DATA605 Discussion Post Week 10

Andrew Bowen

2023-03-29

Exercise 11.1.13

Write a program to compute $u^{(n)}$ given $\mathbf{u}$ and P. Use this program to compute $u^{(10)}$ for the Land of Oz example, with $u = (0, 1, 0)$, and with $u = (1/3, 1/3, 1/3)$.

First, let's define our transition probability matrix P for each weather type (*Rainy, Sunny, Nice*)

```
# First, define our P matrix
n <- 10
oz_matrix <- matrix(c(0.5, 0.25, 0.25, 0.5, 0, 0.5, 0.25, 0.25, 0.5), nrow=3, ncol=3, byrow=TRUE)

rownames(oz_matrix) <- c("R", "N", "S")
colnames(oz_matrix) <- c("R", "N", "S")

print(oz_matrix)
```

```
##      R    N    S
## R 0.50 0.25 0.25
## N 0.50 0.00 0.50
## S 0.25 0.25 0.50
```

Based on Theorem 11.2, if P is our transition matrix, Then the probability that the chain is in state s_i after n steps is the i^{th} entry in the vector:

$$u^{(n)} = \mathbf{u}P^n$$

For the case where $\mathbf{u} = (0, 1, 0)$:

```
u <- c(0, 1, 0)
P <- oz_matrix

for (i in 1:n -1){
  P <- P %*% P
}

(x <- u %*% P)
```

```
##        R   N   S
## [1,] 0.4 0.2 0.4
```

For the case where $\mathbf{u} = (1/3, 1/3, 1/3)$:

```r
u <- c(1/3, 1/3, 1/3)
P <- oz_matrix

for (i in 1:n - 1){
  P <- P %*% P
}

(y <- u %*% P)
```

```
##         R   N   S
## [1,] 0.4 0.2 0.4
```

Since the result of these calculations are probability vectors, we should be able to sum the distribution vectors to 1.

```r
print(sum(x))
```

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

```r
print(sum(y))
```

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

Ideally, we could wrap the above block of code in a function like:

```r
calculate_prob <- function(U, M, N){
  if (n == 1){
    return(U %*% M)
  }
  else{
    for (i in 1:(n)){
      M <- M %*% M
    }

    y <- u %*% M
    return(y)
  }
}

# Test call our function for the above case
calculate_prob(c(0, 1, 0), oz_matrix, 10)
```

```
##         R   N   S
## [1,] 0.4 0.2 0.4
```

```r
calculate_prob(c(1/3, 1/3, 1/3), oz_matrix, 10)
```

```
##         R   N   S
## [1,] 0.4 0.2 0.4
```