# 2.1 Integración numérica: Fórmulas compuestas

# 

Da una aproximación del área de la región acotada por la curva

Y=

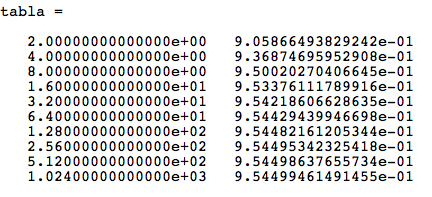
En el intervalo [−2σ, 2σ]:

1. Previo al cálculo numérico haz el cambio de variables t = a la integral que define    el área a calcular.

t=x/ 🡪 3dt= dx / 🡪 dx= donde el intervalo pasa a ser [-2,2]

b)  Usa la regla compuesta de los trapecios para N = 2, 4, 8, 16, 32, 64, . . . .

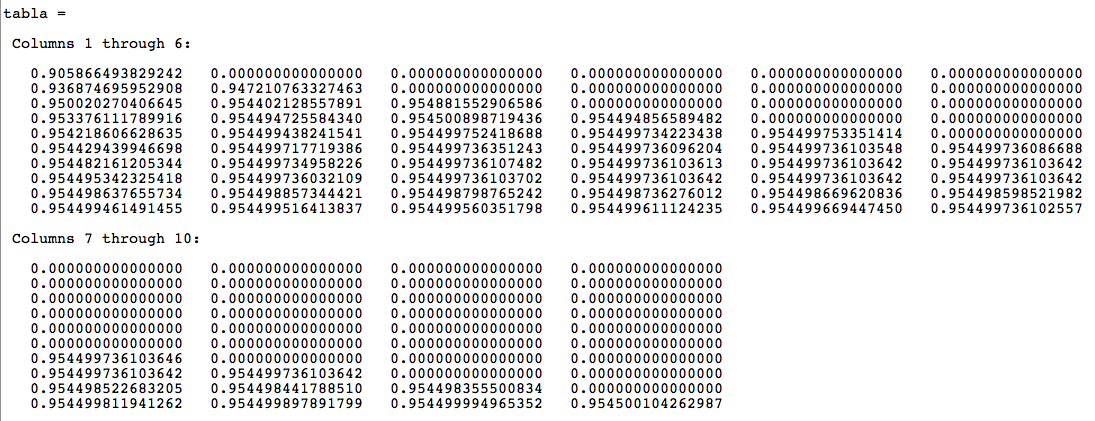
 En este ejercicio he cogido n=10 y he hecho una tabla con los resultados, que es la siguiente:



La primera columna es el valor de las 10 N mientras que la segunda columna es el resultado que da en función de las N.

Este ejercicio se llama trapecio en los códigos de matlab.

c)  Usa el método de Romberg para mejorar la aproximación obtenida.



Haciendo uso del metodo de Romberg para n=10 los resultados obtenidos serán estos.

Romberg en los codigos de matlab

d)  Calcula el valor exacto PRO dado para Matlab

p = normcdf([-sigma, sigma])

PRO = p(2)-p(1)

La probabilidad de que una observación de una distribución normal estándar caiga en el intervalo [-2,2] es :

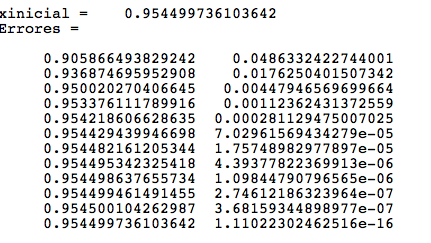
p =
0.0227501319481792 0.9772498680518208

PRO = 0.954499736103642

Lo que quiere decir que la probabilidad de que esto ocurra es de aproximadamente 95.95%

Ej1d en los códigos de matlab

e)  Presenta en tablas los resultados y los errores absolutos obtenidos. Cuantos decimales correctos se obtienen? Comenta los resultados obtenidos.

Donde los 10 primeros son los correspondientes al ejercicio de trapecio, el penúltimo al de Romberg y el último al Normcdf.

|  |  |
| --- | --- |
| TRAPECIO | Decimales correctos |
| 1 | 1 |
| 2 | 1 |
| 3 | 2 |
| 4 | 2 |
| 5 | 3 |
| 6 | 5 |
| 7 | 6 |
| 8 | 7 |
| 9 | 7 |
| 10 | 8 |

|  |  |
| --- | --- |
| ROMBERG | Decimales correctos |
|  | 8 |

|  |  |
| --- | --- |
| Normcdf | Decimales correctos |
|  | 17 |

Vemos que el último método, el de normcdf, es el que tiene el error absoluto más pequeño por lo tanto es el que más se acerca al resultado y la mejor.

El método del Trapecio es el que peor se acerca al resultado y por lo tanto el que tiene los errores más grandes, pero vemos que a medida que la N aumenta, disminuye a su vez el error absoluto.

Errorabsoluto en los códigos de matlab

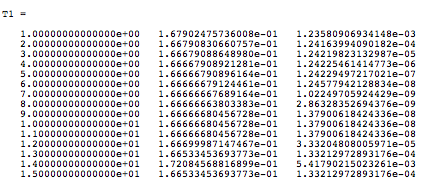
# 2.2 Diferenciación numérica: comportamiento del error

La derivada de la función f(x) = arctan en x = coge el valor f ’( )= 1/6

Considera las dos fórmulas de aproximación de la derivada primera siguientes:

F1: f ’(x0), F2: f ‘(x0)

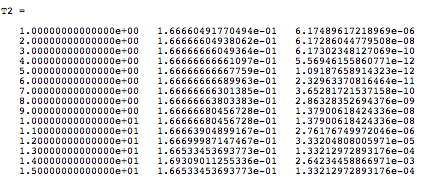
1. Aproxima f ’( ) haciendo uso de la formula F1
2. Para hk = para k=1,2,3…15.
3. Calcula el error absoluto para cada una de las aproximaciones obtenidas
4. Presenta los resultados de los dos apartados previos en una misma tabla (T1).



Donde la primera columna representa el número de iteraciones(15), la segunda la aproximación de la derivada en el punto y la tercera el error absoluto de cada una de ellas.

Ejercicio dosdosuno en los códigos de matlab.

1. Aproxima f ’( ) haciendo uso de la formula F2
2. Para hk = para k=1,2,3…15.
3. Calcula el error absoluto para cada una de las aproximaciones obtenidas
4. Presenta los resultados de los dos apartados previos en una misma tabla (T2).



Donde al igual que la tabla anterior, la primera columna representa el número de iteraciones(15), la segunda la aproximación de la derivada en el punto y la tercera el error absoluto de cada una de ellas.

Ejercicio dosdosdos en los archivos de matlab.

1. Representa los dos errores en una gráfica, con k=1,2,3…15 en el eje de abscisas y log(error) en el eje de ordenadas.



En este gráfico el error absoluto aproximando por F1 esta representado en verde, mientras que el error absoluto aproximado por F2 esta representado en color magenta.

Ejercicio dosdostres en el código de matlab / grafica2

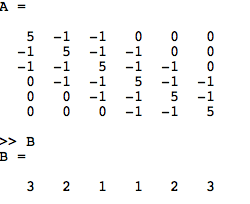
1. Para las dos de derivación hay un paso optimo a partir del cual, si cogemos valores de h más pequeños, los errores comienzan a crecer. Cual es este paso para F1? Y para F2?

En el gráfico anterior vemos claramente que el paso óptimo en caso de F1 será en el punto 5, mientras que para F2 será en el 6, a partir de estos los errores comienzan a crecer.

Todo el ejercicio está hecho con *format long* para poder ver así con mayor claridad la diferencia de resultados de una iteración a la siguiente.

# 2.3 Sistemas lineales: métodos iterativos

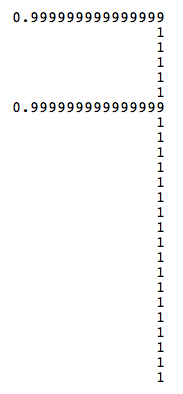
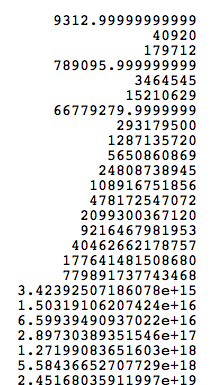
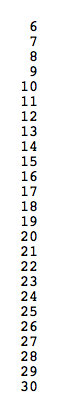
Sea A la matriz y b el vector:



Para todos los ordenes N tal que 6<=N<=30 se pide:

1. Calcula el determinante y el numero de condición de las matrices A.

Para este ejercicio he hecho una tabla con los resultados obtenidos, la primera columna representan las N de 6:30, la segunda columna representa el determinante de A mientras que la última columna representa el numero de condición



Este ejercicio se llama en los códigos de matlab ej3a

Un sistema esta mal condicionado cuando tiene un numero de condición grande, en este caso todos los números de condición se aproximan a 1 por lo que está bien condicionado.

1. Demuestra que X=(1,1,…,1) es solución exacta para cualquier N.

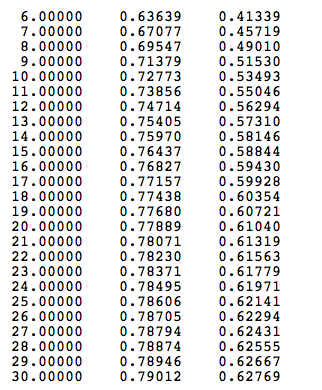
Para este ejercicio he comprobado que A\*x = b, se cumplió esto para todas las N siendo x=(1,1,…,1), por lo que es solución exacta para cualquiera de las N.

Este ejercicio se llama en los códigos de matlab ej3b

1. Estudia la convergencia de los métodos de Jacobi y Gauss-Seidel para la resolución del sistema de ecuaciones lineales. Antes de calcular nada, haz un grafico de evolución del radio espectral de la matriz de iteración de cada uno de los métodos estudiados en función de N.



Donde el radio espectral para el método de Jacobi viene representado en magenta y el radio espectral para el método de Gauss-Seidel en verde. Con este grafico vemos que el método de Gauss-Seidel es el que mejor converge aunque converge para los dos ya que tanto uno como otro es menor que 1.



En esta tabla salen representados en la primera columna las N , en la segunda columna el radio espectral por el método de Jacobi para cada una de las N y en la última en radio espectral por el método de Gauss-Seidel para cada una de las N.

Este ejercicio en los códigos de matlab se llama ej3c

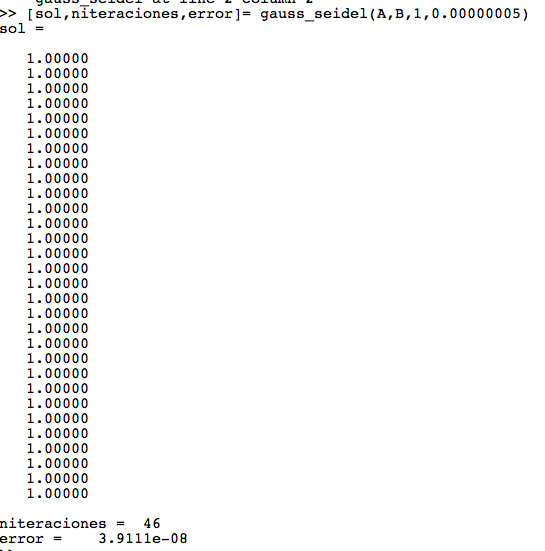
El gráfico se llama graficoej3c

1. Haciendo uso de un método iterativo de resolución de sistemas de ecuaciones lineales, determina la solución de X del sistema A(N)X=B(N) haciendo uso de la aritmética de coma flotante de Matlab y redondeando. Coge N=6…30. Previamente haz un gráfico de la evolución del radio espectral de la matriz de iteración del método escogido en función de N. Explica las ventajas e inconvenientes del método para este caso concreto.

Para este ejercicio he elegido utilizar el método de Gauss-Seidel ya que viendo los resultados anteriores vemos que es más eficiente que el método de Jacobi, para este el radio espectral de sus matrices de iteración, evoluciona de esta manera:



graficoej3d en los códigos de matlab.



Esta sería la solución obtenida por este método, donde se pueden comprobar que todas las x son 1, el numero de iteraciones 46 y el error 9.9111e-08, que es un error muy pequeño.

Ej3d en los apartados de matlab.

# 2.4 Sistemas lineales: mínimos cuadrados

El nivel del agua del Mar del Norte queda determinada fundamentalmente por la llamada marea M2 el periodo es aproximadamente de 12 horas y su expresión aproximada es:

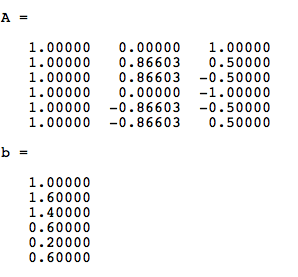
H(t) = h0 + a1sin() + a2cos()

Queremos ajustar los valores de h0, a1, a2 según la tabla de medidas siguiente:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| t(h) | 0 | 2 | 4 | 6 | 8 | 10 |
| H(litros) | 1.0 | 1.6 | 1.4 | 0.6 | 0.2 | 0.6 |

Se pide:

1. Plantea el problema de determinan los valores de h0, a1, a2 como un sistema lineal Ax=b, i.e. define los componentes del vector de incógnitas x, expresa A y b en función de ti y Hi



La matriz A se crea sustituyendo cada punto de t(h) en la fórmula que nos dan. Por ejemplo, la primera fila será el resultado de sustituir en cada t de la formula el 0.

A(1,1)=1 y será igual para toda la primera columna porque h0 nunca cambia, y tiene valor de 1. A(1,2)=0 ya que el seno de 0 es 0, mientras que A(1,3)=1 ya que el coseno de 0 es igual a 1.

1. Resuelve el problema haciendo uso de las ecuaciones normales. Da la solución obtenida y calcula el vector residuo.

En este caso la solución obtenida será igual a h0=0.9000 ; a1=0.63509; a2=0.23333.

Mientras que el valor del vector residuo será igual a 0.21602.

1. Resuelve el problema lineal Ax=b haciendo uso de la factorización QR. Da la solución obtenida y calcula el vector residuo.

Haciendo uso de la factorización QR la solución obtenida es h0=0.014132 ; a1= 0.202289 y a2=-1.142039.

Y el resultado del vector residuo es 3.3154.

1. Comenta las diferencias entre las soluciones encontradas, compara los residuos de los apartados (b) y (c). Da la ecuación H(t) resultante.

Haciendo uso de las ecuaciones normales tenemos que los valores de la solución son mucho mayores que utilizando el método de factorización QR, sin embargo el vector residuo de este último es mucho mayor por lo que es mucho más adecuado para este caso utilizar las ecuaciones normales.

Para comparar los residuos he hecho un gráfico que muestra mejor la diferencia que hay entre ellos y la proximidad a la recta de los mismos.



En esta tabla sale representado el residuo obtenido empleando las ecuaciones normales en magenta mientras que el verde representa el residuo obtenido con el método QR. Vemos que será el residuo del apartado b el que más se aproxime a los puntos por lo tanto será el más favorecedor y el que escojamos.

Por lo tanto la ecuación H(t) resultante será :

H(t)= 0.9000 + 0.63509\*sin() + 0.23333\*cos()

Este ejercicio está guardado como ejercicio4mcentrega en los archivos de matlab, la tabla como grafico 4.