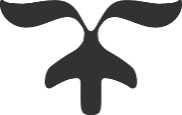


hacettepe üniversitesi

İstatiksel Analiz



Mehmet şenel istatiksel analiz ödevi

2210329077

İçindekiler

[1.0 Kullanılan Paketler 2](#_Toc154146642)

[1.1 Dbplyr 2](#_Toc154146643)

[1.2 Ggiraph 2](#_Toc154146644)

[1.3 E1071 2](#_Toc154146645)

[1.4 Shiny 2](#_Toc154146646)

[2.0 Veri Hakkında 3](#_Toc154146647)

[2.1 Verinin Rstudioya aktarımı 3](#_Toc154146648)

[2.2 Verinin görünümü 4](#_Toc154146649)

[2.3 Veri Hakkında 4](#_Toc154146650)

[3.0 Örneklem Oluşturma 5](#_Toc154146651)

[3.1 Örneklem Oluşturma 5](#_Toc154146652)

[3.2 Örneklem Çıktısı 6](#_Toc154146653)

[4.0 Verinin Analizi 7](#_Toc154146654)

[4.1 Temel Analizler 7](#_Toc154146655)

[4.2 Creating One Scatter Plot for Each Group 8](#_Toc154146656)

[4.2.1 Kullanılan Paketler 8](#_Toc154146657)

[4.2.2 Grafiğin Oluşturulması 8](#_Toc154146658)

[4.3 Creating One Boxplot for Each Factor Level 10](#_Toc154146659)

[4.3.1 Kullanılan Paketler 10](#_Toc154146660)

[4.3.2 Grafiğin Oluşturulması 11](#_Toc154146661)

[4.3.3 Grafik Çıktıları ve Yorumlama 13](#_Toc154146662)

[4.4 Creating a Histogram 14](#_Toc154146663)

[4.4.1 Kullanılan Paketler 14](#_Toc154146664)

[4.4.2 Grafiklerin Oluşturulması 14](#_Toc154146665)

[4.4.3 Grafik Çıktıları ve Yorumlama 15](#_Toc154146666)

[4.4.4 Adding a Density 15](#_Toc154146667)

[15](#_Toc154146668)

[5.0 Changing the Type, Width, or Color of a Line 18](#_Toc154146669)

[5.1 Line Type 18](#_Toc154146670)

[5.2 Change width and color 19](#_Toc154146671)

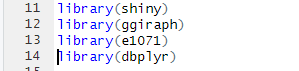
[6.0 Writing Your Plot to a File 20](#_Toc154146672)

[6.1 Uygulamadan Yazdırma 20](#_Toc154146673)

[6.2 Kod ile Yazdırma 21](#_Toc154146674)

[Kaynakça 21](#_Toc154146675)

# Kullanılan Paketler

Yapıcak olduğumuz ödevimizde sunumuzun tamamı R Studio programı ile hazırlanacak olup kullanılan temel paketler ve kütüphaneler görüldüğü gibidir.

## Dbplyr

Dplyr’i, sözlük anlamıyla “veri işlemenin temel prensipleri” veya “veri işleme grameri” olarak çevirebiliriz. Hadley Wickham tarafından geliştirilen R’da sıklıkla kullanılan paketlerin koleksiyonundan oluşan [tidyverse](https://www.tidyverse.org/" \t "_blank) evreninin bir parçası.

Dplyr’a gramer denmesinin sebebi, veri setlerinin düzenlenmesi, filtrelenmesi, sıralanması ve belirli değişkenlerin hesaplanmasını mümkün kılan 6 ayrı fiile (fonksiyona) sahip olması. Bunlar: select (), mutate (), filter (), arrange (), summarize () ve group\_by (). R programlama dilini öğrenmek bu fiilleri kullanmaktan geçiyor.

## Ggiraph

Ggiph, dinamik ggplot grafikleri oluşturmanıza olanak tanıyan bir araçtır. Bu, grafiklere araç ipuçları, vurgulu efektler ve JavaScript eylemleri eklemenizi sağlar. Paket ayrıca parlak uygulamalarda kullanıldığında grafik öğelerin seçilmesine de olanak tanır.

## E1071

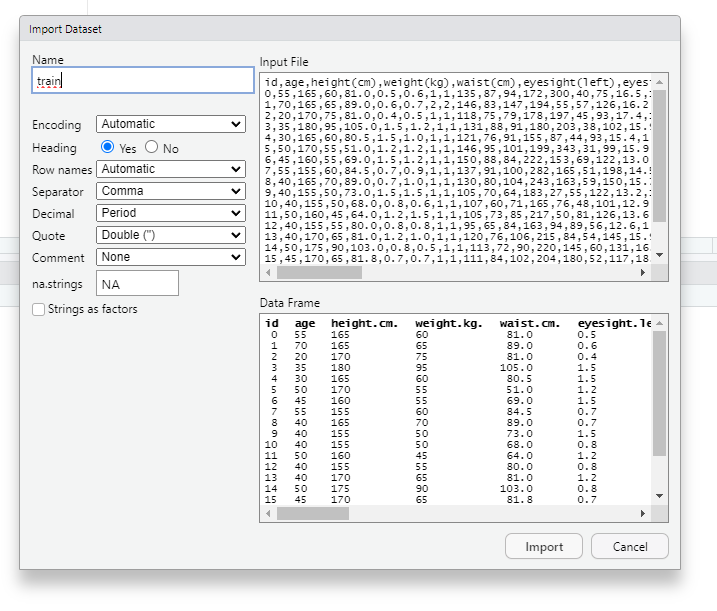
SVM'lerin hızlı ve kolay uygulanmasını sağlar. Doğrusal, polinom, RBF ve sigmoid dahil olmak üzere en yaygın çekirdekleri sağlar. Tahminler için karar ve olasılık değerleri için hesaplama gücü sunar. Ayrıca sınıflandırma modunda ve çapraz doğrulamada sınıfların ağırlıklandırılmasını sağlar.

## Shiny

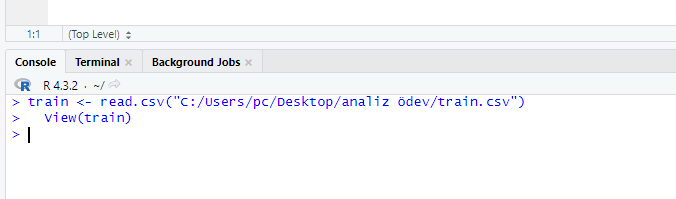
Shiny son yıllarda kullanımı oldukça yaygınlaşan açık kaynaklı bir R paketidir. Shiny ile R programlama dili kullanarak web tabanlı ara yüzlerin oluşturulması sağlanır. Bu sayede HTML, Java Script gibi programlama dillerini bilmeseniz bile analizlerinizi interaktif bir web uygulaması üzerinden gerçekleştirebilirsiniz. İlk versiyonu 28 Şubat 2011 yılında çıkan R shiny Winston Chang önderliğinde bir grup tarafından geliştirilmiştir. Winston’un yanı sıra Joe Cheng, JJ Allaire, Yihui Xie ve Jonathan McPherson R shiny geliştirme sürecinde önemli katkılar sunmuşlardır. Paketin güncelleme ve yeni versiyonlarının geliştirilmesi sürecinde ise pek çok kişinin katkısı vardır ve halen devam etmektedir.

# 2.0 Veri Hakkında

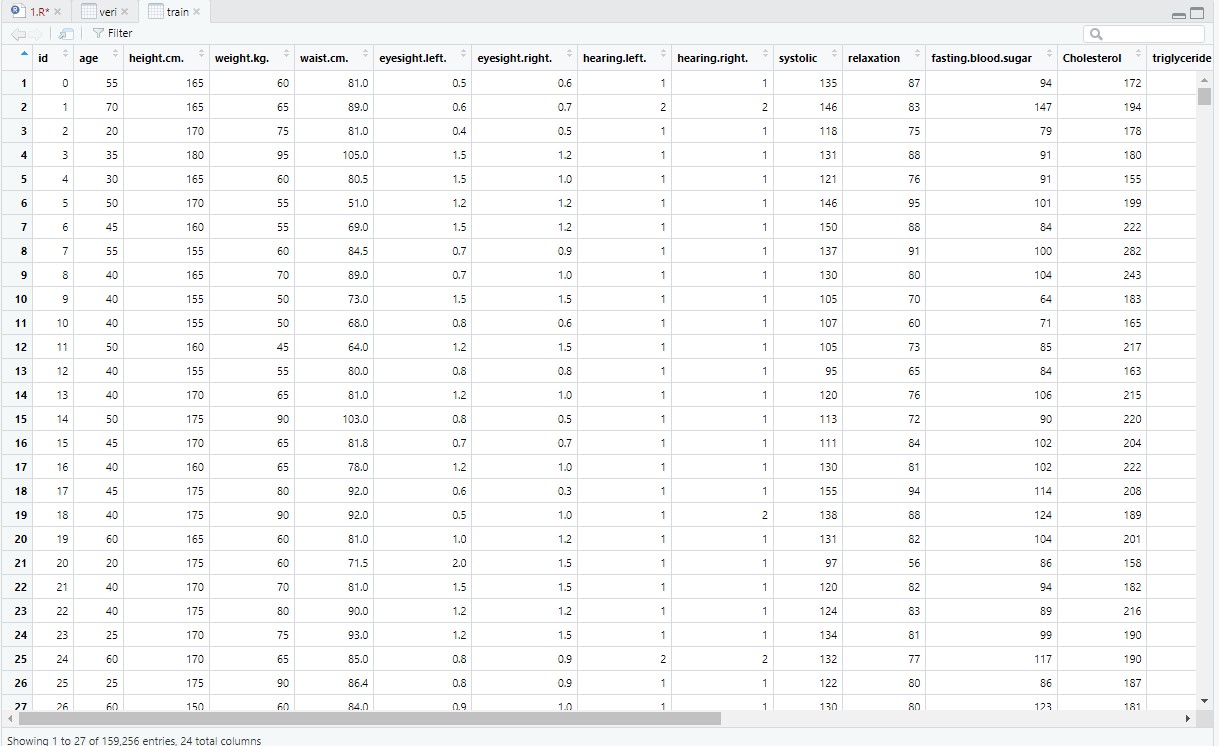
## 2.1 Verinin Rstudioya aktarımı



Rstudio uygulaması üzerinde sırasıyla File, dataset ve son olarak da Import Dataset seçeneği aracılığı ile ya da aşağıda yer alan kod aracılığı ile verimiz uygulamaya çekilir.

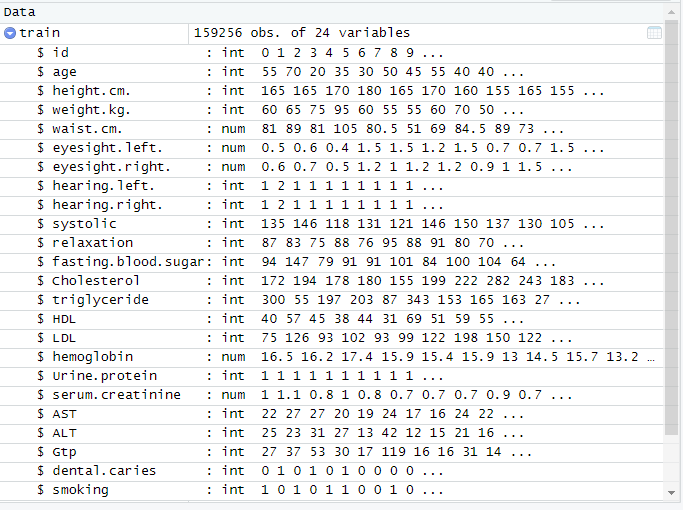


## 2.2 Verinin görünümü



Veri yukarıda belirtildiği şekilde uygulamaya aktarıldıktan sonra çıktısı yukarıdaki gibi olmaktadır.

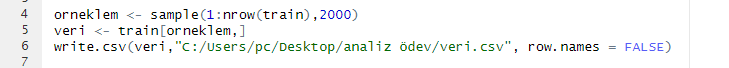
## 2.3 Veri Hakkında

Rstudio uygulamamızın sağ tarafında yer alan kısımdan direkt olarak veri hakkında genel bilgi sahibi olabiliriz.

* Verimiz 24 değişkenden oluşmaktadır.
* Verimizde toplamda 159256 veri yer almaktadır.
* Verimizde yer alan değerler genel anlamda int ve num olarak yer almakta olup bu değerler sayısal ya da sınıflandırılmış verilerdir. (Örneğin smoking değerinin 0 veya 1 olması gibi )

# 3.0 Örneklem Oluşturma

## 3.1 Örneklem Oluşturma



Yukarı görüldüğü üzere “Sample” fonksiyonu aracılığı ile örneklem oluşturma işlemi gerçekleştirilmekte olup, genel kullanımı;

Sample (baş: bitiş (veri kümesi), adet sayısı)

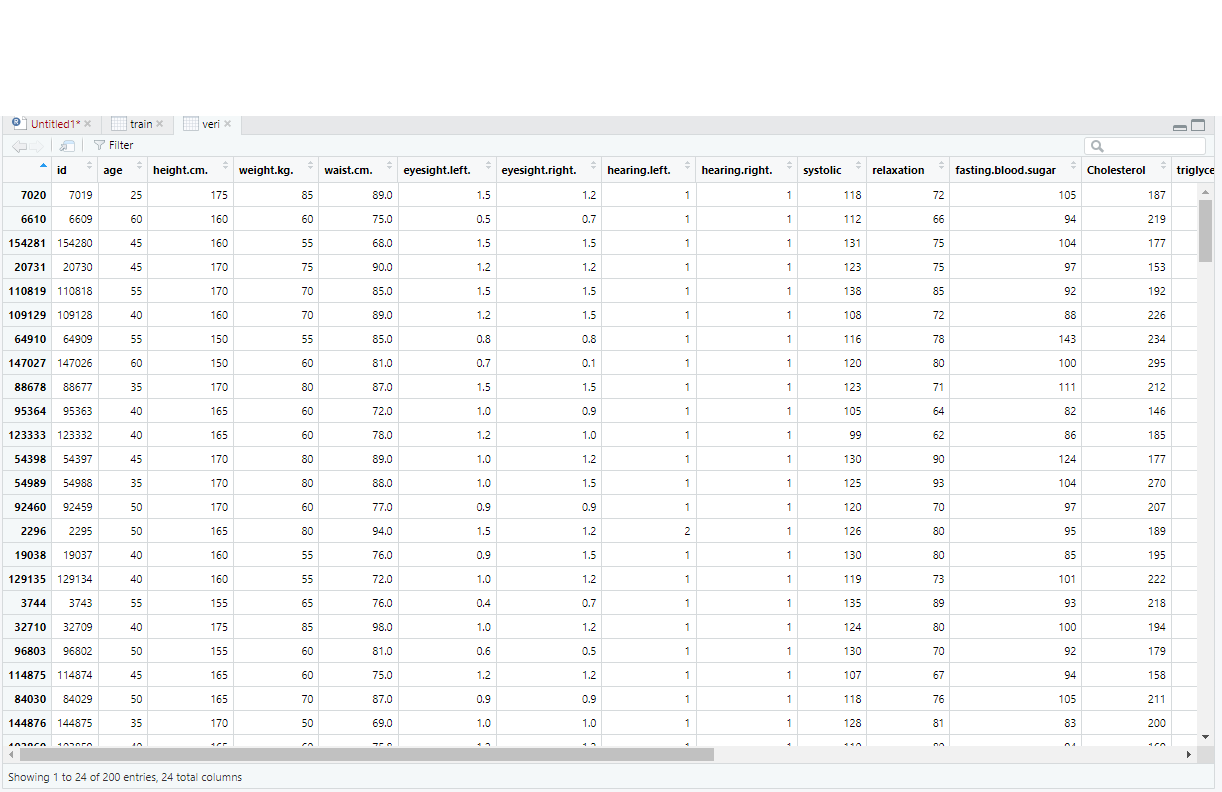
Şeklindedir.

2.adım olarak ise tanımlanmış olan örneklemin bir veri kümesi haline getirilmesi ve son olarak da kaybolmaması veya değişmemesi adına kaydedilmesi aşamaları yer alır.

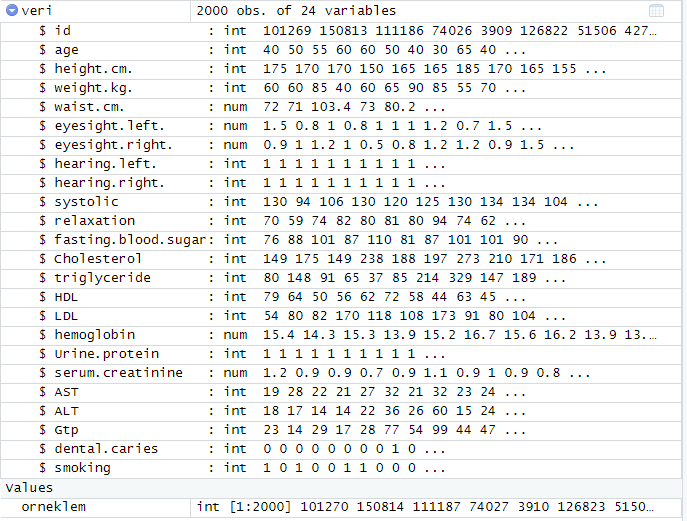
Write.cvs (veri kümesi, “Bilgisayarda kaydedilmesini istediğin alan”)

Şeklindedir.

## 3.2 Örneklem Çıktısı

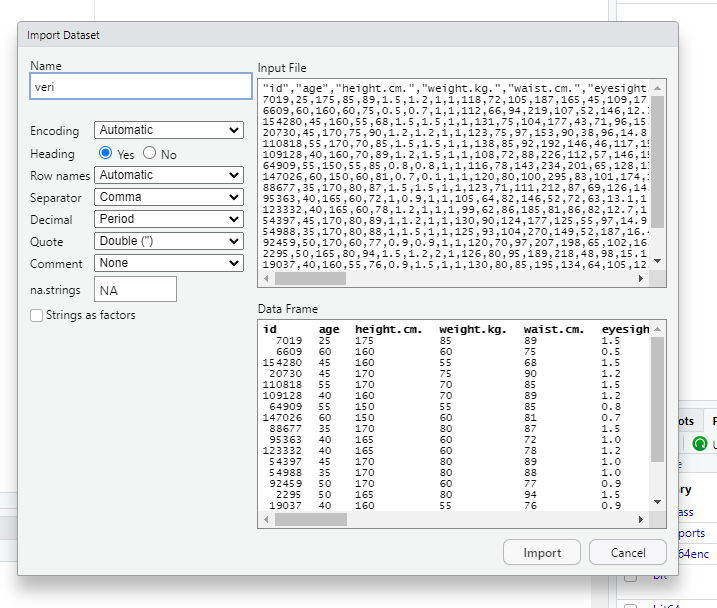


Yazılan kodu çalıştırdıktan sonra yan da yer alan şekilde 2000 adet gözlem taşıyan rastgele seçilmiş bir örneklem kümemiz oluşmuş hale gelir.



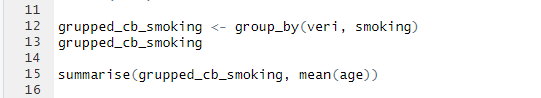
Böylece analizin sağlanması istenilen veri kümemizin incelemesi sağlanılacak olan Örneklemi hazır hale getirmiş bulunmaktayız.

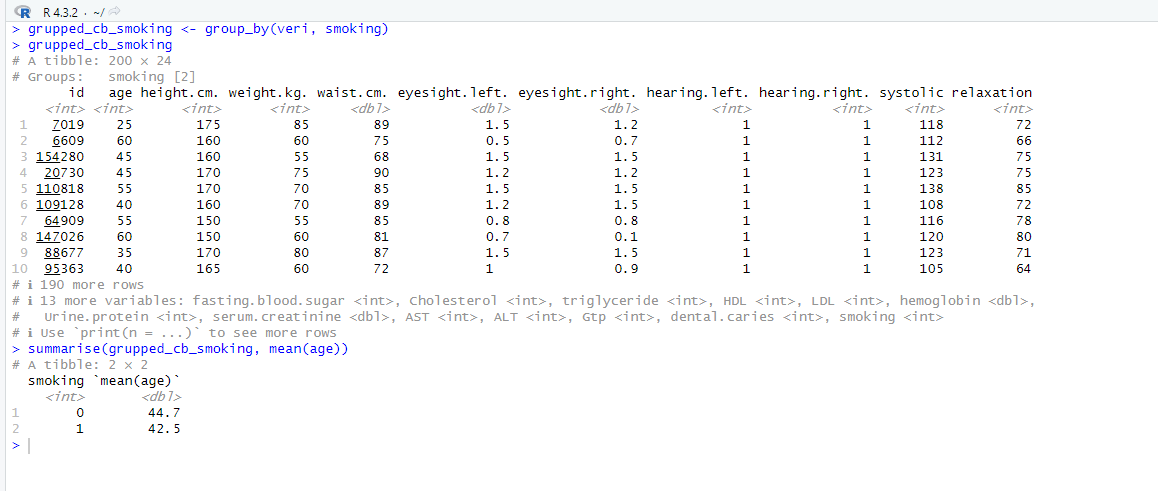
Not:

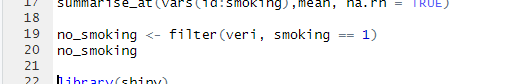
Raporlamanın başında da belirtildiği gibi tekrar vurgulamak gerekirse örneklem kümesi oluşturduktan sonra “cvs” olarak kaydetmemizin temel sebebi projemiz anında bitmeyeceği için uygulama üzerinde kapama açma değişiklik gibi durumlarda sürekli yeni örneklem oluşturmadan mevcut yer alanla devam edebilmektir o yüzden yeniden giriş yapılınca erişim olmaması durumunda “import dataset “kısmında aynı örneklem yüklenerek devam edilir.

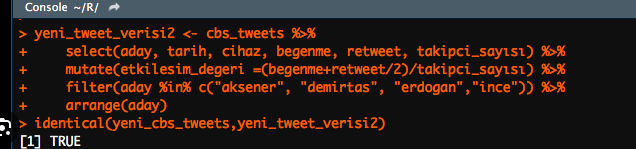
# 4.0 Verinin Analizi

## 4.1 Temel Analizler



Dbplyr kütüphanesinden faydalanarak verileri gruplayarak ortalama, toplam vb. birçok temel analizi gerçekleştirebilmekteyiz, örneğin yukarıda sigara içme durumuna göre sınıflandırmanın ardından sigara içen kişilerin ve içmeyen kişilerin yaş ortalamasın elde etmiş bulunmaktayız.

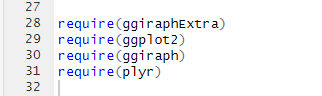
Yanda yer aldığı gibi filtreleme yönetimi aracılığı ile çeşitli verileri sıralayabilir ve kaydedebiliriz. Burada gördüğümüz örnekte “smoking “eşittir 1 olan veri kümesi çekilmiştir.

Bu gibi temel veri analizleri içeren Dbplyr kütüphanesi yardımı ile çoğaltabiliriz.

Yanda yer aldığı gibi “select, mutate, filter, arrange” bunların yanı sıra summerise, mean, gibi birçok fonksiyon ile temel analiz yapılabilir.

## 4.2 Creating One Scatter Plot for Each Group

## 4.2.1 Kullanılan Paketler

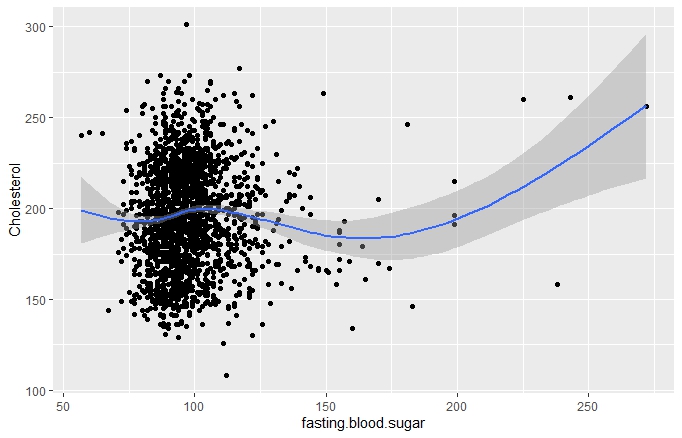
Yan taraftaki resimde yer alan Rstudio kütüphanelerinin temel kullanım amacı 2 ve 3 boyutlu grafiklerde destek ve olanak sağlamak olup oluşturacağımız çoğu grafikte yer alan kütüphanelerden faydalanacağız.

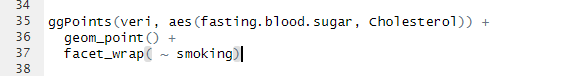
## 4.2.2 Grafiğin Oluşturulması

Rstudio uygulaması üzerinde one scatter grafiği oluşturma adına “ggplot” ya da “ggpoint” fonksiyonlarından faydalanmamız gerekecek.

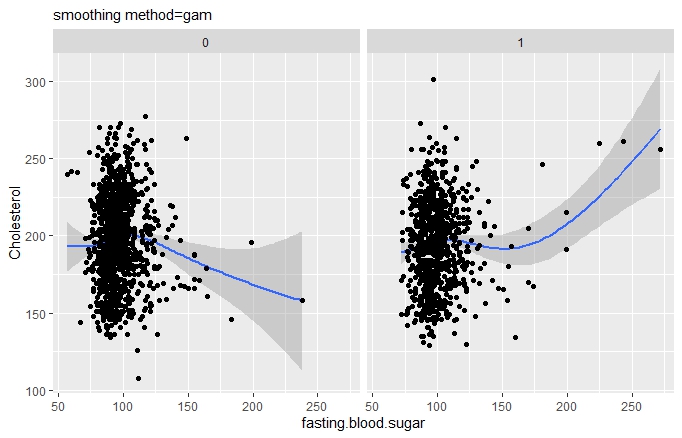


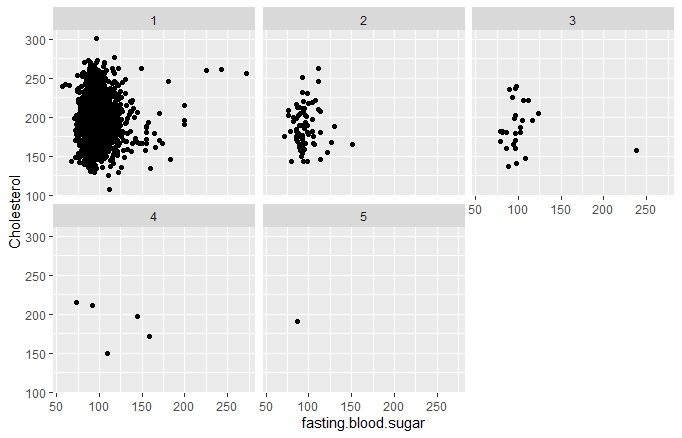
En temel One Scatter Plot grafiğinden başladığımızda yukarıda yer alan ggplot ya da alternatif olarak ggPoint fonksiyonu kullanılarak yanda yer alan grafiğe ulaşım sağlanılır.

* Grafiği inceleme aşamasına geldiğimizde ise Kolesterol değeri ve açlık şekeri diye de tabir edebileceğimiz iki değeri incelemiş oluyoruz.
* İncele sonucunda görüyoruz ki diğer kriterler bir kenara bırakıldığında her iki değerde de ekstrem bir değişim olmadığını ve birbirlerini de olumlu ya da olumsuz bir etkisi yer almamaktadır.



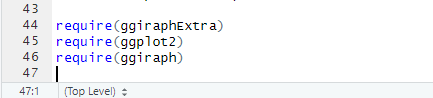
Burada yer alan kodlama aracılığı ile temel incelemesi sağlanacak olan numerik bir verinin çeşitli grafikleri üzerinden inceleme sağlanılacak olup yine aynı şekilde başta belirtilen kütüphanelerden faydalanılmıştır.



* Öncelikli olarak grafiğimize göz geçirdiğimizde x ekseninde açlık şekeri de diyebileceğimiz verilerimiz, y ekseninde ise Kolesterol verilerimiz yer almakla birlikte buradaki temel inceleme koşulumuz kişinin sigara tüketip tüketmemesi ile ilgilidir.
* Sigara tüketen bireyleri yani “1” olan bireyleri incelediğimizde oluşan grafik daha çok kolesterolün değişkenlik gösterdiğini belirtir. Sigara tüketmeyen bireylere baktığımızda ise çok fazla bir değişkenlik olmaksızın hem kolesterol değerinin hem de açlık şekerinin belli düzeylerde sabit kaldığını görmekteyiz. Buradan sonuç olarak sigaranın bireylere kolesterol ve açlık şekeri bakımından aynı etkiyi gösterdiğini belirtmek mümkündür.
* İncelemesini sağlayacağımız bir diğer grafik ise yine aynı x ye y değerlerine istinaden insan vücudundaki bazı protein çeşitlerine etkisini inceleyeceğiz.
* İnceleme sonucunda direkt olarak ilgili protein çeşitlerinden genel olarak 1 numaralı proteinin insan vücudunda yer alması kaynaklı özellikle kolesterol bakımından farklı değerlere rahatlıkla ulaşabileceği kanısına ulaşırız

## 4.3 Creating One Boxplot for Each Factor Level

## 4.3.1 Kullanılan Paketler



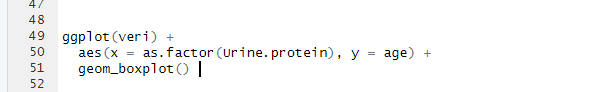
* Yukarıdaki görselde de görüldüğü üzere kullanacağımız ve faydalanacağımız 3 adet kütüphane yer almakta olup aslında baktığımızda aynı kütüphanelerden bir önceki grafiğimizde de yararlandığımızı görmekteyiz.

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Yukarıda görmüş olduğumuz ilgili ggiraph kütüphanesinin G harfi ile başlayan fonksiyonlarını göstermektedir. Burada bu kütüphanenin gösterilmesinin temel nedeni yapacak olduğumuz grafikte geom\_ ile başlayan fonksiyondan faydalanacak olmamızın yanı sıra eğer bir kütüphane içerisinden fonksiyon bulmamız gerekiyor ise nasıl bulunacağı hakkında ufak gir gösterim olmasıdır.

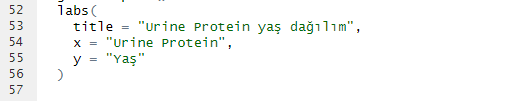
## 4.3.2 Grafiğin Oluşturulması



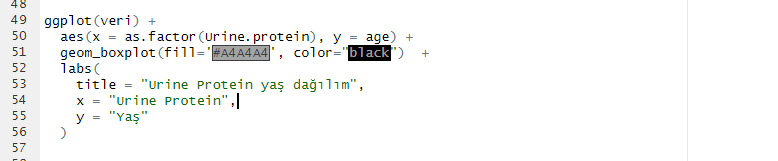
* Yukarıda yer alan “ggplot” fonksiyonu aracılığı ile grafiğe ait yer alan ve aktarım işlemlerini tamamlamış olup ardından bir önceki alanda da belirttiğimiz gibi;

Geom\_boxplot ()

Fonksiyonu ile ggiraph kütüphanesinden faydalanıp grafiğimizi şekillendirmiş olduk.



* Bunun yanı sıra ek olarak yukarıda belirtilen “labs” fonksiyonu aracılığı ile de grafiğimizin başlığı, x eksenin ismi ve y eksenin ismi gibi değişimler yapabilmekteyiz.

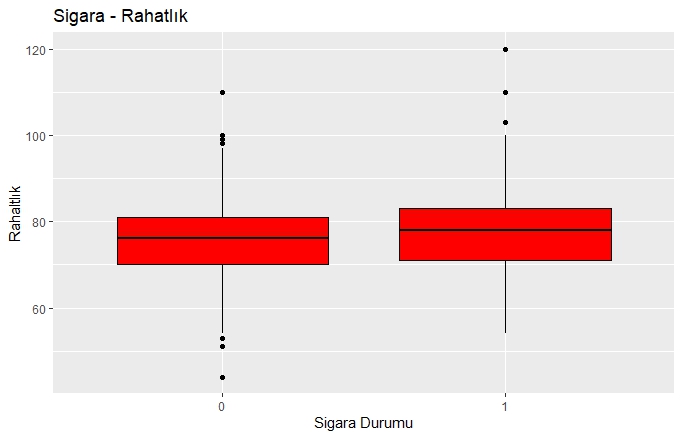


* Bir diğer özelliğimizde yine “geom\_” fonksiyonu içerisinde yer alan “fill” ve” color” araçlarıdır. Renklendirme kısım ve aşamalarında çeşitli renk kodu ve isimleriyle görsel düzen sağlayabilmemize katkı sağlıyor.
* **Kaynakça**

http://www.sthda.com/english/wiki/ggplot2-box-plot-quick-start-guide-r-software-and-data-visualization

## Grafik Çıktıları ve Yorumlama

* Yukarıda kodlamalar aracılığı ile oluşturduğumuz grafik bu şekilde yer almaktadır.
* Grafiğimize genel anlamda baktığımızda Urine Protein miktarının yaş durumuna göre dağılımını baz almaktadır.
* Her beş protein türü içinde inceleme sağladığımızda geçtiğimiz grafiklerde de karşılaştığımız gibi 5 numaralı protein çeşidi az ve istisnai bir şekilde bulunuyor ve 40 yaş dolaylarında yer almaktadır.
* Diğer dört protein çeşidimizi incelediğimizde 1 ve 3 numaralı protein genelde 40-60 yaş aralığında görülmekte, 2 numaralı protein çeşidimiz ise daha çok 40 yaş arasındaki genç kesimde görülmektedir.



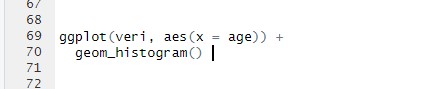
* İkinci bir örnek olarak sigara içme durumunun rahatlık seviyesini etkileme durumunu incelemek adına oluşturulmuştur.
* Grafiğimizi incelediğimizde sigara durumu olarak 0 = içmiyor,1 = içiyor anlamında bir kullanım mevcuttur.
* Grafiğimize aynı zamanda bir bütün olarak da göz gezdirdiğimizde direkt olarak aşırı derecede bir fark yaratacak durum göze çarpmaz. Daha detaylıca kontrol edildiğinde ise sigara içmeyenlerin dağılımında birkaç ufak aykırı değer görmekle beraber daha ortalamaya yakın ve düzgün dağılmış olduğunu görmekteyiz.
* Sonuç olarak inceleme doğrultusunda sigara içip içmemenin rahatlığa çok büyük bir etken yaratmayacağını açığa kavuşturabiliriz

## 4.4 Creating a Histogram

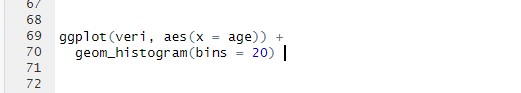
## 4.4.1 Kullanılan Paketler

Kullanılan paketlerde herhangi bir değişim yoktur bir önceki 4.3.1 de ki paketler tamamen aynıdır.

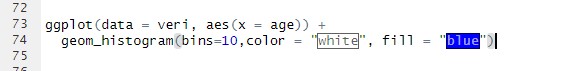
## 4.4.2 Grafiklerin Oluşturulması

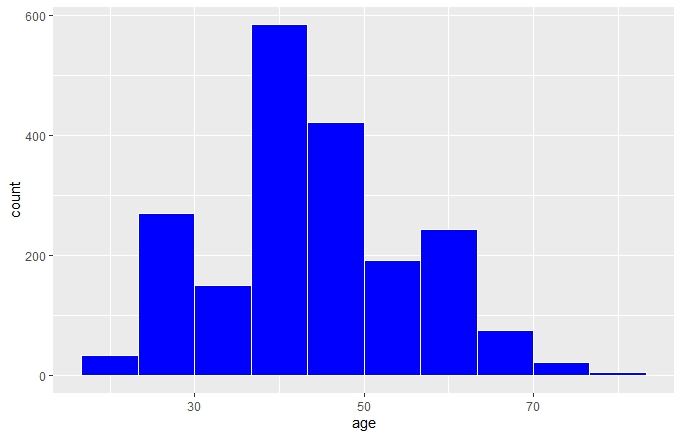
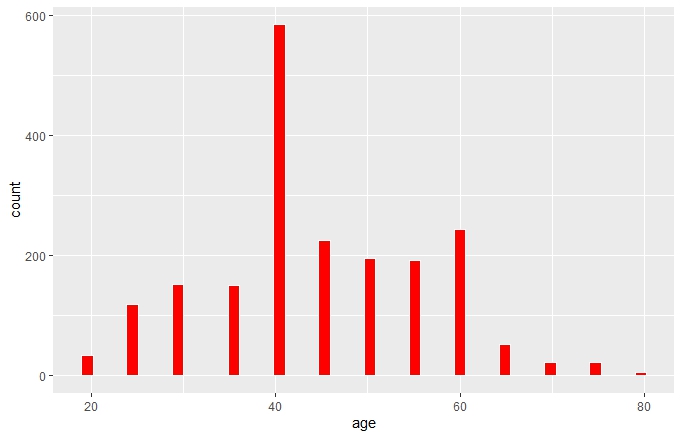


Diğer grafikler de olduğu gibi yine kullandığımız kütüphane kaynaklı “ggplot” ve “geom” fonksiyonlarını kullanmış bulunmaktayız.

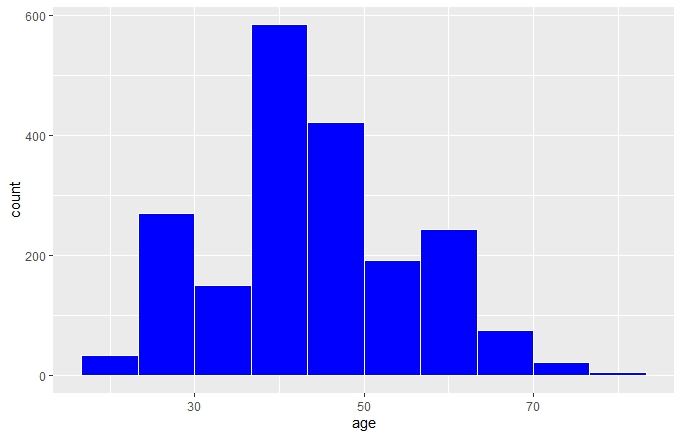


“Bins = “fonksiyonu aracılığı ile x ekseninin sayısal değerlerine düzenleme sağlanır.



“Geom” fonksiyonu içerisinde color, fill gibi yapılarla oluşturduğumuz grafiğin renklendirilebilmesine olanak sağlarız.

## 4.4.3 Grafik Çıktıları ve Yorumlama

* Yanda yer alan grafiğimizde histogram üzerinde yer alan örneklemimizdeki kişilerin yaşlarının dağılımı yer almaktadır.
* Grafiğimizin dağılımını kontrol ettiğimizde daha çok orta yaş ağırlıklı dağılım olduğunu görmekteyiz.

metin, ekran görüntüsü, yazı tipi, doküman, belge içeren bir resim

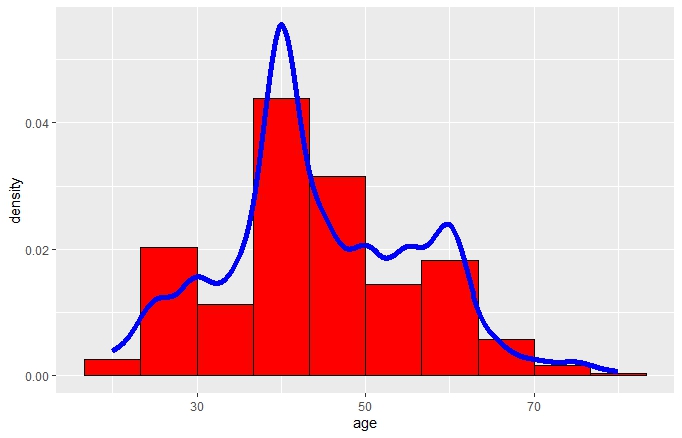
Açıklama otomatik olarak oluşturuldu

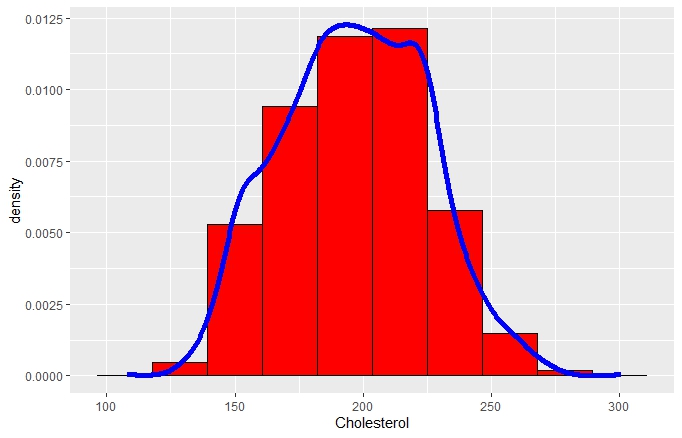
* Dağılımın çarpıklığını öğrenebilmek adına mod, medyan ve ortalama incelemesi yaparız
* Ortalama: 43.625
* Mod: 40
* Medyan: 40
* Bu sonuçlar doğrultusunda grafiğimizin sağa çarpık olduğu sonucuna ulaşabiliriz.

## 4.4.4 Adding a Density

## 

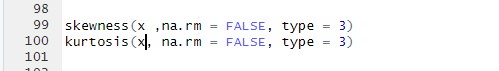
Yukarıda yer alan kodlara ekstra olarak geom\_density fonksiyonu eklenerek gerekli eklemeler yapılmakla beraber bu fonksiyonun içerisinde çizgi rengi, çizgi kalınlığı gibi ekstra eklemeler de yapılabilir.



* Yukarıda genel hatları ile verilen histogramın üzerine çizgi grafiği eklenmiştir.
* Bu oluşturulmuş olan çizgi grafiği özellikle dağılım çarpıklığı inceleme açısından daha verimli ve kullanışlı olmakla beraber bir önceki grafiğimizde mod, medyan ve ortalama hesaplamaları ile dağılımın sağa çarpık olduğunu tespit etmiştik. Şimdi oluşturduğumuz çizgi grafiği ile de sağa çarpıklık durumunu netleştirmiş bulunmaktayız.
* Farklı bir örnek de görebilmek açısından kolesterol değerlerini histogram ve çizgi grafiği incelemesine tabi tuttuğumuzda oransal ve eşit bir dağılım göze çarpmaktadır.
* Dağılım değerleri incelendiğinden;
* Ortalama :197
* Medyan: 200
* Olduğundan çok ufak da olsa bir fark ile bu dağılım sağa çarpık diyebiliriz.

NOT

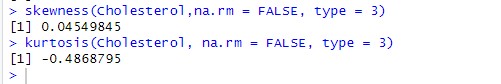
 Burada yer alan paket çağırılarak yukarıda yer alan incelemeler için çarpıklık ve basıklık katsayısı hesaplaması sağlayabiliriz.



metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* Yukarıda görmüş olduğumuz hesaplama “age” yani yaş grafiğimize ait olup hesaplamalar doğrultusunda skewness> 0 olduğundan yukarıda yapmış olduğumuz sağa çarpık sonucumuzu kanıtlamış oluruz.

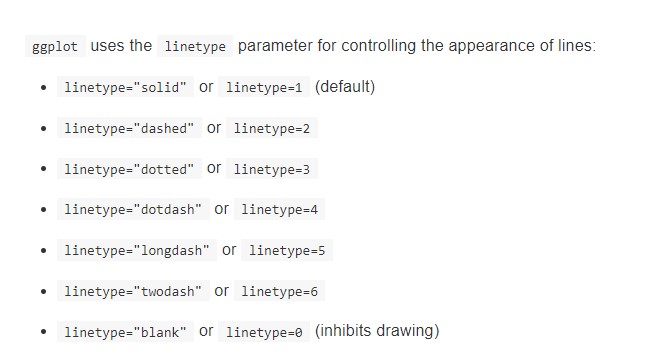


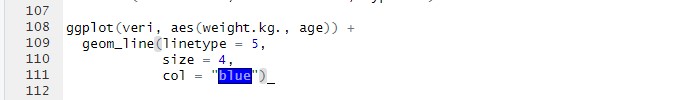
* Aynı işlemi kolesterol grafiğimizi de kanıtlamak adına uygulayabilmekteyiz.
* Yukarıda görüldüğü üzere hesaplamalarımız doğrultusunda kolesterol grafiğmizinde yaş grafiğinde olduğu gibi sağa çarpık olduğunu kanıtlamış ve göstermiş oluruz.
* Sonuç olarak gördüğümüz gibi grafik üzerinden hesaplarken basit yollardan mod ,medyan ve ortalama gibi alanlar değerler kullanırken, aynı zamanda farklı kütüphanelerden daha kısa yollarla hesaplamalarımızda gerçekleştirebilmekteyiz.

# 5.0 Changing the Type, Width, or Color of a Line

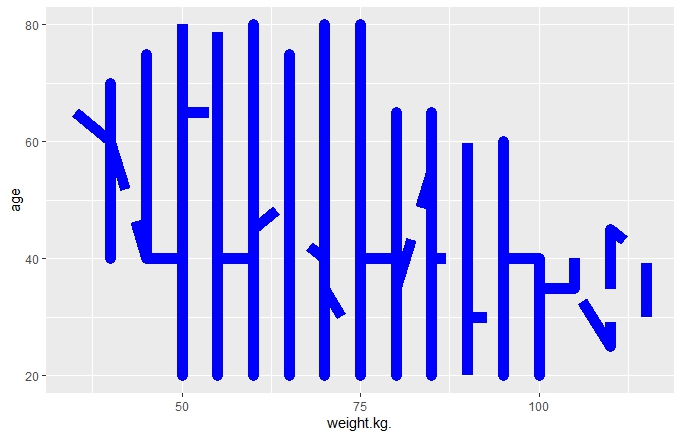
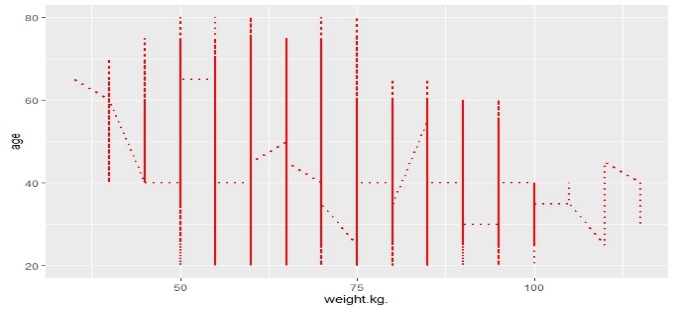
## 5.1 Line Type

* Kullandığımız veri veya oluşturduğumuz grafikler üzerinde değiştirmek isteyeceğimiz birçok şey bulunabilir. Bunlardan birisi de çizgi şekli veya stili adını verdiğimiz kısımdır.
* Öncellikle olarak direkt uygulama kütüphanesi kaynaklı kullanabileceğimiz örnek stillere yer vereceğiz.



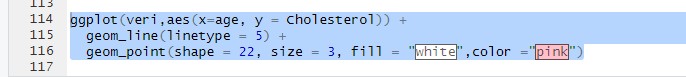


* “geom\_line” fonksiyonu içerisinde yukarıda ki resimde yer alan seçenekler üzerinden seçim yapıp aynı zamanda size, col gibi fonksiyonlarla da çeşitli boyut ve renklendirme işlemleri sağlayabiliriz.

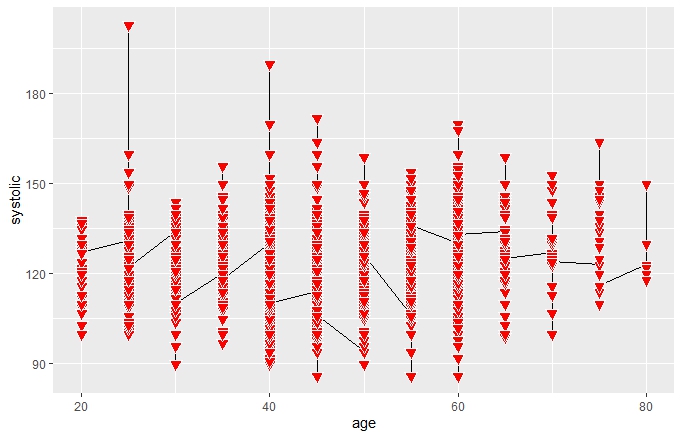
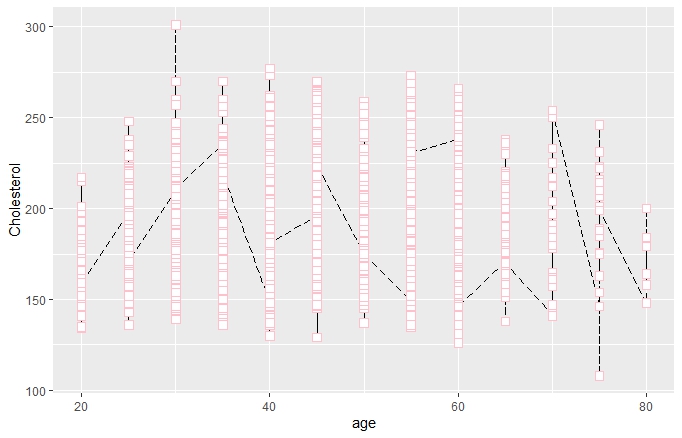


## 5.2 Change width and color

* Bir önceki 5.1 de gördüğümüz kısımda aynı zamanda çizgimizin boyut ve rengini de değiştirebildiğimizi görmüş olmakla beraber bu madde içerisinde ekstradan farklı yollarda boyut ve renk değiştirme işlemleri yapılacaktır.

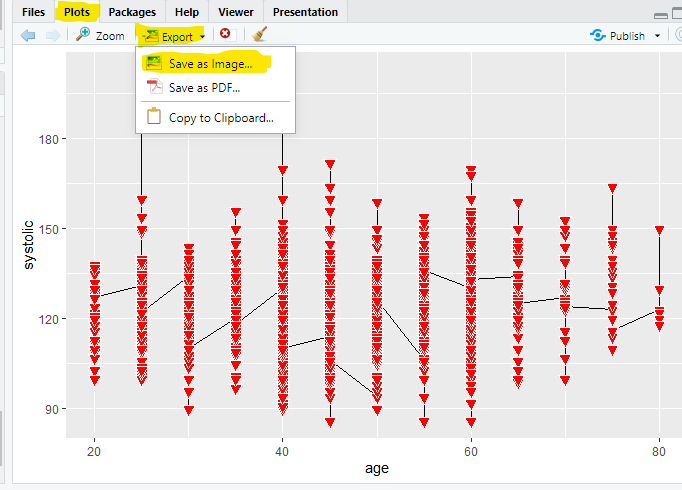


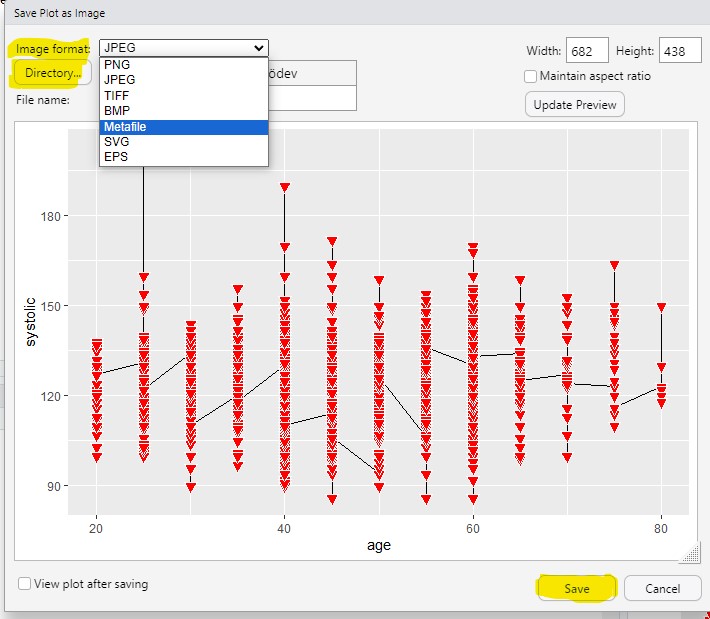
* Yer alan kodlamada da görüleceği üzeri Dbplyt kütüphanesi ve ggiraph kütüphanesi aracılığı ile geom\_line, geom\_point fonksiyonları aracılığı ile kolay bir şekilde oluşturulan grafiğimizin şekil, boyut ve renk değişimini gerçekleştirebilmekteyiz.



# 6.0 Writing Your Plot to a File

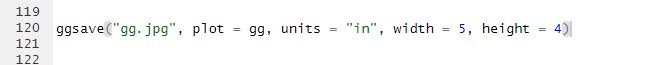
## 6.1 Uygulamadan Yazdırma



* İlk seçeneğimiz direkt olarak Rstudio uygulaması içerisindeki plots konsolu üzerinden ilerlemektir.
* Sırasıyla resimde de belirtildiği şekilde Export ve Save as Image seçenekleri takip etmemiz gerekmektedir.
* Ardından ise gelen ekran üzerinde önemli olan kısımlar özellikle sarı renk içerisine alınmış olup resmin formatı, kaydetmek istediğiniz dosya gibi alanlar seçerek ilerleyip sağ altta yer alan save seçeneği ile kaydetme işlemi başarılı bir şekilde gerçekleştirilir.

## 6.2 Kod ile Yazdırma

* Kod aracılığı ile yazdırma işleminde birçok şekil ve birçok kütüphane yer alabilir. Bunlardan birkaç örnek gösterimi sağlanılacak olup şu an gösterimi sağlanacak olan en temel kütüphane grafikte de faydalanmış olduğumuz ggiraph ve ggplot, ggplot2 kütüphanesinde yer alan “gg” fonksiyonu olacaktır.



* Ekran görüntüsünde de gördüğümüz üzere “ggsave” fonksiyonu aracılığı ile boyutu, kayıt türünü vb. bilgiler aracılığı ile grafiğimizi bilgisayarımıza kaydedebilmekteyiz.

# Kaynakça

* <https://rc2e.com/> (R Cookbook, 2nd Edition)
* <http://www.sthda.com/english/wiki/ggplot2-box-plot-quick-start-guide-r-software-and-data-visualization>
* <https://cran.r-project.org/web/packages/dplyr/vignettes/dplyr.html>
* <https://medium.com/@darnelbolaos/interactive-line-plot-with-ggplot-and-ggiraph-in-r-606d655a1547>