

Funciones anualidades convencionales

Alejandro Emiliano Gaytan Jaramillo, 2408822k@umich.mx (UMSNH)

Alessandro Perez Piña, 2408821b@umich.mx (UMSNH)

Jaime Patricio Guido Rivera, 2408831g@umich.mx (UMSNH)

Mikel Augusto Marin Rodriguez, 2408834a@umich.mx (UMSNH)

Table of contents

Anualidades Vencidas (VF)	1
Anualidades Venciadas (VA)	2
Anualidades Anticipadas (VF)	3
Anualidades Anticipadas (VA)	4
Funciones Generales	5



Anualidades Vencidas (VF)

Lo primero que haremos será crear las diferentes funciones para las anualidades vencidas con valor futuro:

1. Valor futuro (VF)

```
AnualidadesVencidasVFVF = function(A, r, T){
  VF = A * ( ( (1 + r)^T - 1 ) / r )
  return(VF)
}
```

2. Monto del pago en la anualidad (A)

```
AnualidadesVencidasVFA = function(VF, r, T){
  A = VF * ( r / ( (1 + r)^T - 1 ) )
  return(A)
}
```

3. Tasa de interes del periodo (r)

```
AnualidadesVencidasVFr = function(VF, A, T, guess = 0.1){

  f <- function(r){
    A * (((1+r)^T - 1) / r) - VF
  }

  sol <- uniroot(f, interval = c(1e-8, 1), extendInt = "yes")

  return(sol$root)
}
```

4. Número de periodos de pagos (T)

```
AnualidadesVencidasVFT = function(VF, A, r){
  T = log(1 + (VF * r / A)) / log(1 + r)
  return(T)
}
```

Anualidades Vencidas (VA)

Ahora lo que haremos será definir las funciones para anualidades vencidas con valor actual:

1. Valor Actual (VA)

```
AnualidadesVencidasVAVA = function(A, r, T){
  VA = A * (1 - (1 + r)^(-T)) / r
  return(VA)
}
```

2. Monto del pago en la anualidad (A)

```
AnualidadesVencidasVAA = function(VA, r, T){  
  A = VA * r / (1 - (1 + r)^(-T))  
  return(A)  
}
```

3. Tasa de interes del periodo (r)

```
AnualidadesVencidasVAr = function(VA, A, T, guess = 0.1){  
  
  f <- function(r){  
    A * (1 - (1+r)^(-T)) / r - VA  
  }  
  
  sol <- uniroot(f, interval = c(1e-8, 1), extendInt = "yes")  
  
  return(sol$root)  
}
```

4. Número de periodos de pagos (T)

```
AnualidadesVencidasVAT = function(VA, A, r){  
  T = -log(1 - (VA * r / A)) / log(1 + r)  
  return(T)  
}
```

Anualidades Anticipadas (VF)

Lo siguiente que haremos será definir las funciones de Anualidades anticipadas con Valor futuro:

1. Valor futuro (VF)

```
AnualidadesAnticipadasVFVF = function(A, r, T){  
  VF = A * (((1 + r)^T - 1) / r) * (1 + r)  
  return(VF)  
}
```

2. Monto del pago en la anualidad (A)

```
AnualidadesAnticipadasVFA = function(VF, r, T){
  A = VF * r / (((1 + r)^T - 1) * (1 + r))
  return(A)
}
```

3. Tasa de interes del periodo (r)

```
AnualidadesAnticipadasVFr = function(VF, A, T, guess = 0.1){

  f <- function(r){
    A * (((1+r)^T - 1) / r) * (1+r) - VF
  }

  sol <- uniroot(f, interval = c(1e-8, 1), extendInt = "yes")

  return(sol$root)
}
```

4. Número de periodos de pago (T)

```
AnualidadesAnticipadasVFT = function(VF, A, r){
  T = log(1 + (VF * r / (A * (1 + r)))) / log(1 + r)
  return(T)
}
```

Anualidades Anticipadas (VA)

Casi por terminar, vamos a crear las funciones para cada parte de anualidades anticipadas con valor actual:

1. Valor actual (VA)

```
AnualidadesAnticipadasVAVA =function(A, r, T){
  VA = A * (1 - (1 + r)^(-T)) / r * (1 + r)
  return(VA)
}
```

2. Monto del pago en la anualidad (A)

```
AnualidadesAnticipadasVAA = function(VA, r, T){
  A = VA * r / ((1 - (1 + r)^(-T)) * (1 + r))
  return (A)
}
```

3. Tasa de interes del periodo (r)

```
AnualidadesAnticipadasVAr = function(VA, A, T, guess = 0.1){

  f <- function(r){
    A * (1 - (1+r)^(-T)) / r * (1+r) - VA
  }

  sol <- uniroot(f, interval = c(1e-8, 1), extendInt = "yes")

  return(sol$root)
}
```

4. Número de periodos de pago (T)

```
AnualidadesAnticipadasVAT = function(VA, A, r){
  T = -log(1 - (VA * r / (A * (1 + r)))) / log(1 + r)
  return(T)
}
```

Funciones Generales

Por último, necesitamos incluir las ultimas dos funciones generales “valorActualAnualidades” y “valorFuturoAnualidades” para que esas dos funciones puedan llamar a las otras dependiendo el valor que nos pidan encontrar:

1. valorFuturoAnualidades

```
valorFuturoAnualidades <- function(A = NA, r = NA, T = NA, VF = NA, anticipada = FALSE) {

  if (!is.na(r) && r < 0) stop("Error: la tasa de interés (r) no puede ser negativa.")
  if (!is.na(T) && T <= 0) stop("Error: el número de periodos (T) debe ser mayor que cero.")
  if (!is.logical(anticipada)) stop("Error: 'anticipada' debe ser TRUE o FALSE.")

  cantidad_NA <- sum(is.na(c(A, r, T, VF)))
```

```

if (cantidad_NA != 1) {
  stop("Error: Debes dejar exactamente uno de los valores como NA (A, r, T o VF).")
}

if (is.na(VF)) {
  if (anticipada) {
    VF <- AnualidadesAnticipadasVFVF(A, r, T)
  } else {
    VF <- AnualidadesVencidasVFVF(A, r, T)
  }
  return(VF)
}

if (is.na(A)) {
  if (anticipada) {
    A <- AnualidadesAnticipadasVFA(VF, r, T)
  } else {
    A <- AnualidadesVencidasVFA(VF, r, T)
  }
  return(A)
}

if (is.na(r)) {
  if (anticipada) {
    r <- AnualidadesAnticipadasVFr(VF, A, T)
  } else {
    r <- AnualidadesVencidasVFr(VF, A, T)
  }
  return(r)
}

if (is.na(T)) {
  if (anticipada) {
    T <- AnualidadesAnticipadasVFT(VF, A, r)
  } else {
    T <- AnualidadesVencidasVFT(VF, A, r)
  }
  return(T)
}

```

```

}

stop("Error inesperado: revisa los valores nuevamente.")
}

```

2. valorActualAnualidades

```

valorActualAnualidades <- function(A = NA, r = NA, T = NA, VA = NA, anticipada = FALSE) {

  if (!is.na(r) && r < 0) stop("Error: la tasa de interés (r) no puede ser negativa.")
  if (!is.na(T) && T <= 0) stop("Error: el número de periodos (T) debe ser mayor que cero.")
  if (!is.logical(anticipada)) stop("Error: 'anticipada' debe ser TRUE o FALSE.")

  cantidad_NA <- sum(is.na(c(A, r, T, VA)))

  if (cantidad_NA != 1) {
    stop("Error: Debes dejar exactamente un valor como NA (A, r, T o VA).")
  }

  if (is.na(VA)) {
    if (anticipada) {
      VA <- AnualidadesAnticipadasVAVA(A, r, T)
    } else {
      VA <- AnualidadesVencidasVAVA(A, r, T)
    }
    return(VA)
  }

  if (is.na(A)) {
    if (anticipada) {
      A <- AnualidadesAnticipadasVAA(VA, r, T)
    } else {
      A <- AnualidadesVencidasVAA(VA, r, T)
    }
    return(A)
  }
}

```

```

if (is.na(r)) {
  if (anticipada) {
    r <- AnualidadesAnticipadasVAr(VA, A, T)
  } else {
    r <- AnualidadesVencidasVAr(VA, A, T)
  }
  return(r)
}

if (is.na(T)) {
  if (anticipada) {
    T <- AnualidadesAnticipadasVAT(VA, A, r)
  } else {
    T <- AnualidadesVencidasVAT(VA, A, r)
  }
  return(T)
}

stop("Error inesperado: revisa los datos ingresados.")
}

```

Ejemplos:

1.

```
valorFuturoAnualidades(r=0.02,VF=1000000,T=360,anticipada=TRUE)
```

```
[1] 15.72955
```

2.

```
valorFuturoAnualidades(r=0.02,A=1500,T=360,anticipada=TRUE)
```

```
[1] 95361926
```

3.


```
valorFuturoAnualidades(VF=30000, A=400, T=48, anticipada=TRUE)
```

```
[1] 0.01721927
```

4.

```
valorActualAnualidades(anticipada=FALSE, VA=23000, r=0.0125, T=48)
```

```
[1] 640.1072
```