

‘dplyr’ Paketine Giriş: 2019 Yılı PTF - SMF Verisi

Baran Dogru

1/7/2020

Contents

1	Giriş	1
2	dplyr	1
2.1	Hazırlıklar	1
2.2	Önemli Fonksiyonlar	3
3	RStudio Cloud	7

1 Giriş

Bu dosyanın temel amacı, çok önemli bir data manipülasyonu paketi olan **dplyr** paketinin önemli fonksiyonlarına dair örnekler vererek nasıl kullanılabileceklerini göstermektir. Bu yazıda ele alınacak fonksiyonlar şu şekildedir:

```
select/rename  
filter  
distinct  
arrange  
mutate/transmute  
group_by/summarise
```

Yukarıda bahsedilen bütün fonksiyonlar ayrı ayrı ele alınacak ve her biri için örnek kullanımlar gösterilecektir. Ayrıca “pipe operator” olarak adlandırılan “Bağlantı Operatörü” (`%>%`) de kısaca anlatılacak ve bütün döküman boyunca kullanılacaktır.

2 dplyr

2.1 Hazırlıklar

Başlamadan önce yapılması gereken iki şey var. Birincisi **dplyr** paketini indirmek ve `setwd()` fonksiyonunu kullanarak çalışma dizinini belirlemek. Paketi indirmek için `install.packages("dplyr")` , paketi yüklemek için ise `library(dplyr)` komutları kullanılabilir.

İkinci hazırlık aşaması ise, EPİAŞ’ın Raporlama Sayfası ’ından 2019 yılı Piyasa Takas Fiyatı (PTF) - Sistem Marjinal Fiyatı (SMF) dosyasının .xls formatında belirlediğimiz çalışma dizini dosyasına indirilmesi ve okunması.

```
# dplyr paketinin yüklenmesi  
library(dplyr)  
# 2019 yılı PTF - SMF verisinin okunması  
ptfsmf <- read_excel("ptf-smf.xls")
```

Tablonun ilk haline `print()` fonksiyonu ile göz atılabilir.

```
print(ptfsmf)
```

```
##           Tarih    PTF    SMF Pozitif Dengesizlik Fiyatı (TL/MWh)
## 1: 01.01.19 00:00 100,38    5,00                                4,85
## 2: 01.01.19 01:00  96,72   95,04                                92,19
## 3: 01.01.19 02:00  81,60   79,60                                77,21
## 4: 01.01.19 03:00  38,58    0,00                                0,00
## 5: 01.01.19 04:00  11,52    0,00                                0,00
## ---
## 8756: 31.12.19 19:00 327,82 241,18                                233,94
## 8757: 31.12.19 20:00 319,57 221,00                                214,37
## 8758: 31.12.19 21:00 316,54 196,18                                190,29
## 8759: 31.12.19 22:00 314,70 195,00                                189,15
## 8760: 31.12.19 23:00 311,84 195,00                                189,15
##           Negatif Dengesizlik Fiyatı (TL/MWh)    SMF Yön
## 1:                                103,39 ↓Enerji Fazlası
## 2:                                99,62 ↓Enerji Fazlası
## 3:                                84,05 ↓Enerji Fazlası
## 4:                                39,74 ↓Enerji Fazlası
## 5:                                11,87 ↓Enerji Fazlası
## ---
## 8756:                                337,65 ↓Enerji Fazlası
## 8757:                                329,16 ↓Enerji Fazlası
## 8758:                                326,04 ↓Enerji Fazlası
## 8759:                                324,14 ↓Enerji Fazlası
## 8760:                                321,20 ↓Enerji Fazlası
```

Aynı işlem `glimpse()` fonksiyonu ile de yapılabilir.

```
ptfsmf %>% glimpse()
```

```
## Observations: 8,760
## Variables: 6
## $ Tarih           <chr> "01.01.19 00:00", "01.01.19 0...
## $ PTF             <chr> "100,38", "96,72", "81,60", "...
## $ SMF             <chr> "5,00", "95,04", "79,60", "0,...
## $ `Pozitif Dengesizlik Fiyatı (TL/MWh)` <chr> "4,85", "92,19", "77,21", "0,...
## $ `Negatif Dengesizlik Fiyatı (TL/MWh)` <chr> "103,39", "99,62", "84,05", "...
## $ `SMF Yön`       <chr> "↓Enerji Fazlası", "↓Enerji F...
```

Fark edileceği üzere, bu fonksiyon tabloda var olan sütunları satırlarda gösterirken değişken tiplerinin ne olduğuna ve totalde kaç gözlem olduğuna da değiniyor. Burada “Bağlantı Operatörü”nün kullanımına dikkat edilmelidir. Bu operatör sayesinde kullanılan `data.table` tek bir kez yazılarak istenildiği kadar işlem yapılabilir ve bu sayede daha tertipli bir koda ulaşılabilir.

Bu aşamada karakter tipindeki değişkenlerin tarih sütunu için “POSIXct” (`datetime`), geri kalanlar sütunlar için ise “numeric” tipine dönüşümü sağlanmıştır. Bu dökümanın amacı bu dönüşümleri göstermek olmadığından bahsedilen önemli fonksiyonlar tablonun düzenlenmiş hali üzerinden anlatılacaktır. (İlgilenenler için `data` manipülasyonu aşamasındaki kodlara buradan ulaşılabilir.)

Şimdi tablonun bahsedilen düzenlenmiş haline bir kez daha göz atacak olunursa,

```
ptfsmf %>% glimpse()
```

```
## Observations: 8,760
## Variables: 5
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
```

2.2 Önemli Fonksiyonlar

2.2.1 select/rename

`select` fonksiyonu belirli sütunları seçmek için kullanılır. Örneğin tablonun sadece Tarih ve Piyasa Takas Fiyatı'nı (PTF) göstermesi isteniyorsa,

```
ptfsmf %>% select(Tarih, PTF) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 2
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
```

Veya (bu örnek için pek faydalı gibi görünmese de) içinde “P” harfini barındıran sütunlar seçilmek istendiğinde `contains` kelimesi kullanılabilir.

```
ptfsmf %>% select(contains("P")) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 2
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.50,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0...
```

Bu kelime gibi `select` fonksiyonu içinde kullanılabilecek diğer kelimeler: `starts_with`, `ends_with`, `matches` olarak sıralanabilir.

Aynı zamanda, eğer sütunların sırası biliniyorsa iki sütun arasındaki her sütunu seçmek için `:` operatörü kullanılabilir. Örneğin Sistem Marjinal Fiyatı (SMF) ve Negatif Dengesizlik Fiyatı (NDF) arasındaki sütunları seçmek için,

```
ptfsmf %>% select(SMF:NDF) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 3
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.53,...
```

`rename` fonksiyonu ise adından da anlaşılacağı üzere bir sütunun adını değiştirmek için kullanılır. Örneğin “PTF” sütununun ismini “ptf” olarak değiştirmek isteniyorsa `rename` kullanılabilir.

```
ptfsmf %>% rename(ptf = PTF) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 5
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ ptf <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
```

2.2.2 filter

`filter` fonksiyonu temel olarak istenilen koşulları sağlayan satırları seçmek için kullanılır. Örneğin Piyasa Takas Fiyatı'nın 250'den düşük olduğu satırlar incelenebilir.

```
ptfsmf %>% filter(PTF < 250) %>% glimpse()
```

```
## Observations: 2,453
## Variables: 5
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
```

Gözlem sayısının 8670'den 2453'e düştüğü görülebiliyor. Aynı şekilde bir değerin büyüklüğü $>$, küçüklük veya eşitliği \leq , büyüklük veya eşitliği \geq , eşitliği ise $==$ sembolleriyle incelenebilir.

Birden fazla koşulun sağlanması gerekiyorsa ise, "VE" operatörü için $\&$, "VEYA" operatörü için $|$ sembolleri kullanılmalıdır. Örneğin Pozitif Dengesizlik Fiyatı'nın 200'den küçük, Negatif Dengesizlik Fiyatı'nın ise 200'den büyük olduğu satırlar incelenebilir.

```
ptfsmf %>% filter(PDF<200 & NDF>200) %>% glimpse()
```

```
## Observations: 1,256
## Variables: 5
## $ Tarih <dtm> 2019-01-01 17:00:00, 2019-01-01 18:00:00, 2019-01-01 21:00:0...
## $ PTF <dbl> 286.93, 291.52, 285.18, 204.89, 287.66, 296.99, 296.86, 293.1...
## $ SMF <dbl> 172.00, 172.00, 173.00, 172.98, 168.60, 173.00, 190.00, 190.0...
## $ PDF <dbl> 166.84, 166.84, 167.81, 167.79, 163.54, 167.81, 184.30, 184.3...
## $ NDF <dbl> 295.54, 300.27, 293.74, 211.04, 296.29, 305.90, 305.77, 301.9...
```

Bu koşulları sağlayan 1256 satır olduğu görülüyor. Eğer bir koşulu sağlamayan satırlar aranıyorsa ise koşulun başına $!$ yazılmasıyla bu ters etki taratılabilir.

2.2.3 distinct

`distinct` fonksiyonu bir veya birden fazla sütunu baz alarak özgün satırları bulmak için kullanılır. Örneğin var olan tabloda her bir saat için 2 tane satır var ve bunlardan kurtulmak isteniyor. Öncelikle tabloya göz atılırsa,

```
ptfsmf %>% head(8) # head fonksiyonu da print ve glimpse fonksiyonlari ile benzer islevde
```

```
##           Tarih    PTF    SMF    PDF    NDF
## 1 2019-01-01 00:00:00 100.38  5.00  4.85 103.39
## 2 2019-01-01 00:00:00 100.38  5.00  4.85 103.39
## 3 2019-01-01 01:00:00  96.72 95.04 92.19  99.62
## 4 2019-01-01 01:00:00  96.72 95.04 92.19  99.62
## 5 2019-01-01 02:00:00  81.60 79.60 77.21  84.05
## 6 2019-01-01 02:00:00  81.60 79.60 77.21  84.05
## 7 2019-01-01 03:00:00  38.58  0.00  0.00  39.74
## 8 2019-01-01 03:00:00  38.58  0.00  0.00  39.74
```

Bu durumdan kurtulmak için `distinct` fonksiyonu kullanılabilir. Örneğin,

```
ptfsmf %>% distinct(Tarih) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 1
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
```

Dikkat edileceği üzere yalnızca Tarih sütunu korundu. Diğer sütunları da korumak için `.keep_all = TRUE` komutu kullanılmalıdır.

```
ptfsmf %>% distinct(Tarih, .keep_all = TRUE) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 5
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
```

2.2.4 arrange

`arrange` fonksiyonu Excel'deki sıralama özelliğine benzetilebilir. Varsayılan durumunda A'dan Z'ye ya da küçükten büyüğe sıralanır. Tam tersi sıralama için `desc()` fonksiyonu kullanılmalıdır. Örneğin artan Piyasa Takas Fiyatlarına göre sıralamak için,

```
ptfsmf %>% arrange(PTF) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 5
## $ Tarih <dtm> 2019-02-17 09:00:00, 2019-03-24 12:00:00, 2019-03-24 13:00:0...
## $ PTF <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ SMF <dbl> 30, 5, 0, 0, 0, 10, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ PDF <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ NDF <dbl> 30.90, 5.15, 0.00, 0.00, 0.00, 10.30, 0.00, 0.00, 0.00, 0.00,...
```

Bir başka örnekte Tarih, Pozitif Dengesizlik Fiyatı (PDF) ve Negatif Dengesizlik Farkı (NDF) sütunları seçilip azalan NDF'ye göre sıralanabilir.

```
ptfsmf %>% select(Tarih, PDF, NDF) %>%
  arrange(desc(NDF)) %>%
  glimpse()
```

```
## Observations: 8,760
## Variables: 3
## $ Tarih <dtm> 2019-06-27 15:00:00, 2019-06-27 16:00:00, 2019-06-27 14:00:0...
## $ PDF <dbl> 436.50, 436.50, 453.74, 385.24, 303.44, 307.22, 369.73, 339.5...
## $ NDF <dbl> 515.00, 514.99, 502.40, 501.76, 462.47, 449.82, 438.96, 432.6...
```

Dikkat edileceği üzere NDF değerleri azalarak devam ediyor. Ayrıca `arrange` fonksiyonunun içine birden fazla değer girerek ilk değerın eşitliği durumunda ikinci değerin karar vermesi sağlanabilir.

2.2.5 mutate/transmute

`mutate` fonksiyonu genellikle var olan değişkenlerle yapılan operasyonlar sonucu yeni değişkenler (sütunlar) yaratmak için kullanılır. Örneğin Pozitif Dengesizlik Fiyatı (PDF) ve Negatif Dengesizlik Fiyatı (NDF) arasındaki Dengesizlik Fiyatı (DF) sütununda gösterilmek istenirse,

```
ptfsmf %>% mutate(DF = PDF - NDF) %>% glimpse()
```

```
## Observations: 8,760
## Variables: 6
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
## $ DF <dbl> -98.54, -7.43, -6.84, -39.74, -11.87, -11.47, -11.47, -25.10,...
```

Bu fonksiyonun içerisinde base R fonksiyonları da kullanılabilir. Örneğin yeni bir Denge sütununda PDF NDF'den büyükse "Pozitif", tam tersiye "Negatif, eşitlik varsa ise"Dengeli" yanıtı alabilmek için,

```
ptfsmf %>% mutate(Denge = ifelse(PDF>NDF,"Pozitif","Negatif"),
  Denge = ifelse(PDF==NDF,"Dengeli",Denge)) %>%
  glimpse()
```

```
## Observations: 8,760
## Variables: 6
## $ Tarih <dtm> 2019-01-01 00:00:00, 2019-01-01 01:00:00, 2019-01-01 02:00:0...
## $ PTF <dbl> 100.38, 96.72, 81.60, 38.58, 11.52, 11.14, 11.14, 24.37, 34.5...
## $ SMF <dbl> 5.00, 95.04, 79.60, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ PDF <dbl> 4.85, 92.19, 77.21, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,...
## $ NDF <dbl> 103.39, 99.62, 84.05, 39.74, 11.87, 11.47, 11.47, 25.10, 35.5...
## $ Denge <chr> "Negatif", "Negatif", "Negatif", "Negatif", "Negatif", "Negat...
```

`transmute` fonksiyonu da `mutate` fonksiyonu ile benzer bir işleve sahip olmasının yanında `select` fonksiyonunun belirli sütunları seçme işlevine de sahiptir. Örneğin PTF/SMF oranını Tarih'in de gösterileceği şekilde sergileyip bu orana göre artan şekilde sıralamak için,

```
ptfsmf %>% transmute(Tarih, PTF, SMF, PTF_SMF_Oran = PTF/SMF) %>%
  arrange(PTF_SMF_Oran) %>%
  glimpse()
```

```
## Observations: 8,760
## Variables: 4
## $ Tarih      <dtm> 2019-02-17 09:00:00, 2019-03-24 12:00:00, 2019-03-24 ...
## $ PTF        <dbl> 0.00, 0.00, 0.00, 2.00, 1.00, 0.99, 0.99, 1.00, 1.01, ...
## $ SMF        <dbl> 30.00, 5.00, 10.00, 244.00, 100.00, 90.99, 75.99, 76.0...
## $ PTF_SMF_Oran <dbl> 0.000000000, 0.000000000, 0.000000000, 0.008196721, 0....
```

2.2.6 group_by/summarise

Bu iki fonksiyon çoğunlukla özetleme tabloları çıkarmak için kullanılır. `group_by` sütunlara göre veriyi gruplamak, `summarise` ise gruplanan bu verilere göre istenen özeti çıkarmakla görevlidir. Örneğin günlük ortalama PTF fiyat bilgisi için (Burada “POSIX” formatında olan Tarih sütununu “Date” formatına çevirebilmek için `as.Date()` fonksiyonu kullanılmıştır.),

```
gunluk_ort_ptf <- ptfsmf %>%
  mutate(Gun = as.Date(Tarih)) %>%
  group_by(Gun) %>%
  summarise(Gunluk_Ort_PTF = mean(PTF)) %>%
  glimpse()
```

```
## Observations: 365
## Variables: 2
## $ Gun        <date> 2019-01-01, 2019-01-02, 2019-01-03, 2019-01-04, 201...
## $ Gunluk_Ort_PTF <dbl> 121.0229, 228.7592, 238.6304, 212.3496, 244.3354, 23...
```

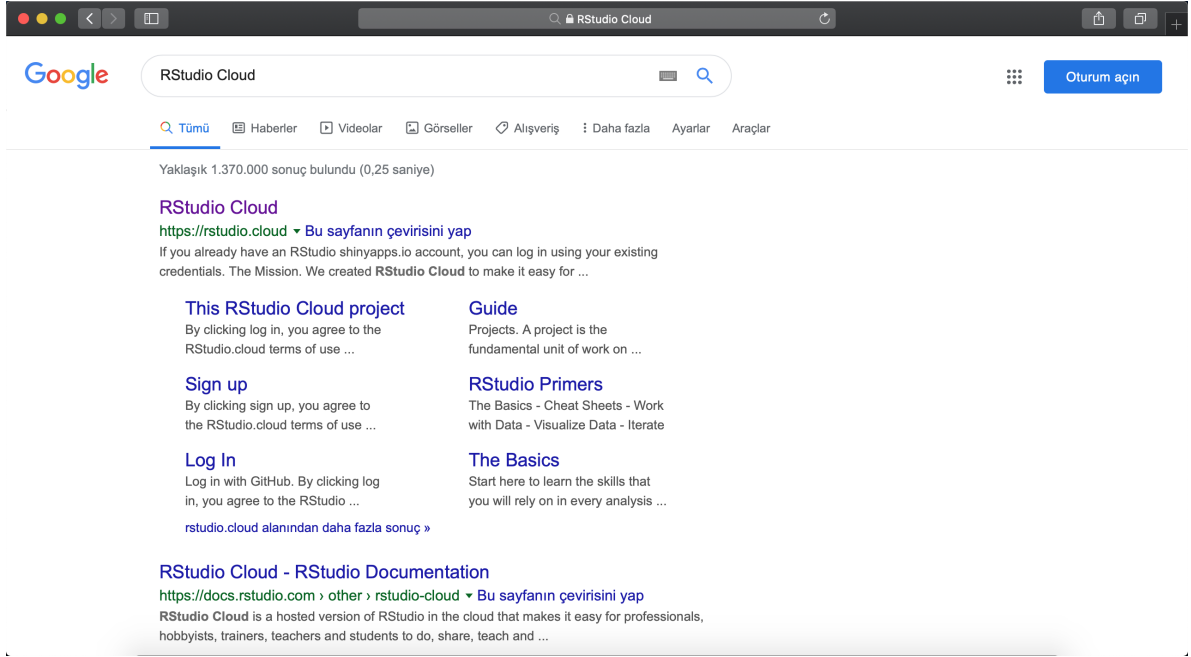
Burada kullanılan `mean` fonksiyonu dışında maksimum değer için `max`, minimum değer için `min`, medyan için `median`, total değer için `sum` ve grupladığımız değişkene ait gözlem sayısını saymak için ise `n()` fonksiyonları kullanılabilir.

3 RStudio Cloud

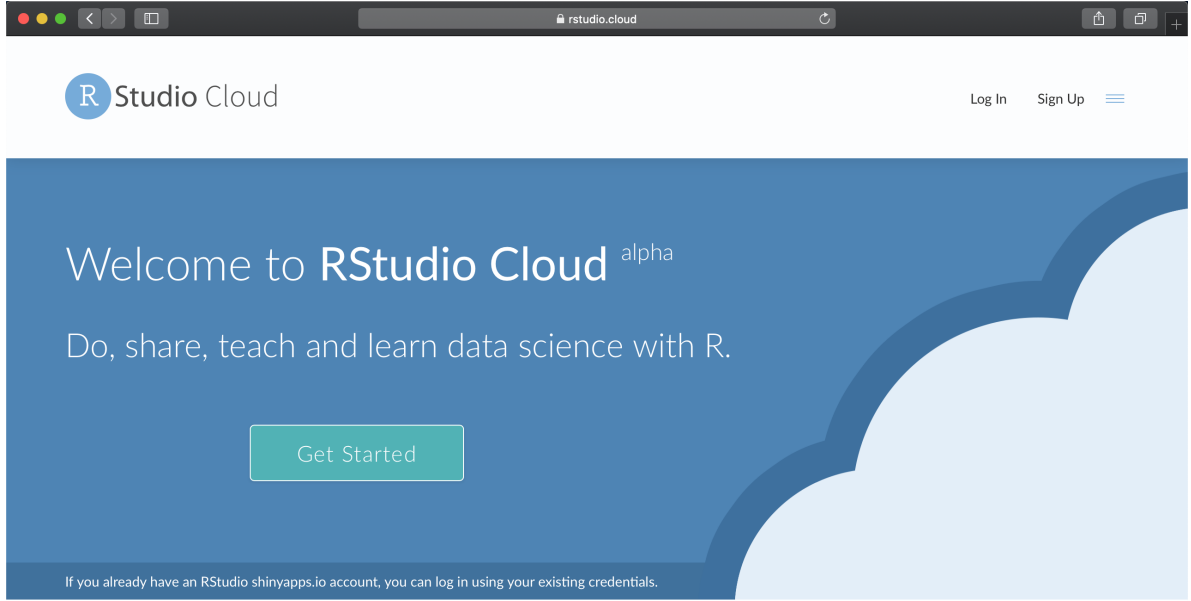
RStudio Cloud, R ve RStudio’yu bilgisayarınıza indirmeden çevrimiçi olarak kodlarınızı yazabileceğiniz, çalışmalarınızı is arkadaşlarınızla rahatça paylaşabileceğiniz, gerekli paketleri bilgisayarınıza yükleyip yüklemediğinizi dert etmeyeceğiniz tamamen ücretsiz bir platformdur.

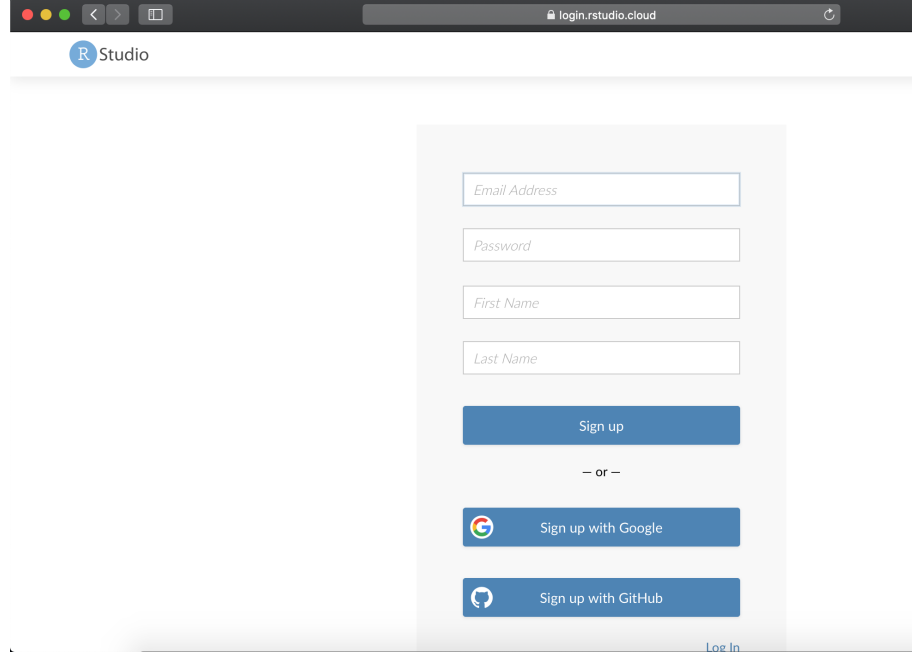
RStudio Cloud’a erişmek için aşağıdaki adımları izleyebilirsiniz.

1. Tercih ettiğiniz web tarayıcıyı açın ve “RStudio Cloud” yazarak aratın. En üstte çıkan linke tıklayın.



2. Açılan ana ekranın sağ üst köşesindeki “Sign Up” butonuna tıklayarak kayıt ekranına ulaşın.





login.rstudio.cloud

R Studio

Email Address

Password

First Name

Last Name

Sign up

— or —

Sign up with Google

Sign up with GitHub

Log In

3. Karşınıza çıkan kayıt formunu doldurun. .
4. Kaydınızı tamamlayın ve hesabınıza giriş yapın. Artık RStudio Cloud'u rahatça kullanabilirsiniz.

