**4.1.Servet dağılım analizi: Milyarderlerin servetinin farklı sektörlere, ülkelere ve bölgelere dağılımını keşfedin.**

dplyr, veri çerçeveleri üzerinde veri manipülasyonu ve filtreleme işlemleri için tasarlanmıştır. Bu paket, filter(), select(), mutate(), group\_by() gibi fonksiyonlar aracılığıyla veri setlerini hızlı ve anlaşılır bir şekilde işlemenizi sağlar.

tidyr, veri setlerini düzenleme ve temizleme işlemleri için kullanılır. Bu paket, özellikle veri çerçevelerindeki geniş formatlı veriyi daha uzun formatlı bir hale getirme veya tam tersini yapma (veri normalleştirme) gibi işlemleri gerçekleştirmek için tasarlanmıştır.

Veri görselleştirmesi için Rstudioda ggplot2 paketi kullanılır. Paketi indirdikten sonra library(ggplot2) ile paketi kullanıma hazır hale getiririz.

Geom\_col ile sütun grafiği çizdirdim. Show.legend=FALSE ile lejantları göstermesini istemedim.

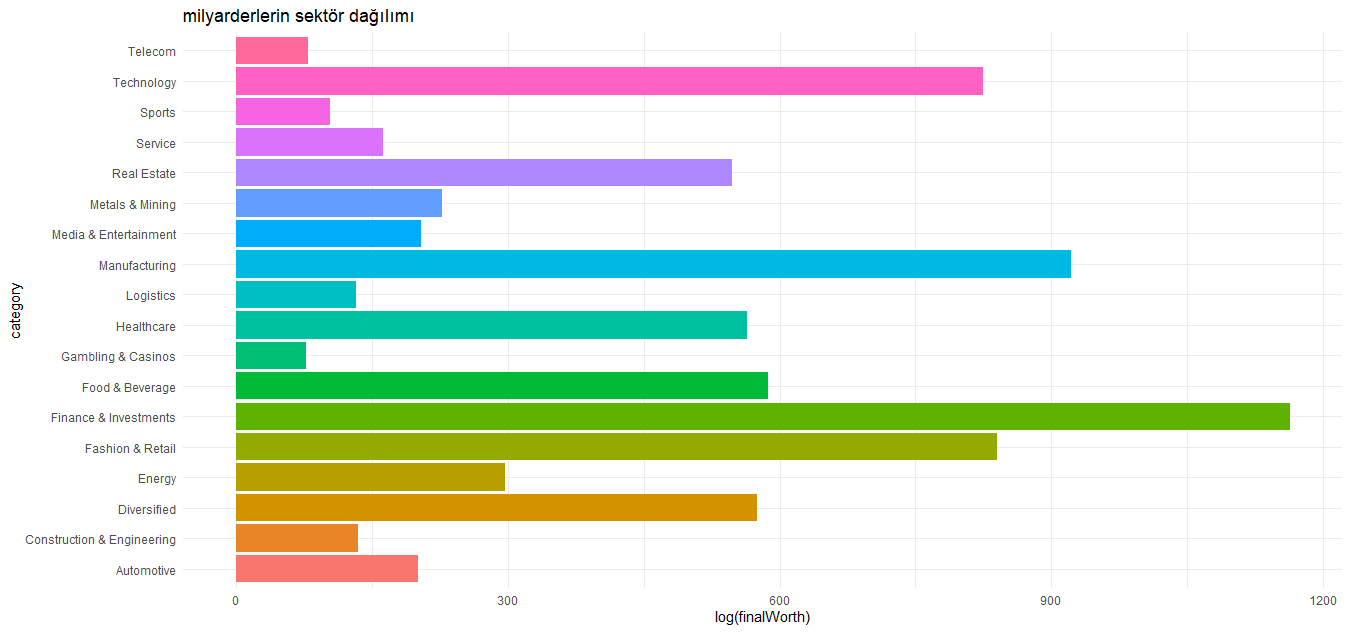
Coord\_flip() ile eksenlerin yerini değiştirdim. Böylece grafik daha okunaklı oldu.

Theme\_minimal() ile arka plan temasını seçtim.

**Sektör:**

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu



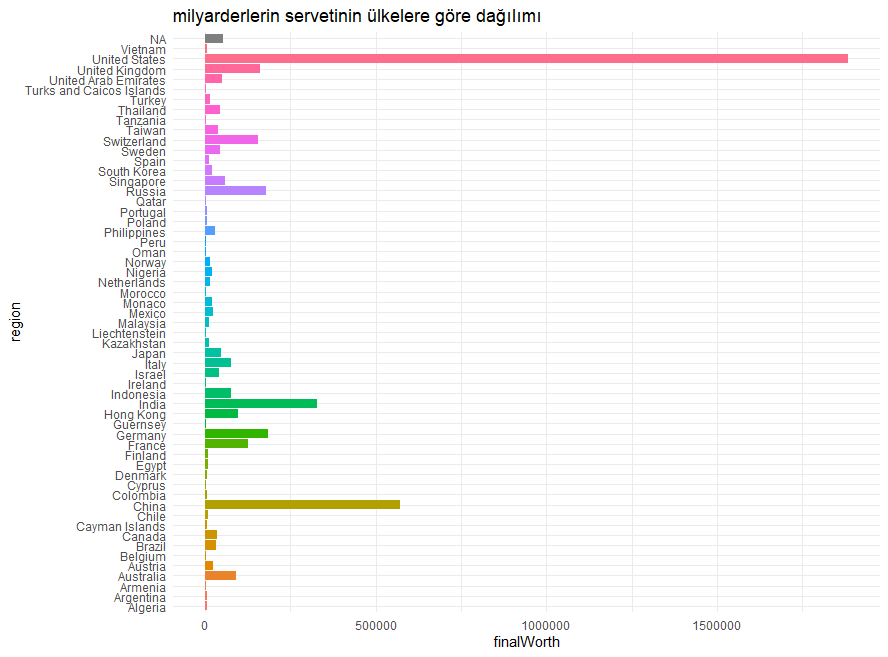
Çıktıya göre milyarder servetinin en fazla olduğu sektör finance & ınvestments (finans ve yatırım). Yani en çok bu sektörde iş yapan kişiler daha zengin olmuş denebilir. En az ise gambling and casinos (kumar ve kumarhaneler) olduğunu görüyoruz.

**Ülke:**

Billionaries veri setinde “county” sütunun adını 4.5 ‘deki soruda sorun yaşamamız nedeniyle “region” olarak değiştirdim.

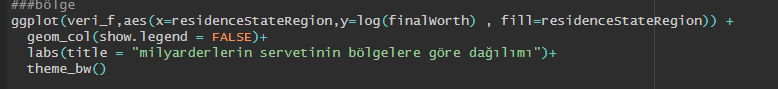
**metin, ekran görüntüsü, yazı tipi içeren bir resim

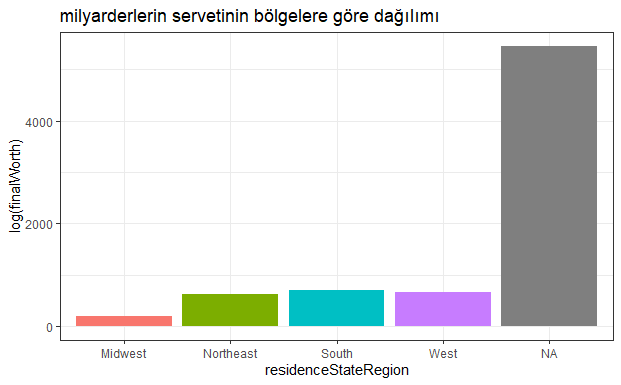
Açıklama otomatik olarak oluşturuldu**

****

Çıktıya göre milyarderlerin servetinin en fazla olduğu United States. Yani Amerika Birleşik Devletlerinde yaşayan milyarderlerin kazandığı servet daha fazla.

**Bölgelere göre:**

****

****

Çıktıya göre güney, batı ve kuzeydoğu bölgesinde yaşayan milyarderlerin serveti orta batıda yaşayan milyarderlerin servetinden daha fazla.

**4.3.Kendi kendine yapılan ve miras alınan zenginlik: Kendi kendine milyarder olanların ve onların servetini miras alan kişilerin oranını analiz edin.**

veri3\_soru adında bir veri çerçevesi oluşturduk. Bu çerçeve, "category" adında bir sınıf ve "count" adında bir sayım içeriyor. "category" sütunu "Kendi Milyarder" ve "Miras Alınan Zenginlik" olarak iki kategori içerirken, "count" sütunu bu kategorilere ait kişi sayılarını içerir. veri\_f verisinden "status" sütunundaki değerlere göre kişi sayılarını hesaplayarak oluşturuluyor. Status sütunuda D kendi kendine edinilen zenginlik , U miras alınan zenginliği ifade ediyor. summarise() fonksiyonu ile veri setlerini özetleyebilir ve arrange() ile sıralama işlemleri gerçekleştirebilirsiniz.

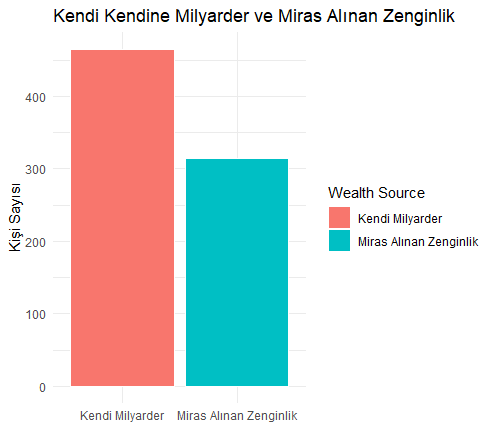
geom\_bar() fonksiyonu, çubuk grafiğini oluşturmak için kullanılır. "stat = "identity"" parametresi, veri çerçevesindeki "count" sütununun direkt olarak çubuk yüksekliklerin

karşılık gelmesini sağlar. Veri3\_soru değişkeninden category ve count değişkenlerini aes içine atadım.

**metin, yazılım, yazı tipi, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Bar grafiği:**

****

Kendi kendine milyarder olan kişi sayısı daha fazladır.

**Pasta grafiği:**

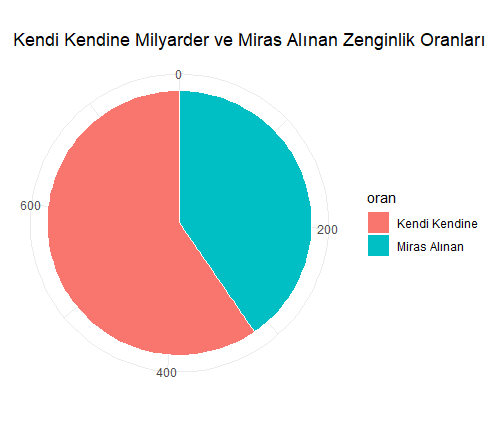
**metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

İlk olarak, "Kendi kendine Milyarder(D)" ve "Miras Alınan Zenginlik(U)" kategorilerine ait kişi sayılarını toplanıp , **veri\_yeni** adında yeni bir veri çerçevesi oluşturdum.

**coord\_polar("y")** fonksiyonu, polar koordinatlarda bir grafik oluşturulmasını sağlar, bu da pasta grafiğini oluşturur. Yani bar grafiğinden pasta grafiğine dönüşüyor.

**theme(legend.position = "right")** ise grafikteki lejantın sağ üst köşeye yerleştirilmesini sağlar.

****

**metin, yazı tipi, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

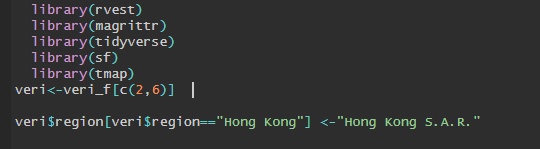
Kendi kendine milyarder olan kişi oranı 45.6 iken miras ile milyarder olan kişi oranı 34.3’tür.

**4.5.Mekansal analiz: Milyarderlerin coğrafi dağılımını ve servetlerini bir harita üzerinde görselleştirin.**

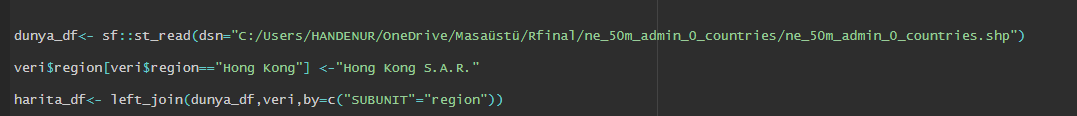
**1.yol:**

Gerekli kütüphaneleri yükledim. Veri\_f veri setinin sadece iki sütunuyla(finalworth ve region) ilgilendiğimiz için o ikisini seçip “veri” adlı değişkene atadım.

**1)**

****

**2)**

****

**Rvest** paketi ,web sitesinden bilgiyi almak için yani web sitesinden bilgiyi “scrape(kazımak)” ediyoruz.

**Magrittr** paketi, veri manipülasyonunu daha okunabilir ve zincirleme tarzında (chaining) yapmayı sağlayan bir pakettir. Bu paket kodun daha anlaşılır hale gelmesini ve işlemlerin sıralı bir şekilde gerçekleştirilmesini sağlar. Burada gerekli düzenleme yapmak için kullandım.

**Tidyverse** paketini bilgileri filtreleme yapmak için kullandım.

**Sf** paketi yani shape file dediğimiz haritalarda şekil dosyalarını oluşturması için işlem gördüğümüz kütüphanedir.

**tmap** paketi, tematik haritalar oluşturmak ve görselleştirmek için kullanılan bir pakettir. Bu paket, harita oluşturma sürecini kolaylaştırmak ve çeşitli özelleştirme seçenekleri sunmak amacıyla geliştirilmiştir. Tmap paketi, tematik haritalar oluşturmak için **tmap** ve **tm** adlı iki ana sınıf içermektedir.

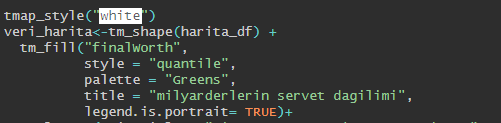
**dunya\_df<sf::st\_read(dsn="C:/Users/HANDENUR/OneDrive/Masaüstü/Rfinal/ne\_50m\_admin\_0\_countries/ne\_50m\_admin\_0\_countries.shp"):** sf paketinden st\_read isimli fonksiyonu ile coğrafi verileri çeşitli formatlardan okumak için çağırdık. Dsn, veri kaynağının adresini belirtir. Oluşturulan bu veriyi dunya\_df değişkene atadım.

Bu dunya\_df verisinde ülkelerin isimleri,ölçüleri,konumları ,sınırları gibi ülkelerle alakalı coğrafik verileri gösterir.

**veri$region[veri$region=="Hong Kong"] <-"Hong Kong S.A.R." :** burada “veri” isimli verimiz ile “dunya\_df “ verisindeki ülke isimlerini (subunit sütunu) ile karşılaştırdım ve aynı olmayanları düzenleyip aynı isim ile gösterilmesini sağladım. Yani “veri “deki Hong Kong ismi yerine Hong Kong S.A.R. yazdırdım.

**harita\_df<- left\_join(dunya\_df,veri,by=c("SUBUNIT"="region")):**  **dplyr** paketinin **left\_join** fonksiyonunu kullanarak iki veri çerçevesini birleştirme işlemi gerçekleştirir. İki veri çerçevesi arasında birleştirme işlemi, **SUBUNIT** sütunundaki değerlerle **region** sütunundaki değerlerin eşleştirilmesine dayanır. Yani dünya veri çerçevesinden (dunya\_df) hepsini al, “veri ”den ise eşleşenleri al ve birleştir. Ama SUBUNIT=region’ a göre eşleştir.

3)



**tmap\_style("white")**: Bu kod satırı, harita stili için kullanılacak temayı belirler. Haritanın arka plan rengini beyaz olarak ayarlar.

Veri haritamızı oluşturmak istediğimizde, tm\_shape fonksiyonu ile daha önce tanımladığımız harita\_df’yi aldık. Böylece tm kütüphanesi bana bir veri haritası oluştuşturacak. Enlem ve boylamlarını yerleştirmesi gerekiyor.

Tm\_fill , içini “finalworth” ile dolduracak. Stil, renk paleti,başlık gibi farklı parametreler kullandık. **style = "quantile**.” değerlerin yüzdelik dilimlerine göre renklendirme yapar. **legend.is.portrait = TRUE**: Bu, harita üzerindeki renk skalasının dikey (portrait) olarak görüntülenmesini sağlar.

4)

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

Açıklama otomatik olarak oluşturuldu

* **main.title**: Haritanın başlığını belirtir.
* **main.title.position**: Başlığın pozisyonunu belirtir. "center" değeri, başlığın haritanın ortasında yer almasını sağlar.
* **main.title.size**: Başlık font boyutunu belirtir.
* **legend.height** ve **legend.width**: Bu değerler, renk skalasının boyutunu kontrol eder.
* **legend.outside**: Renk skalasının harita dışında olup olmamasını belirtir. **FALSE** değeri, renk skalasının harita içinde yer alacağını gösterir.
* **legend.position**: Renk skalasının konumunu belirtir. Bu durumda, sol alt köşede (**"left", "bottom"**) yer alacaktır.
* **legend.frame**:Renk skalasının etrafındaki çerçevenin kullanılmasını istediğimiz için **TRUE** yazdık.
* **legend.bg.color** ve **legend.bg.alpha**: Renk skalası arka planının rengi ve şeffaflığı belirtilir.
* **bg.color**: Haritanın genel arka plan rengini belirtir. Bu durumda, "skyblue" olarak ayarlanmıştır.
* **inner.margins**: Harita içerisindeki kenar boşluklarını belirtir. Bu değerler, sırasıyla sol, üst, sağ ve alt kenarlardaki boşlukları temsil eder.

Bütün bu değerler tek tek denenerek özelleştirilmiş ve son hali paylaşılmıştır.

5)

metin, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

**tm\_borders** fonksiyonu, harita üzerindeki ülke sınırlarını çizmek için kullanılır. **alpha** parametresi, sınırların şeffaflığını belirler. Burada **alpha = 0.5** seçerek sınırların yarı şeffaf bir şekilde görüntülenmesini sağladım.

**tm\_text("ISO\_A3", size = "AREA")**: Bu satır, harita üzerine ülke etiketlerini ekler. **tm\_text** fonksiyonu, "ISO\_A3" sütunundaki değerleri harita üzerine yazdırmak için kullanılır. **size = "AREA"** parametresi, etiketlerin boyutunu, harita üzerindeki alanları temsil eden bir özellik olan "AREA" sütununa göre belirler. Yani, büyük ülkelerin etiketleri daha büyük, küçük ülkelerin etiketleri daha küçük olacaktır.

6)



**tm\_grid(alpha = 0.2)**: harita üzerine ızgara eklemek için kullanılır. **alpha = 0.2** seçerek, ızgaranın şeffaflığını belirledik.Burada ızgara, haritanın altında hafif şeffaflıkla görünür olacaktır.

Tm\_compass ile de hairtaya istediğimiz şekil ve formatta pusula ekleriz.

* **type = "8star"**: Pusula tipini belirtir. Burada "8star", sekiz yıldızlı bir pusula kullanılacağını gösterir.
* **position = c(.65, .15)**: Pusulanın konumunu belirtir. İki elemanlı bir vektör kullanılarak pusulanın x ve y koordinatları belirlenir. Burada, pusula sağ üst köşede (**x = 0.65, y = 0.15**) görünecektir.
* **size = 3**: Pusula boyutunu belirtir.
* **color.light = "grey90"**: Pusula üzerindeki ışıkların rengini belirtir. Burada gri tonları kullanılır.

7)

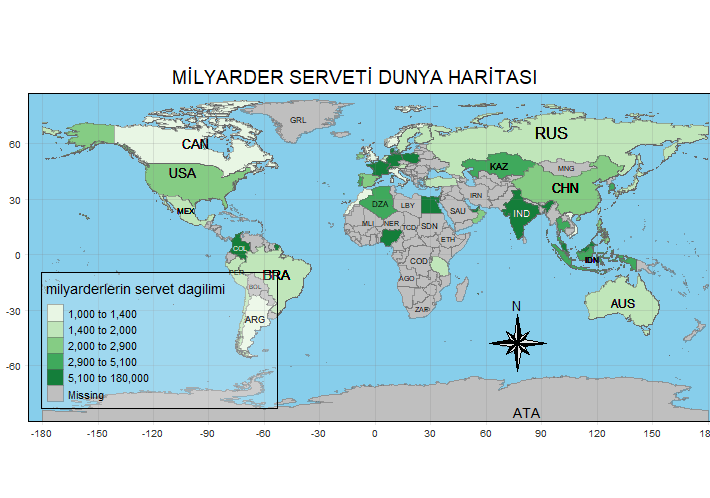
metin, yazı tipi, ekran görüntüsü, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Haritayı plot(sabit bir çizim) şeklinde ya da interaktif şekilde göstermemizi sağlayan kod satırlarıdır.

Plot modunu çalıştıralım: (Grafiklerin çalışması biraz zaman alacaktır.)





İnteraktif hale getirelim:



metin, harita, diyagram, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Dilediğimiz gibi yakınlaşabilir, istediğimiz ülkeyi inceleyebiliriz.

metin, ekran görüntüsü, harita, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Kaynakça:**

* <https://campus.datacamp.com/courses/>
* <https://www.data-to-viz.com/>
* <https://bookdown.org/ugurdar/rileverigorsellestirme/ggplot2-k%C3%BCt%C3%BCphanesi.html>