göstermektedir. Bunun nedeni olarak da featureler arası fazla korelasyonun bulunması ve verilerin az olması gösterilebilir.

```
successRatio: Float = 0.51369864
basarılı tahmin 150
  toplam veri seti 292
basarı=0.51369864
```

Şekil 4.4. SVM Algoritması Başarılı Tahmin Oranı

5.Projenin Katkıları

Bu lisans tezi kapsamında bazı katkılar sağlandı. Artımlı ML algoritmalarının uygulanmasını önleyen mevcut sistemlerin ve programlama modellerinin literatür çalışması. Uygulanan prototiplerin ve algoritmaların, Apache Flink üzerinde uygulanan ölçeklenebilir makine öğrenmesi algoritmasına göre performansının değerlendirilmesi. Değerlendirme, çözümümüzün zaman içinde iyi ölçeklendiğini ve kabul edilebilir bir doğruluğu koruduğunu gösterdi.

5.1.Ekip Üyelerine Katkıları

Apache Flink teknolojisine, makine öğrenmesini, makine öğrenmesi algoritmalarını uygulamalarını, scala dilini, Apache Zeppelin teknolojisini de öğrenme imkânı bulundu. Grup çalışmasını dair tecrübe edinerek iş bölümü yapma imkânı bulundu.

5.2.Bölüme Katkıları

Bu projeyle bitcoine olan ilgi daha da artmıştır. Böylelikle bitcoinle ilgili daha geniş kapsamlı yeni projelerin oluşturulmasına zemin hazırlanmış oldu.

5.3.Ülkemize Katkıları

Uygulama, açık kaynak projesi olan Apache'ye bir katkı olacaktır. Böylece uygulamaya serbestçe erişilmiş olur. Bitcoin verileri üzerinde fiyat durum tahmini yapılmasıyla insanların bitcoine yaklaşımı değişir. Böylelikle daha risk almış olurlar. Ortalama tahmine göre hareket ederler.

6.Projeye Yönelik Gelecek Çalışmaları

Bu projede bitcoin değerinin artıp artmayacağına dair tahminlerde bulunulmuştur. Gelecek çalışmamızda Apache Flinkin temel amacı olan streaming data üzerinde çalışma yapılacakatır. Bitconin fiyat değerinin tahminin yapılmasına çalışılacaktır. Test sonuçlarının daha iyi sonuçlar vermesi sağlanacaktır. Ayrıyeten bu aşamaları gerçekleştirdikten sonra artık sadece bitcoin için değil diğer sanal paralar için bu tahminler yapılması sağlanabilir hale getirilmesi amaçlanmaktadır.

KAYNAKLAR

[1]İnternet: Veri Madenciliği, httpshttp://tugbaguder.blogspot.com.tr/2014/05/veri-madenciligi-ve-kullanilan.html , (2017)

[2] İnternet: http://www.iticu.edu.tr/uploads/yayin/fen21_dosyalar/1-23.pdf(2017)

[3]İnternet:http://www.isites.info/PastConferences/ISITES2016/ISITES2016/papers/B6-ISITES2016ID109.pdf, (2017)

[4] Traffic Count Methodology, Orange County Government, Florida (http://www.orangecountyfl.net/cms/DEPT/pw/traffic/method.htm, ùubat 2009) (2017)

[5] İnternet: https://flink.apache.org/(2017)

[6]İnternet:http://data-flair.training/blogs/create-run-first-apache-flink-application-java-eclipse/(2017)

[7] İnternet: https://www.pcsteps.com/5492-how-to-install-java-linux-mint-ubuntu/(2017)

[8] İnternet:http://data-flair.training/blogs/introduction-apache-flink-quickstart-tutorial/(2017)

[9] İnternet:http://data-flair.training/blogs/apache-flink-tutorial-comprehensive-guide/(2017)

[10]İnternet: http://data-flair.training/blogs/apache-flink-ecosystem-components/(2017)

[11]İnternet:http://data-flair.training/blogs/run-apache-flink-wordcount-program-eclipse/(2017)

[12]İnternet:https://ci.apache.org/projects/flink/flink-docs-release-1.3/quickstart/setup_quickstart.html(2017)

[13]. Internet: http://cagriemreakin.com/veri-bilimi/support-vector-machine-svm-8.html (2017).