

## GÖRSELLEŞTİRME

Görselleştirme analizleri ve sonuçları paylaşmak ve anlatmak için en güçlü yöntemdir. Bir görsel dakikalarca sürececek bir açıklamanın yerini kolayca alabilir. Bu nedenle dersimizde görselleştirmenin üzerinde oldukça durduk. ggplot paketi görselleştirme için bize büyük kolaylıklar sağlayan bir pakettir. Bu paketin kullanımı da oldukça kolaydır. Dersimizde de bahsettiğimiz gibi ggplot fonksiyonlarını bir iskelet üzerine oturtabiliriz.

İlk olarak hangi verinin hangi sütunlarıyla işlem yapacağımızı R'a söylememiz gerekiyor.

Sonrasında bu veri ile nasıl bir işlem yapacağımızı söylememiz gerekiyor. Örneğin, bir bar chart yapacaksak `geom_bar` komutu, bir histogram yapacaksak `geom_hist` komutunu bu bölümde kullanmamız gerekiyor.

Sonrasında da eğer görselimizi özelleştirmek istiyorsak kodumuzun sonuna gereken komutları eklememiz gerekiyor. Örneğin oluşturacağımız görselin verimizde bulunan bir kategorik değişkenin seviyelerine göre, her seviye için oluşmasını istiyorsak `facet_wrap` fonksiyonunu bu kısımda ekleyebiliriz. 'Ekleme' kelimesini kullanma sebebim gerçekten de bunları birbirine + işareti ile ekliyor olmamız. Yani

```
ggplot(...) + geom_point(...) + facet_wrap(...)
```

gibi bir iskeletimiz oluşuyor.

Görseldeki verilerimizi farklı gruplara göre farklı renklere sahip olacak şekilde ayarlayabiliriz. Örneğin `geom_bar` fonksiyonundaki `color` parametresi burada kullanılabilir. `color` parametresine bir kategorik değişken atadığımızda görseldeki veriler ait oldukları kategorilere göre farklı renkler alacaktır.

### Örnek kodlar

```
p1 <- ggplot(train, aes(x=Pclass,fill=Survived)) + geom_bar(position='fill')
p2 <- ggplot(train,aes(x=Age,color=Survived)) + geom_freqpoly(binwidth=1) +
theme(legend.position = "none")
p3 <- ggplot(train,aes(x=Age,color=Survived)) + geom_density()
+ theme(legend.position = "none")
p4 <- ggplot(train,aes(x = SibSp+Parch,fill=Survived)) + geom_bar(position='fill')
+ theme(legend.position = "none")
p5 <- ggplot(train,aes(x = Fare, color = Survived)) + geom_freqpoly(binwidth=0.04) +
scale_x_log10() + theme(legend.position = "none")
ggplot2.multiplot(p1,p2,p3,p4,p5)
```

Çıktımız,

