

TAREA 3

Desarrollo de la librería QuiCa

Funciones para química y cálculo científico

Escuela de Informática y Computación

Información de la tarea

Curso: EIY403 - Introducción al análisis de datos para otras carreras

Valor: 5 % de la nota final

Modalidad: Grupal

Fecha de entrega: Miércoles 1 de octubre, 6:00 PM

Formato: R Markdown compilado (.Rmd + .html)

Entrega: Carga a Aula virtual

1. Introducción: Creando la librería QuiCa

Contexto profesional:

En esta tarea simularemos el proceso de desarrollo de una librería de R para contribuir al repositorio CRAN (Comprehensive R Archive Network). Su misión es crear **QuiCa** (Química + Cálculo), una librería especializada que integre funciones útiles para estudiantes y profesionales de ciencias exactas, especialmente aquellos que trabajan con química y necesitan herramientas matemáticas complementarias.

1.1. ¿Qué es CRAN y por qué es importante?

CRAN es el repositorio oficial de librerías de R, donde desarrolladores de todo el mundo contribuyen con herramientas especializadas. Tiene estándares muy rigurosos de documentación, testing y mantenimiento. Crear una librería que cumpla estos estándares es una habilidad valiosa en el mundo profesional del análisis de datos.

1.2. Objetivos de aprendizaje

Al completar esta tarea, habrás desarrollado:

- Competencias en creación de funciones robustas con validación completa
- Habilidades de documentación técnica estilo CRAN
- Experiencia en testing y validación de software científico
- Capacidad de integrar conocimientos de matemática y química en código
- Práctica en la aplicación de funciones propias para resolver problemas reales

2. Especificaciones técnicas de la librería QuiCa

Características de QuiCa:

- **Nombre completo:** QuiCa - Chemistry and Calculus Tools for R
- **Versión:** 1.0.0
- **Autor:** Su nombre completo
- **Descripción:** Una colección de funciones especializadas para cálculos químicos y matemáticos básicos
- **Total de funciones:** 6 (3 química + 3 cálculo)
- **Documentación:** Estilo CRAN con ejemplos y referencias

3. Funciones requeridas

3.1. Módulo de Química (3 funciones)

Función 1: `convertir_unidades_quimicas()`

Propósito: Convertir entre diferentes unidades químicas comunes

Parámetros:

- **valor:** Valor numérico a convertir
- **de:** Unidad origen (string)
- **a:** Unidad destino (string)

Conversiones requeridas:

- **Temperatura:** celsius, kelvin, fahrenheit
- **Presión:** atm, torr, bar, psi
- **Volumen:** L, mL, gal, m3

Validaciones: Unidades válidas, valores numéricos, temperaturas físicamente posibles

Función 2: calcular_concentracion()**Propósito:** Calcular diferentes tipos de concentración química**Parámetros:**

- masa_solute: Masa del soluto en gramos
- masa_molecular: Masa molecular del compuesto (g/mol)
- volumen_solucion: Volumen de la solución en litros
- tipo: molaridad, molalidad, ppm (opcional, default = "molaridad")

Retorna: Lista con diferentes tipos de concentración calculados**Validaciones:** Valores positivos, masa molecular mayor a 0**Función 3: validar_condiciones_laboratorio()****Propósito:** Validar si las condiciones de laboratorio están en rangos seguros**Parámetros:**

- ph: Valor de pH
- temperatura: Temperatura en Celsius
- presion: Presión en atm (opcional, default = 1)
- humedad: Humedad relativa % (opcional, default = 50)

Retorna: Lista con evaluación de seguridad y recomendaciones**Rangos seguros:** pH 0-14, Temp 15-40°C, Presión 0.8-1.2 atm, Humedad 30-70 %

3.2. Módulo de Cálculo (3 funciones)

Función 4: derivada_numerica()**Propósito:** Calcular la derivada numérica de una función en un punto**Parámetros:**

- f: Función a derivar
- x: Punto donde calcular la derivada
- h: Paso para diferencias finitas (opcional, default = 1e-7)
- metodo: central, adelante, atras (opcional, default = central)

Retorna: Valor numérico de la derivada**Validaciones:** Función válida, x numérico, h positivo

Función 5: `resolver_cuadratica()`**Propósito:** Resolver ecuaciones cuadráticas con análisis completo**Parámetros:**

- a: Coeficiente de x^2
- b: Coeficiente de x
- c: Término independiente

Retorna: Lista con raíces, discriminante, tipo de soluciones, vértice**Validaciones:** $a \neq 0$, coeficientes numéricos**Función 6: `integral_simpson()`****Propósito:** Calcular integral numérica usando regla de Simpson**Parámetros:**

- f: Función a integrar
- a: Límite inferior
- b: Límite superior
- n: Número de subdivisiones (opcional, default = 100, debe ser par)

Retorna: Valor numérico de la integral**Validaciones:** $a < b$, n par y positivo, función válida

4. Estructura del documento R Markdown

Configuración YAML obligatoria:

```
1 ---
2 title: "Libreria QuiCa: Chemistry and Calculus Tools"
3 author: "Su nombre completo y cedula"
4 date: "`r Sys.Date()"
5 output:
6   html_document:
7     toc: true
8     toc_float: true
9     toc_depth: 3
10    number_sections: true
11    theme: flatly
12    highlight: tango
13    code_folding: show
14    df_print: paged
15    fig_caption: true
16 ---
```

4.1. Estructura obligatoria del documento

1. Introducción a QuiCa (200-300 palabras)

- Propósito de la librería
- Audiencia objetivo
- Casos de uso principales
- Instalación e importación (simulada)

2. Módulo de Química

- Introducción al módulo (100 palabras)
- Las 3 funciones de química con documentación completa
- Ejemplo de uso de cada función

3. Módulo de Cálculo

- Introducción al módulo (100 palabras)
- Las 3 funciones de cálculo con documentación completa
- Ejemplo de uso de cada función

4. Testing de funciones

- Suite de tests para validar cada función
- Casos extremos y manejo de errores
- Verificación de resultados esperados

5. Casos de uso integrados

- 3 problemas que usen múltiples funciones de QuiCa
- Soluciones paso a paso con interpretación

6. Conclusiones y trabajo futuro

- Evaluación de la librería desarrollada
- Funciones adicionales propuestas
- Mejoras posibles

5. Estándares de documentación CRAN

Cada función debe incluir:

- **Descripción:** Qué hace la función en 1-2 líneas
- **Parámetros:** Cada parámetro con tipo, descripción y valores válidos
- **Valor de retorno:** Qué retorna y en qué formato
- **Detalles:** Información técnica adicional
- **Ejemplos:** Mínimo 2 ejemplos de uso
- **Referencias:** Fuentes científicas si aplica
- **Ver también:** Funciones relacionadas
- **Validaciones:** Qué errores puede generar

5.1. Formato de documentación requerido

```
1 #' Convertir entre unidades quimicas comunes
2 #'
3 #' Esta funcion convierte valores entre diferentes unidades utilizadas
4 #' frecuentemente en quimica: temperatura, presion y volumen.
5 #'
6 #' @param valor Valor numerico a convertir
7 #' @param de Unidad origen (character). Ver detalles para opciones validas
8 #' @param a Unidad destino (character). Ver detalles para opciones validas
9 #'
10 #' @return Valor numerico convertido a la unidad destino
11 #'
12 #' @details
13 #' Unidades soportadas:
14 #' - Temperatura: "celsius", "kelvin", "fahrenheit"
15 #' - Presion: "atm", "torr", "bar", "psi"
16 #' - Volumen: "L", "mL", "gal", "m3"
17 #'
18 #' @examples
19 #' # Convertir temperatura
20 #' convertir_unidades_quimicas(25, "celsius", "kelvin")
21 #'
22 #' # Convertir presion
23 #' convertir_unidades_quimicas(1, "atm", "torr")
24 #'
25 #' @references
26 #' NIST. (2023). Guide for the Use of the International System of Units.
27 #'
28 #' @seealso
29 #' \code{\link{calcular_concentracion}}
30 #'
31 convertir_unidades_quimicas <- function(valor, de, a) {
32   # Codigo de la funcion aqui
33 }
```

6. Casos de uso integrados

Problemas obligatorios a resolver:

1. **Problema de laboratorio:** Un químico necesita preparar una solución de NaCl 0.5 M a 25°C. Validar condiciones, calcular concentraciones y convertir unidades necesarias.
2. **Problema de cinética:** Dada una función de concentración vs tiempo $C(t) = 10e^{-0,5t}$, calcular la velocidad de reacción (derivada) en $t=2$ y el área bajo la curva (integral) de 0 a 4.
3. **Problema de optimización:** Un reactor químico tiene eficiencia dada por $E(T) = -0,01T^2 + 0,8T - 10$ donde T es temperatura en °C. Encontrar la temperatura óptima y convertirla a Kelvin.

7. Criterios de evaluación

Criterio	Puntos	Descripción
Implementación de funciones	40	Código correcto, validaciones completas, manejo de errores, eficiencia
Documentación CRAN	25	Documentación completa estilo CRAN, ejemplos claros, referencias apropiadas
Testing y validación	15	Tests exhaustivos, casos extremos, verificación de resultados
Casos de uso integrados	15	Solución correcta de los 3 problemas, integración de múltiples funciones
Presentación y estructura	5	Formato R Markdown correcto, organización lógica, estilo profesional
Total	100	

8. Especificaciones de entrega

Archivos requeridos:

1. **Archivo principal:** Tarea3_Apellido1_Apellido2_Nombre.Rmd
2. **Archivo compilado:** Tarea3_Apellido1_Apellido2_Nombre.html
3. **Fecha límite:** Miércoles 1 de octubre, 6:00 PM

9. Recursos de apoyo

9.1. Referencias técnicas

- **Writing R Extensions:** Manual oficial de CRAN para desarrollo de paquetes

- **R Packages by Hadley Wickham:** Guía práctica para desarrollo de librerías
- **Documentación de roxygen2:** Para generar documentación automática
- **testthat package:** Para implementar testing automatizado

9.2. Fórmulas de referencia

Conversiones de temperatura:

$$K = C + 273,15 \quad (1)$$

$$F = C \times \frac{9}{5} + 32 \quad (2)$$

Conversiones de presión:

$$1 \text{ atm} = 760 \text{ torr} = 1,01325 \text{ bar} = 14,696 \text{ psi} \quad (3)$$

Diferencias finitas:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \quad (\text{central}) \quad (4)$$

Regla de Simpson:

$$\int_a^b f(x)dx \approx \frac{h}{3} \left[f(x_0) + 4 \sum_{i=1,3,5\dots}^{n-1} f(x_i) + 2 \sum_{i=2,4,6\dots}^{n-2} f(x_i) + f(x_n) \right] \quad (5)$$

10. Consejos para el desarrollo

Buenas prácticas:

- Desarrolle una función a la vez y pruébela completamente antes de continuar
- Use nombres de variables descriptivos y consistentes
- Implemente validaciones robustas para todos los parámetros de entrada
- Incluya mensajes de error informativos y útiles
- Pruebe casos extremos y situaciones límite
- Mantenga el código limpio y bien comentado
- Verifique que todos los ejemplos funcionen correctamente

Esta tarea simula el proceso profesional de desarrollo de software científico, preparando a los estudiantes para contribuir a la comunidad de código abierto y desarrollar herramientas especializadas para sus disciplinas.