Assignment 1 for Stat 280

Maharaj Teertha Deb , Student id: 40227747

2024-01-29

Part 1:

Question (1) :

Using one line of R code, calculate the interest earned on an investment of $2000, assuming an interest rate of 3% compounded annually, for terms of 1, 2, . . . , 30 years.

Answer:

```
interest_earned <- 2000 * (1 + 0.03)^ (1:30) - 2000 # Interest = P * (1 + R)^N - P
interest_earned
```

```
##  [1]   60.0000  121.8000  185.4540  251.0176  318.5481  388.1046  459.7477
##  [8]  533.5402  609.5464  687.8328  768.4677  851.5218  937.0674 1025.1794
## [15] 1115.9348 1209.4129 1305.6953 1404.8661 1507.0121 1612.2225 1720.5891
## [22] 1832.2068 1947.1730 2065.5882 2187.5559 2313.1825 2442.5780 2575.8554
## [29] 2713.1310 2854.5249
```

Question (2):

Using one line of R code, calculate the respective areas of the circles having radii 3,4,. . . , 100.

Answer:

```
circle_areas <- pi * (3:100)^2
circle_areas
```

```
##  [1]    28.27433    50.26548    78.53982   113.09734   153.93804   201.06193
##  [7]   254.46900   314.15927   380.13271   452.38934   530.92916   615.75216
## [13]   706.85835   804.24772   907.92028  1017.87602  1134.11495  1256.63706
## [19]  1385.44236  1520.53084  1661.90251  1809.55737  1963.49541  2123.71663
## [25]  2290.22104  2463.00864  2642.07942  2827.43339  3019.07054  3216.99088
## [31]  3421.19440  3631.68111  3848.45100  4071.50408  4300.84034  4536.45979
## [37]  4778.36243  5026.54825  5281.01725  5541.76944  5808.80482  6082.12338
## [43]  6361.72512  6647.61005  6939.77817  7238.22947  7542.96396  7853.98163
## [49]  8171.28249  8494.86654  8824.73376  9160.88418  9503.31778  9852.03456
```

```
## [55] 10207.03453 10568.31769 10935.88403 11309.73355 11689.86626 12076.28216
## [61] 12468.98124 12867.96351 13273.22896 13684.77760 14102.60942 14526.72443
## [67] 14957.12262 15393.80400 15836.76857 16286.01632 16741.54725 17203.36137
## [73] 17671.45868 18145.83917 18626.50284 19113.44970 19606.67975 20106.19298
## [79] 20611.98940 21124.06900 21642.43179 22167.07776 22698.00692 23235.21927
## [85] 23778.71480 24328.49351 24884.55541 25446.90049 26015.52876 26590.44022
## [91] 27171.63486 27759.11269 28352.87370 28952.91790 29559.24528 30171.85585
## [97] 30790.74960 31415.92654
```

Question (3):

Calculate the sum $\sum_{j=0}^{n} r^j$, where r has been assigned the value 1.08, and compare with $\frac{1-r^{(n+1)}}{1-r}$, for $n = 10, 20, 30, 40$. Repeat for $r = 1.06$.

Answer:

```
calculate_sum <- function(r, n) {
    return(sum(r^(0:n)))
}

r_power_n_without_formula <- function(r, n_values) {
    results <- sapply(n_values, calculate_sum, r = r)
    return(results)
}


r_power_n_using_formula <- function(r, n_values) {
    results <- (1 - r^(n_values + 1))/(1 - r)
    return(results)
}

r <- 1.08
n_values <- c(10, 20, 30, 40)

# for r = 1.08
print("For r = 1.08")
```

```
## [1] "For r = 1.08"
```

```
results_for_r1_without_formula <- r_power_n_without_formula(r, n_values)
cat("Results without formula: ", results_for_r1_without_formula, "\n")
```

```
## Results without formula:  16.64549 50.42292 123.3459 280.781
```

```
results_for_r1_using_formula <- r_power_n_using_formula(r, n_values)
cat("Results using formula: ", results_for_r1_using_formula, "\n")
```

```
## Results using formula:  16.64549 50.42292 123.3459 280.781
```

```r
# for r = 1.06

cat("\n\n\n" , "For r = 1.06\n")
```

```
##
##
##
##  For r = 1.06
```

```r
results_for_r1_without_formula <- r_power_n_without_formula(r, n_values)
cat("Results without formula: ", results_for_r1_without_formula, "\n")
```

```
## Results without formula:  16.64549 50.42292 123.3459 280.781
```

```r
results_for_r1_using_formula <- r_power_n_using_formula(r, n_values)
cat("Results using formula: ", results_for_r1_using_formula, "\n")
```

```
## Results using formula:  16.64549 50.42292 123.3459 280.781
```

Question (4):

Referring to the above question, utilize the quick formula to calculate $\sum_{j=0}^{n} r^j$ for $r = 1.08$, considering all values of n in the range from 1 to 100. Save these 100 values in a vector.

Answer:

```r
# Set the values
r <- 1.08
n_values <- 1:100

# Calculate the sum for each value of n
result_vector <- sapply(n_values, function(n) sum(r^(0:n)))

# Print the result vector
print(result_vector)
```

```
##   [1]     2.080000     3.246400     4.506112     5.866601     7.335929
##   [6]     8.922803    10.636628    12.487558    14.486562    16.645487
##  [11]    18.977126    21.495297    24.214920    27.152114    30.324283
##  [16]    33.750226    37.450244    41.446263    45.761964    50.422921
##  [21]    55.456755    60.893296    66.764759    73.105940    79.954415
##  [26]    87.350768    95.338830   103.965936   113.283211   123.345868
##  [31]   134.213537   145.950620   158.626670   172.316804   187.102148
##  [36]   203.070320   220.315945   238.941221   259.056519   280.781040
##  [41]   304.243523   329.583005   356.949646   386.505617   418.426067
##  [46]   452.900152   490.132164   530.342737   573.770156   620.671769
##  [51]   671.325510   726.031551   785.114075   848.923201   917.837058
##  [56]   992.264022  1072.645144  1159.456755  1253.213296  1354.470360
##  [61]  1463.827988  1581.934227  1709.488966  1847.248083  1996.027929
```

```
##  [66]  2156.710164  2330.246977  2517.666735  2720.080074  2938.686480
##  [71]  3174.781398  3429.763910  3705.145023  4002.556624  4323.761154
##  [76]  4670.662047  5045.315011  5449.940211  5886.935428  6358.890263
##  [81]  6868.601484  7419.089602  8013.616770  8655.706112  9349.162601
##  [86] 10098.095609 10906.943258 11780.498718 12723.938616 13742.853705
##  [91] 14843.282002 16031.744562 17315.284127 18701.506857 20198.627405
##  [96] 21815.517598 23561.759006 25447.699726 27484.515704 29684.276961
```

Question (5):

Calculate the sum $\sum_{i=1}^{N} \frac{1}{i}$, and compare it with $\log(N) + 0.6$, for $N = 500, 1000, 2000, 4000, 8000$.

Answer:

```r
# Define the values of n
n_values <- c(500, 1000, 2000, 4000, 8000)

# Function to calculate the sum
calculate_sum <- function(n) {
    return(sum(1 / (1:n)))
}

# Calculate sums for each value of n
sums <- sapply(n_values, calculate_sum)

# Print the sums
cat("Sums:", sums, "\n")
```

```
## Sums: 6.792823 7.485471 8.178368 8.87139 9.564475
```

```r
# Function to calculate log(n) + 0.6
logn_0.6 <- function(n) {
    return(log(n) + 0.6)
}

# Calculate log(n) + 0.6 for each value of n
log_sum <- sapply(n_values, logn_0.6)

# Print log(n) + 0.6
cat("log(n) + 0.6:", log_sum, "\n")
```

```
## log(n) + 0.6: 6.814608 7.507755 8.200902 8.89405 9.587197
```

```r
# Print the comparison
cat("Comparison:", log_sum - sums, "\n")
```

```
## Comparison: 0.02178467 0.02228442 0.02253436 0.02265934 0.02272184
```

Question (6):

Using `rep()` and `seq()` as needed, create the vectors:

$$0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,2\,2\,2\,2\,3\,3\,3\,3\,4\,4\,4\,4$$
$$1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5$$

Answer:

To get the sequence :

$$0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,2\,2\,2\,2\,3\,3\,3\,3\,4\,4\,4\,4$$

```
rep(seq(1 : 4) , each = 5)
```

```
##  [1] 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4
```

To get the sequence :

$$1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5\,1\,2\,3\,4\,5$$

```
rep(1 : 5 , 5)
```

```
##  [1] 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5
```

Question (7):

Using `rep()` and `seq()` as needed, create the vector:

$$1\,2\,3\,4\,5\,2\,3\,4\,5\,6\,3\,4\,5\,6\,7\,4\,5\,6\,7\,8\,5\,6\,7\,8\,9$$

Answer:

```
  seq(1 : 5) + rep(0 : 4 , each = 5)
```

```
##  [1] 1 2 3 4 5 2 3 4 5 6 3 4 5 6 7 4 5 6 7 8 5 6 7 8 9
```

Part 2:

Question (1):

Using one line of R code, calculate the respective areas of the circles having radii 3,4,..., 100. Create the following vectors in R using `seq()` and `rep()`:

(i) $1, 1.5, 2, 2.5, \ldots, 12$

(ii) $1, 8, 27, 64, \ldots, 1000$

(iii) $1, -\frac{1}{2}, \frac{1}{3}, -\frac{1}{4}, \ldots, -\frac{1}{100}$

(iv) $1, 0, 3, 0, 5, 0, 7, \ldots, 0, 49$

(v) $1, 3, 6, 10, 15, \ldots, \sum_{i=1}^{n} i, \ldots, 210$ [Look up `?cumsum`]

(vi) $1, 2, 2, 3, 3, 3, 4, 4, 4, 4, \ldots, 9, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10$

Answer:

(i) Answer :

```
seq(1 , 12 , by = 0.5)
```

```
##  [1]  1.0  1.5  2.0  2.5  3.0  3.5  4.0  4.5  5.0  5.5  6.0  6.5  7.0  7.5  8.0
## [16]  8.5  9.0  9.5 10.0 10.5 11.0 11.5 12.0
```

(ii) Answer :

```
seq(1 , 10) ^ 3
```

```
##  [1]    1    8   27   64  125  216  343  512  729 1000
```

(iii) Answer:

```
n <- (1 : 100)

seq <- 1/n * (-1)^(n)

seq * (-1)
```

```
##   [1]  1.00000000 -0.50000000  0.33333333 -0.25000000  0.20000000 -0.16666667
##   [7]  0.14285714 -0.12500000  0.11111111 -0.10000000  0.09090909 -0.08333333
##  [13]  0.07692308 -0.07142857  0.06666667 -0.06250000  0.05882353 -0.05555556
##  [19]  0.05263158 -0.05000000  0.04761905 -0.04545455  0.04347826 -0.04166667
##  [25]  0.04000000 -0.03846154  0.03703704 -0.03571429  0.03448276 -0.03333333
##  [31]  0.03225806 -0.03125000  0.03030303 -0.02941176  0.02857143 -0.02777778
##  [37]  0.02702703 -0.02631579  0.02564103 -0.02500000  0.02439024 -0.02380952
##  [43]  0.02325581 -0.02272727  0.02222222 -0.02173913  0.02127660 -0.02083333
##  [49]  0.02040816 -0.02000000  0.01960784 -0.01923077  0.01886792 -0.01851852
##  [55]  0.01818182 -0.01785714  0.01754386 -0.01724138  0.01694915 -0.01666667
##  [61]  0.01639344 -0.01612903  0.01587302 -0.01562500  0.01538462 -0.01515152
##  [67]  0.01492537 -0.01470588  0.01449275 -0.01428571  0.01408451 -0.01388889
##  [73]  0.01369863 -0.01351351  0.01333333 -0.01315789  0.01298701 -0.01282051
##  [79]  0.01265823 -0.01250000  0.01234568 -0.01219512  0.01204819 -0.01190476
##  [85]  0.01176471 -0.01162791  0.01149425 -0.01136364  0.01123596 -0.01111111
##  [91]  0.01098901 -0.01086957  0.01075269 -0.01063830  0.01052632 -0.01041667
##  [97]  0.01030928 -0.01020408  0.01010101 -0.01000000
```

(iv) Answer:

```
c((1:49) * (1:49 %% 2))
```

```
##  [1]  1  0  3  0  5  0  7  0  9  0 11  0 13  0 15  0 17  0 19  0 21  0 23  0 25
## [26]  0 27  0 29  0 31  0 33  0 35  0 37  0 39  0 41  0 43  0 45  0 47  0 49
```

(v) Answer:

```
cumsum(1 : 15)
```

```
##  [1]   1   3   6  10  15  21  28  36  45  55  66  78  91 105 120
```

(vi) Answer:

```
n_values <- 1 : 10
rep(n_values , n_values)
```

```
##  [1]  1  2  2  3  3  3  4  4  4  4  5  5  5  5  5  6  6  6  6  6  6  7  7  7  7
## [26]  7  7  7  8  8  8  8  8  8  8  8  9  9  9  9  9  9  9  9  9 10 10 10 10 10
## [51] 10 10 10 10 10
```