



Hacettepe Üniversitesi

Fen Fakültesi

İstatistik Bölümü

İST-374 - Olasılıksal Süreçler

Ara Sınav Ödevi

Bünyamin AKIN – 2210329044

Ders Sorumlusu – Prof. Dr. Gamze ÖZEL KADILAR

18.04.2025

Gerekli Paketlerin Yüklmesi

```
library(quantmod)      # Finansal veri çekmek için
library(markovchain)    # Markov zinciri oluşturmak için
library(dplyr)          # Veri işleme için
library(ggplot2)        # Grafik çizimi
library(reshape2)       # Heatmap için veri formatlama
library(DiagrammeR)     # Diyagram çizimi
library(expm)           # Matris üs alma işlemleri için
```

1. Gerçek hayattan en az 30 gözlem içeren bir veri kümesi seçiniz. Bu verilere dayanarak en az üç durum içeren bir stokastik süreç tanımlayınız. Bu süreçte ait durum uzayını (S) ve zaman parametresini (T) açıklayınız. Kullandığınız verinin kaynağını, zaman aralığını, tanımladığınız durumları belirtiniz.

Cevap: NASDAQ-100 endeksinden Door Dash Inc (\$DASH) şirketine ait veriler kullanılmıştır.

- Veri Kümesi: Door Dash (\$DASH) hisse senedinin kapanış fiyatları.
- Zaman Aralığı: 4 Ocak 2021 – 4 Ocak 2024 tarihleri arası. (769 iş günü)
- Kaynak : Yahoo Finance

Verinin çekilmesi

```
getSymbols("DASH", from = "2021-01-04", to = "2024-01-05")
## [1] "DASH"
dash_close <- Cl(DASH)
```

- **Süreç Tanımı:** Her bir işlem günü için, hisse senedi kapanış fiyatının bir önceki güne göre logaritmik durumu.

X_t = Door Dash hisse senedinin t-1. güne göre logaritmik getirisi.

Logaritmik Getirinin Hesaplanması

```
dash_returns <- diff(log(dash_close))
dash_returns <- na.omit(dash_returns)
```

- Durum Uzayı:

Durumlar üç kategoride tanımlanmıştır.

$S = \{\text{Yükseliş, Düşüş, Sabit}\}$

- Yükseliş: $P_t > P_{t-1}$
- Düşüş: $P_t < P_{t-1}$
- Sabit: $P_t = P_{t-1}$

Durumları Tanımlama

```
states <- ifelse(dash_returns > 0, "Yukselis",
                ifelse(dash_returns < 0, "Dusus", "Sabit"))
states <- factor(states, levels = c("Yukselis", "Dusus", "Sabit"))
```

- Zaman Parametresi

Zaman parametresi kesikli parametrelidir ve işlem gününü temsil eder. (Tatil günleri hariç tutulmuştur.)

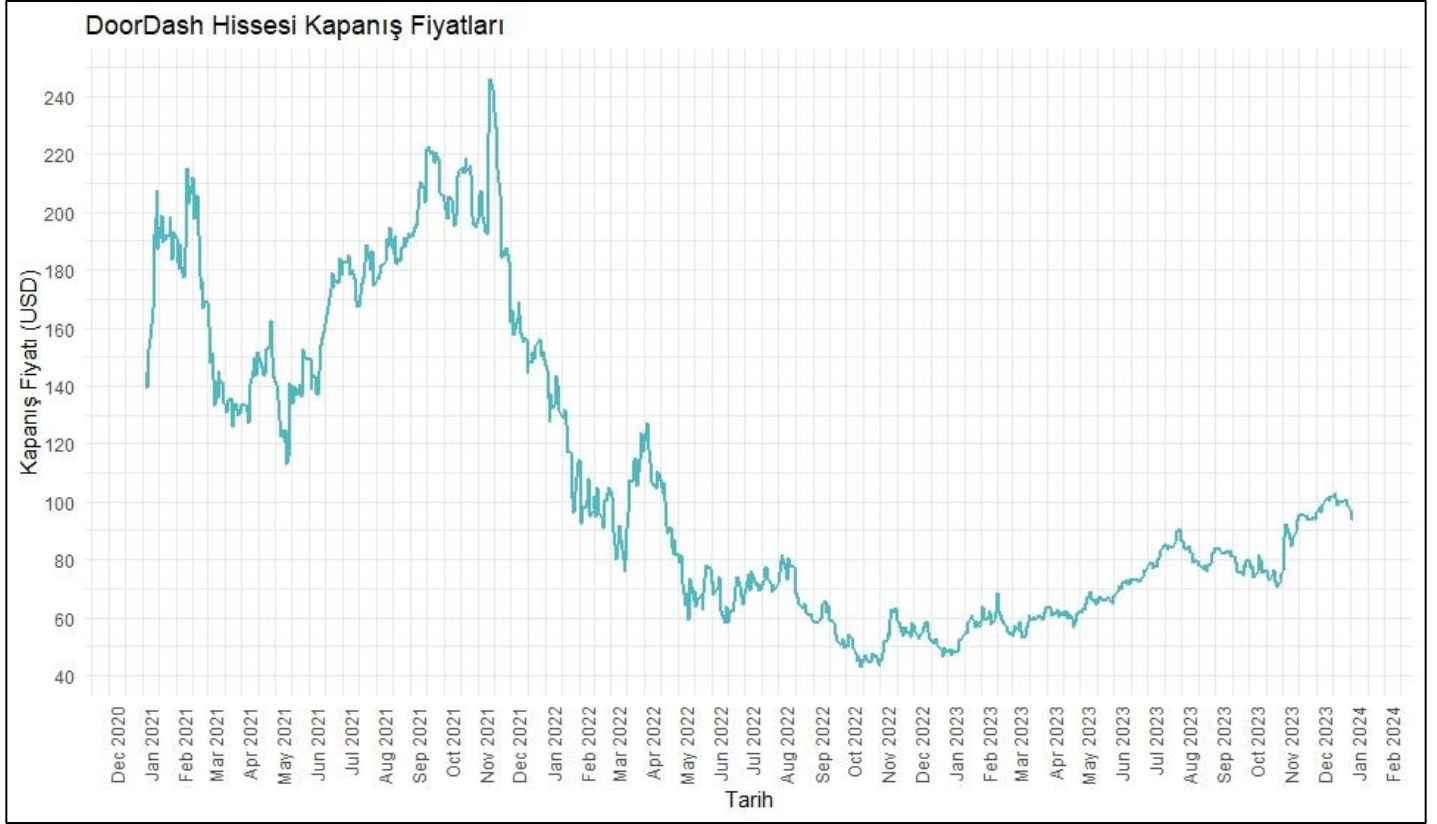
$T = \{1, 2, \dots, 756\}$

```
length(dash_close)
```

```
## [1] 756
```

2.Tanımladığınız Markov zincirinin durum uzayı (y eksen) ve zaman parametresine (x eksen) bağlı grafiğini çiziniz. Yorumlayınız.

```
ggplot(DASH, aes(x=index(dash_close), y=dash_close)) +
  geom_line(color="#4BBAC3", size=1) +
  labs(title="DoorDash Hissesi Kapanış Fiyatları",
       x="Tarih", y="Kapanış Fiyatı (USD)") +
  scale_x_date(date_labels="%b %Y", date_breaks="1 month") +
  scale_y_continuous(breaks=seq(0, max(dash_close), by=20)) +
  theme_minimal() +
  theme(axis.text.x=element_text(angle=90, hjust=1))
```



Hisse senedi, 2021'in başlarında 140-160 dolar seviyelerinden işlem görmeye başlamış. 2021 ortasından itibaren yükseliş eğimi gözükmüş ve Kasım 2021'de tüm zamanların en yüksek fiyatını görmüştür. 2022'den 2023 yılına kadar ise sert bir düşüş yaşamış ve 2023'ten sonra ise fiyatta ufak bir yükseliş gözükmiştir. Volatilesi yüksek bir hisse senedi olduğu görülmektedir.

3.Tanımladığınız Markov zincirine ait bir-adım geçiş matrisini P oluşturunuz. Bu matrise ait geçiş diyagramını çiziniz. Bu geçiş matrisindeki en yüksek ve en düşük olasılık değerlerini yorumlayınız.

Markov Zincirini Tanımlama

```
mc_fit <- markovchainFit(data = states)
markov_model <- mc_fit$estimate
transition_matrix <- markov_model@transitionMatrix
```

Geçiş Matrisini Yazdırma

```
print(transition_matrix)

##           Dusus      Sabit  Yükselis
## Dusus    0.4919786 0.00802139 0.5000000
## Sabit     0.6666667 0.00000000 0.3333333
## Yükselis 0.4986737 0.00000000 0.5013263
```

Geçiş Matrisi Grafiği

```
transition_df <- melt(transition_matrix)
colnames(transition_df) <- c("Nereden", "Nereye", "Olasilik")

ggplot(transition_df, aes(x = Nereye, y = Nereden, fill = Olasilik)) +
  geom_tile(color = "white") +
  geom_text(aes(label = round(Olasilik, 8)), color = "black", size = 5) +
  scale_fill_gradientn(colors = c("white", "salmon", "tomato", "orangered",
"darkorange")) +
  labs(title = "Gecis Olasilik Matrisi",
       x = "Bir Sonraki Durum", y = "Simdiki Durum") +
  theme_minimal(base_size = 12)
```



En yüksek olasılık değeri: 0.667 ile Sabit'ten Düşüş'e geçme olasılığıdır. En düşük olasılık değeri: 0 ile Yükseliş'ten Sabit'e ve Sabit'ten Sabit'e geçme olasılığıdır.

4.Tanımladığınız Markov zincirine ait P3, P10, P100 olasılık matrislerini oluşturunuz. Diyagramlarını çiziniz.

Öncelikle diyagram yazdıran fonksiyonu tanımlayalım.

```
diagram=function(transition_matrix){ states <- colnames(transition_matrix)

graph_code <- "digraph markov_chain {
  rankdir=LR;
  node [shape=box, style=filled, fontname=Helvetica, fontsize=14];

"

graph_code <- paste0(graph_code,
"\n", states[1], "\n" [fillcolor="#A8E6CF\"];\\n",
"\n", states[2], "\n" [fillcolor="#FFD3B6\"];\\n",
"\n", states[3], "\n" [fillcolor="#FFAA5\"];\\n")

for (i in 1:nrow(transition_matrix)) {
  for (j in 1:ncol(transition_matrix)) {
    prob <- transition_matrix[i, j]
    if (prob > 0.01) {
      pen_width <- round(2, 2)
      font_size <- 10
      pen_color <- "#9CC8E2" #
      graph_code <- paste0(
        graph_code,
        states[i], " -> ", states[j],
        " [label=", round(prob, 10),
        "\n", fontsize=", font_size,
        "\n", penwidth=", pen_width,
        "\n", color=", pen_color, "\n"];\\n"
      )
    }
  }
}

graph_code <- paste0(graph_code, "\\n}")

DiagrammeR::grViz(graph_code)

}
```

```

P3 <- transition_matrix %>% 3
cat(cat("\nP3 olasilik matrasi:\n"))

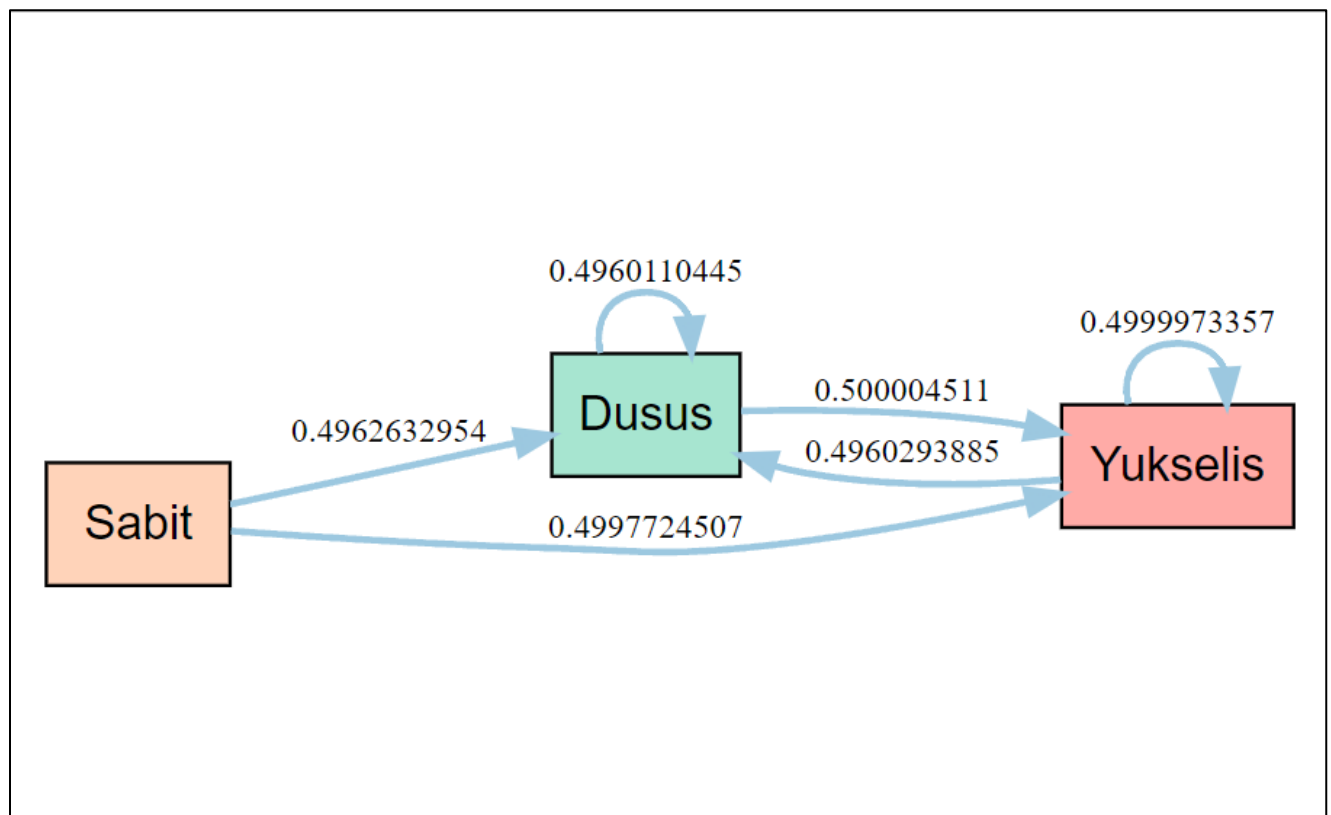
##
## P3 olasilik matrasi:

P3

##           Dusus      Sabit  Yukselis
## Dusus    0.4960110 0.003984445 0.5000045
## Sabit     0.4962633 0.003964254 0.4997725
## Yukselis  0.4960294 0.003973276 0.4999973

diagram(P3)

```



```

P10 <- transition_matrix %% 10
cat(cat("\nP10 olasilik matrиси:\n"))

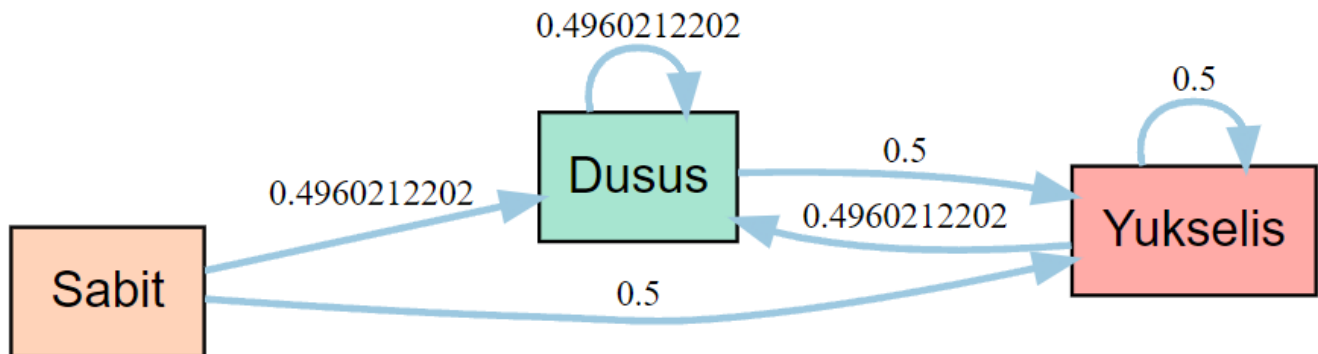
##
## P10 olasilik matrиси:

P10

##           Dusus      Sabit Yukselis
## Dusus    0.4960212  0.00397878    0.5
## Sabit     0.4960212  0.00397878    0.5
## Yukselis  0.4960212  0.00397878    0.5

diagram(P10)

```




```

P100 <- transition_matrix %%% 100
cat(cat("\nP100 olasılık matrisi:\n"))

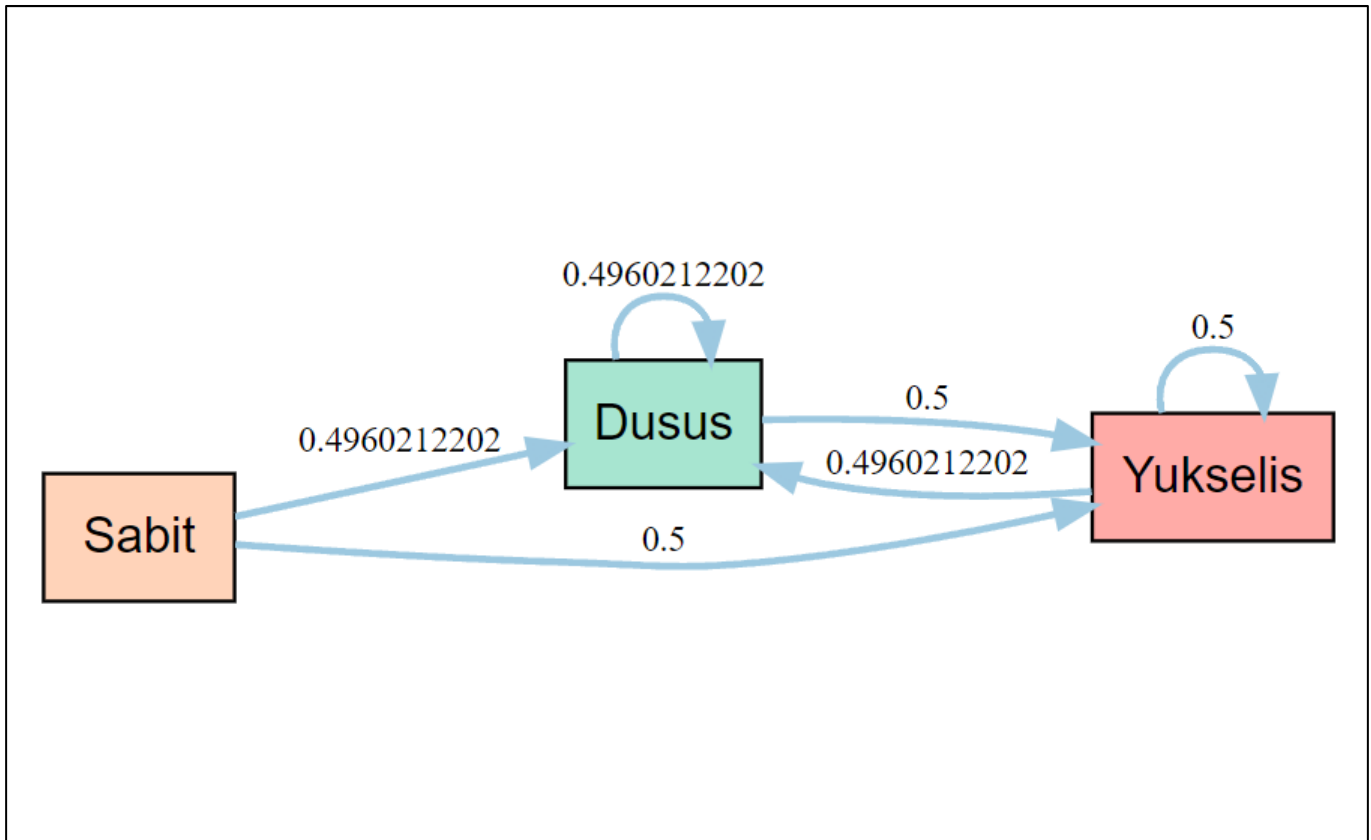
##
## P100 olasılık matrisi:

P100

##           Dusus      Sabit Yukselis
## Dusus    0.4960212  0.00397878    0.5
## Sabit     0.4960212  0.00397878    0.5
## Yukselis  0.4960212  0.00397878    0.5

diagram(P100)

```



5.İncelediğiniz süreç indirgenebilir midir? Kapalı küme, periyodik durum, yutucu durumları var ise bulunuz. Açıklayıp yorumlayınız.

```
print(transition_matrix)

##           Dusus      Sabit  Yukselis
## Dusus    0.4919786 0.00802139 0.5000000
## Sabit     0.6666667 0.00000000 0.3333333
## Yukselis 0.4986737 0.00000000 0.5013263
```

Bu süreç indirgenemezdir çünkü tüm durumlar birbirine ulaşmıyor. Örneğin Sabit'ten Sabit'e veya Yükseliş'ten Sabit'e ulaşma durumu olasılığı sıfır.

{Düşüş, Yükseliş} kümesi kapalıdır çünkü Sabit durumuna geçiş bu kümelerden mümkün değil.

Bu süreçte yutucu durum yoktur. Hiçbir satırda "1" olasılığı yoktur.

Bu süreçte periyodik durum yoktur. Her durumda kendine dönüş ihtimali olduğundan, hepsi aperiyoiktir.

6.İncelediniz süreç indirgenemez midir? Denge dağılımı var ise bulunuz. Açıklayıp yorumlayınız.

Bu süreç indirgenemezdir çünkü tüm durumlar birbirine ulaşmıyor.

Denge Dağılımı

```
stationary <- steadyStates(markov_model)
cat("\n Denge Dağılımı:\n")

##
## Denge Dağılımı:

print(stationary)

##           Dusus      Sabit  Yukselis
## [1,] 0.4960212 0.00397878      0.5
```