**Bölüm 4**

**KONVEKS KÜMELER-FONKSİYONLAR**

Bu bölümde doğrusal programlama teoreisinin anlaşılabilmesi için gerekli olan geometrik bazı kavramlara değinilecektir.

**1. KONVEKS KÜMELER**

**Tanım4.1:** Rn uzayında S kümesi verilsin. olmak üzere bu iki noktadan geçen doğru;

şeklinde verilen S noktalar kümesi olarak tanımlanır. Burada lambda skaler bir değerdir.

Eğer ise bu durumda S, yi birleştiren doğru parçası adını alır.

**Örnek .** için S kümesi aşağıda gösterilmiştir.

**Tanım4.2 (Konveks Küme):** olsun. Eğer her ve için oluyorsa, başka bir deyişle X kümesindeki her x1, x2 elemanı için bu noktaları birleştiren doğru parçası üzerindeki tüm noktalar yine X kümesinin elemanları ise bu durumda X kümesine konveks (dışbükey-yukarıya doğru konveks) küme adı verilir.

**Tanım4.3 (Pozitif kombinasyon):** olsun. Eğer ise bu durumda

(4.1)

ifadesine ’ lerin bir pozitif kombinasyonu denir.

**Tanım 4.4 ( Konveks kombinasyon):** olsun.

Eğer ise bu durumda

(4.2)

ifadesine ’ lerin bir konveks kombinasyonu adı verilir. ise bu durumda (4.2)’ye sıkı konkveks kombinasyon denir.

**Not:** Lineer kombinasyonun tanımını hatırlanırsa, lineer kombinasyondan pozitif kombinasyona ve ordan da konveks kombinasyona genelden daha özele bir geçiş olduğu görülebilir. Noktaların ya da vektörlerin lineer kombinasyonu, katsayıların herhangi bir reel sabit olarak seçilebilmesine imkan verir. Diğer taraftan bir pozitif kombinasyonda katsayılar pozitif olmaya zorlanırken konveks kombinasyonda katsayılar pozitif ve toplamları 1 olmaya kısıtlanmıştır.

**Örnek .** Aşağıdaki şekillerde konveks ve konveks olmayan kümelere örnekler verilmiştir.

**Şekil 4.1.** Soldaki elips bir konveks kümedir; Elipsin içindeki her nokta çifti için bunları oluşturan doğru parçası yine elips tarafından içerilmektedir. Yandaki şekilde ise konveks olmayan bir küme örneği vardır.

**Teorem 4.1:** Rn de sonlu sayıda konveks kümenin arakesiti de konvekstir.

**İspat.**  konveks kümeler olsunlar. konveks kümlerin kesişim kümesi olsun. ‘i göz önüne alalım. ise C’nin tanımından ’dir. Konvekslik tanımından olup bu da demektir. Buradan C konveks bir kümedir.

**Uyarı:** Konveks kümelerin birleşimi konveks değildir!

* Konveks bölgeleri sınırlı ve sınırsız olarak ikiye ayırabiliriz.
* 2-boyutlu reel uzayın kendisi konveks bir kümedir.
* 2-boyutlu uzayda hangi çeyrek kadrandan alınırsa alınsın herhangi iki noktayı birleştiren doğru parçası sözü edilen bölgede kalacaktır.
* 2-boyutlu uzay sınırsızdır. Sınırsız konveks kümedir.
* Bu durum n-boyuta doğru genelleştirilebilir.
* n-boyutlu uzay sınırsız konveks kümedir.
* Üçgenler, dörtgenler, daha genel olarak çokgenler, elips, daire vd. sınırlı konveks bölgedir.
* Hiperdüzlemler, kapalı-açık yarıuzaylar tek başlarına düşünüldüğünde sınırsız konveks kümelerdir.
* Arakesitleri düşünüldüğünde sınırlı ya da sınırsız olabilirler.

**2. KONVEKS VE KONKAV FONKSIYONLAR**

**Tanım 4.5 (Konveks Fonksiyon):** Bir fonksiyonu her için

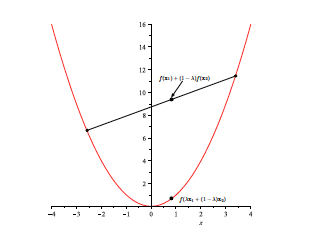
(4.4)

eşitliğini sağlıyorsa konveks (yukarıya doğru konveks-dışbükey) fonksiyon adını alır.

**Tanım 4.6 (Konkav Fonksiyon):** Bir fonksiyonu her için

(4.5)

eşitliğini sağlıyorsa konkav (Aşağıya doğru konveks-içbükey) fonksiyon adını alır.



Şekil 4.2. Bir konveks fonksiyonun grafiği.

**Teorem 4.2** konveks bir fonksiyon olsun. Bu durumda konveks bir kümedir.

**İspat.** ….

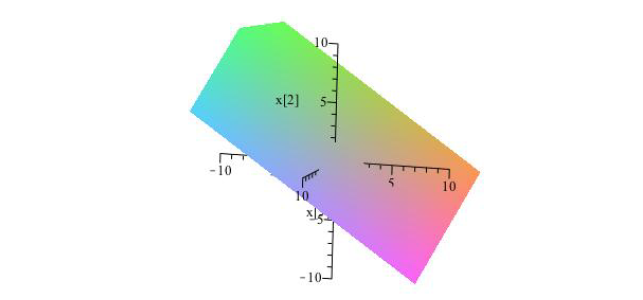
**3. POLİHEDRAL (ÇOK YÜZLÜ) KÜMELER**

Konveks kümelerin önemli örnekleri, düzlemde poligonların-çokgenlerin çok boyutlu analogları olan polihedral kümelerdir. Bu yapıları anlayabilmek için ilk olarak hiperdüzlem ve yarı düzlem kavramları iyi bilinmelidir.

**Tanım 4.7 Hiperdüzlem**: aRn n boyutlu uzayda bir sabit vektör ve bir sabit skaler olsun.

nokta kümesi n boyutlu uzayda bir hiperdüzlem (çok boyutlu düzlem) adını alır. Burada a ve x kolon vektörler olarak kullanılmaktadır.

Örnek: hiperdüzlemini gözönüne alalım.



Şekil 4.3. 3 Boyutlu Uzayda Bir Hiperdüzlem. Bir hiperdüzlem denklemini sağlayan x noktalarının kümesidir. Burada a sabit vector b de sabit bir skalerdir