

Pokemon, Friends, ve Disney'e İlişkin Veri Görselleştirmeleri

Buğracan Tanrıverdi

17/11/22

Özet

Pokemon oyunlarındaki pokemonların efsanevi olup olmadıklarına göre ve bulundakları jenerasyonlara göre çeşitli istatistikler, Friends TV dizisinin sezonlarına göre IMDB puanlarının kıyaslaması, ve Walt Disney filmlerinin süreleri ile IMDB puanları arasındaki ilişkiler üzerine bir veri görselleştirmesi çalışması.

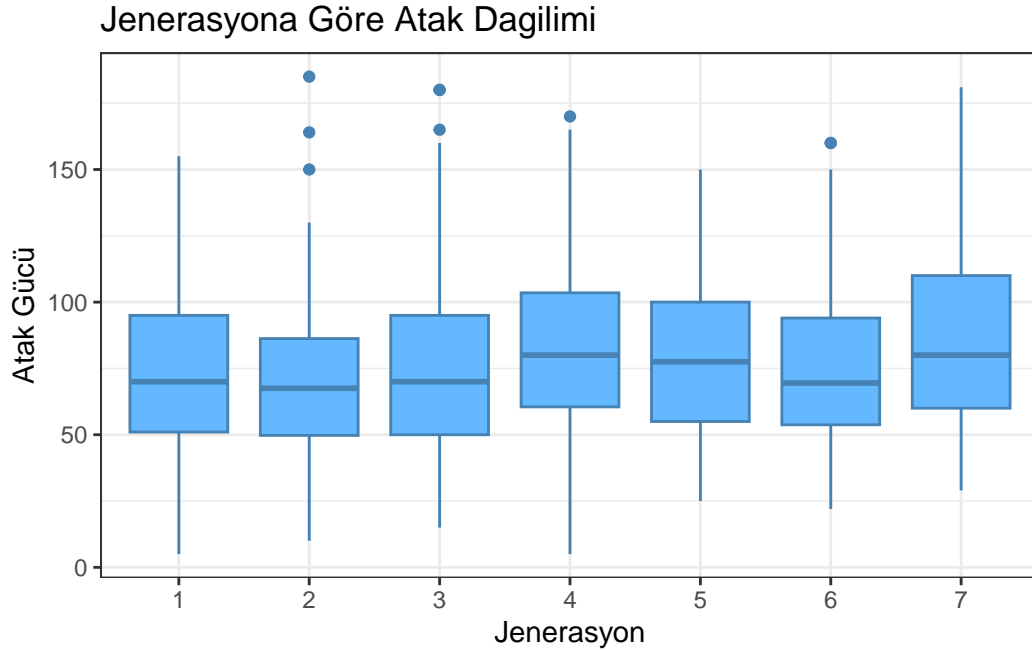
1. Pokemon

Pokemon oyununda yer alan bütün pokemonların atak güçleri, defansları, hızları gibi istatistiklerinin yanı sıra, pokemonun hangi jenersyona ait olduğu ve efsanevi olup olmadığı bilgisine de yer veren veri setine [Kaggle](#)'dan ulaşabilirsiniz.

1.1 Jenerasyonlarına Göre Pokemonların Atak Güçlerinin Dağılımı

Pokemon oyununa her jenerasyon ile birlikte yeni pokemonlar eklenir. Oyunun yapımcılarının, eski pokemonların da kullanılmalarını istedikleri için jenerasyonlar arasında bir güç dengesi kurmalarını bekleriz.

```
ggplot(pokemon, aes(x = as.factor(generation), y = attack)) +  
  geom_boxplot(color = "steelblue", fill = "steelblue1") +  
  labs(  
    title = "Jenerasyona Göre Atak Dağılımı",  
    x = "Jenerasyon",  
    y = "Atak Gücü"  
  ) +  
  theme_bw()
```

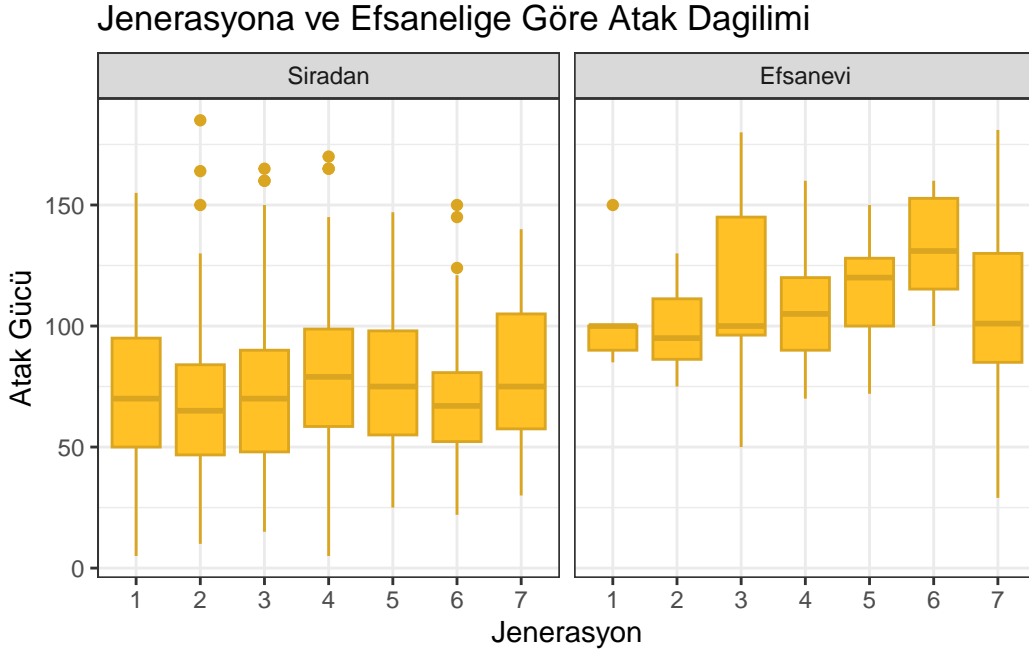


Her bir jenerasyona karşılık gelen atak güçlerinin dağılımları birbirlerinden farklı görünmüyor. Beklendiği gibi, yeni jenerasyonlarda eklenen pokemonlar, eski jenerasyonlardaki pokemonların daha az kullanılmasını önlemek amacıyla eski jenerasyonlarla denk seviyede saldırı güçleriyle eklenmiş.

1.2 Jenerasyon ve Efsane Olup Olmadıklarına Göre Atak Güçlerinin Dağılımı

Pokemon oyununda en nadir karşılaşılan ve en güçlü olarak bilinen pokemon sınıfı *efanevi* olarak bilinir. Her jenerasyonda ancak bir kaç adet sunulan bu pokemonların gerçekten de diğerlerinden daha güçlü olup olmadıklarını görelim.

```
ggplot(pokemon, aes(x = as.factor(generation), y = attack)) +  
  geom_boxplot(color = "goldenrod", fill = "goldenrod1") +  
  labs(  
    title = "Jenerasyona ve Efsanelige Göre Atak Dagilimi",  
    x = "Jenerasyon",  
    y = "Atak Gücü"  
  ) +  
  facet_grid(~pokemon$is_legendary) +  
  theme_bw()
```

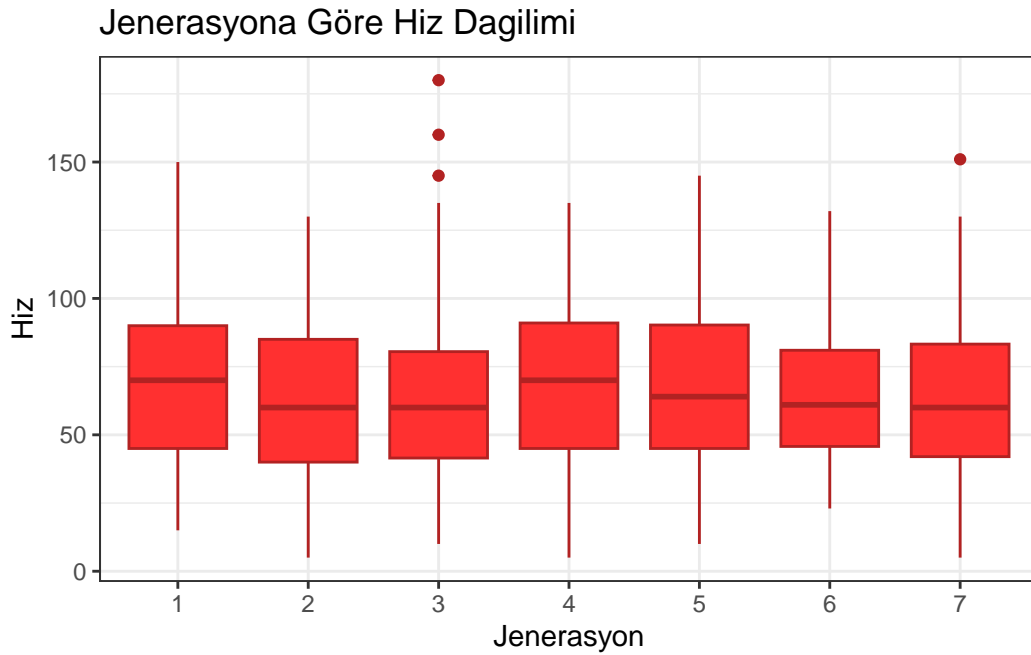


Beklediğimiz gibi, her jenerasyonda eklenen efsanevi pokemonlar, o jenerasyondaki diğer pokemonlardan daha yüksek atak güçlerine sahipler gibi görünüyor.

1.3 Jenerasyonlarına Göre Hızlarının Dağılımı

Bir pokemonun en önemli özelliklerinden biri de hızıdır. Her bir jenerasyonda eklenen pokemonların hızlarını da kıyaslayalım.

```
ggplot(pokemon, aes(x = as.factor(generation), y = speed)) +  
  geom_boxplot(color = "firebrick", fill = "firebrick1") +  
  labs(  
    title = "Jenerasyona Göre Hiz Dagilimi",  
    x = "Jenerasyon",  
    y = "Hiz"  
  ) +  
  theme_bw()
```



Atak gücü ile aynı şekilde, beklendiği gibi pokemonların hızları da jenerasyonlar arasında dengede tutulmuş. Ancak 4. jenerasyon diğerlerinden daha hızlı gözüküyor.

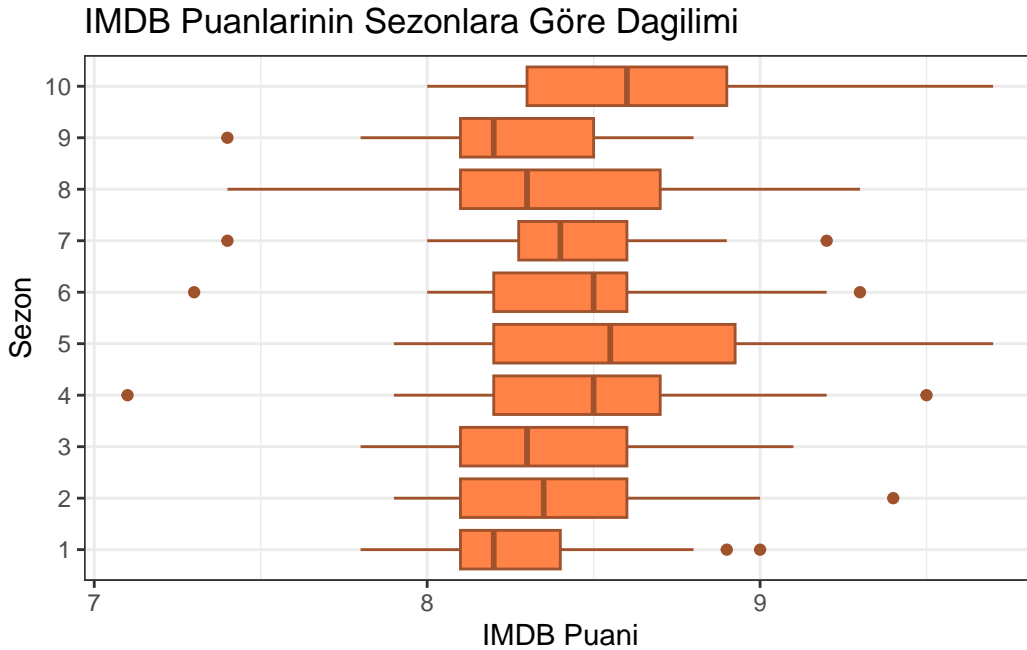
2. Friends

Friends dizisinin her bir bölümü için sezon ve IMDB puanları bilgisini içeren veri setine [Kaggle](#)'dan ulaşabilirsiniz.

2.1 Sezonlara Göre IMDB Puanlarının Dağılımı

Friends dizisi çok popüler bir dizi olduğu için IMDB puanlarının yüksek olmasını bekleriz.

```
ggplot(friends, aes(x = imdb_rating, y = as.factor(season))) +  
  geom_boxplot(fill = "sienna1", color = "sienna") +  
  labs(  
    title = "IMDB Puanlarının Sezonlara Göre Dağılımı",  
    x = "IMDB Puanı",  
    y = "Sezon"  
  ) +  
  theme_bw()
```

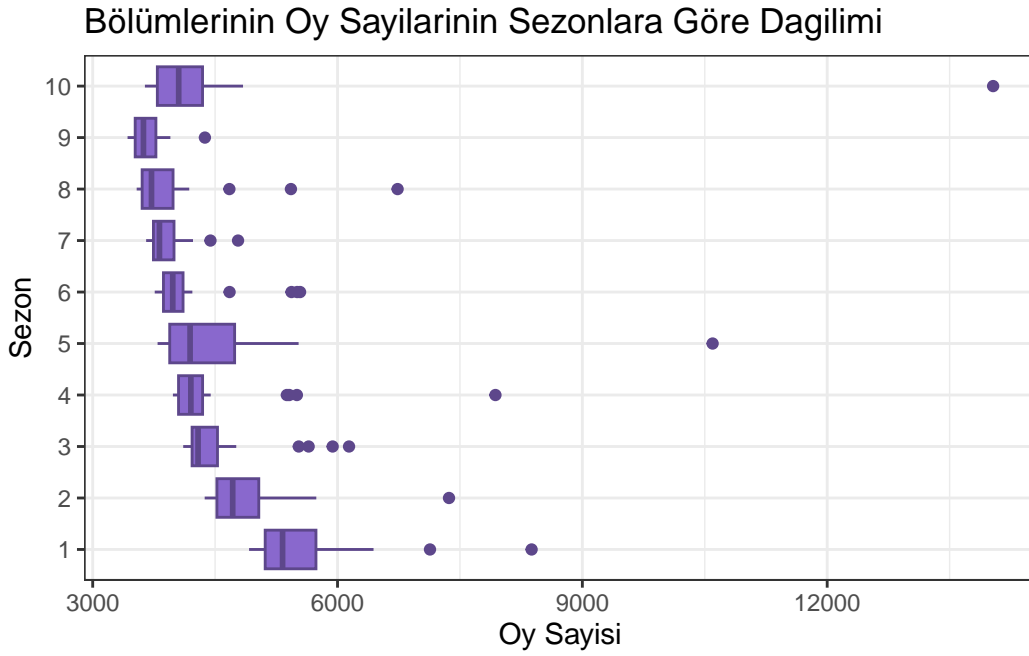


Sezonların IMDB puanları arasında ciddi bir farklılaşma gözüküyor. 1. sezon ve 9. sezon diğer sezonlara göre daha düşük bir ortalamaya sahip iken, 5. ve 10. sezonlar en yüksek puanlara sahip. Ayrıca en asimetric puan dağılımına sahip sezon 6. sezon gibi gözüküyor.

2.2 Sezonlara Göre Oy Sayısının Dağılımı

Şimdi de sezonlara göre, IMDB oy sayılarını inceleyelim.

```
ggplot(friends, aes(x = total_votes, y = as.factor(season))) +  
  geom_boxplot(fill = "mediumpurple3", color = "mediumpurple4") +  
  labs(  
    title = "Bölümlerinin Oy Sayılarının Sezonlara Göre Dağılımı",  
    x = "Oy Sayısı",  
    y = "Sezon"  
  ) +  
  theme_bw()
```



Puan sayıları arasında en ciddi farkın gözüktüğü sezon 10. sezon. Bunun en önemli sebebi ise dizinin son bölümünün çok fazla kişi tarafından oylanmış olması.

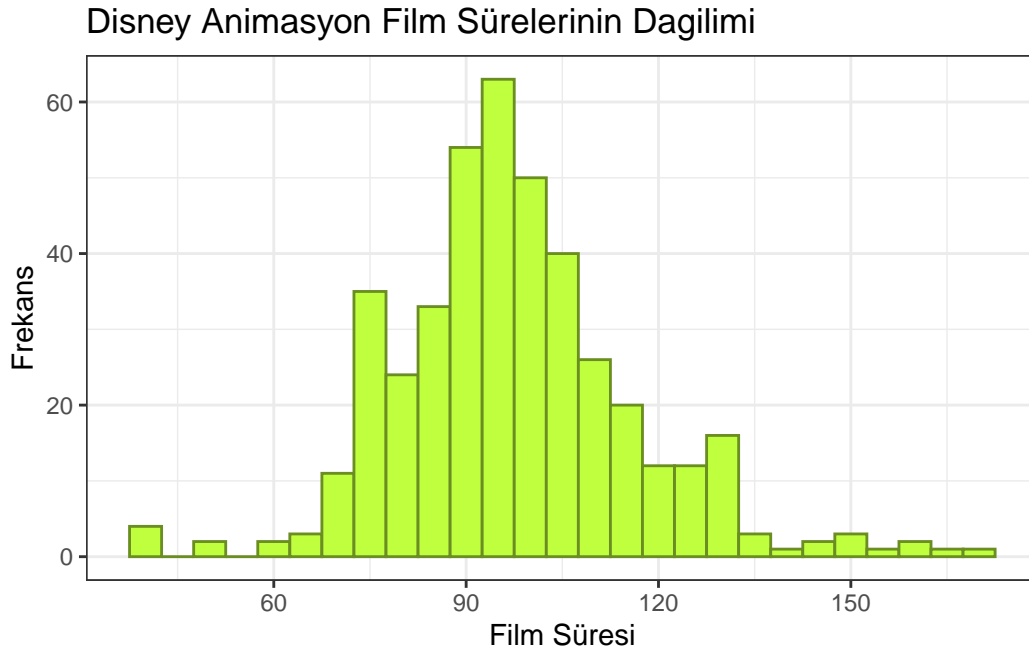
3. Disney

Walt Disney filmlerinin IMDB puanları ve filmlerin sürelerini içeren veri setine [Kaggle](#)'dan ulaşabilirsiniz.

3.1 Film Sürelerinin Dağılımı

Walt Disney filmlerinin sürelerinin dağılımını inceleyelim.

```
ggplot(disney.1, aes(x = `Running time (int)`) +  
  geom_histogram(binwidth = 5, color = "olivedrab", fill = "olivedrab1") +  
  labs(  
    title = "Disney Animasyon Film Sürelerinin Dagilimi",  
    x = "Film Süresi",  
    y = "Frekans"  
  ) +  
  theme_bw()
```

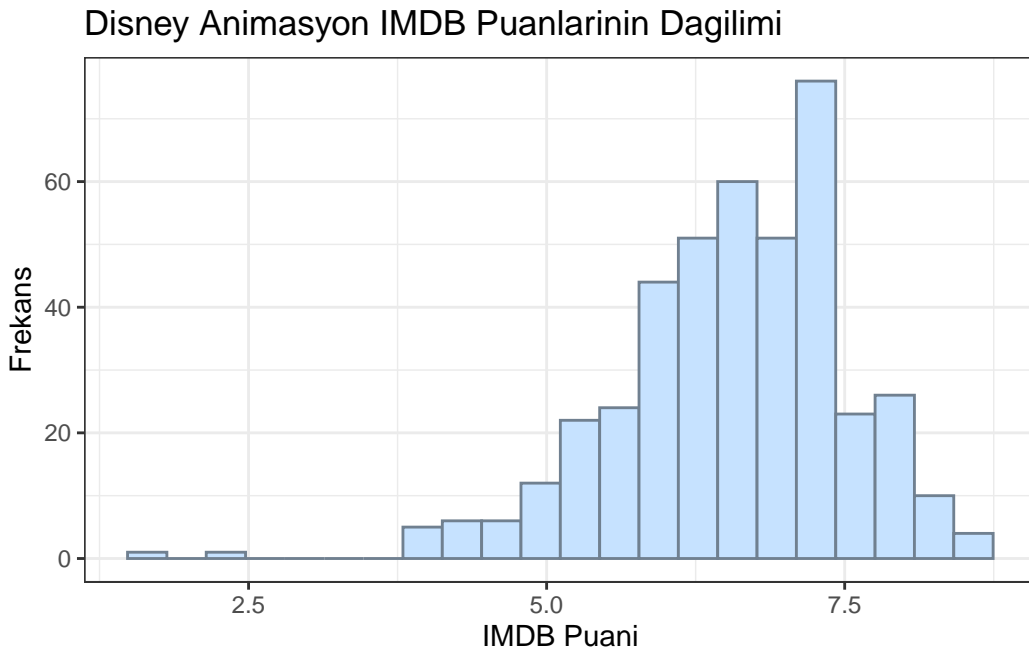


Çok uzun ve çok kısa filmler gözükmemekle birlikte, filmlerin sürelerinin normal dağılıma sahip olduğuna dair güçlü bir görsel karşımıza çıkıyor.

3.2 Filmlerin IMBD Puanlarının Dağılımı

Şimdi de filmlerin IMDB Puanlarının dağılımını inceleyelim.

```
ggplot(disney.1, aes(x = as.numeric(imdb_rating))) +  
  geom_histogram(binwidth = 0.33, color = "slategray", fill = "slategray1") +  
  labs(  
    title = "Disney Animasyon IMDB Puanlarının Dağılımı",  
    x = "IMDB Puanı",  
    y = "Frekans"  
  ) +  
  theme_bw()
```

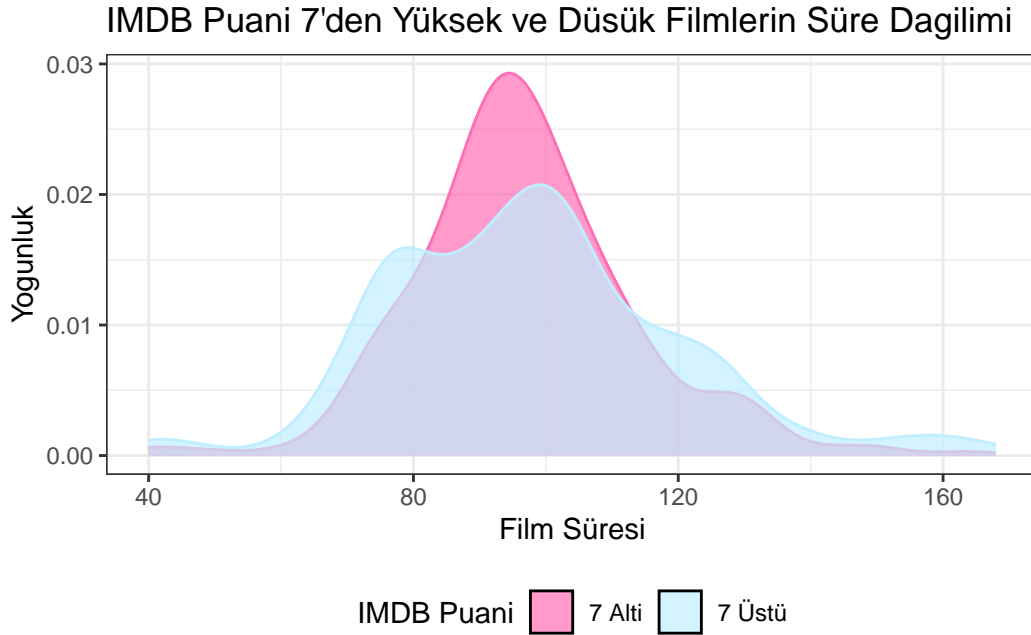


Dağılım bir iki tane 2.5'ten daha düşük puana film hariç gayet simetrik gözüküyor. Aykırı değerleri saymaz isek IMDB puanlarının da normal dağılıma sahip olabileceği gözlemleniyor.

3.3 IMDB Puanı 7'den Yüksek ve Düşük Olan Filmlerin Sürelerinin Dağılımı

Şimdi de IMBD puanlarına göre filmleri iki gruba ayırıp, bu iki grubun sürelerinin dağılımlarını birbirleriyle kıyaslayalım.

```
ggplot(disney.1, aes(fill = as.factor(above_7),  
                      color = as.factor(above_7),  
                      x = `Running time (int)`) +  
geom_density(alpha = 0.7, bw = 5) +  
labs(  
  title = "IMDB Puanı 7'den Yüksek ve Düşük Filmlerin Süre Dağılımı",  
  x = "Film Süresi",  
  y = "Yogunluk",  
  fill = "IMDB Puanı"  
) +  
scale_fill_manual(values=c("hotpink1", "lightblue1")) +  
scale_color_manual(values=c("hotpink1", "lightblue1")) +  
theme_bw() +  
guides(color = FALSE) +  
theme(legend.position = "bottom")
```



İki grubun süreleri arasında ciddi bir değişkenlik gözlemlenmiyor. Buradan filmlerin beğenilip beğenilmediğinin filmlerin süreleri ile ilgili olmadığı sonucuna varabiliriz.