Facultad de Ingeniería,
Universidad de Buenos Aires

[75.12 -95.04] Análisis Numérico I,
Departamento de Computación

Trabajo práctico 1:

1er cuatrimestre, 2019

Grupo 2

Nombre	Padrón	Mail
Rodrigo Cardella Manzur	102596	rodricardellamanzur@gmail.com
Federico del Mazo	100029	delMazoFederico@gmail.com
Froy Dennis Sarasi Merino	101691	froysarasi@gmail.com
Valentina Trotta	102604	Valentina.trotta98@gmail.com

Fecha de entrega: 24 de Abril de 2019

Temática a tratar: Aproximación de problemas matemáticos con diferentes métodos numéricos, y su subsecuente comparación contra algoritmos computacionales.

Palabras Clave: aproximación, cotas, errores, precisión, exactitud

Introducción:

El presente trabajo tiene por objetivo la aproximación de funciones no lineales a partir del uso de diferentes métodos numéricos de aproximación. Adicionalmente, se pretende realizar una comparación y evaluar la efectividad de los diferentes métodos de aproximación utilizados en la teoría, a través del uso de algoritmos propuestos (escritos) en el lenguaje de programación y paquete de software Octave. Finalmente, se propone realizar un análisis en base a los resultados obtenidos de las distintas fuentes de error.

Conceptos teóricos:

En todo momento en el que se desarrolla un cierto cálculo, existen diferentes tipos de errores asociados, sea por la naturaleza de los datos en sí o por cómo se manejan.

Por empezar, siempre se pueden arrastrar errores provenientes de la obtención de los mismos. En ese sentido, no importa cuán bien se adapta una determinada función modelo al comportamiento de lo observado en la experiencia, siempre va a ser impreciso, ya que no es un error de la función modelada, sino de los datos en los que se basa el algoritmo para su construcción. En ese caso, se habla de errores del tipo <u>inherentes</u>, es decir, la incertidumbre que presentan los datos con los que se trabajan, que bien pueden ser considerados como <u>parámetros de entrada</u> de esa función obtenida.

Dado que la optimización y rapidez de los recursos resultan fundamentales en todo cálculo a realizar o en toda proyección a estimar (y de la que se desconocen datos), es quizás lo que inclina a realizar cualquier tipo de operaciones desde la informática, a través de la programación. Ahora bien, las computadoras tienen, sin excepción, una capacidad finita para almacenar datos, por lo que, frente a un resultado infinito para determinado cálculo, puede que el número no se represente exactamente, y la misma realice un <u>truncamiento</u>, es decir, redondear el resultado a determinada cantidad de cifras decimales, y que consecuentemente derive en lo que se conoce como <u>errores de representación</u>. Pero si además se concibe a la operación matemática como una sucesión de cuentas para llegar a un resultado final, en cada paso se puede llegar a estar realizando el truncamiento de un resultado obtenido, y, por lo tanto, ir incrementando la incerteza del cálculo operación a operación. Al error que queda fijado por las sucesivas operaciones que se realizan en determinado procedimiento se lo conoce como <u>error aritmético</u>, que es, al igual que el resto de los casos, inevitable en todo procedimiento matemático que se haga. Todos estos errores afectan a la precisión de determinado <u>parámetro de salida</u>, es decir, un dato con el que, por ejemplo, alguien querría determinar el comportamiento físico de determinado objeto, en función de aquellos valores experimentales que haya medido. Pero además se infiere que en todo momento se hace lo que se denomina una <u>aproximación</u>, puesto que en base a todos los errores que se comentan se obtendrá un resultado más o menos preciso de aquel resultado "exacto", desconocido y codiciado.

Muchas veces, las relaciones que establecen los distintos parámetros de entrada no guardan una relación de linealidad estricta entre sí, describiendo una gráfica mucho más compleja sobre la cual evaluar el comportamiento de la misma en determinado punto desconocido. Consecuentemente, en estos casos, se realiza lo que se denominan aproximaciones por funciones para obtener la información deseada a partir de los datos ingresados. Ello significa que se toman o analizan distintos métodos que permiten obtener un resultado cercano al de la función, sin tener que calcularlo a partir de ella, a propósito de la complejidad algebraica que ello supondría.

A lo largo de esta práctica, se van a proponer diferentes métodos cuyo fundamento de aplicación o no, en experiencias que sean posteriores, dependerán de los resultados y su comparación con el valor esperado teóricamente.

Desarrollo:

En este trabajo se utilizaron tres métodos de aproximación distintos, que se pasan a exponer a continuación. Para facilitar la relación del escrito con las consignas presentadas, se dividirá el apartado en *parte 1* y *parte 2*.

A. Parte 1:

Se utilizó el desarrollo de Taylor truncado a su segundo polinomio. El polinomio de Taylor permite llevar una expresión del estilo de e^x a una función polinómica, según:

$$P_2(x,t) := \sum_{i=1}^{2} \left[\frac{\frac{d^i}{dt^i} f(t)}{i!} \cdot (x-t)^i \right] + f(t)$$

donde $\mathbf{P_n}$ es el polinomio de Taylor a desarrollar para la función **F,** a partir de un $\mathbf{x_0}$=t, y para el que se puede calcular un valor en **x**.

En sí, el polinomio desarrolla su estructura en base a la imagen y derivadas de una determinada función, que se pretende aproximar a través de este método. Como se dijo anteriormente, al ser un truncamiento al segundo polinomio desarrollado en un punto x_0=t, se pueden cometer errores groseros a medida que se pretenda aproximar la función en un x cada vez más lejano a *t*., según queda definido en la función del error del cálculo a través de éste polinomio:

Considerese $F(x)=y=P_2(x)+f(e_x)$. Si $f(e_x)$ resulta despreciable en comparación de $P_2(x)$, y además el desarrollo por Taylor se hace en un $t=\bar{x}$; $F(x)=f(\bar{x})$: [1]

$$y=\bar{y}+(x-\bar{x}).\frac{d(f(\bar{x}))}{dx}+\frac{1}{2}*(x-\bar{x})^2.\frac{d2(f(\bar{x}))}{dx2}+(...)$$ [2]

Como los términos mayores a los primeros dos resultan muy chicos en comparación, resultan en aproximación despreciables obteniéndose:

$$y=\bar{y}+(x-\bar{x}).\frac{d(f(\bar{x}))}{dx}$$

De ello se deduce que:

- El <u>error absoluto</u> es:

$$e_y=y-\bar{y}=error\ de\ x*\frac{d(f[\bar{x}])}{dx},$$

siendo $f[\bar{x}]$ la función aproximada en el punto donde se desarrolla el polinomio, e $y-\bar{y}$ la diferencia que surge de la aproximación de la imagen de la función por el polinomio de Taylor.

- El <u>error relativo</u> es:

$$eR(y)=\frac{ey}{\bar{y}}=\frac{ex}{\bar{x}}*\frac{f'(\bar{x})}{f(\bar{x})}*\bar{x}=\bar{x}*\frac{f'(\bar{x})}{f(\bar{x})}*eR(x)$$

Los resultados obtenidos previamente son muy importantes, no solo porque en ellos radica la base teórica del error por polinomio de Taylor, sino también porque son conceptos que se van a utilizar al resolver los cálculos en el próximo apartado, y en base a ellos esbozar el pertinente análisis.

Para la parte práctica se optó por hacer un algoritmo que reciba como parámetro de entrada el valor al cual aproximar la función, y opcionalmente (gracias a los parámetros nombrados de Octave) el orden del polinomio de Taylor.

---

[1] El punto elegido de desarrollo es un “x” aproximado al que se quiere evaluar la función. Si bien con t=1 puede que la aproximación no quede del todo clara, pensar el siguiente supuesto: se pide encontrar cuánto vale la función para $f(\pi)$, utilizando el polinomio de Taylor. Π es un número irracional infinito, por tanto, en ese caso, se podría adoptar como x aproximado el término truncado al cuarto decimal (3,1416).
[2] Desarrollo en serie infinito.

En vez de traducir todo el desarrollo matemático que se hizo teóricamente, el algoritmo lo que hace es aplicar Taylor mismo, y es por eso que puede recibir cualquier número como orden posible, en vez de limitarnos a dos. Esto permitió hacer distintas comparaciones para ver las aproximaciones para distintos órdenes del polinomio de Taylor. Adicionalmente, programáticamente se hacen los gráficos donde se compara la aproximación por taylor a la función original. Para poder hacer uso de las derivaciones nativas de Octave se tuvo que recurrir al paquete `symbolic.`

El código se encuentra en `aproximacion_por_taylor - ParteI.m`

B. Parte 2:

A propósito de algunas investigaciones hechas por los autores del informe, se llegaron a encontrar algunas fórmulas interesantes en el cálculo de la imagen de una función que tiene una integral definida, y que pueden ser de extrema utilidad partiendo de la función a utilizar:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

En todo caso, lo interesante podría ser comprender por qué es necesario llevar a cabo una aproximación, donde la respuesta más lógica radica en comprender la tediosidad que puede llegar a presentarse en el tener esa integral impropia de segundo orden en el cálculo, dados esos límites de integración.

Los métodos propuestos para la integración radican en:

1. Método del trapecio
2. Método de simpson

Método del trapecio:

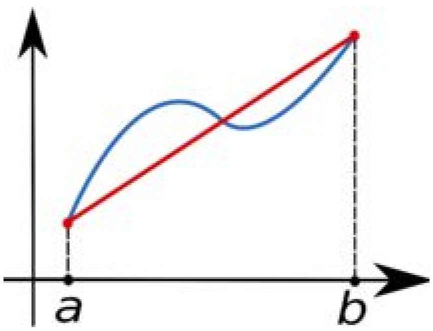
Dados dos límites de integración *a* y *b*, el método de los trapecios consiste en aproximar el valor de la función a partir del integrando:

$$\textbf{Función} = \int_a^b f(x)dx \approx \int_a^b P_1(x)\, dx$$

En el caso del ejemplo previamente ilustrado:

$$f(x) = \frac{2}{\sqrt{\pi}} * e^{-t^2} \quad (1)$$

El razonamiento para llegar a la expresión de cálculo por este método es el siguiente:



Sea una f(x) cuya traza queda descrita por la curva azul, se puede lograr unir a los puntos *a* y *b* (límites de integración) a partir de una recta (P_1(x)) que pase por los puntos:

- (a, f(a))
- (b, f(b))

Para lo que el polinomio de la aproximación queda dado según:

$$P_1(x) = f(a) + \frac{f(b) - f(a)}{b - a}(x - a)$$

Si ahora se reemplazan los resultados obtenidos en **Función,** se obtendrá que:

$$\textbf{Función} \approx \int_a^b \left[f(a) + \frac{f(b) - f(a)}{b - a}(x - a)\right] dx$$

de lo que se termina deduciendo, al resolver la integral que:

$$\int_a^b f(x)dx \approx (b - a)\frac{f(a) + f(b)}{2}$$

Si bien los cálculos correspondientes a la función de aplicación queda reservada para la próxima parte, es conveniente llegar a la generalidad que, de querer aplicar la fórmula para erf(x), entonces la aproximación por este método queda fijada según:

$erf(x) = (x - 0) * \frac{f(0) + f(x)}{2}$, donde *f* viene dada por (1) previamente definido.

Los problemas que podrían surgir en torno a la función considerada, podrían venir dadas según se considera la presencia del número irracional π, aún en la función *f* utilizada para aproximar a erf. A partir de ello, lo que se decidió fue llevar a cabo tres cálculos distintos, con 3 cantidades diferentes de decimales del irracional considerado, para evaluar el efecto que tiene en la precisión del cálculo de la función el hecho de utilizar un π con mayor cantidad de decimales.

<u>Método de Simpson:</u>
Si bien sería incorrecto alegar que Simpson es una mejora a la regla de los trapecios (puesto que no es que el estudio de un método deriva en el otro), lo cierto es que Simpson tranquilamente podría ser visto como un método que tiene en cuenta el avance de la f(x) considerada a lo largo del intervalo *a*,*b*, lo cual supone una mejora en base a cómo se realiza la aproximación. No solo influye de dónde se parte hasta dónde se llega, sino también la forma en la que se llega de un punto a otro.
La demostración de cómo obtener las expresiones finales del método puede resultar tediosa, dado que involucra tener en cuenta al método de interpolación polinómica de Lagrange. Por ello, debe bastar con saber que el polinomio P(x) con el que se trabaja puede aproximarse como:

$$\textbf{Función} = \int_a^b P_2(x)\,dx + \text{término error}$$

Si ahora despreciamos el término del error, P(x) depende sólo de f(x), que, bajo el mismo pretexto del razonamiento del caso anterior, Simpson logró expresar según:

$$\textbf{Función} = \int_a^b P_2(x)\,dx + \text{término error} = \frac{b-a}{6}\left[f(a) + 4f(m) + f(b)\right] + E(f).$$

$$\approx \frac{b-a}{6}\left[f(a) + 4f(m) + f(b)\right]$$

A pesar del cambio de método, los errores de truncamiento por decimales de **π** siguen estando presentes, como así también el error de truncamiento del resultado final de la operación realizada. A lo largo de las próximas secciones, se evaluará, entonces, cuál de estos métodos presentados resulta mejor para la aproximación y se intentará delinear una tendencia común en cuanto a si resulta mejor o peor, a los fines de la aproximación, considerar uno o varios decimales significativos.

Resultados:

A. Parte 1:

En la teoría, la obtención del polinomio de segundo grado se logró de la siguiente manera:

$$p_2(x) := e^t\cos(t) + \frac{(x-t)^1}{1!}\cdot\left(\frac{d}{dt}e^t\cdot\cos(t)\right) + \frac{(x-t)^2}{2!}\cdot\left(\frac{d^2}{dt^2}e^t\cdot\cos(t)\right)$$

Siendo los términos del polinomio en t=0:

$$\frac{d}{dt}e^t\cdot\cos(t) \rightarrow \exp(0)\cdot\cos(0) - \exp(0)\cdot\sin(0) = 1$$

$$\frac{d^2}{dt^2}e^t\cdot\cos(t) \rightarrow -2\cdot\exp(0)\cdot\sin(0) = 0$$

De lo que resulta que:

$$p_2(x) \rightarrow 1 + x$$

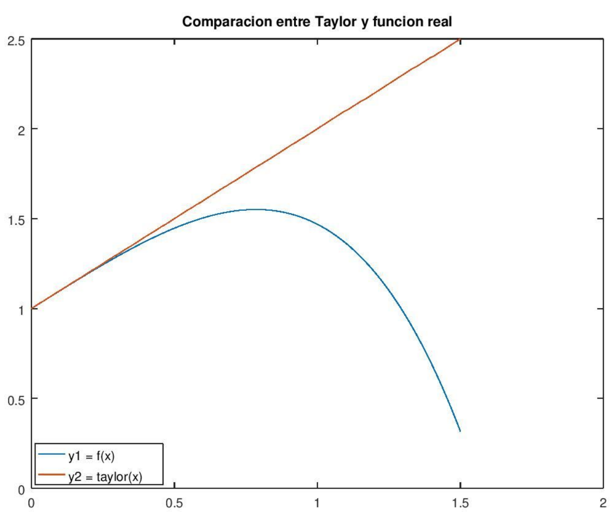
De aplicar la fórmula obtenida para el cálculo de f(0.5) se llega a que:
f(0.5)=1.4469
$P_2(0.5)=1.5$
De lo que se obtiene que la cota de error de aproximación es: $|e_{P(x)}|$=1.5000-1.4469=**0,0531**

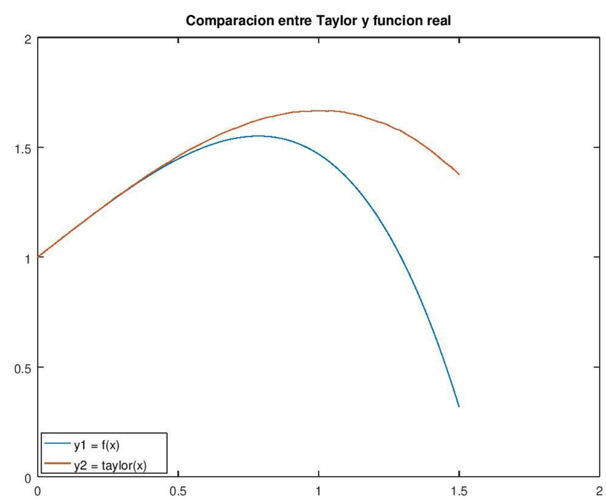
En la práctica, Octave se encarga de los cálculos, y, por ende, no hay que preocuparse por el orden del polinomio considerado para realizar los cálculos. Se pueden ver las siguientes aproximaciones, para el valor 0.5, con distintos órdenes:

```
>> aproximacion_por_taylor(0.5, orden_taylor=2)
valor_real =  1.4469
ans =  1.5000
```

```
>> aproximacion_por_taylor(0.5, orden_taylor=4)
valor_real =  1.4469
ans =  1.4492
>> aproximacion_por_taylor(0.5, orden_taylor=5)
valor_real =  1.4469
ans =  1.4473
>> aproximacion_por_taylor(0.5, orden_taylor=10)
valor_real =  1.4469
ans =  1.4473
```

Como se puede apreciar, a mayor el orden de taylor, mejor la aproximación, como era de esperar. Esto es perfectamente visualizado una vez que se hacen los gráficos de comparación. La diferencia entre un polinomio segundo y un polinomio tercero es inmensa. Para un polinomio décimo la aproximación es tan buena que no se llega a ver una diferencia entre los valores.





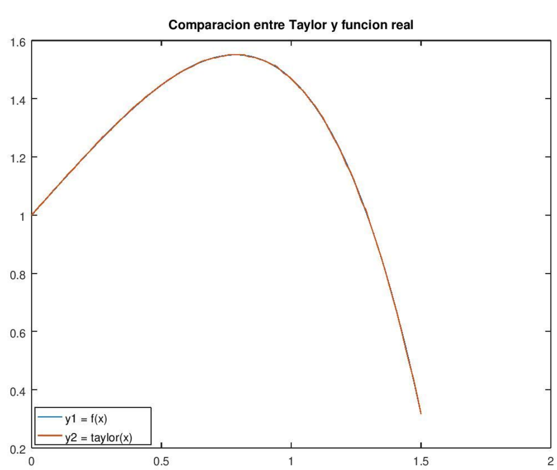


Gráfico I: Grados ascendentes de polinomios de Taylor vs. f(x) a aproximar

Para finalizar este inciso, podría resultar interesante encontrar la cota del error de aproximación de f(x) al usar $P_2(x)$. Para ello, podría ser útil calcular el punto de mayor error relativo del polinomio de Taylor usado como función aproximante en el intervalo [0;1], y de esa manera lograr advertir aquellos puntos donde el error de la función de aproximación resulte mayor.

Se encuentra que el polinomio de Taylor genérico para la función utilizada es:

$$P_2(x)=e^t.\cos(t)+[e^t.\cos(t)-\text{sen}(t).e^t].(x-t)-\text{sen}(t).e^t.(x-t)^2$$

Como se dedujo en el apartado anterior, hay una expresión teórica de cálculo del error relativo en la que $f(\bar{x})$ es $P_2(x)$ para nuestro caso, por lo que, de efectuar los reemplazos correspondientes se llega a que:

$$eR(P_2(x))=\frac{x*e^t(cos(t)-sen(t)-2*sen(t)*x+2t*sen(t))}{e^t.cos(t)+e^t.cos(t)-sen(t).e^t.(x-t)-sen(t).\ e^t.(x-t)^2}*eR(x)$$

Si ahora enfocamos en especializar esa expresión cuando menos confusa y larga en el polinomio hallado (es decir, con t=0), se obtiene una expresión mucho más simple, que se pasa a presentar:

$$eR(P_2(x))=\ \frac{x}{2}*eR(x)\,,$$

De lo que se puede extraer información mucho más clara, dado que en el intervalo considerado, el error relativo va a ser mayor cuanto más grande sea x. Específicamente, el error relativo va a ser lo más grande posible cuando x=1.

Si se calcula el error que existe en la valuación de la función para $P_2(1)$ se podrá, entonces, calcular la tan añorada cota para el error de aproximación:

La cota de error de aproximación para $P_2(x)$ entre [0;1] es: $|e_{P(x)}|=2.0000-1.4687=$**0,5313**

En este inciso, se encontró que el problema matemático era calcular $F(x)=e^x.\cos(x)$ y problema numérico el cómo realizar una aproximación de dicha función, obteniendo como método numérico elegido el cálculo a través del polinomio de Taylor.

B. Parte 2:

En `aprocimacion_erf.m` se encuentra el código de la segunda parte del trabajo. Aquí se definen ambos métodos de aproximación según lo analizado en el apartado anterior, y se comparan contra la función natifa de error integrada en Octave.

Aproximando a erf(1) obtenemos:

a. Utilizando el valor de π provisto por Octave

```
>> aproximacion_erf(1)
valor_erf =  0.84270
aproximacion_trapecio =  0.77174
aproximacion_simpson =  0.84310
```

b. Cálculo con una precisión de 10^{-6}

```
>> aproximacion_erf(1, precision = 6)
valor_erf =  0.842701
aproximacion_trapecio =  0.771743
aproximacion_simpson =  0.843103
```

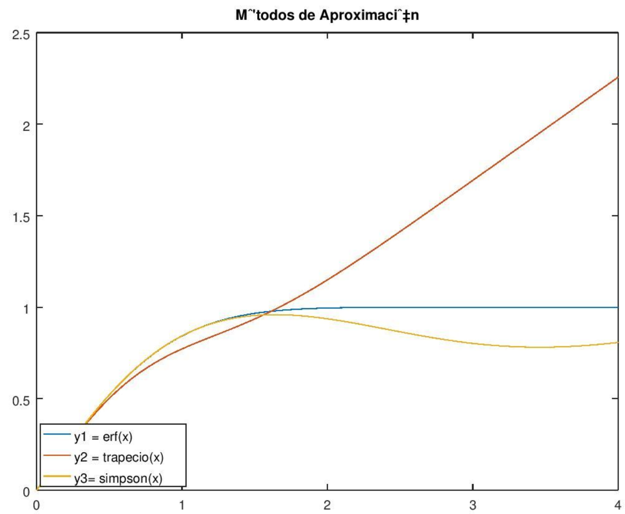
c. Cálculo con una precisión de 10^{-7}

```
>> aproximacion_erf(1, precision = 7)
valor_erf =  0.8427008
aproximacion_trapecio =  0.7717433
aproximacion_simpson =  0.8431028
```

d. Cálculo con una precisión de 10^{-8}

```
>> aproximacion_erf(1, precision = 8)
valor_erf =     8.4270079e-01
aproximacion_trapecio =    7.7174333e-01
aproximacion_simpson =     8.4310283e-01
```

En ocasiones, una representación gráfica permite ratificar algunas posturas, como por ejemplo si efectivamente una función resulta aproximar mejor que la otra a erf(x). Los gráficos realizados en ambos archivos superponiendo las gráficas de las 3 funciones cuya imagen en 1 fueron calculadas son:



<u>Gráfico II:</u> Representación de erf(x) con las dos funciones utilizadas para aproximar su valor.

En este apartado, se aprecia que el problema matemático reside nuevamente en el cálculo de una función (erf(x)), siendo el problema numérico la forma de aproximar su valor. El método numérico, sin embargo, y a diferencia de lo utilizado en la Parte I, ya no resultó ser un polinomio de Taylor, sino el método de Trapecio y Simpson.

Conclusiones

En conclusión, en lo que respecta a los errores a tener en consideración para éste trabajo, se aprecia que, como no se trabajó con ningún dato medible que pudiera afectar a alguno de los métodos utilizados y aproximados "de antemano", entonces resulta absurdo hablar de errores inherentes.
No así, al hablar de errores de redondeo, se debe considerar que como se trabajó con una computadora, en ocasiones la misma pudo tener problemas en la representación de unidades con infinitos números de cifras decimales. Esto es así dado que la misma tiene un límite de memoria, y, por ende, llega un momento donde no puede seguir representando las cifras de un determinado

número infinito en cuestión. Un ejemplo ideal son los irracionales. π, por ejemplo, es un número que tiene interminable cantidad de cifras, pero que en la práctica resulta absurdo aplicarlo en cuentas como tal. Por ende, en algún momento, pasados los n dígitos (delimitados por el sistema de punto flotante, sistema de representación del dispositivo usado), el número indefectiblemente será redondeado, y de allí en más se propagará un error en el cálculo, que justificará decir que, en toda operación donde se lo aplique, en rigor sólo se estará calculando una *aproximación* del resultado verdadero de la misma. Es allí donde, entonces, se ve que en la segunda parte del trabajo se estuvo constantemente arrastrando un error de aproximación por redondeo debido a este número infinito. Otros redondeos, también, han de haberse hecho presentes como denominador común en los resultados de ambas partes al representar el resultado final de las operaciones. Para ponerlo en ejemplos concretos, en la parte 1 se quiso obtener una aproximación de una función donde estaban presentes los irracionales con el número *e*, y en la parte 2 el cálculo de una función con π incluido. La pregunta del vigor, entonces, es si es posible obtener un número no infinito en cuanto a cantidad de cifras a partir de irracionales, donde la trivialidad de la respuesta nos lleva a concluir que todo resultado presente en el informe ha de haber sufrido de un error de aproximación, ya sea con el cálculo hecho por el humano de las operaciones aquí presentes, o ya sea que las mismas hayan sido calculadas en computadoras. En este punto, lo único que se puede destacar es que la única operación de la que se podrá obtener un número entero a partir de un irracional, es elevándolo a la 0, cuando por definición dará 1.
Por otro lado, cabe la mención de los errores discretización. Un ejemplo de este tipo de problemas se encuentra en la primera parte de este informe en el cálculo de la imagen de la función F(x) dada por el polinomio de Taylor. Si uno quiere aproximar a, dígase, por ejemplo, el segundo polinomio de Taylor de una determinada función, ahí se está truncando el problema. En el momento que alguien marca un límite (arbitrario) desde el cual ya no se van a seguir computando términos subsiguientes, allí es donde se cometen este tipo de errores. Ciertamente no se puede correr una aproximación por polinomio de Taylor infinita, y es por eso que, eventualmente, se tendrá que hacer un truncamiento. No así, la segunda parte no presentó errores de truncamiento dado que, como se dijo, el mayor eje de conflictos pasaba por el irracional en la definición de erf(x) que únicamente constaba de errores de redondeo.

Para concluir, se realizarán algunos comentarios, ahora por separado, de algunos detalles interesantes a tener en cuenta al respecto de los métodos presentados:

<u>Parte I</u>:

Polinomios de Taylor: Al utilizar el polinomio de Taylor como método de aproximación, se desarrolla alrededor de un determinado punto (t=0 en este caso), de manera tal que al realizar el cálculo para un cierto valor de *t* que se encuentre muy cercano al punto de desarrollo del polinomio mencionado, el error que se cometa sea relativamente chico, siendo el valor obtenido por este método una buena aproximación al real. Mientras que, si se desea analizar el valor para un punto *t* lejano al mismo (lo cual se ve ejemplificado en este informe al aproximar en t=1), se obtendrá un resultado con un error mayor. Es decir, cuanto más alejado se encuentra el punto de estudio del punto alrededor del cual se desarrolla el polinomio, los resultados que se obtengan serán menos precisos, aproximando con más error y alejándose más el polinomio desarrollado a la función de estudio. En el caso de elegir un polinomio de Taylor de grado mayor derivaría en la mayor precisión de los resultados obtenidos. Es decir, para cada valor fijo de *t*, no importa si está cerca o no de 0, la aproximación va mejorando conforme aumentamos el grado del polinomio, pues, aunque infinitesimal entre términos del polinomio de grados sucesivos, el error de truncamiento será menor.

Parte II:

Como se dijo en "Desarrollo", no es posible ver un método como una mejoría del otro, dado que no son enfoques "evolucionistas" de una misma hipótesis, sino que son herramientas independientes entre sí utilizadas como herramientas de cálculo para un mismo tipo de función. Para cada ocasión, se podrá juzgar oportunamente si un método resultó simplemente mejor que otro.
El método de Simpson, en este caso, presenta una mejor aproximación respecto del de trapecios. Esto se puede atribuir al hecho de que tanto la gráfica de la función definida para Simpson como la de función dada (erf) son similares (ambas son curvas en el intervalo de análisis. Ver gráfico II). En este método, entre dos puntos dados, se traza una curva de forma similar a la que presenta la función original, y, por lo tanto, se encuentran más cercanas entre sí punto a punto, dando un rendimiento de aproximación mucho mejor, en comparación. Si a esto se le suma que en el caso del método de los trapecios, solo se tienen en cuenta los límites de integración iniciales y finales, mientras que Simpson toma una imagen en el punto intermedio entre estos extremos, lo cual permite realizar un análisis de la forma en la que la función avanza entre un punto y otro, parecería tentador decir que Simpson es un mejor método para calcular integrales definidas que los trapecios.
Capaz es cuando menos inapropiado ser tajantes en nuestro punto de vista, puesto que para evaluar efectividades comparativas de los métodos tendrían que haberse efectuado muchos más cálculos y casos de prueba como para esgrimir una conclusión con sustento de argumentos. Por lo pronto, entonces, lo que sí se puede decir es que para el cálculo de integrales de funciones curvilíneas, el método de Simpson probó ser más preciso para el cálculo.

Los métodos numéricos estudiados en este informe sirvieron para calcular aproximaciones de las funciones e integrales propuestas. Quizás llegar a una proximidad y exactitud mayor en los *cálculos requiera de algunos avances en conceptos matemáticos por hacer, como por ejemplo, definir la exactitud. Por lo pronto, los autores del informe nos damos por conformes habiendo logrado cumplir exitosamente los objetivos planteados al comienzo de la práctica, habiendo resultado satisfactoria la aplicación de métodos numéricos en la resolución aproximada de las operaciones matemáticas indicadas-.*

Referencias

- **https://slideplayer.es/slide/2306265/ (14.35-18/4/19)**
- **https://es.wikipedia.org/wiki/Regla_del_trapecio (15.30 - 18/4/19)**
- **https://es.wikipedia.org/wiki/Regla_de_Simpson (16.37 - 18/4/19)**