

Apuntes de *Matemáticas*

Eduardo Espuch



Universitat d'Alacant
Universidad de Alicante

Resumen

Documento que reunirá todos los conceptos matemáticos vistos siguiendo un orden alternativo al del grado, siendo la intención de esto el poder entender conceptos de una forma progresiva y mucho mas clara, ya que están relacionado entre si.

Ya que habrán conceptos pertenecientes a matemáticas 1, matemáticas 2 y matemáticas discreta en este documento, ademas de conceptos generales, se indicará al comienzo si han sido usados en cada asignatura (m1, m2 y md hará referencia a cada asignatura, entenderemos que todo lo que se vea serán conceptos generales).

PD: habrán comentarios/referencias o erratas ortográficas, no tenerlas en cuenta ya que el hacer un documento así es un tedio y a veces se cuelan tonterias, pero los conceptos de por si están revisados y se puede comprobar que son ciertos.

M1 esta pendiente de cambio ya que sería interesante ver con mas calma cada concepto.

Índice

1. Aritmética y álgebra	4
1.1. Introducción a lo básico	4
1.1.1. Método matematico	4
1.1.2. Introducción a la lógica	5
1.1.3. Herramientas de demostracion WIP	6
1.1.4. Teoria de los conjuntos	7
1.2. Grafos, md	11
1.2.1. Accesibilidad y conectividad	13
1.2.2. Arboles	15
1.2.3. Grafos ponderados	16
1.2.4. Método de Floy-Warshall	19
1.2.5. Arboles generadores de minimo peso	21
1.2.6. Ejemplo agrupado	22
1.3. Álgebra y aritmética numérica simple	25

1.4. Ampliación de los conjuntos numéricos, md	25
1.4.1. Aritemtica modular	26
1.4.2. Congruencias, relacion binaria de equivalencia	30
1.5. Repaso simbologia	33
2. Geometría	33
3. Introducción al álgebra lineal, m1	34
3.1. Resolucion de SL por metodos directos	35
3.2. Resolucion de SL por metodos iterativos	37
4. Matrices, m1	37
4.1. Conocimientos básicos	37
4.2. Matrices compuestas	39
5. Espacio vectorial, m1	40
5.1. Espacio y subespacio vectorial	40
5.2. Matrices y espacios vectoriales	41
6. Valores y vectores propios de una matriz, m1	42
7. Análisis	43
7.1. Conceptos básicos	43
7.1.1. Propiedades de una funcion	44
7.1.2. Limites, continuidad y asymptotas	45
7.2. Derivadas	47
7.2.1. Aplicaciones para una variable	49
7.2.2. Derivadas con varias variables	49
7.2.3. Aplicaciones para varias variables	49
7.3. Integrales	49
8. Resolucion por medios numericos	50
8.1. Teoria de errores	50
8.1.1. Digitos significativos y digitos exactos	51
8.1.2. Relacion entre error relativo y digitos exactos	52
8.1.3. Error de redondeo	52
8.1.4. Operaciones con errores	53
8.2. Resolucion de ecuaciones, algoritmos	54
8.2.1. Teorema de conservacion del signo	54
8.2.2. Teorema de bolzano	54

8.2.3. Metodo del punto fijo	55
8.2.4. Metodo de la secante	55
8.2.5. Metodo de regula false	56
8.2.6. Metodo de newton	56
8.2.7. Ejemplos de implementacion	57
9. Estadística y probabilidad	57

1. Aritmética y álgebra

Con este tema nos introduciremos en el mundo de las matemáticas para comprender que no son solo números, tratamos con elementos y conjuntos de estos y que las operaciones con las que solemos tratar fácilmente contienen conceptos que utilizamos en otros ámbitos más complicados

Para explicar este tema seguiremos una estructura en la que:

1. Debemos entender que siempre trabajamos con conjuntos y elementos, teniendo que conocer el significado de estos conceptos y su utilidad en las matemáticas aplicadas.

Para ello estudiaremos la teoría de conjuntos, la lógica, el álgebra de conjuntos y las estructuras algebraicas. También introduciremos brevemente los conjuntos numéricos que usamos a diario.

2. Partiendo del punto anterior, nos introduciremos en el mundo del álgebra y aritmética numérica, aquella en la que trabajamos sobre conjuntos numéricos,

Trataremos los polinomios, los sistemas de ecuaciones y el concepto de matrices, siendo un buen ejemplo para mostrar la importancia de las estructuras algebraicas.

3. Con intención de ampliar nuestros conocimientos sobre los conjuntos numéricos, estudiaremos con más detalle propiedades que se observan en dos conjuntos muy importantes.

El conjunto de números complejos y la trigonometría nos proporcionará mayor información respecto a este campo de las matemáticas y el conjunto de números enteros y la congruencia aritmética será un punto muy denso, aunque tratemos con los números enteros de forma habitual, existen muchos conceptos de los cuales desconocemos por ahora su significado pero lo usamos igualmente.

Se recomienda que, a la hora de seguir la lectura, se vayan probando con varios ejemplos en una hoja aparte y comprender el concepto que se está viendo.

1.1. Introducción a lo básico

Si partimos de lo básico, debemos conocer que es el método matemático, introducirnos a la lógica y finalmente ver la teoría de conjuntos y todo lo asociado a ésta. Es importante saber que todo símbolo tiene un significado, no es lo mismo usar $[]$ que $\{ \}$, con lo cual fijarse en cómo se definen todos los conceptos.

1.1.1. Método matemático

Método matemático, el método cuya conclusión de la veracidad de un resultado matemático debe pasar por la realización de su demostración.

La notación abstracta es de gran importancia, de ahí la formalización de un lenguaje cuya simbología es muy precisa y clara respecto a su significado. Es como se ha dicho anteriormente, existe una gran importancia en conocer este lenguaje de símbolos y notaciones, además de las demostraciones.

Llamaremos proposición en general a una oración del lenguaje de la que tiene sentido afirmar que es verdadera o falsa. Atendiendo a su naturaleza, en matemáticas se denominan:

- Axioma: proposición que se considera siempre cierta debido a su fácil demostración (una demostración es trivial si es obvia y sencilla de obtener) y construirá el punto de partida para otros tipos de proposiciones.
- Teorema: sentencia relativa a objetos matemáticos (elementos, conjuntos, axiomas, ...)

- Proposición: Teorema que se considera trivial o de menor importancia.
- Corolario: Teorema que se considera consecuencia directa o inmediata del resultado obtenido.
- Lema: Teorema que va a ser utilizado como recurso en la resolución de otro teorema. Podríamos considerar el lema como objeto matemático perteneciente a la sentencia de objetos matemáticos de un teorema más amplio.

Aunque suene sencillo, llamaremos definición al significado de un término u objeto matemático. Debemos saber que la tesis (lo que podemos afirmar que es cierto si se cumple la hipótesis y es objeto de estudio) y la hipótesis (conjunto de condiciones que suponemos que son ciertas) pueden formar parte de la definición o del proceso de definir algún objeto.

Las proposiciones no pueden ser ambiguas, o son ciertas para unas condiciones o falsas, no cabe lugar para la subjetividad (p.e. $a + b = c$ tienen condiciones que hace la proposición cierta y otras que las hace falsa, pero 'la habitación es grande' el relativo al observador y, por lo tanto, de veracidad ambigua). A las proposiciones las denotaremos por letras minúsculas (p.e. $p : a + b = c$).

1.1.2. Introducción a la lógica

La lógica es el campo de las matemáticas con el que se han definido un número de relaciones y operaciones con los que podemos estudiar la veracidad de una demostración, sirven de cementación para el método matemático. En ocasiones se hablarán de elementos y conjuntos, estos por ahora basta saber que consisten en objetos y agrupaciones de estos, se ampliará su significado en la teoría de conjuntos.

La base de la lógica reside en los operadores (o conectores) lógicos, nexos con los que combinar proposiciones y obtener proposiciones más complejas. Dadas las proposiciones p y q , veamos algunos operadores:

Operadores lógicos monarios

Actúan sobre una proposición

Negación: $\neg p$ o \bar{p} (no p)

Si p es cierta,

$\neg p$ será falsa

y viceversa

Operadores lógicos binarios

Combinan varias proposiciones, pueden encadenarse

Disyunción: $p \vee q$ (p o q), falsa si ambas lo son

Conjunción: $p \wedge q$ (p y q), cierta si ambas lo son

Implicación: $p \Rightarrow q$ (p solo si q ,...) falsa si p cierta y q falsa

Equivalencia: $p \Leftrightarrow q$ (p equivale a q ,...) hace falta?

Se verá un ejemplo más adelante pero cabe destacar que hay muchas formas de denotar cada conector, se recomienda buscar y anotar varios ejemplos de cada tipo de operador lógico.

ESPACIO PARA ANOTAR EJEMPLOS DE DENOTACIONES

Las tablas de verdad serán una forma de expresar las operaciones lógicas, considerándose todas las posibles combinaciones. Si tenemos n proposiciones distintas, tendremos 2^n posibles combinaciones, veamos lo con un ejemplo:

		$a :$	$b :$	$c :$	$d :$	$e :$	$f :$
p	q	$\neg p$	$\neg p \vee q$	$p \wedge q$	$p \Rightarrow q$	$(p \Rightarrow q) \wedge (q \Rightarrow p)$	$p \Leftrightarrow q$
1	1	0	1	1	1	1	1
1	0	0	0	0	0	0	0
0	1	1	1	0	1	0	0
0	0	1	1	0	1	1	1

a: No p

b: No p o q

c: p y q

d: p es suficiente para q

e: q solo si p y p solo si q

f: p es necesaria y suficiente para q

Las proposiciones e y f son equivalentes, eso es porque f es composición de e, se intentara mas adelante realizar una tabla con mas relaciones así.

Una tabla de verdad sera una tautología (o cumple la ley lógica) si para todas las interpretaciones, sin importar las condiciones iniciales, el resultado es cierto. Si no todas son ciertas diremos que es contingencia y si todas son falsas entonces será una contradicción.

Diremos que una expresion viene en su forma proposicional cuando dicha expresion aparece una variable x , cuyos valores pertenecen a un conjunto referencial E (i.e. $x \in E$), y en la que podemos sustituir la variable x por cualquier elemento de E obteniendo de tal forma una proposicion que depende de una variable ($p(x)$ con $x \in E$).

Si esta expresión os suena es porque la usamos a diario asumiendo que se cumple pero si nos parasemos a pensarlo y demostrarlo invertiríamos mucho tiempo en ello. Las funciones, geometria, analisis,... parten de proposiciones, los conjuntos y elementos (dos conceptos que veremos a continuacion) y de unas reglas demostradas de como estos interaccionan entre ellos, todo definido por la teoria de conjuntos.

1.1.3. Herramientas de demostracion WIP

Reunimos el significado de diversos conceptos matematicos para realizar demostraciones, ademas de tambien explicar varios de los metodos para demostrar una proposicion. Este apartado no sera obligatorio pero se recomienda mirarlo ya que en posibles explicaciones es posible que se necesiten y salvo que sea estrictamente necesario ver el desarrollo, se daran por sabido.

1. Tecnicas de demostracion:

- Directa
- Indirecta
- Por contraposicion o el contrareciproco
- Por contraejemplo
- Por reduccion al absurdo (RAA)
- Por casos
- Por induccion

- Dado un conjunto incluido en un cuerpo en el que se define una relacion binaria de orden (RBO), conceptos que mas adelante veremos pero basta con saber que hablamos de un conjunto de numeros ordenados como el de los reales, diremos que $a \in \mathbb{R}$ es cota (superior o inferior) del subconjunto dado (D) si se cumple que $\forall d \in D \leq a$ (superior, es el elemento mas alto del conjunto) o $a \leq d$ (inferior, es el elemento mas bajo del conjunto).

La menor de todas las cotas superior recibe el nombre de extremo superior o supremo, es decir, el primer elemento que podamos definir como cota superior y que pertenezca al conjunto D sera supremo. Por otro lado, llamaremos extremo inferior o infimo a la mayor de todas las cotas inferiores, siendo el primer elemento que podamos definir como cota inferior y que pertenezca al conjunto D .

- Teorema de conservacion del signo:

4. Teorema de Bolzano

5. Teorema de valor intermedio

1.1.4. Teoria de los conjuntos

Rama de las matematicas que estudia las propiedades y relaciones de objetos matemáticos conocidos por conjuntos: colecciones abstractas de objetos. Si un conjunto contiene una coleccion de conjuntos se le denomina familia de conjuntos. Los conjuntos y sus operaciones mas elementales son una herramienta basica en la formulacion de cualquier teoria matematica, siendo la teoria de conjuntos mas que suficiente para construir el resto de objetos y estructuras de interes en matematicas.

Algebra de conjuntos: operaciones basicas para manipulas los conjuntos y sus elementos. Los conceptos basicos para iniciarse en el algebra de conjuntos son:

$x \in A$	x pertenece a A	$A = \{a, b, c\}$	Conjunto por extension
$A = \{x \in \{a, b\} / x \neq b\}$	Conjunto por comprehension	$A_i \subset \mathcal{C} / \mathcal{C} = \{A_i, i \in I\}$	A_i esta incluido en la familia \mathcal{C}
$A = \emptyset$	Conjunto vacio	$A \neq \emptyset$	Conjunto no vacio
$A = \{a\}$	Conjunto singular	$A = \{a, b\}$	Pareja
I es el conjunto de indices, notar como se define con $\{\}$ los conjuntos, \emptyset es NULL			

IMPORTANTE, cuando un elemento pertenece a un conjunto, usamos el simbolo \in pero cuando un conjunto (llamado tambien subconjunto) pertenece a otro conjunto, o mejor dicho, esta INCLUIDO en otro conjunto, se usa el simbolo \subset . Estos simbolos no deben de mezclarse ya que sus significados son muy especificos.

Sabiendo al menos esto, definamos ahora las funciones elementales que tenemos en algebra de conjuntos:

1. Union: combina dos o varios conjuntos en uno de tal manera que se forma un conjunto destino que contiene todos los elementos de sus conjuntos fuente, siendo E un conjunto referencial en el que todo elemento y conjunto esta definido, se denota la union para un par de conjuntos o para una sucesion de conjuntos de las siguientes maneras:

Un par de conjuntos: $A \cup B, \{x \in E | x \in A \vee x \in B\}$

Una familia de conjuntos: $\cup_{i \in I} A_i, \{x \in E | x \in A_i, \exists i \in I\}$

2. Interseccion: selecciona los elementos que estan a la vez en todos los conjuntos fuente. Dado un conjunto referencial E , denotaremos la interseccion de un par de conjuntos y de una familia de conjuntos de las siguientes maneras:

Un par de conjuntos: $A \cap B, \{x \in E | x \in A \wedge x \in B\}$

Una familia de conjuntos: $\cap_{i \in I} A_i, \{x \in E | x \in A_i, \forall i \in I\}$

3. Particion de A: dado una familia de conjuntos $\mathcal{C} = \{A_i, i \in I\}$, si la union de todos los subconjuntos dan lugar a un conjunto A , y la interseccion de dos subconjuntos cualesquiera de el vacio, estos subconjuntos A_i formaran una particion de A .

$$\mathcal{C} = [A_i, i \in I], \cup_{i \in I} A_i = A, A_i \cap A_j = \emptyset \forall i, j \in I, i \neq j$$

4. Diferencia: conjunto que se forma a partir de los elementos de A pero no esten en B .

$$A \setminus B = \{x \in E | x \in A \wedge \neg(x \in B)\}$$

5. Conjunto complementario: la diferencia de A entre el conjunto referencial, es decir, todos los elementos que no pertenezcan a A . $\bar{A} = A$
 $E = \{x \in E | \neg(x \in A)\}$

6. Inclusion: comentado por encima anteriormente, diremos que A esta incluido en B cuando $\forall x \in A, x \in B$. Todos los conjuntos que definamos deberan estan incluidos en el conjunto referencial y si $A \subset B \wedge B \subset A \Leftrightarrow A = B$.

Describamos las que seran las propiedades elementales, estas nos sonaran porque, como ya se ha comentado anteriormente, las usamos dandolas como ciertas gracias a la teoria de conjuntos:

1. Reflexiva: $A \subset A$. Un conjunto esta incluido en si mismo. Esta propiedad basicamente se usa para indicar que un elemento esta en dos conjuntos,
2. Antisimetrica: $[(A \subset B) \wedge (B \subset A)] \Leftrightarrow (A = B)$ es trivial.
3. Transitiva: $[(A \subset B) \wedge (B \subset C)] \Rightarrow (A \subset C)$ es trivial.
4. Conmutativa: $A \oplus B = B \oplus A$ siendo \oplus una operación binaria elemental definida. Significa que el orden de los operandos no altera la solucion.
5. Leyes de Morgan: $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$ y $\overline{(A \cap B)} = \overline{A} \cup \overline{B}$, demuestran que existe una operacion equitativa al negar otra operacion.
6. Asociativa: $(A \oplus B) \oplus C = A \oplus (B \oplus C)$, una cadena de la misma operacion se puede alterar el orden de ejecucion y se obtendra el mismo resultado.
7. Distributiva: $(A \oplus B) \ominus C = (A \ominus C) \oplus (B \ominus C)$ Podemos distribuir una operacion compleja en varias simples y viceversa.
8. Partes de un conjunto dado: definido tal que $\mathcal{P}(A) := \{x | x \subset A\}$, las partes de un conjunto dado con n elementos son las 2^n combinaciones posibles que se pueden formar con todos los elementos. Basicamente son todos los subconjuntos que seamos capaces de expresar con los elementos de un conjunto. Si el conjunto es vacio, podemos decir que el propio vacio es una parte del conjunto y $\mathcal{P}(\emptyset)$ seria un conjunto singular, siempre debemos considerar el elemento vacio como una posible combinacion.

Definamos mas conceptos para poder iniciar la explicacion de que es una correspondencia... , todo lo que veremos ahora tendra mucha similitud con las funciones que acostumbramos a usar pero se debe de tener claro que con las funciones, trabajamos sobre un concepto llamado cuerpo y con propiedad demostradas, lo que veremos ahora son los componentes que nos permiten realizar la demostracion de estas propiedades y el poder definir facilmente elementos.

El par ordenado, denotado con $(a, b) = \{\{a\}, \{a, b\}\}$, $a \in A \wedge b \in B$, es un conjunto en el cual el orden de los elementos es de gran importancia, Para hacerte a la idea, si tenemos (x, y) , el valor y dependera de que valor tenga x , es posible que y tenga varios x relacionados y viceversa, se vera mas adelante que pasa si esto sucede.

El producto cartesiando, denotado con $A \times B = \{(a, b) | a \in A \wedge b \in B\}$ es un conjunto compuesto por pares ordenados (o componentes del producto cartesiano) en los que se toman todos los elementos de los conjuntos iniciales. Un claro ejemplo y con el que solemos trabajar seria el producto cartesiano $\mathbb{N} \times \mathbb{N} = \mathbb{N}^2$, es decir, tomamos cada punto positivo que se pueda definir.

1. El producto cartesiando de un conjunto cualquiera y un conjunto vacio, sin importar el orden de como se operen, dara como resultado el vacio,
2. Para el producto cartesiano, la propiedad asociativa y conmutativa no se cumple. Si se cumpliera, tratamos con casos especiales o que todos los conjuntos son iguales.
3. El producto cartesiano de un numero finito de conjuntos finitos es, a su vez, finito. La demostracion se realiza usando el cardinal de los conjuntos ($\text{card}(A) = n^\circ$ de elementos del conjunto).

Dado dos conjuntos A y B y una relacion binaria (operacion que relacion dos conjuntos) \mathbb{R} , llamaremos correspondencia al subconjunto $G \subset A \times B$, $G := \{(a, b) \in A \times B \mid a \mathbb{R} b \wedge (a \in A \wedge b \in B)\}$. Para la definicion anterior debemos entender:

1. A es el Dominio o conjunto inicial.
2. B es el Codominio o conjunto final.
3. b es imagen de a y a es preimagen de b si $(a, b) \in G$
4. Llamaremos conjunto imagen al conjunto de imagenes o de elementos del codominio que esten definidos en la correspondencia, es decir, $Im = \{b \mid (a, b) \in G\}$
5. Llamaremos conjunto preimagen al conjunto de preimagenes o de elementos del dominio que esten definidos en la correspondencia, es decir, $Pm = \{a \mid (a, b) \in G\}$

Si cada elementos de del dominio tiene una unica imagen, diremos que la correspondencia es una aplicacion. En matematicas aplicadas, una aplicacion es una funcion y la simbologia es la misma, denotamos por f a la aplicacion y la definimos tal que:

$$\begin{aligned} f : A &\longmapsto B \\ x &\longmapsto f(x) \end{aligned}$$

Es mas, diremos que es inyectiva si elementos distintos del dominio tienen imagenes distintas, o dicho de otra maneras, dos imagenes iguales existitran si y solo si sus preimagenes son las mismas. Tambien diremos que una aplicacion es suprayectiva si cada elementos elemento del codominio tiene al menos una preimagen.

Diremos que es biyectiva si cada elemento del codominio tiene exactamente una preimagen o, lo que es lo mismo, es inyectiva y suprayectiva a la vez. Es importante aclarar algunas cosas para continuar, entender los siguientes enunciado para correspondencias con aclaraciones sobre las aplicaciones:

1. La aplicacion te dice que el conjunto preimagen y el dominio (el conjunto A), son lo mismo pero el conjunto imagen y el codominio (el conjunto B) no tienen porque ser estrictamente lo mismo. Si lo son, entonces la aplicacion tiene que ser suprayectiva o biyectiva.
2. Llamaremos identidad a la correspondencia cuya preimagen equivale a la imagen, ambos pertenecientes al mismo conjunto.
3. Llamaremos composicion a la encadenacion de correspondencias usando la imagen de una de ellas como preimagen de la otra. Es importante entender que, usando aplicaciones, la primera aplicacion al menos sea inyectiva, ya que la segunda requiere coger cada elementos del conjunto inicial para ser aplicacion. Existen muchas propiedades sobre la composicion de aplicaciones pero todas se pueden obtener a raiz de probar cada aplicacion si es inyectiva, suprayectiva o biyectiva.
4. La restriccion de una aplicacion consiste en definir un subconjunto del dominio y definir una con el subconjunto como conjunto inicial.
5. Dada una correspondencia, definiremos a la correspondencia inversa como al conjunto de componentes con el orden invertido que pertenecen a esta. En aplicaciones hay que asegurarse al menos que es suprayectiva para definir la aplicacion inversa y, si fuese biyectiva, podemos asegurar que la aplicacion inversa tiene inversa y esta es igual a la aplicacion inicial.

Puede darse el caso que trabajemos sobre un producto cartesiano de un conjunto por si mismo, A^2 , en ese caso denotaremos a la relacion binaria con (A, G) con G siendo una correspondencia incluida en el producto cartesiano. Diremos que la relacion binaria puede ser:

- R. Reflexiva $(x, x) \in G, \forall x \in A$, para entendernos, los elementos de una diagonal en una matriz cuadrada son iguales tanto si lees a_{ij} como a_{ji} ya que $i = j$, es una explicación burra pero ayuda.
- S. Simétrica $(x, y) \in G \Rightarrow (y, x) \in G, \forall x, y \in A$, prácticamente lo que viene a ser un espejo, en las posiciones inversas se ve lo mismo.
- A. Antisimétrica $[(x, y) \in G \wedge (y, x) \in G] \Rightarrow x = y, \forall x, y \in A$, requiere que sea reflexiva ya que lo que dice es que la única forma de que dos pares ordenados el cual uno de ellos es el inverso del otro, y ambos están definidos en el grafo, se debe a que estos pares sean los mismos y se cumpla la reflexiva.
- T. Transitiva $[(x, y) \in G \wedge (y, z) \in G] \Rightarrow (x, z) \in G, \forall x, y, z \in A$, un ejemplo sería si $a < b$ y $b < c$, entonces afirmamos que $a < c$ con $a, b, c \in \mathbb{R}$.

Con estas propiedades descritas, podemos estudiar las RBE y las RBO según que propiedades cumpla:

RBE: Relación binaria de equivalencia, cumple Reflexiva, Transitiva y Simétrica podríamos definirlo como el conjunto de elementos equivalentes entre sí.

RBO: Relación binaria de orden, cumple Reflexiva, Transitiva y Antisimétrica indica que en un conjunto hay un determinado orden a seguir, las cotas son un ejemplo de esta relación de orden.

NOTA TEMPORAL: Para estudio de grafos no es necesario pero basta con saber que las RBE te dicen igualdades y las RBO desigualdades, dicho muy a lo bruto y sin nada en lo que basarme, únicamente un recordatorio propio para futuras explicaciones.

Sabiendo que son correspondencias, que son las relaciones binarias y las propiedades descritas para ellas veamos el resultado final que se obtiene con la teoría de conjuntos, las estructuras algebraicas.

Las estructuras algebraicas son una lista ordenada de elementos cuyo primer elemento es un conjunto no vacío y los siguientes a este son aplicaciones aplicables a los elementos del conjunto dado. Las aplicaciones, también llamadas operaciones binarias en un conjunto no vacío A , son aplicaciones de A^2 en A , que denotamos mediante el par ordenado (A, \oplus) , siendo \oplus una relación binaria definida.

$$(A, \oplus) := \{(x, y) \in A^2, x \oplus y \in A\}$$

y podemos decir que la correspondencia es A^2 porque, al ser una aplicación se da que todo el dominio está incluido.

Veamos las propiedades que se pueden dar para la operación binaria y a continuación definiremos dos operaciones con las que trabajamos en estructuras algebraicas:

1. Asociativa	$a * (b * c) = (a * b) * c$
2. Conmutativa	$a * b = b * a$
3. Elemento neutro ($\exists! e \in A$)	$a * e = e * a = a, \forall a \in A$
4. Elemento simétrico ($\exists! a^{-1} \in A$)	$a * a^{-1} = a^{-1} * a = e, \forall a \in A$
5. Distributiva	Dado $(A, *)$ y (A, \oplus) \oplus es distributiva respecto a $*$ si: $a \oplus (b * c) = (a \oplus b) * (a \oplus c) = (b \oplus a) * (c \oplus a) = (b * c) \oplus a$

Veamos las estructuras algebraicas con el siguiente esquema:

TIPOS DE ESTRUCTURAS ALGEBRAICAS

EN ESTE CASO ESTAMOS APLICANDO LA SUMA Y LA MULTIPLICACIÓN, TENGASE EN CUENTA QUE SE PUEDE APLICAR EXACTAMENTE LO MISMO CON OTRAS OPERACIONES; HAY QUE TENER EN CUENTA QUE OPERACIÓN SE TOMA COMO 1º, 2º.

TIPOS DE ESTRUCTURAS ALGEBRAICAS						
TIPO DE ESTRUCTURA	TIPO DE OPERACIÓN	PROPIEDADES DE LAS OPERACIONES				
LAS DOS OPERACIONES SON INTERNAS $\forall a, b, c \in M; a + b \in M$ $\forall a, b, c \in M; a \cdot b \in M$		PROPIEDAD 1 <i>Asociativa</i> $(a + b) + c = (a + b) + c$	PROPIEDAD 2 <i>Neutro</i> $a + N = N + a = a$	PROPIEDAD 3 <i>Opuesto</i> $a + (-a) = (-a) + a$	PROPIEDAD 4 <i>Conmutativa</i> $a + b = b + a$	PROPIEDAD 5 <i>Distributiva</i> $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$
GRUPO $(M, +)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	NO SE CUMPLE	n/a
GRUPO CONMUTATIVO $(M, +)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	SE CUMPLE	N/a
ANILLO $(M, +, \cdot)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	SE CUMPLE	N/A
	2º PRODUCTO	SE CUMPLE	NO SE CUMPLE	NO SE CUMPLE	NO SE CUMPLE	SE CUMPLE
ANILLO CONMUTATIVO $(M, +, \cdot)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	SE CUMPLE	n/A
	2º PRODUCTO	SE CUMPLE	NO SE CUMPLE	NO SE CUMPLE	SE CUMPLE	SE CUMPLE
ANILLO CONMUTATIVO UNITARIO $(M, +, \cdot)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	SE CUMPLE	n/A
	2º PRODUCTO	SE CUMPLE	SE CUMPLE	NO SE CUMPLE	SE CUMPLE	SE CUMPLE
CUERPO $(M, +, \cdot)$	1º SUMA	SE CUMPLE	SE CUMPLE	SE CUMPLE	SE CUMPLE	n/A
	2º PRODUCTO	SE CUMPLE	SE CUMPLE	SE CUMPLE SIN TENER EN CUENTA EL NÚMERO "0" $M - \{0\}$	SE CUMPLE	SE CUMPLE

Hemos usados las operaciones aditiva y multiplicativa usuales, las que usamos a diario, pero podemos usar otras. Para las matemáticas aplicadas, solemos usar cuerpos usando estas operaciones y conjuntos numéricos (naturales, enteros, racionales, irracionales, reales o complejos).

1.2. Grafos, md

Hemos visto las estructuras algebraicas, y por lo tanto, podemos crear un entorno de trabajo con unas propiedades que se cumplen para los elementos y operaciones definidos en él. Los grafos, en particular, consisten en estructuras algebraicas en las que, dado un conjunto de vértices o nodos ($V = \{v_1, v_2, v_3, \dots, v_n\}$) y dado un conjunto de relaciones ($A = \{a_1, a_2, \dots, a_m\}$), que más adelante categorizaremos como arcos o aristas, estas relaciones conectan un vértice con otro.

Si los conecta sin importar el orden tal que $a_i = \{x, y\} | x, y \in V, i \in I$ las llamaremos **aristas** y, si por el contrario, el orden sí tiene importancia y se tiene que cada relación viene indicada con el par tal que $a_i = (x, y) | x, y \in V, i \in I$ a las relaciones las llamaremos **arcos**. Los extremos de una relación se dicen que son **incidentes** con la relación (arco o arista), dos vértices que están relacionados se dicen **adyacentes** y si los extremos de una relación están situados en un mismo vértice diremos que esta relación es un **bucle**.

Llamaremos **grafo no dirigido** al grafo $G = (V, A)$ con A siendo un conjunto de aristas y llamaremos **grafo dirigido** al grafo $G = (V, A)$ con A siendo un conjunto de arcos. Podemos convertir los arcos en aristas y definir un **grafo no dirigido asociado** (a un grafo dirigido) si ignoramos el par ordenado de los arcos y lo interpretamos como relación de doble sentido. Diremos que un **grafo es mixto** si contiene arcos y aristas a la vez.

Los **grafos simples** son grafos sin bucles y en la que no hay dos relaciones iguales (dos aristas iguales o dos arcos con el mismo par de vértices). Si no son simples diremos que son **multigrafos**.

Definiremos **grafo completo** (dirigido o no) si hay al menos una relación entre cada par de vértices posible, es decir, dados dos vértices cualesquiera del conjunto de vértices debe haber al menos una relación entre ellos. Denotaremos por K_n al **grafo completo simple y no dirigido** (podría ser el grafo no dirigido asociado(?)) con n vértices.

Un **grafo bipartido** es un grafo no dirigido (asociado) en la que existe una particion $\{X, Y\}$ del conjunto de vertices de tal forma que toda arista (trabajamos con grafos no dirigidos) tiene un extremo en cada parte. Los grafos dirigidos seran bipartidos si su grafo no dirigido asociado lo es. Si tratamos de definir un grafo bipartido B podemos hacerlo tal que:

$$B = (\{X, Y\}, A), \{x, y\} \in A \text{ con } x \in X, y \in Y \text{ y } X \cup Y = V \wedge X \cap Y = \emptyset \text{ dado } G = (V, A)$$

Podremos definir un **grafo bipartido completo** si cada vertice de la particion X esta unico con cada vertice de la particion Y.

NOTA: aunque dijéramos que un grafo completo tenían que estar conectados cada vertice con el resto de vertices, en los grafos bipartidos es unicamente que se conecte con todos los vertices de la otra particion. Revisar explicacion aunque parece que sera por propiedades de los caminos o alguna cosa por el estilo.

Diremos que $H = (V', A')$ es subgrafo de $G = (V, A)$ si $V' \subseteq V \wedge A' \subseteq A$, si se diese que $V' = V$ entonces podremos afirmar que el sugrafo H es generador de G.

Para saber el grado de un vertice, debemos considerar si hablamos de grafo dirigido o no dirigido. Si es no dirigido, el **grado de un vertice** corresponde al numero de aristas incidentes con el, contando los bucles 2, denotandolo por $d_G(v)$ y denotaremos con $\Gamma(v)$ al conjunto de vertices adyacentes a v y **diremos que la suma de todos los grados es el doble del numero de aristas**. Si es dirigido, distinguiremos el **grado de entrada**, $d_e(v)$, al numero de arco con v como extremo final y denotaremos con $\Gamma^{-1}(v)$ al conjunto de vertices que son extremo inicial de estos arcos, y **grado de salida**, $d_s(v)$, al numero de arcos con extremo inicial en v y denotaremos con $\Gamma(v)$ al conjunto de vertices que son extremo final de estos arcos. Si se trata de un grafo simple, $\text{card}(\Gamma(v)) = d_s(v)$ y $\text{card}(\Gamma^{-1}(v)) = d_e(v)$. Para los grafos dirigidos, **la suma de los grados de entrada o de salida valdra lo mismo que el numero de arcos**.

Hablando sobre grafos no dirigidos (asociados), llamaremos **cadena** a una sucesion finita de vertices y aristas que se alternan uniformemente de tal forma que, iniciandose desde un vertice y el siguiente vertice deben de ser incidentes con la arista en medio, debe cumplirse para toda la cadena. Llamaremos **longitud** de una cadena al numero de aristas que contiene. Sera una **cadena simple** si todas las aristas son distintas y un **camino** si es una cadena con todos los vertices distintos. Las **cadenas cerradas** son cadenas de longitud no nula cuyo vertice inicial es tambien vertice final. Si se cumple que es un camino simple y cerrado, lo llamaremos **ciclo**. Si tratamos ahora con grafos dirigidos, los conceptos son los mismo pero teniendo en cuenta que los arcos siguen una direccion y hay que respetarla y que en lugar de llamarse ciclo, se llama **circuito**. Si existe un ciclo de longitud impar, podremos afirmar que el grafo no dirigido no es bipartido. Diremos que un **grafo es ciclico** si contiene un ciclo o circuito y **aciclico** si no.

Diremos que dos vertices estan conectados si y solo si existe un camino entre ellos en ambos sentidos, es decir, dado a y b y la arista $\{a, b\}$ y el arco (a, b) , a y b estan conectados por la arista ya que el direccion es ambigua pero no estan conectados por el arco ya que no hay manera de llegar a a desde b con los arcos disponibles.

Un **grafo es conexo** si todo par de vertices estan conectados, por lo general los grafos no dirigidos suelen ser conexos pero los dirigidos deficiilmente lo son, podremos decir que son **debilmente conexos** si su grafo no dirigido asociado es conexo.

Es mas, si definimos las **componentes conexas** como elementos de una particion del grafo donde cada partes descrita esta compuesta por vertices conectados entre si pero no con los vertices de otras partes, diremos que un grafo es conexo si y solo si tiene un unico componente conexo.

Llamaremos **matriz de adyacencia** a la matriz cuadrada de orden n con n siendo el numero de vertices de un grafo en la cual a_{ij} corresponde al numero de relaciones existentes entre los vertices v_i y v_j , contando los bucles por 2 en los grafos no dirigidos. Definimos las siguientes propiedades:

1. La matriz sera simetrica para grafos no dirigidos y la suma de los elementos de una fila (columna)

dada el grado del vertice correspondiente a dicha fila (columna).

2. La suma de los elementos de una fila corresponde al grado de salida del vertice correspondiente a dicha fila y la suma de los elementos de una columna corresponde al grado de entrada del vertice correspondiente.
3. La matriz de adyacencia $A^1 = A$ comprueba unicamente los caminos de longitud 1 entre dos vertices, si elevamos a r la matriz de adyacencia obtenemos los caminos de longitud r que conectan dos vertices dados.

Llamaremos **matriz de incidencia** para grafos no dirigidos a la matriz $M_{n \times m}$ con n el numero de vertices y m el numero de aristas y en la que el valor de m_{ij} depende de si v_i no incide (0), incide (1) o es bucle(2) en relacion a a_j . Si tratamos con un grafo dirigido, denotaremos a la matriz con $B_{n \times m}$ donde si no incide o es bucle tendra el mismo valor que antes pero si incide habra que ver si v_i es extremo inicial (1) o final (-1) al arco a_j . Definimos las siguientes propiedades:

1. Siendo las filas correspondientes a los vertices, la suma de los elementos de una fila sera igual al grado de dicho vertice, ademas, la suma de todos los elementos de cada columna debe de ser igual a 2.
2. La suma de todas las columnas de un grafo dirigido sin bucles debe de ser 0, si hubiesen bucles, la columna correspondiente al arco en cuestion sera 2.

La **tabla de aristas incidentes** consiste en un listado en el que se indican todos las aristas que inciden sobre cada vertice, la **tabla de arcos salientes** es una lista de todos los arcos agrupados por los vertices que actuan como extremos iniciales y la **tabla de arcos entrantes** una lista que agrupa arcos por extremos finales.

Diremos que dos **grafos son isomorfos** si, dados $G_1 = (V_1, A_1)$ y $G_2 = (V_2, A_2)$ existe una aplicacion biyectiva $f : V_1 \mapsto V_2$ tal que dada una relacion en el primer grafo existe una unica relacion equitativa en el segundo y viceversa (definicion de aplicacion biyectiva), la relacion ya sea arco o arista. Para entendernos, es una reorganizacion espacial de un grafo cambiando de nombres o no a los vertices de un conjunto.

1.2.1. Accesibilidad y conectividad

Para grafos dirigidos, diremos que v_i alcanza a v_j o que v_j es alcanzable desde v_i si existe un camino dirigido (cadena con vertices distintos) con v_i como extremo inicial y v_j como extremo final. Es mas, definiremos a la **matriz de accesibilidad** asociada a un grafo como la matriz R de orden n con n el numero de vertices donde r_{ij} valdra 1 si v_i alcanza v_j o 0 si no lo alcanza. La traspuesta de la matriz de accesibilidad se llama **matriz de acceso** y comprueba si es alcanzable en lugar de si alcanza, se denota con $Q = R^T$.

Hay varios metodos para obtener la matriz de accesibilidad:

1. Con el **algoritmo de Warshall**, usaremos la matriz de adyacencia A y a partir de esta creamos R_0 o C que es igual a A pero los elementos mayores que 1 valdran 1 directamente (tambien se le llama matriz de conexion).

Se puede desarrollar de dos formas, un sumatorio desde 0 a $n - 1$ con n el numero de vertices tal que $\sum_{i=0}^{n-1} C^i = I_n + C^1 + \dots + C^{n-1} = R$ o elemento a elemento considerando R_i con $i = 0, 1, \dots, n - 1$ matrices intermedias hasta obtener R_n que sera la matriz de accesibilidad sin considerar los caminos de longitud 0 los cuales añadimos añadiendo 1 en la diagonal de esta. Para este caso se considera:

$$R_k = [r_{ij}^{(k)}]_{1 \leq i, j \leq n} \text{ tal que } r_{ij}^{(k)} = 1 \Leftrightarrow \begin{cases} r_{ij}^{(k-1)} = 1 \\ \text{ó} \\ r_{ik}^{(k-1)} = r_{kj}^{(k-1)} = 1 \end{cases}$$

2. Usando los conjuntos de vertices que son extremo final de forma anidada. Por ejemplo, dado $\Gamma(v_1) = \{v_2, v_3\}$ y tenemos que $\Gamma(v_2) = \{v_4, v_5\}$ y $\Gamma(v_3) = \{v_1, v_2\}$, si hacemos $\Gamma(\Gamma(v_1)) = \Gamma^2(v_1)$ lo que haremos sera estudiar los caminos de longitud 2 y haria

$$\Gamma(\{v_2, v_3\}) = \Gamma(v_2) \cup \Gamma(v_3) = \{v_4, v_5\} \cup \{v_1, v_2\}$$

en el caso de que $V = \{v_1, v_2, v_3, v_4, v_5\}$, no nos haria falta seguir ya que comprobamos que v_1 alcanza a todos los vectores, si hubiesen mas vectores, seguiriamos hasta alcanzarlos todos o alcanzar longitud $n-1$ (ya que como son caminos, no se pueden repetir vertices y por lo tanto la longitud de un camino no puede superar el numero de vertices).

Podriamos decir que $R(v_i) = \{v_i\} \cup \Gamma(v_i) \cap \dots \Gamma^p(v_i), p \leq n$ como el conjunto de vertices alcanzables desde v_i

Podemos estudiar los componentes conexos de un grafo dirigido usando la matriz de accesibilidad y la de acceso. Para ello, tenemos dos metodos:

1. Dado $G = (V, A)$

· Etapa 1: inicializar $i=1, V^{(1)} = V$

· Etapa 2: tomar $v_j \in V^{(i)}$

· Etapa 3: calcular $R(v_j) \cap Q(v_j)$ que sera la componente conexa.

Hacer $V^{(i+1)} = V^{(i)} \sim R(v_j) \cap Q(v_j)$ y $i++$, aunque se use el simbolo de similitud, realmente hace una diferencia, es decir, cogere los elementos del primer conjunto que no esten en el segundo.

· Etapa 4: finalizar si $V^{(i)} = \emptyset$. Si no se da este caso, se vuelve a la etapa 2.

La simbologia y explicacion de este metodo lo veo mal redactado, lo que se quiere hacer es que vas calculando componentes conexas a partir de ver que elementos estan conectados en ambos sentidos desde las matrices de accesibilidad y acceso, de ahi que saquemos la componente conexa, y hace la diferencia entre el conjunto de vertices y los vertices conexos para seguir probando el resto de vertices si existe un componente conexo entre ellos.

2. Usando las matrices de accesibilidad y acceso, y una funcion logica AND denotado por \oplus , siendo las filas de $R \oplus Q = R \oplus R^T$ las componentes conexas.

Para grafos dirigidos bastaria con comprobar el conjunto de vertices alcanzables desde un vector para saber las componentes conexas.

Dado un grafo conexo y por lo general no simple (es importante esto), llamaremos **tour** a una cadena cerrada (vertice inicial igual a vertice final) que atraviesa cada arista del grafo al menos una vez, si atraviesa cada arista exactamente una vez entonces se llama **tour euleriano**. Lo llamaremos **camino euleriano** si es una cadena por lo general simple que atraviesa cada arista al menos una vez. Podemos decir que todos los tour euleriano son caminos euleriano pero no todos los caminos euleriano son tour euleriano. Si encontramos un tour euleriano, podemos decir que es un **grafo euleriano**.

Para un grafo no dirigido (y obviamente conexo) podemos decir que es euleriano si y solo si no tiene vertices de grado impar o si unicamente tiene dos. Para un grafo dirigido (que al menos es debilmente conexo) podemos decir que es euleriano si, para todo vertice en el grafo su grado de entrada es igual al grado de salida o si contiene un camino euleriano los extremos de inicio y final tendra que el grado de

entrada es menor por uno al de salida y el de salida es menor por uno al de entrada, respectivamente, mientras que los componentes internos del camino tendra igual valor para el grado de entrada como para el de salida.

Para obtener un tour o camino euleriano, se usa el algoritmo de Fleury, Para grafos no dirigidos consiste en situarnos en un vertice cualquiera (a poder ser en uno de los dos de grado impar) e ir avanzando por una cadena simple eliminando las aristas por las que pasamos y considerando que al eliminarla no se desconecte el grafo (las aristas que provoquen la separacion de un grafo en dos partes se llaman aristas de corte). Con grafos dirigidos nos situaremos en el vertice que sera extremo inicial con grado de entrada uno menor que de salida y avanzamos por el camino de igual forma que con el grafo no dirigido pero en lugar de tour sera un camino y alcanzara el extremo final.

Un **camino Hamiltoniano** en un grafo es un camino que atraviesa cada vertice del grafo exactamente una vez, en el caso de que el camino sea simple (relaciones distintas) y pasa exactamente una vez por todos los vertices (es decir, todos los vertices distintos y todas las relaciones distintas, es un ciclo) sera un **ciclo Hamiltoniano**. Diremos que es un **grafo Hamiltoniano** si y solo si contiene un ciclo Hamiltoniano. Para construir un camino y ciclo Hamiltoniano consideramos las siguientes reglas:

1. Si el grafo no es conexo, no hay ciclos hamiltonianos
2. Si contiene n vertices, el camino sera de longitud $n-1$ y el ciclo de n .
3. Dado cualquier vertice del grafo, el camino tendra al menos un relacion incidente y como mucho dos y el ciclo exactamente 2.
4. Si el grafo es hamiltoniano, todos los vertices deben de ser de grado igual o mayor que 2.
5. Los vertices de grado igual a 2, las relaciones incidentes a estos deben de aparecer en el camino/ciclo.
6. Los vertices con grado mayor a 2 sobre los que incide el camino/ciclo se omitiran el resto de relaciones que no se hayan usado para avanzar en ese vertice.
7. Si se obtiene un ciclo o camino Hamiltoniano para un grafo, el unico subgrafo capaz de poder definir tambien un ciclo o camino es aquel que contenga todos los vertices del grafo original.

Si el grafo fuese bipartido podemos afirmar que si es Hamiltoniano (contiene un ciclo hamiltoniano) entonces hay igual numero de vertices en una particion que en la otra pero si solamente contiene un camino hamiltoniano entonces el numero de vertices en una particion difiere como mucho en 1 de la otra particion. El reciproco es cierto para grafos biparticos completos con mas de 2 vertices.

El teorema de Dirac afirma que todo grafo simple con 3 o mas vertices, con todo vertice de grado $n/2$, n siendo el total de vertices, tendra un ciclo hamiltoniano y, por lo tanto sera un grafo hamiltoniano. Si es un grafo completo simple (cada par de vertices esta conectado por una unica relacion igual y sin bucles) y hay 3 o mas vertices entonces el grafo tiene un ciclo hamiltoniano y, por lo tanto, sera un grafo hamiltoniano.

El codigo Gray basicamente es que puedes crear un ciclo ya que el ciclo los conecta de tal manera que sigue el orden correcto (00, 01, 11, 10, 00) (000, 001, 011, 111, 101, 100, 110, 010, 000)...

1.2.2. Arboles

Los arboles son grafos conexos y aciclicos y diremos que es un arbol generador si es arbol y subgrafo generador a la vez.

1. Dos vertices cualesquiera estan conectados por un unico camino.
2. Un grafo sera conexo si y solo si tiene un arbol generador.

3. El numero de relaciones en un arbol difiere en uno al numero de vertices de este
4. Todo arbol no trivial tiene al menos 2 vertices de grado 1.

Dado un vertice r_0 el cual actua como extemo inicial unicamente y todo vertice esta conectado con este, podemos definir un arbol enraizado a r_0 y distinguiremos los distintos niveles o alturas n a los vertices cuyos caminos sean de longitud n .

Los arboles binarios son aquellos en los que tienen uno, dos hijos a la izquierda y/o a la derecha o ningun hijo y sera completo si tienen dos o ningun hijo. Un arbol binario completo con n vertices internos(vertices con hijos) tendra $n+1$ vertices terminales (sin hijos) y un total de $2n+1$ vertices. Dado un arbol binario de altura h y n vertices terminales, se cumple que $t \leq 2^h$.

El arbol binario de busqueda y el algoritmo de busqueda esta raro... no lo veo claro solo entiendo que distribuyes en cada nivel de derecha a izquierda y de izquierda a derecha alternando (para las posiciones) y le asignas un dato de una frase dada y con el algoritmo de busqueda, si el elemento que buscas esta delante del dato actual (es mayor) te mueves al hijo de la derecha, si esta detras (es menor) te mueves al hijo de la izquierda, si es el valor ya tienes la posicion y paras o si no se encuentra en el arbol tambien paras.

Un arbol enraizado ordenado es un tal en el que el conjunto de hijos de un padre estan ordenados linealmente de izquierda a derecha, todo arbol binario esta ordenado.

La notacion polaca infija considera cada subarbol de un arbol, para escribirlo relacionas los hijos por el padre y con la notacion escribes tal relacion, si un hijo tiene hijos, dejara pasar primero al de la izquierda y dejara atras al de la derecha. Esta notacion puede ser ambigua asique se utilida la de preorden o directa (se dibuja una linea alrededor del arbol que toca todos los vertices y se sigue de izquierda a derecha empezando por la raiz y esvribiendo los vertices no repetidos que tocan dicha linea) o postorden o inversa (la misma linea pero se sigue de derecha a izaquierda y se escribe de derecha a izquierda empezando por la raiz y anotando los vertices no repetidos que tocan dicha linea).

1.2.3. Grafos ponderados

Dado un grafo simple, sea dirigido o no, tal que $G = (V, A)$ con V el conjunto de vertices y A el conjunto de relaciones entre los vertices, diremos que es ponderado si dado un conjunto que denotamos por R , constituidos por elementos a los que llamaremos pesos tal que $w_{ij} \in R$, existe una funcion tal que $W : A \mapsto R, \{v_i, v_j\} \text{ o } (v_i, v_j) \mapsto w_{ij}$ a la que llamamos funcion de podneracion.

Si la arista o el arco estan definidos en el conjunto A , el peso dentra un valor determinado, pero si no lo estuviesen o, mejor dicho, no son parte del grafo, el peso respectivo a este seria infinito. Los pesos se pueden considerar como el tiempo o esfuerzo invertido en atravesarlo.

Llamaremos matriz de peso de orden n con $n = \text{card}(V)$ a la siguiente matriz:

$$\Omega = [a_{ij}]/a_{ij} = \begin{cases} w_{ij} & \text{si } \{v_i, v_j\} \text{ o } (v_i, v_j) \in A \\ \infty & \text{si } \{v_i, v_j\} \text{ o } (v_i, v_j) \notin A \end{cases}$$

Si para un grafo ponderado definimos una cadena con todos sus vertices distintos (es decir, un camino, hay que ir recordando definiciones xd) llamaremos peso de camino a la suma de los pesos de cada arista o arcos que los forman y se denotara tal que, teniendo $C \equiv v_1 v_3 v_2$, el peso del camino sera $w(C) = w_{13} + w_{32}$.

Sabiendo que los caminos pueden ahora caracterizarse no solo por longitud (recordad que usando la matriz de adyacencia se podian cuanto caminos de longitud variada se podian dar entre varios vertices si la elevabamos al numero de longitud), si no por peso, destacaremos el camino de menor peso o mas corto y de mayor peso o mas largo (critico). Para ellos primero debemos conocer el algoritmo de numeración, el cual usaremos sobre grafos dirigidos y sigue los siguientes pasos:

1. Inicializamos $i=1$ y tenemos que $V_i = V$.
2. Tomamos $v \in V_i$ tal que $d_e(v) = 0$ en $G[V_i]$. Buscaremos el vertices sin arcos entrantes en el grafo que considera el conjunto de vertices V_i .
3. Denotamos el vertice v por el valor de i , hacemos $V_{i+1} = V_i$ e incrementamos el valor de i por uno($i++$).
4. Si se da que el nuevo conjunto $V_i = \emptyset$, hemos numerados todos los vertices. En caso de seguir habiendo elementos volvemos al paso 2 hasta que no hayan mas elementos.

Este metodo permite estudiar un orden a seguir con caminos alternativos.

Ahora que sabemos numerar los vertices, estudiaremos ahora las ecuaciones de Bellman y el PERT, algoritmos basicos y sencillos para estudiar los caminos mas cortos y largos de un grafo.

Para una renumeracion del conjunto V , tal que $1, \dots, n \in V$ con n el ultimo vertice, veremos que para los caminos mas cortos (cmc) denotaremos por u_j el peso del cmc entre 1 y j , con el vertice 1 siendo el origen. Cualquier camino comprendida en una cmc sera, ademas, otra cmc obtenida de partir por los vertices de interes.

La ecuacion de Bellman consiste en obtener el peso de un cmc desde 1 a j , para ello prueba todos los caminos de 1 a cada vertice (el peso del cmc a 1 es 0 siempre). Vamos a ver como funciona el algoritmo para calcular los cmc a todos los vertices (ayuda para obtener componentes conexas en grafos dirigidos y despues veremos otra forma en la que existe un orden (evitando asi los componentes ciclicos), empecemos:

1. $u_1 = 0$
2. Ponemos los pesos de todos los caminos en orden tal que:

$$\begin{aligned} u_2 &= \min\{u_1 + w_{12}, \dots, u_k + w_{k2}\}, k \leq n \wedge (k \neq 1, 2) \\ u_3 &= \min\{u_1 + w_{13}, \dots, u_k + w_{k3}\}, k \leq n \wedge (k \neq 1, 3) \\ &\dots \\ u_n &= \min\{u_1 + w_{1n}, \dots, u_k + w_{kn}\}, k \leq n \wedge (k \neq 1, n) \end{aligned}$$

Estos calculos vienen dados por la funcion:

$$u_j = \min_{k \neq j} \{u_k + w_{kj}\}, 1 < j \leq n$$

Es decir, comprueba de entre todos los elementos que cumplan las condiciones $k \neq j$ y $1 < j \leq n$ cual es el minimo.

3. Si al realizar el paso 2, algunos valores quedan en funcion de otros pesos, por lo general sera porque para este conjunto de vertices (ya que deberan de darse con al menos un par de vertices) seran una componente conexas. Para omitir esto, consideraremos los pesos por debajo de la diagonal con valor infinito.
4. Se comprueba cual es el valor minimo para alcanzar cada vertice.

En el caso de no estar interesados en probar si existen componentes conexas, modificamos la funcion dada anteriormente en el paso dos, siendo ahora

$$u_j = \min_{k < j, v_k \in \Gamma^{-1}(v_k)} \{u_k + w_{kj}\}, 1 < j \leq n$$

Es decir, comprueba de entre todos los elementos que cumplan las condiciones $k < j$, con $1 < j \leq n$ y que $(v_k, v_j) \in A$ (tratamos con arcos, el orden es importante), cual es el minimo.

Para obtener los caminos mas largos usamos el PERT (Project Evaluation Research Task), que se basa en el tiempo maximo de realizacion de una tarea. Parte del concepto de que para empezar una tarea, se debe de haber hecho una antes (para hacerte una tostada, necesitas cortar pan, sacar la tostadora, el aceite,... pero no puedes hacerte una tostada y luego sacar la tostadora para hacer una tostada, ERROR 404 ?!). Aunque sea el camino mas largo, su interpretacion corresponde al tiempo minimo necesario para completar el proyecto, ya que es el tiempo que nos interesa para alcanzar el final del proyecto. Veamoslo mas orientado a matematicas y a menos comida, que tengo hambre.

Conceptos claros que debemos tener es que el vertice de inicio, s, y el del final, t, se pueden incorporar si es necesario (en el caso de multiples opciones de inicio o final), y sus pesos son 0 y el peso del camino mas largo, respectivamente. Nosotros deberemos construir el grafo y numerarlo. PERT se asimila a las ecuaciones de Bellman con las condiciones extendidas, pero usando el maximo en lugar del minimo tal que:

1. $u_1 = 0$
2. Ponemos los pesos de todos los caminos en orden tal que:

$$\begin{aligned} u_2 &= \max\{u_k + w_{k2}\} \\ u_3 &= \max\{\dots, u_k + w_{k3}\} \\ &\dots \\ u_n &= \max\{\dots, u_k + w_{kn}\} \end{aligned}$$

$$u_j = \max_{k < j, v_k \in \Gamma^{-1}(v_j)} \{u_k + w_{kj}\}, 1 < j \leq n$$

Es decir, comprueba de entre todos los elementos que cumplan las condiciones $k < j$, con $1 < j \leq n$ y que $(v_k, v_j) \in A$ (tratamos con arcos, el orden es importante), cual es el maximo.

3. El peso obtenido en el ultimo vertice, ya que tenemos un vertice final t (dado o creado) es necesariamente la suma de todos los caminos mas largos partiendo del vertice inicial s. Este sera el camino critico y el camino usado (el arco correspondiente al peso maximo en cada camino) seran las actividades criticas.

Podemos calcular retrasos maximos, es decir, un camino alternativo y paralelo al camino critico el cual debiera de ser el de mayor peso posible pero inferior o igual al del camino critico o al de su punto de incision.

A parte de las ecuaciones de Bellman, podemos usar el algoritmo de Dijkstra. Este algoritmo encuentra los caminos mas cortos desde el vertice inicial hasta el resto, con lo cual se deben de hacer tantas iteraciones como vertices haya para probar el cmc a cada vertice. Hay que entender que Dijkstra no te dara el camino mas corto entre dos vertices, te dara todos los caminos mas cortos, pero el conjunto de vertices modificados que ahora veremos es unicamente un listado, no el cmc en si. Dijkstra no tiene porque trabajar con un grafo dirigido si o si, puede con uno no dirigido o mixto, si entendemos los arcos como dos aristas que conectan los mismos vertices pero en sentidos opuestos. Veamos que hacer:

1. Inicializamos la iteracion con un conjunto de vertices fijos (nuestro listado de vertices con su cmc asignado) denotado por P y con el de conjunto de vertices variables (listado de vertices de los cuales desconocemos su cmc) denotado por T. Para comodidad siempre aplicamos el algoritmo de numeracion y por lo tanto afirmaremos que, al inicializar, $P=\{1\}$ y $T=\{2, \dots, n\}$. Esto es lo que puede resultar lioso, ya que el camino unicamente se forma por el conjuntos de vertices pertenecientes a P, teniendo entonces:

$$\begin{cases} P = \{1\}, T = \{2, \dots, n\}, \text{conjunto de vertices fijo y variable para la primera iteracion} \\ u_1 = 0, \text{el peso de 1 a 1 es 0 siempre} \\ u_j = w_{1j}/j \in \Gamma(1), \text{si } j \text{ es extremo final de un arco con 1 extremo inicial, el peso sera el del arco} \\ u_j = \infty/j \notin \Gamma(1), \text{si } j \text{ no es extremo final de un arco con 1 extremo inicial, el peso es infinito} \end{cases}$$

- De entre los $u_j/j \in T$, algunos de valor infinito y otros un valor real distinto de infinito, tomaremos el mínimo de estos tal que $u_k = \min_{j \in T} \{u_j\}$, habiendo encontrado así el cmc para k .

Ahora que sabemos su cmc, lo añadimos a P y lo quitamos de T , tal que:

$$P = P \cup \{k\} \text{ y } T = T \setminus \{k\}$$

Si $T = \emptyset$ finalizamos el algoritmo y hemos encontrados el peso de todos los caminos mas cortos desde 1 a cada vertice (de normal hara falta hacer tantas iteraciones como vertices)

NOTA: hemos indicado la diferencia con el simbolo \setminus y no con \sim como indican en los apuntes porque esta es la simbologia correcta para la diferencia de conjuntos, el otro simbolo es para equivalencia, podeis encontrarlo en cualquier pagina de matematicas si buskais diferencia de conjuntos.

- Se actualizan los pesos de los cmc, este es el paso mas confuso ya que lo que se hace es considerar los caminos ya existentes anteriormente de longitud x y ver nuevos caminos considerando haber avanzado un vertice mas, siendo de longitud $> x$.

$$\begin{aligned} \forall j \in \Gamma(k) \cap T & \leftrightarrow j \text{ debe de estar en } T \text{ y existir } (k, j) \in A \\ u_j = \min\{u_j, u_k + w_{kj}\} & \leftrightarrow u_j \text{ sera el menor entre el camino ya existente y los que caminos que pasan por } k \end{aligned}$$

Al actualizar los caminos con un nuevo vertice de paso (que se ha podido o no usar), se vuelve al paso 2.

Puede ser lioso pero por logica es sencillo de sacar, se asimila al Bellman solo que se tendra que hacer iteraciones, a partir de la segunda debemos de considerar los caminos anteriores y aunque estemos en la 5 iteracion, alomejor estamos probando un cmc para un vertice de longitud igual a 3. El ejemplo en los apuntes es medianamente claro (Leccion4b pag 141), en la primera iteracion, el camino de menor peso es el de 1 a 4, pero en la segunda no se escoge un camino que pase por 4, se coge el camino de 1 a 2 ya que en la iteracion anterior este tenia un valor mas pequeño que el que se podria lograr pasando por 4. Esto se ha quedado algo largo pero mejor el aclararlo que quedarse en blanco.

1.2.4. Método de Floyd-Warshall

FW lo explicaremos aparte ya que tiene cierta dificultad y, a mi parecer, en los apuntes carece de sentido, asique lo veremos con calma. Este algoritmo permite estudiar el cmc de un par de vertices i, j , denotado por u_{ij} . Para ello, usaremos las variables $u_{ij}^{(m)}$, que significa el camino entre i, j con la restriccion de que no pasamos por los vertices $\geq m$, obviando los propios i, j , que son extremos por definicion. NO ES NECESARIO EL ALGORITMO DE NUMERACION PERO PUEDE AYUDAR. Veamos como actuar (este algoritmo se obtiene de aplicar toda la broza puesta en los apuntes):

- Inicializamos con $m=1$, haciendo:

$$u_{ij}^{(1)} = w_{ij}, \forall i, j \in V$$

Esta primera iteracion consiste en obtener la misma matriz de pesos ya que estamos indicando que el peso del camino de longitud 1 entre cada par de vertices es el del arco que los conecta.

- Calculamos $u_{ij}^{(m+1)}$ sobrescribiendo en la matriz de peso los valores que se modifiquen, siendo la definicion ahora:

$$u_{ij}^{(m+1)} = \min\{u_{ij}^{(m)}, u_{im}^{(m)} + u_{mj}^{(m)}\}, \forall i, j \in V$$

Hay dos cosas que dejar indicado, al obtener los pesos de los caminos con una restriccion mayor a 1 ($m > 1$), la interpretacion de los elementos en la matriz puede ser ambigua, es decir, puede significar muchas cosas. Por ejemplo, los elementos de la diagonal indicaran si existe un ciclo por su peso. Por otro lado, lo que la formula trata de hacer es comprobar si es mas corto el camino ya existente o uno en el que pasa por el vertice m (si estamos calculando $m+1$).

- Actualizamos el valor de m a $m+1$ ($m++$) y volvemos al paso 2, en el caso de que obtengamos $m=n+1$ con $n=\text{card}(V)$, detenemos el algoritmo y podemos afirmar que tenemos los pesos de los cmc entre cada par de vertices del grafo ponderado.

Pero este algoritmo unicamente permite saber los pesos de los cmc, pero sin saber los componentes de este. Para ello definimos una segunda matriz, que nos permitira saber al menos el componente del camino anterior al vertice final, siendo:

$$\ominus^{(m)} = [\theta_{ij}^{(m)}]$$

Esta matriz se va obteniendo a la vez que aplicabamos el algoritmo de FW, por lo tanto veamoslo entendiendo su aplicacion paralela a FW:

- Inicializamos $m=1$, todos los caminos de longitud 1 existentes sabemos que el vertice anterior a j sera i (por definicion), por lo tanto tenemos:

$$\theta_{ij}^{(1)} = \begin{cases} \theta_{ij}^{(1)} = i & \text{si } u_{ij}^{(1)} < \infty \\ \theta_{ij}^{(1)} = \emptyset & \text{si } u_{mj}^{(1)} = \infty \end{cases}$$

- Calculamos para $m+1$, actualizando las posiciones segun el siguiente criterio:

$$\theta_{ij}^{(m+1)} = \begin{cases} \theta_{ij}^{(m)} & \text{si } u_{ij}^{(m+1)} = u_{ij}^{(m)} \\ \theta_{mj}^{(m)} & \text{si } u_{ij}^{(m+1)} < u_{ij}^{(m)} \end{cases}$$

con esto probamos si el cmc no se altera dejamos el valor ya existente en esa posicion y si es menor, entonces actualizamos con el valor de en la posicion $\theta_{mj}^{(m)}$.

- Incrementamos el valor de m y comprobamos si hemos alcanzado la condicion de salida.

Hay que destacar que en los apuntes es confuso el hecho de que señalicen filas y columnas pero se actualicen valores fuera de esta señalizacion, por ahora no le veo sentido pero bueno. . . , siguiendo estos pasos con calma puedes obtener todas las matrices con restricciones de vectores y la matriz con el vertice anterior para el cmc con una restriccion dada.

Vamos a entrar en detalles con algunos conceptos complicados:

- Cuando hablamos de $u_{ij}^{(m)}$ hablamos del peso del cmc que no pasa por vertices $m, m+1, \dots, n$ y $\theta_{ij}^{(m)}$ indicara el vertice anterior a j del cmc que no pasa por estos vertices dichos (recordad que no es necesario aplicar el algoritmo de numeracion con lo cual podemos tratar con letras o cualquier otro tipo de identificador, pero siempre se puede aplicar para facilitarlo).
- Sera importante mantener registro de todas las tablas que se obtiene e indicar respecto a que restriccion se aplican. Con estas tabla podremos saber los caminos que no incluyen a vectores que se encuentra por encima (numericamente o de forma ordenada) de la restriccion.
- Si se nos plantea un problema donde nos dan varias restricciones, podemos reordenar la matriz de pesos para reducir las iteraciones que se deben realizar. Si se nos da $V = \{A, B, C, D\}$ y se nos pide el cmc entre A y D sin pasar por B , entonces podemos reordenar la matriz tal que:

$$\Omega = \left(\begin{array}{c|cccc} & A & B & C & D \\ \hline A & w_{AA} & w_{AB} & w_{AC} & w_{AD} \\ B & w_{BA} & w_{BB} & w_{BC} & w_{BD} \\ C & w_{CA} & w_{CB} & w_{CC} & w_{CD} \\ D & w_{DA} & w_{DB} & w_{DC} & w_{DD} \end{array} \right) \simeq \left(\begin{array}{c|cccc} & C & A & D & B \\ \hline C & w_{CC} & w_{CA} & w_{CD} & w_{CB} \\ A & w_{AC} & w_{AA} & w_{AD} & w_{AB} \\ D & w_{DC} & w_{DA} & w_{DD} & w_{DB} \\ B & w_{BC} & w_{BA} & w_{BD} & w_{BB} \end{array} \right)$$

Esta reordenación es posible ya que tenemos en cuenta que:

- Los vertices que no queremos que pertenezcan al camino los podemos poner al final, asegurándonos el omitir las iteraciones partir del primer vertice restringido. En este ejemplo pasamos de ABCD a ADCB, omitiendo la iteración 4.
- Los vertices internos permitidos deben de ir delante de los vertices extremos, siendo esto iteraciones obligadas a hacer. En nuestro ejemplo pasamos de ADCB a CDAB, estamos seguros de que para considerar C debemos hacer al menos una iteración.
- El vertice inicial debe de ir antes del vertice final, o dicho de otra manera, se debe de poner STM (S vertice inicial, T vertice final y M conjunto de vertices restringidos). Esto nos hara decir en que iteración no detenemos ya que, por definicion, el vertice final esta incluido en el camino (se podría pensar que el inicial tambien pero hay que considerar los casos en los que se pasa por los vertices anteriores, alomejor el camino directo es mas largo que pasando por otros vertices). En nuestro ejemplo, pasamos de CDAB a CADB, y detendremos la iteración en $m=2$ (vamos, solo habremos hecho $\Omega^{(1)}$ y $\Omega^{(2)}$ para saber el cmc de A a D.

1.2.5. Arboles generadores de minimo peso

Dado un grafo ponderado y no dirigido (si, ahora trabajamos con no dirigido), diremos que T es un arbol generador de minimo peso si es un arbol generador y la suma de los pesos asociados a sus aristas es minima. Recordad que los arboles son aciclicos y siempre se puede intepretar como grafos dirigidos si indicamos una raíz.

Podemos usar dos algoritmos para obtener los arboles generadores de minimo peso, Kruskal y Prim (prefiero Kruskal, parece mas comodo).

Kruskal consiste en tomar la definicion de arbol genreador al pie de la letra, un arbol es un grafo aciclico y el numero de aristas en un arbol difiere en uno al numero de vertices de este, es decir, $\text{card}(V)=n$ entonces habran $n-1$ aristas en el arbol. Veamos que hacer:

- Iniciamos el algoritmo con $T = \emptyset$, no tenemos ninguna arista que forme el arbol.
- Ordenamos de menos a mayor los pesos de las aristas en A, tal que

$$e_1, e_2, \dots, e_n \text{ si y solo si } w_1 \leq w_2 \leq \dots \leq w_n$$

con esto lo que tratamos de hacer es decir que renombramos a las aristas y ponemos ese orden en concreto para que cumpla la condicion, el dejar expresado tal condicion seria muy compleja con lo cual interpretamos el que hemos cambiado el nombre para ajustarlo y apañado.

- Añadimos de forma ordenada las aristas a T hasta que $\text{card}(T) = \text{card}(V) - 1$, evitando que una arista añadida forme un ciclo.

Prim, por otro lado, trabaja con los vertices y los pesos mas que con las aristas, y parece que complica el algoritmo de ahi que vea mas comodo Kruskal. Lo que va haciendo es fijar una raíz y crea un conjunto para cada vertice que consiste en ir probando cada par de vertice adyacente es el que tiene el menos peso. Este algoritmo podría entenderse como dijkstra con la condicion añadida de que no se pueden formar ciclos y en la que no considera el camino anterior (se asemeja en la parte

en la que tiene que recordar el camino anterior en funcion de un vertice y compararlo con un vertice nuevo, escogiendo el de menor peso), por eso te crean dos conjuntos, el de los vertices fijos (los que ya sabemos que estan y entre ellos solo los conecta un camino. Veamos el algoritmo que nos dan en las transparencias y tratemos de traducirlo:

1. Inicializamos con un conjunto de vertices fijos $U = \{v^*\}$ y otro de aristas incluidas $T = \emptyset$, donde $v^* \in V(G)$ y este es la raiz. Ademas, definiremos todos los caminos directos entre la raiz y el resto de vertices tal que

$$L(u) = w(\{u, v^*\}), \infty \text{ si no existe la arista, } \forall u \in V(G)$$

2. De entre todos los caminos definidos, escogemos el de menor peso para un vertice el cual no este incluido en el conjunto de vertices fijos. Esto lo denotamos matematicamente de la siguiente forma:

$$\text{Encontramos } u^* \in V(G)/u^* \notin U, L(u^*) = \min_{u \notin U} \{L(u)\}$$

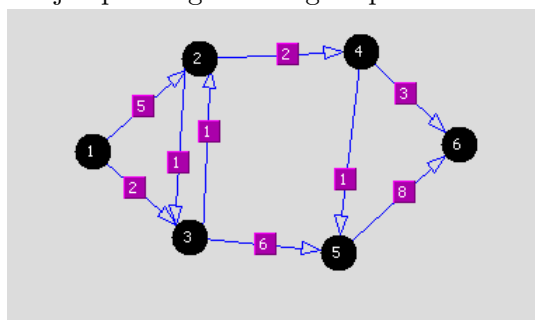
3. Añadimos el vertice destacado (u^*) al conjunto de vertices fijos, sabiendo asi que hemos encontrado el cmc para llegar a el y, ademas, no se da ningun ciclo hasta llegar a este. Al encontrarlo, debemos añadirlo a U, tal que $U := U \cup \{u^*\}$ y añadimos la arista e incidente con el peso $L(u^*)$ a T, tal que $T := T \cup \{e\}$
4. Comprobamos si el conjunto U engloba todos los vertices, y si lo hace, hemos obtenido el arbol generador. En caso contrario, actualizamos los pesos de los camino, al igual que dijsktra, ahora considerando el vertice intermedio el recientemente añadido a U y comprobando si el camino para llegar a un vertice era mas corto el ya existente o el nuevo, con la condicion de que estos vertices no esten en U. AL obtener estos caminos, volvemos al paso 2. Matematicamente, este enunciado se expresaria tal que:

$$L(u) := \min_{u \notin U} \{L(u), w(\{u^*, u\})\}$$

1.2.6. Ejemplo agrupado

Vamos a usar un solo grafo para probar los distintos algoritmos ademas de probar algunos casos especiales, el grafo que usaremos es el siguiente:

Ejemplo de grafo dirigido ponderado:



	1	2	3	4	5	6
1	∞	5	∞	2	∞	∞
2	∞	∞	1	2	∞	∞
3	∞	1	∞	∞	6	∞
4	∞	∞	∞	∞	1	3
5	∞	∞	∞	∞	∞	8
6	∞	∞	∞	∞	∞	∞

1. **ALGORITMO DE NUMERACION:** En MaGraDa diria que el grafo no es aciclico, es decir, que existe algun componente conexo (de 2 a 3) y daria error a la hora de realizar el algoritmo de numeracion. Pero esto se puede interpretar de varias formas, para ello omitiremos en el algoritmo de numeracion uno de los arcos pero lo tendremos en cuenta para futuras aplicaciones. En este caso bastaria con ir mirando cuales de los vertices no numerados tienen grado de entrada igual

a 0 y en el caso de ninguno, hemos encontrado componente conexas, omitimos uno de los vertices y seguimos.

2. **Ecuaciones de Bellman,CMC:** Con la numeracion (o renumeracion si los vertices ya estan dados en numeros, no hay ninguna diferencia realmente) procedemos a sacar el cmc entre el vertice inicial 1 y el resto de vertices. Este algoritmo no tiene iteraciones y se hace considerando todos los caminos de los vertices renumerados anteriores a el. AUNQUE, si se usa la condicion $k \neq j$ vista en la explicacion, esta tmbn te deja indicada mas logicamente la dependencia entre un conjunto de vertices y, por lo tanto, se pueden dejar indicadas componentes conexas. Para este ejemplo usaremos la condicion $k < j$ y omitiremos el arco del grafo anterior (2,3), SE APLICA EL ALGORITMO DE NUMERACION (2-3,3-2):

$$u_1 = 0$$

$$u_2 = \min(u_1 + w_{12}) = \min(0 + 2) = 2$$

$$u_3 = \min(u_1 + w_{13}, u_2 + w_{23}) = \min(0 + 5, 2 + 1) = 3$$

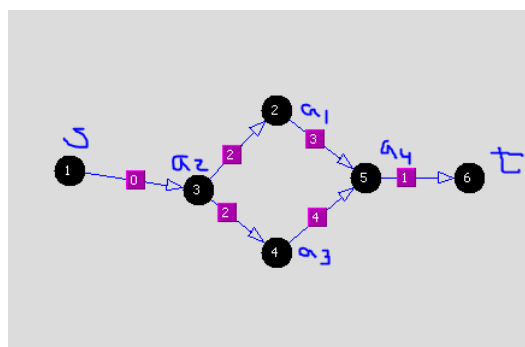
$$u_4 = \min(u_1 + w_{14}, u_2 + w_{24}, u_3 + w_{34}) = \min(0 + \infty, 2 + \infty, 3 + 2) = 5$$

$$u_5 = \min(u_1 + w_{15}, u_2 + w_{25}, u_3 + w_{35}, u_4 + w_{45}) = \min(0 + \infty, 2 + 6, 3 + \infty, 5 + 1) = 6$$

$$u_6 = \min(u_1 + w_{16}, u_2 + w_{26}, u_3 + w_{36}, u_4 + w_{46}, u_5 + w_{56}) = \min(\infty, \infty, \infty, 5 + 3, 6 + 8) = 8$$

Los cmc a cada vertice estan relacionados con el vertice del cual depende cada peso de camino, por ejemplo, para u_6 , debemos de pasar por 1 2 3 4 6. Si usaramos la condicion $k \neq j$, en u_2 y u_3 no se podria haber definido un valor exacto si se diese que si consideramos el arco (2,3), y ahi podriamos identificar un componente conexo

3. **PERT:** son ecuaciones de Bellman pero con maximo en lugar de minimo. En algunos casos se puede dar una tabla que nos dice el tiempo necesario para realizar una actividad (asociada al vertice indicado) y nos dan sus prerrequisitos, en lugar de venir un grafo ya hecho. Para estos casos debemos de añadir dos vertices adicionales, s (todos los arcos con s como extremo inicial seran de peso 0 y tendran como extremo final aquellos que vertices (o actividades) que no tienen prerrequisitos) y t (todos los arcos con t como extremo final seran de peso igual al tiempo necesario para finalizar una tarea la cual no es prerrequisito de otro vertice dado). Veamos esta tabla y hagamos al menos el grafo, el algoritmo se deja a realizar ya que, como se ha dicho, es como Bellman pero usando maximos en lugar de minimos.



Actividad	a_1	a_2	a_3	a_4
Tiempo necesario	3	2	4	1
Prerrequisito	a_2	-	a_2, a_1	a_3

Cabe decir que PERT se preocupa en obtener el camino entre s y t, por lo tanto el resultado final debe de ser el camino critico entre s y t, siendo no necesario que pase por algun vertice. Con el algoritmo de numeracion, 2-3 y 3-2. Y el camino critico seria de valor 7.

Si quisiéramos obtener el maximo retraso permitido para realizar una actividad, sin que este retraso afecte al tiempo minimo necesario para hacer el camino de s a t, entonces consideramos todos los caminos posibles con inicio en los vertices anteriores al de interes y final posteriores a este, y que obviamente pasen por el, y comprobamos el camino pasando por este vertice mas una incognita es menor o igual que el camino hasta donde hace incision. La incognita se obtendra por un sistema de inecuaciones y significara el tiempo maximo que se podria retrasar la actividad (un incremento de su tiempo necesario) sin alterar el tiempo para el camino critico.

4. **Dijkstra:** Es otra forma para aplicar las ecuaciones de Bellman usando iteraciones, donde cada iteración es una actualización de los caminos a cada vertice pasando por un vertice que sabemos fijo haber encontrado su cmc. Además, con Dijkstra los componentes conexos son más sencillos de tener en cuenta ya que también puede trabajar con grafos no dirigidos (en lugar de $\Gamma(v)$ considerarlos como el conjunto de extremo final conectado a v , serán los vertices adyacentes a v).

Para el grafo dado al principio tendríamos (recordad que 2-3,3-2 por la numeración):

Primera iteración, INICIALIZACION

$$P = \{1\}$$

$$T = \{2, 3, 4, 5, 6\}$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = w_{13} = 5$$

$$u_4 = w_{14} = \infty$$

$$u_5 = w_{15} = \infty$$

$$u_6 = w_{16} = \infty$$

$$u_2 = \min_{j \in T}(u_j) = \min(u_2, u_3, \dots)$$

$$P = \{1\} \cup \{2\}$$

$$T = V \setminus P = \{3, 4, 5, 6\} \neq \emptyset$$

Tercera iteración, ACTUALIZACION

$$P = \{1, 2, 3\}$$

$$T = \{4, 5, 6\}$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = u_2 + w_{23} = 3$$

$$u_4 = \min(u_4, u_3 + w_{34}) = \min(\infty, 3 + 2) = 5$$

$$u_5 = u_2 + w_{25} = 9$$

$$u_6 = w_{16} = \infty$$

$$u_4 = \min_{j \in T}(u_j) = \min(u_4, u_5, u_6)$$

$$P = \{1, 2, 3\} \cup \{4\}$$

$$T = V \setminus P = \{5, 6\} \neq \emptyset$$

Quinta iteración, ACTUALIZACION

$$P = \{1, 2, 3, 4, 5\}$$

$$T = \{6\}$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = u_2 + w_{23} = 3$$

$$u_4 = u_3 + w_{34} = 5$$

$$u_5 = u_4 + w_{45} = 6$$

$$u_6 = \min(u_6, u_5 + w_{56}) = \min(\infty, 6 + 8) = 8$$

$$u_6 = \min_{j \in T}(u_j) = \min(u_6)$$

$$P = \{1, 2, 3, 4, 5\} \cup \{6\}$$

$$T = V \setminus P = \emptyset$$

Segunda iteración, ACTUALIZACION

$$P = \{1, 2\}$$

$$T = \{3, 4, 5, 6\}$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = \min(u_3, u_2 + w_{23}) = \min(5, 2 + 1) = 3$$

$$u_4 = w_{14} = \infty$$

$$u_5 = \min(u_5, u_2 + w_{25}) = \min(\infty, 3 + 6) = 9$$

$$u_6 = w_{16} = \infty$$

$$u_3 = \min_{j \in T}(u_j) = \min(u_3, u_4, \dots)$$

$$P = \{1, 2\} \cup \{3\}$$

$$T = V \setminus P = \{4, 5, 6\} \neq \emptyset$$

Cuarta iteración, ACTUALIZACION

$$P = \{1, 2, 3, 4\}$$

$$T = \{5, 6\}$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = u_2 + w_{23} = 3$$

$$u_4 = u_3 + w_{34} = 5$$

$$u_5 = \min(u_5, u_4 + w_{45}) = \min(9, 5 + 1) = 6$$

$$u_6 = \min(u_6, u_4 + w_{46}) = \min(\infty, 5 + 3) = 8$$

$$u_5 = \min_{j \in T}(u_j) = \min(u_5, u_6)$$

$$P = \{1, 2, 3, 4\} \cup \{5\}$$

$$T = V \setminus P = \{6\} \neq \emptyset$$

Sexta iteración, FINALIZACION

$$P = \{1, 2, 3, 4, 5, 6\}$$

$$T = \emptyset$$

$$u_1 = 0$$

$$u_2 = w_{12} = 2$$

$$u_3 = u_2 + w_{23} = 3$$

$$u_4 = u_3 + w_{34} = 5$$

$$u_5 = u_4 + w_{45} = 6$$

$$u_6 = u_4 + w_{46} = 8$$

Tenemos los pesos de todos los cmc con 1 origen

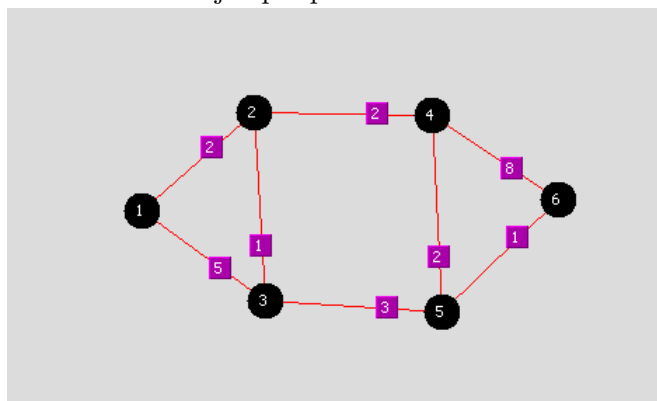
Los componentes del cmc se obtienen de las sumas

	De 1 a 2	De 1 a 3	De 1 a 4	De 1 a 5	De 1 a 6
Camino	1 2	1 2 3	1 2 3 4	1 2 3 4 5	1 2 3 4 6
Peso	2	3	5	6	8

5. **Floyd-Warshall (FW):** Al hacerlo entre cada par de vertice, el algoritmo que usamos comprueba en cada iteración si el camino pasando por un vertice en particular es más corto que el anterior, y al hacerlo de forma ordenada (al usar el algoritmo de numeración) debido a que es largo de

realizar y carezco de algo que los mortales llaman TIEMPO, dire que la idea es la misma que la de dijkstra pero en lugar de usar el peso de un arco usamos el de un camino, con el que indicamos la incision en un vertice en particular. Finalizaremos el algoritmo cuando $m = \text{card}(V) + 1$ o cuando m alcanza el valor del vertice inicial $+1$ si hemos reorganizado el orden para evitar pasar por unos vertices (recordad que el vertice inicial se ponía al final de los vertices por los que podemos poner y a continuacion de el poniamos el vertice final y/o los vertices restringidos).

Ejemplo para arboles:



WIP

	1	2	3	4	5	6
1	∞	2	5	∞	∞	∞
2	∞	2	5	∞	∞	∞
3	∞	2	5	∞	∞	∞
4	∞	2	5	∞	∞	∞
5	∞	2	5	∞	∞	∞
6	∞	2	5	∞	∞	∞

1. Kruskal: Si no necesitamos una arista en particular, las aristas se ordenan de menor a mayor peso, en el caso de si necesitarse, esta arista se pone la primera. Se iran añadiendo a un conjunto las arista de menor peso que no formen ningun ciclo.
2. Prim: Similar a dijkstra considerando que no se formen ciclos y al actualizar solo consideramos el arco, no el peso del camino hasta alcanzar el vertice de interseccion.

1.3. Álgebra y aritmética numérica simple

1.4. Ampliación de los conjuntos numéricos, md

Por ahora, vamos a centrar este tema al conjunto de los numeros enteros. Debeis de saber que este se trata de una estructura algebraica $(\mathbb{Z}, +, *)$ en la que, al definirse una serie de propiedades, podemos categorizar como anillo conmutativo unitario, y ademas, al no existir divisores de cero, diremos que es dominio de integridad(Revisar si estais interesados en la materia, no necesario). Estas propiedades para las dos operaciones internas definidas (los operandos y el resultado pertenecen al conjunto sobre el que se define, las operaciones son la aditiva $+$ y la multiplicativa $*$ usuales) con las que podemos trabajar asumiendo que son ciertas son:

1. Asociativa (para ambas) $(a \oplus b) \oplus c = a \oplus (b \oplus c)$, con $a, b, c, a \oplus b \oplus c \in \mathbb{Z}$, basicamente se dice que la prioridad de operaciones del mismo tipo dara el mismo resultado.
2. Conmutativa (para ambas) $a \oplus b = b \oplus a$, $a, b, a \oplus b \in \mathbb{Z}$, diciendo que el orden con el que operar no altera el resultado
3. Existencia del elemento neutro (para ambas) basicamente, es un valor para cada operacion en el que el resultado de un operando y el elemento neutro sea el propio operando.
4. La multiplicativa es distributiva respecto a la aditiva $a * (b + c) = a * b + a * c$
5. Existencia del elemento opuesto (solo aditiva), un valor no necesariamente unico con el que, dado un operando y el elemento opuesto, el resultado obtenido es el elemento neutro.

6. No existen divisores de cero, es decir, dado $a \neq 0$, $a * b = a * c$ la unica solucion posible seria $b = c$.
A primera vista tiene sentido pero, si trabajaras con matrices, $(1, 0) * (0, 1)^t = (1, 0) * (0, 4)^t$ tiene b y c distintos y aun asi dan el mismo valor.

Es mas, no solo diremos que trabajamos sobre esta estructura, sino que definiremos una relacion binaria sobre este conjunto (\mathbb{Z}, \leq) que cumple las propiedades de una Relacion Binaria de Orden, siendo estas:

1. Reflexiva: $\forall n \in \mathbb{Z}, n \leq n$ en particular $n = n$
2. Transitiva: $\forall x, y, z \in \mathbb{Z}, x \leq y \wedge y \leq z \Rightarrow x \leq z$, basicamente $x \leq y \leq z$
3. Antisimetrica: $\forall x, y \in \mathbb{Z}, x \leq y \wedge y \leq x \Leftrightarrow x = y$, basicamente $x \leq y \leq x$

En particular, diremos que es de orden total ya que $\forall x, y \in \mathbb{Z}, x \leq y \vee y \leq x$. Lo que quiere decir esque dados dos elementos cualesquiera del conjunto, la RBO se cumplira.

Podemos afirmar mas cosas para esta RBO para futuras definiciones como:

1. Denota que al hacer un subconjunto de este, podria estar acotado definiendo un minimo y un maximo (valores incluidos en el conjuntos pero que son las primeras cotas inferior y superior)
2. Podemos interpretar que si se cumple una igualdad, esa igualdad al tratarla con constante a ambos lados seguira siendo cierta, p.e., $a \leq b \Rightarrow a * c \leq b * c$, con $0 \leq c$

Con estas bases del conjunto numerico de los enteros, vamos a ver por un lado la aritmetica modular (que consiste en simplificar el estudio de un problema en partes, usando n° enteros) y definiremos las congruencias (en este caso definiremos lo que son las RBE)

1.4.1. Aritemtica modular

Algoritmo de la division (teorema) Dados $a, b \in \mathbb{Z}, b \neq 0, \exists! q, r \in \mathbb{Z} | a = b * q + r$ con $0 \leq r < |b|$.
El calculo para obtener q y r se llama division euclidea de a por b, siendo q el cociente, r el resto, a el dividendo (numerador) y b el divisor (denominador).

Aplicacion, representacion en base t de un entero Dado $t \geq 2, t \in \mathbb{Z}$, que indica la base sobre la que queremos estudiar, y dado $x_{10} \in \mathbb{Z}$ (un valor representado en la base usual, la 10 (se suele omitir el subindice)), si usamos de forma reiterada el algoritmo de la division usando t como divisor y los cocientes como el dividendo en la siguiente iteracion obtendremos la representacion en base t el valor x, considerando los restos como los digitos en dicha base $x_{10} \equiv (r_n r_{n-1} \dots r_0)_t$. Vamos a ver como seria el pasar de una base cualquiera a usual y, de usual a cualquier base:

- De usuarl a cualquier base: hacemos el algoritmo iterado de la division, para la primera iteracion el dividendo sera el valor en base usual pero para el resto sera el cociente de la anterior iteracion.

$$\begin{aligned} x &= q_0 * t + r_0 \\ q_0 &= q_1 * t + r_1 \\ &\dots \\ q_{n-1} &= q_n * t + r_n \quad q_n = 0 \end{aligned}$$

$$0 \leq r_i < |t|, \forall i \in I$$

La iteracion finaliza cuando el cociente es 0, en tal caso, sabremos que el valor x representado en la base t, siendo r_n el digito de mayor peso, tal que $x \equiv (r_n r_{n-1} \dots r_0)_t$.

- De cualquier base a usual: en este caso, entendemos el algoritmo al revés y resolvemos un sistema de forma regresiva, es decir, del valor x unicamente conocemos los restos que se obtienen de hacer la anterior interacción. Si vamos resolviendo de forma regresiva vamos obteniendo los cocientes con los que se iba despejando que división. Esta propiedad nos permite obtener el valor de x en función de los restos y de la base, tal que:

$$\sum_{i=0}^n t^i * r_i = t^n * r_n + \dots + t^0 * r_0 = x$$

Otras propiedades de la división Si al aplicar el algoritmo de la división observamos que el resto es nulo, la función $a = b * q$ define a a como múltiplo de b , o desde otro punto de vista, b es un divisor (o divide) a a (recordad que tratamos con números enteros). Cuando esto se cumple, se denota con $b|a$, siendo $\frac{a}{b} = q$.

Sabiendo esto, miremos algunas propiedades con $a, b, c \in \mathbb{Z}$

1. $1|a \Rightarrow a = 1 * q, a = q // a | 0 \Rightarrow 0 = a * q, 0 = q // a | a \Rightarrow a = a * q, 1 = q$
2. $a|b \wedge b|a \Leftrightarrow b = a * q \wedge a = b * p \Leftrightarrow a = b$
3. $a|b \wedge b|c \Leftrightarrow b = a * q \wedge c = b * p \Leftrightarrow a * p * q = c$
4. $a|b \wedge a|bx, \forall x \in \mathbb{Z} \Leftrightarrow b = a * q \wedge b * x = a * p \Leftrightarrow a * q * x = b * x$
5. $a|b \wedge a|c \Rightarrow a|(b * x + c * y), \forall x, y \in \mathbb{Z}$ siendo

$$b = a * q \wedge c = a * p \Leftrightarrow b * x + c * y = a * z \Leftrightarrow a * q * x + a * p * y = a * (q * x + p * y) = a * z$$

Estudio del máximo común divisor, mcd Llamamos mcd a $c \in \mathbb{Z}$ de dos enteros no nulos a y b si se cumple que c es divisor de los dos a la vez ($c|a \wedge c|b$) y múltiplo de todos los divisores comunes de a y b ($d|a \wedge d|b \Leftrightarrow d|c$, tal que podríamos definirlo:

$$D = \{x \in \mathbb{Z} | a = q * x \wedge b = p * x, q, p \in \mathbb{Z}\}$$

siendo D el conjunto de todos los divisores comunes de a y b . Diremos que $\exists! mcd(a, b) \in D, x|mcd(a, b) \Leftrightarrow mcd(a, b) = x * q, \forall x \in D, q \in \mathbb{Z}$.

El mcd lo podemos representar siempre en positivo sin importar el signo de a y b , ya que los cocientes podemos cogerlos tanto positivos como negativos.

Números primos ENTRE SI Dados dos valores $a, b \in \mathbb{Z}$, si se cumple que el mcd entre ellos es 1, diremos que son primos entre si. OJO solo podemos afirmar que lo son entre ellos, mas adelante veremos como afirmar que un número es primo.

Identidad de Bezout Usando en conjunto con otros algoritmos (veremos mas adelante) la identidad de Bezout permite obtener una solución particular a la combinación lineal de a y b para obtener su mcd mediante la suma de productos.

$$\exists s, t \in \mathbb{Z}, mcd(a, b) = a * s + b * t$$

Es mas, si consideramos $a = d * p$ y $b = d * q$, con $d = mcd(a, b)$ observamos

$$d = d * p * s + d * q * t = d * (p * s + q * t) \Leftrightarrow p * s + q * t = 1$$

Algoritmo de Euclides Este algoritmo consiste en utilizar de forma reiterada el algoritmo de la division con la finalidad de obtener el mcd. La idea consiste en, si queremos obtener $\text{mcd}(a,b)$, el divisor sera el menor de los dos y el dividendo el mayor. En cada iteracion que se haga, el resto de la anterior sera el nuevo divisor y el nuevo dividendo sera el anterior divisor. Cuando el resto obtenido en una iteracion es 0, diremos que el $\text{mcd}(a,b)$ sera el resto obtenido en la iteracion anterior a esta.

Es mas, si despejamos los restos de forma regresiva, considerando que el divisor en una iteracion es el resto de la iteracion anterior, podemos obtener la identidad de Bezout.

Veamos mas o menos como se haria para obtener el $\text{mcd}(a,b)$ y su identidad de Bezout.

$$\begin{array}{lll}
 a = q_0 * b + r_0 & & \text{De forma regresiva} \\
 b = q_1 * r_0 + r_1 & & \text{mcd}(a,b) = r_{n-1} = r_{n-3} - q_{n-1} * r_{n-2} \\
 \dots & & \\
 r_{n-2} = q_n * r_{n-1} + r_n & r_n = 0 & r_{n-1} = a * s + b * t \\
 \\
 r_{n-1} = \text{mcd}(a,b) & 0 \leq r_i < |r_{i-1}|, \forall i \in I &
 \end{array}$$

Ecuacion diofantica es el nombre que recibe la ecuacion $a * x + b * y = d$, con las incognitas y los coeficientes siendo enteros y el resultado un entero positivo, en la que podremos confirmar que existe una solucion (recordad que las incognitas son enteros) si y solo si existe $\text{mcd}(a,b)$ y es divisor del resultado. Al realizar una ecuacion diofantica se debe tener en cuenta los siguientes pasos:

1. Muchas veces la ecuacion se obtiene de igualar dos descripciones distintas para un valor que se quiere calcular, p.e., $X = a * x$ y $X = b * y + d$, si igualamos tendriamos $a * x = b * y + d \Leftrightarrow a * x - b * y = d$
2. En algunos casos, por no decir la mayoria, puede que la ecuacion venga en formato de resta y no de suma. No pasaria nada pero, para mantener el proceso lo mas estricto posible, puedes redefinir las incognitas tal que $x' = |x| \wedge y' = |y|$, solo tendras que tener en cuenta al finalizar el cambiar el signo donde sea necesario. (en las siguientes explicaciones consideraremos que no ha hecho falta, solo por ne tener que escribir x' e y' todo el rato)
3. Al obtener el mcd, tambien obtener la identidad de bezout ya que esta permite obtener el resultado de la ecuacion para un caso particular (puede ser el resultado para nuestro problema o no, depende de las condiciones dadas). Para ello, si tenemos $c = \text{mcd}(a,b)$, $c = a * s + b * t$ y tenemos que $a * x_0 + b * y_0 = d = c * k$, entonces

$$a * x_0 + b * y_0 = c * k = (a * s + b * t) * k = a * s * k + b * t * k$$

Si el mcd obtenido no es divisor del resultado de la ecuacion que estamos estudiando, entonces no existe solucion para dicha ecuacion con valores enteros.

4. Usaremos el siguiente sistema con el que poder comprobar todos los casos de valores enteros para las incognitas que cumplirian la igualdad de la ecuacion:

$$\begin{array}{l}
 x = s + p * \frac{b}{\text{mcd}(a,b)} \\
 y = t - p * \frac{a}{\text{mcd}(a,b)}
 \end{array}$$

$$\begin{array}{l}
 a * x + b * y = n * d \text{ solucion general} \\
 a * x_0 + b * y_0 = d \text{ solucion particular} \\
 a * s + b * t = \text{mcd}(a,b) \text{ identidad de Bezout}
 \end{array}$$

Recordad que si habeis redefinido el signo de alguna de las incognitas, debeis de tenerlo en cuenta al operar mas adelante.

5. Obtener el conjuntos de valores para p en los que se obtiene un resultado dentro de las condiciones dadas. Recordemos que en el apartado 1 habiamos dicho que queriamos obtener X , el cual

podíamos obtener mediante una ecuación con x , y otra con y . En ocasiones, este valor puede no ser único, si no que está delimitado por alguna condición.

Para ello, deberemos probar que valores de p en $(x = s + p * \frac{b}{\text{mcd}(a,b)})$, por ejemplo, obtiene una X dentro de dichas condiciones.

Mínimo común múltiplo Dados dos enteros positivos, a y b , llamaremos mcm a c si este es el menor entero positivo que sea múltiplo común de a y b , es decir,

$$c = \text{mcm}(a, b) \Leftrightarrow d \in \mathbb{Z}^+, a|d \wedge b|d, c \leq d$$

recordando que $a|b \Leftrightarrow b = a * q$, b es múltiplo de a .

1. La identidad de Bezout y todas las aplicaciones que le hemos dado son aplicables a mcm
2. Todos los múltiplos del $\text{mcm}(a,b)$ serán múltiplos comunes de a y b .

Definición de número primo Dado cualquier entero positivo mayor que 1, diremos que es primo si y solo si tiene dos divisores enteros, el 1 y el mismo.

Descomposición en factores primos de n Como cualquier entero positivo mayor o igual que 2 se puede escribir como una única combinación (orden no considerado) de producto de primos, podemos descomponer el entero n en factores primos.

Para descomponer n en factores primos se va probando si son primos entre sí, es decir, dado $p \in \mathbb{Z}^+$ un primo cualquiera, si el $\text{mcd}(n,p)=1$, podemos afirmar que es un factor. PERO, es más cómodo si en lugar de ir probando si son primos entre sí, aplicamos el algoritmo de la división de forma reiterada siendo el divisor el menor primo que haga que el resto sea 0. Cuando el cociente no pueda ser otro más que 0 ya que ningún primo (1 no es primo). Por ejemplo:

$$\left. \begin{array}{l} n = q_0 * p_0 + 0 \\ q_0 = q_1 p_1 + 0 \\ \dots \\ q_m = 0 * p_{m+1} + 1 \end{array} \right\} \begin{array}{l} n \\ q_0 \\ \vdots \\ q_m \end{array} \left| \begin{array}{l} p_0 \\ p_1 \\ \vdots \end{array} \right. \Leftrightarrow n = \prod_{i=0}^m p_i = p_0 * p_1 * \dots * p_m$$

Es más, es posible que varios números primos se repitan y se obtengan potencias de dicho primos, con lo cual el valor n lo podemos descomponer de la siguiente manera:

$$n = \prod_{i=0}^m p_i^{e_i} = p_0^{e_0} * p_1^{e_1} * \dots * p_m^{e_m}$$

Obtener mcm y mcd mediante descomposición Un método muy rápido para obtener el mcm y mcd de dos valores es descomponerlos a sus factores primos. Ya que por definición el mcm y el mcd son enteros positivos, por lo general mayores de 1, estos también son producto de primos. El mínimo común múltiplo tomara de entre los dos valores, el primo con mayor exponente y, el máximo común divisor, tomara aquellos con menos exponente (el 0 será obviamente el menor), quedándose tal que:

Dado $a = p_0^{e_0} * p_1^{e_1} * \dots * p_m^{e_m}$ y $b = p_0^{r_0} * p_1^{r_1} * \dots * p_m^{r_m}$ obtenemos el $\text{mcm}(a,b)$ y el $\text{mcd}(a,b)$ con

$$\left. \begin{array}{l} \text{mcm}(a, b) = \prod_{i=0}^m p_i^{\max(e_i, r_i)} \\ \text{mcd}(a, b) = \prod_{i=0}^m p_i^{\min(e_i, r_i)} \end{array} \right\} \text{Recordad que consideramos todo primos con } p^0 = 1$$

Es más, el producto de a y b es igual al producto de mcm y mcd ya que se consideran en ambos lados el producto con los mismos primos.

1.4.2. Congruencias, relacion binaria de equivalencia

Congruencias Dado un entero mayor que 1, n , y dos enteros a y b , diremos que a es congruente con b modulo n si $a - b = k * n, k \in \mathbb{Z}$ y lo denotamos con $a \equiv b(\text{mod } n)$

La congruencia define una relacion binaria de equivalencia las cuales define las siguientes propiedades(no son necesarias de conocer):

1. Reflexiva: $\forall a \in \mathbb{Z}, a \equiv a(\text{mod } n)$ en particular $a - a = 0 = 0 * n$
 2. Transitiva: $\forall x, y, z \in \mathbb{Z}, (x \equiv y(\text{mod } n) \Leftrightarrow x - y = p * n) \wedge (y \equiv z(\text{mod } n) \Leftrightarrow y - z = q * n) \Rightarrow (x \equiv z(\text{mod } n) \Leftrightarrow x - z = k * n)$, basicamente $(x - y) + (y - z) = p * n + q * n = (p + q) * n = k * n$
 3. Simetrica: $\forall x, y \in \mathbb{Z}, (x \equiv y(\text{mod } n) \Leftrightarrow x - y = p * n) \Rightarrow (y \equiv x(\text{mod } n) \Leftrightarrow y - x = q * n)$, basicamente se afirma que si se puede hacer x congruente con y modulo n , el reciproco tambien se obtiene.
- Es mas, diremos que, dado $a \equiv b(\text{mod } n)$, a es equivalente a b modulo n . Para ser mas concreto, el conjunto de elementos equivalente generan las llamadas clases de equivalencia, en la que podremos tomar un representante de clase. Para un n dado, tenemos

$$[a] = \{b \in \mathbb{Z} | a \equiv b(\text{mod } n) \Leftrightarrow a - b = k * n, k \in \mathbb{Z}\}$$

Diremos que $[a]$ es la clase de equivalencia que agrupa todos los elementos equivalentes a a , siendo a el representante de la clase.

Congruencia modulo 9 Para cuando tratamos con base usual (10) y modulo 9, podemos sustituir un valor por el sumatorio de sus digitos hasta obtener un solo digito.

Esto sirve para estudiar operaciones ya que puedes simplificarlo mediante el equivalente, es decir, si tenemos $a * b = c$, si consideramos $a \equiv a'(\text{mod } 9) \wedge b \equiv b'(\text{mod } 9) \wedge c \equiv c'(\text{mod } 9)$, se tiene que dar que $a' * b' = d'$ simplificado al digito y tiene que darse que $d' = c'$. Indicandote que pueden ser equivalente pero la operacion puede que no sea correcta. (Sinceramente, dudo que esto se use porque es algo muy concreto)

Enteros modulo n Podemos definir un subconjunto de los enteros que esta compuesto por las clases de equivalencia cuyos representantes van desde 0 hasta $n - 1$, es decir,

$$\mathbb{Z}_n = \{[0], [1], \dots, [n - 1]\}$$

Es mas, por el algoritmo de la division $a = q * n + r \Leftrightarrow a - r = q * n$ que es la definicion de congruencia, indicando que a y r son equivalentes. Como son equivalentes, podemos usar tanto a como r como el representante de la clase de equivalencia.

Vamos a recalcar los pasos a hacer para obtener un nuevo representante de clase:

1. Nos dan el entero de modulo n y el representante de clase $[a]$, con esta informacion sabremos que habran $n - 1$ clases de equivalencia distintas.
2. Usamos el algoritmo de division para descomponer a , tal que $a = q * n + r$ En ocasiones, si se da que el coeficiente y el resto estan en negativo(o no hay coeficientes y el resto esta negativo), podemos aprovechar la propiedad de $0 = n - n$ para tomar el equivalente al negativo del resto pero en positivo (ya que tambien es el resto)
3. Por definicion, debiera de ser menor que n con lo cual aseguramos que es un posible representante para una clase de equivalencia en concreto.

Operaciones inducidas en \mathbb{Z}_n consisten en las operaciones aditiva y multiplicativa de los representantes y posteriormente reducir a un representante menor.

Elementos inversibles Definimos \mathbb{Z}_n^* como conjunto de elementos inversibles de \mathbb{Z}_n para el producto, siendo equivalentes:

1. $[a] \in \mathbb{Z}_n^*$ a sera elemento invertible
2. $\exists [b] \in \mathbb{Z}_n$ tal que $[a][b] = [1]$ a sera invertible para un b, siendo [1] el representante neutro.
3. $\exists b, k \in \mathbb{Z}$ tal que $ab - kn = 1$, es una demostracion trivial ya que $[a] \Leftrightarrow a = a + 0 * n \wedge [b] \Leftrightarrow b = b + 0 * n$ entonces podria darse $(a + 0) * (b + 0) = 1 + kn \Leftrightarrow ab - kn = 1$
4. $\text{mcd}(a, n) = 1$, el elemento inverso y el modulo deben de ser primos entre si

Para hallar el equivalente a un elemento inverso seguiremos los siguientes pasos:

1. Dado el representante del elemento inverso, $[a]^{-1}$ y el conjunto de enteros modulo n, \mathbb{Z}_n . Procedemos a obtener el $\text{mcd}(a, n)$. Se puede hacer mediante el algoritmo de euclides o descomposicion en factores, como se prefiera (pero por el siguiente paso se recomienda por euclides.

$$n = q_0 * a + r_0$$

$$a = q_1 * r_0 + r_1$$

...

$$r_{n-1} = q_{n+1} * r_n + 0$$

El valor para $r_{n+1} = 0$ y, si n y a son primos entre si, $r_n = 1 = \text{mcd}(a, n)$

2. Si son primos entre si, obtenemos la identidad de Besout que, casualmente, tendra la siguiente estructura $1 = a * b - k * n$, el multiplicando de a sera el representante equivalente del elemento inversible en \mathbb{Z}_n .
3. Si el valor esta en negativo o es mayor que el valor de n, lo reducimos a un mejor representante mediante el algoritmo de la division y la propiedad $0 = n - n$.

Funcion de euler Permite obtener el numero de representantes que son primos respecto al modulo del conjunto de enteros, o visto de otra manera, la cardinalidad del conjunto de enteros, que denotaremos con $\varphi(n)$. Para estudiarlo suponemos un conjunto \mathbb{Z}_n y su conjunto de elementos inversibles \mathbb{Z}_n^* con $n \geq 1$. Podemos interpretarlo de dos maneras:

1. Demostrandolo para el conjunto \mathbb{Z}_n usando:

$$\varphi(n) = \text{card}\{x \in \mathbb{Z}^+ | x \leq n \wedge \text{mcd}(x, n) = 1\}$$

Con esta expresion estamos primero creando un conjunto de enteros positivos que son menor o igual a n y donde se cumple que el elemento es primo respecto a n y posteriormente contando el numero de elementos que cumplen dicha condicion.

2. Conociendo que lo que acabamos de hacer basicamente es definir el conjunto de elementos reversibles para el \mathbb{Z}_n . Anteriormente habiamos probado que todo representante del conjunto \mathbb{Z}_n^* es primo respecto al modulo, con lo cual es equivalente decir:

$$\varphi(n) = \text{card}\{[a] \in \mathbb{Z}_n^* | a \leq n \wedge \text{mcd}(a, n) = 1\} = \text{card}(\mathbb{Z}_n^*)$$

En la primera igualdad estamos confirmando que cada representante del conjunto menor que n (por definicion todos lo seran) son primos con el, y en la segunda igualdad estamos directamente considerando cuantos elementos hay en este conjunto.

- Es mas, si consideramos un conjunto cuyo modulo es un primo, los elementos reversibles son todos aquellos que son primos respecto a este, es decir, $\mathbb{Z}_p^* = \{x \in \mathbb{Z}^+ | x \leq p \wedge \text{mcd}(x, p) = 1\} = \{[1], \dots, [p-1]\}$, siendo su cardinalidad $\varphi(p) = p - 1$

A partir de la funcion podemos obtener las siguientes propiedades que definen el teorema de Euler, veamoslas:

1. Dado un representante de clase $[x] \in \mathbb{Z}_n^*$, podemos afirmar que $[x]^{\varphi(n)} = [1]$. Esto se debe a que lo que hacemos es probar para un caso trivial $x = x + 0 * n \wedge 1 = 1 + q * n \Leftrightarrow (x + 0 * n)^{\varphi(n)} = 1 + q * n \Leftrightarrow x^{\varphi(n)} - q * n = 1$. Podeis probar con cualquier valor que esto sera asi, y si se cumple con el representante se cumplira con todos los equivalentes al representante.
2. De lo anterior podemos comprender dos cosas, el valor x y n son primos entre si (por definicion del \mathbb{Z}_p^*) y mas importante, el valor $x^{\varphi(n)}$ y 1 son equivalentes. Esto lo suponemos a partir de la formula anterior, $x^{\varphi(n)} - q * n = 1 \Leftrightarrow x^{\varphi(n)} - 1 = q * n$ lo cual indica $x^{\varphi(n)} \equiv 1 \pmod{n}$.
3. Es mas, si consideramos el modulo un primo, recordamos que $\varphi(p) = p - 1$ lo que nos simplifica la operacion (recordad que tratamos con elementos de \mathbb{Z}_p^* , si fuese cualquier entero deberemos de probar que es minimo respecto a p con $\text{mcd}(x, p) = 1$). Esto recibe el nombre del teorema de fermat.
4. Dado un modulo p que es primo y un entero positivo u, el valor de la funcion de euler de p^u sera $\varphi(p^u) = p^{u-1}(p - 1)$. Debeis de considerarlo cierto ya que para simplificar el calculo de la funcion de Euler

Dado un entero positivo cualquiera, podemos descomponerlo en factores primo y si cumpliria que el producto de las funciones de euler para cada primo. Si ademas consideramos que pueden haber varios primos, podemos entender que:

1. Dado $n \in \mathbb{Z}^+$ tal que $n = n_1 n_2 \dots n_k$ siendo todo n_i primo respecto a n y, es mas, $n_i = p_i^{r_i}$ siendo p_i un primo, podemos obtener la funcion de euler tal que

$$\varphi(n) = \prod_{i=1}^k \varphi(n_i) = \prod_{i=1}^k \varphi(p_i^{r_i}) = \prod_{i=1}^k (p_i - 1) p_i^{r_i - 1}$$

Criptografia Es tremenda mierda, consiste en una funcion biyectiva (que tiene inverso) en la que tenemos

$$f : \mathbb{Z}_n \mapsto \mathbb{Z}_n \\ x \mapsto x'$$

PERO OJO, no es una funcion inverso como con las que tratamos de normal, es una en la que pueden ser dos funciones con las que tomar todo el elemento de un conjunto y relacionarlo con el otro y viceversa.

El sistema criptografico esta constituido por:

1. Dos conjuntos de mensajes, uno para transmitir y otro cifrados, que cuyos valores pertenecieran al mismo conjunto. Los conjuntos se definiran en el momento y habra que considerar las condiciones pedidas.
2. Un conjunto de parametros que actuaran como las claves, por ahora lo denotamos como K pero mas adelante veremos como usarlo. OJO, no tiene porque ser un valor, puede ser un par ordenado o una matriz, depende de como se defina el conjunto)

3. El conjunto de funciones de cifrado, las cuales agrupan todas las funciones de cifrado con todas las claves definidas en K , se denota tal que

$$C = \{C_k : M \mapsto M^*, k \in K\}$$

C_k sera una funcion para una clave k dada que se debiera de definir en funcion de k .

4. El conjunto de funciones de descifrado, las cuales seran la 'inversa' de las de cifrado. Este conjunto agrupara todas las funciones para cada clave k , tal que

$$D = \{D_k : M^* \mapsto M, k \in K\}$$

Como en el caso anterior, esta funcion debiera de definirse.

- Se da que para una clave dada, si hacemos $D_k(C_k(m)) = D_k(m^*) = m, \forall m \in M$

Se dan dos tipos de criptosistemas segun como es la clave:

1. La clave es privada, mutua para emisor (que usara las funciones de encriptado) y el receptor (que usara las de descifrado). Este tipo de sistemas permiten obtener la funcion de descifrado a partir de la de cifrado y viceversa.

Esto se debe a que, tal como se describen las dos funciones respecto a la clave, existe cierto nivel de equivalencia en la que, sabiendo que $D_k(C_k(m)) = D_k(m^*) = m, \forall m \in M$ para $k \in K$, mientras tengamos una funcion podemos obtener la otra.

2. La clave es 'publica', pero unicamente para la funcion de cifrado, para el descifrado unicamente el receptor adecuado tendra la clave que descifra el mensaje. Dicho de otra manera, se nos tienen que dar las dos funciones pero sabremos que la de descifrado, para la clave que tenemos, es correcta, si cumple unas condiciones que depende de dicha clave.

Se nos da un ejemplo en el que, si os fijais, primero nos dicen las condiciones sobre las que la clave de la que disponemos sera valida, siendo $\text{mcd}(t, \varphi(p * q)) = 1$, los tres enteros t y p y q primos, se da que $t * s - 1 = k * \varphi(p * q) \Leftrightarrow t * s \equiv 1 \pmod{\varphi(p * q)}$, de esta forma tenemos t , s , p , q como claves por ahora.

Trabajando sobre $\mathbb{Z}_n^* = M = M^*$, siendo $n = p * q$ definiremos las funciones que dependeran de las claves tales que

$$\begin{aligned} C : M &\mapsto M^* | C([m]_n) = [m]_n^t \\ D : M^* &\mapsto M | D([m^*]_n) = [m^*]_n^s \end{aligned}$$

Observamos que el par (n, t) se usa en el cifrado, pero (n, s) se usa en el descifrado, siendo esta la clave secreta. Es mas, deberemos de confirmar que esta semiclave es valida usando la condicion inicial, con lo cual deberemos de conocer los valores de p, q y s (y el de $\varphi(p * q)$ si te da pereza calcularlo o ya viene dado). Este sistema es el RSA y no creo que lo pregunte, en todo daran las funciones.

1.5. Repaso simbologia

2. Geometría

Conociendo que podemos definir estructuras algebraicas y que según las propiedades descritas en dichas estructuras podemos operar con los elementos en estos de distintas formas, vamos a ver uno de los ejemplos mas claros a los que aplicar todos estos conocimientos, la geometría. En este campo de las matemáticas definimos un cuerpo en la que existe una determinada relación binaria y veremos como podemos aplicarlo. Para explicar este tema, tendremos en cuenta:

1. La concepción de los espacios vectoriales de diversas dimensiones, cuya estructura algebraica son los cuerpos.
Demostraremos las propiedades fundamentales para poderlas considerarlas ciertas en todo momento durante las explicaciones siguientes y el uso de ecuaciones y distintas formas de representarlas.
2. Mostrar ejemplos del plano y el espacio (espacios vectoriales de 2 y 3 dimensiones) y definir conceptos que se definen en estos.
3. La utilidad de los vectores y valores propios.

3. Introducción al álgebra lineal, m1

El álgebra es una rama de las matemáticas que estudia conceptos que podemos representar gráficamente, es decir, los vectores (unidad más sencilla), matrices y sistemas lineales (ambas están correlacionadas y se componen de vectores, y espacios vectoriales (nuestra área de trabajo, nos ofrece los componentes y herramientas en forma de propiedades con las que operar) y sus transformaciones lineales. Hagamos una breve explicación para entenderlos mejor y poder relacionar los conceptos entre sí (cabe destacar que esta no es parte del temario pero se ve necesario):

1. Vectores: elementos básicos de un espacio vectorial que unen dos puntos que se encuentran en él, sabemos además que todo espacio vectorial contiene al menos uno ya que por definición este contiene al vector nulo, un punto en sí. Los vectores se pueden entender como un polinomio, una función con varias incógnitas (por lo general, estas incógnitas ayudan a saber cuántas dimensiones posee un espacio vectorial ya que corresponden a los valores en cada eje).
2. Sistemas lineales: conjunto de ecuaciones paramétricas que pueden o no compartir soluciones para las incógnitas, siendo esta la condición principal para distinguir distintos tipos. Gráficamente se consideran un conjunto de rectas (ecuaciones de rectas y planos en el anexo) y los valores de las soluciones indican en qué puntos se cruzan, las . Pueden realizarse operaciones entre las ecuaciones, debido a la combinación lineal descrita en el espacio vectorial (se verá más adelante).
3. Matrices: pueden verse como una reorganización de los sistemas lineales que actúan como un elemento de por sí. Se definen operaciones y propiedades en particular para estos elementos y son de utilidad para relacionar sistemas lineales o para descomponerlos en sus partes básicas, las incógnitas, sus coeficientes y los términos independientes.
4. Espacio vectorial: la explicación más ampliada de este concepto se verá más adelante pero bastará con saber que es una región de un determinado número de dimensiones (cuántos ejes tiene, para entendernos) con la que estaremos trabajando en todo momento y estas descritas unas propiedades que consideraremos ciertas para todo momento (no es lo mismo trabajar en un espacio vectorial formado por los números reales que por el conjunto de números binarios, la suma convencional no tiene porque dar lo mismo en ambos casos pero ambas describirían un espacio vectorial).

Los vectores y las ecuaciones de la recta no se consideran necesarias estudiarlas pero repasarlas no estaría mal. Hemos comentado que podemos clasificar los sistemas lineales según la solución para las incógnitas o parámetros (según como interpretes cada ecuación las puedes llamar de una forma u otra pero son lo mismo al final), veamos qué métodos de resolución tenemos disponibles. Cabe destacar que, como ya se ha dicho, las matrices se pueden entender como SL reorganizados, por lo tanto se operará mucho con estas, se recomienda revisar el siguiente tema, al menos los conocimientos básicos de las matrices, para poder ir con soltura en las explicaciones (se indicará en **negrita y cursiva** cualquier concepto que consideremos sabido a partir de haber realizado dicha revisión).

Según el número de soluciones, distinguimos:

1. Sistemas compatibles (o SL consistentes): aquellos sistemas en los que existe soluciones que relacionen las rectas. Si existen mas de una solucion hablaremos de un Sistema compatible indeterminado (SCI), donde estas soluciones definiran tramos de la recta en la que se sobreponen. Si es una unica solucion hablaremos de Sistemas compatibles determinados, donde se cruzan en un unico punto y no mas. Es mas, diremos que un SC es homoganeo si el termino independiente en todas las ecuaciones es 0 (una solucion trivial seria considerar las incognitas 0, llamaremos trivial a los procedimientos sencillos de obtener).
2. Sistemas incompatibles (SLI y no, no es la tecnologia de Nvidia para aparear tarjetas graficas): si no se encuentra ninguna solucion, las ecuaciones graficamente no se tocan, no existe relacion entre ellas alguna.

Entenderemos dos metodos de resolucion de SL, el directo y el iterativo, veamoslos:

3.1. Resolucion de SL por metodos directos

Usados para sistemas relativamente pequeños, los metodos directos consisten en aplicar un numero finito de pasos conocidos a priori con los que calcular la solucion exacta. No se recomienda realizar redondeo ya que estan sujetos a error. Distinguimos 4 metodos al menos:

Gauss y Gauss-Jordan

Consideramos un Sistema lineal tal que:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = d_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = d_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = d_m \end{cases}$$

donde a_{ij} son los coeficientes para las x_j incognitas y d_i los terminos independientes correspondientes a cada operacion. Sabiendo los componentes de una sistema lineal, distinguiremos las siguientes matrices:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{pmatrix} \Leftrightarrow \begin{pmatrix} a_{11}x_1 & a_{12}x_2 & \cdots & a_{1n}x_n & d_1 \\ a_{21}x_1 & a_{22}x_2 & \cdots & a_{2n}x_n & d_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1}x_1 & a_{m2}x_2 & \cdots & a_{mn}x_n & d_m \end{pmatrix}$$

o, mas simplificada, $Cx = D \Leftrightarrow A$ donde C es la matriz de coeficientes formada por $a_{ij} \in \mathbb{R}, \forall i, j \in \mathbb{N}$, x es la matriz de incognitas, D es la matriz de terminos independientes y A es la matriz ampliada que podemos escribir tambien como $[Cx|D] = A$.

Para resolver por Gauss y Gauss-Jordan se deben de obtener primero **la matriz escalonada** y, en el caso de usar Gauss-Jordan, operarla con operaciones internas (llamalas combinaciones o transformaciones lineales, combinas dos ecuaciones) para obtener **la matriz reducida**. Debido a que todo lo que se ha hecho hasta hora son combinaciones entre las ecuaciones, la matriz inicial (la ampliada en este caso) y las matrices escalonada y escalonada reducida son **matrices equivalentes**, y por lo tanto los resultados obtenidos en unos son validos para los otros.

Ademas, si nos damos cuenta, estas matrices siguen siendo SL, con lo cual lo que estamos realmente haciendo es una sustitucion regresiva para despejar las incognitas. Si tenemos esto en mente el metodo de Gauss y Gauss-Jordan es muy util para despejar los sistemas compatibles (determinado si se nos da una unica solucion e indeterminado si una incognita depende de otra) o incompatible si vemos que dado el valor 0 para todas las incognitas (toda una fila de 0 en la matriz Cx escalonada/reducida) se un valor no nulo en la misma fila para los terminos independientes.

Método de la inversa

Habiendo visto lo que es una **matriz invertible** y una **matriz inversa**, además de saber que operando con matrices, **la matriz identidad** es el elemento neutro en operaciones multiplicativas, el método de la inversa se basa en la siguiente propiedad:

Dado una matriz de coeficientes invertible C , una matriz de coeficientes x y una matriz de términos independientes d , ya que $CC^{-1} = I$ y el sistema es $Cx=d$, podemos despejar la matriz incógnitas de la siguiente forma:

$$x = C^{-1}d \Leftrightarrow Ix = C^{-1}d \Leftrightarrow C^{-1}Cx = C^{-1}d$$

Es importante respetar las propiedades de operar con matrices y, obviamente, el conocer como despejar una matriz inversa, con lo cual se recomienda mirarse el tema siguiente.

Descomposición LU

Para entender este método de m... si o si necesitas mirarte el tema siguiente, miratelo... vuelve y ya hablamos porque vamos...

Método que requiere de que se den unas condiciones iniciales y de gran utilidad para despejar incógnitas cuando se nos dan una gran variedad de matrices de términos independientes para una misma matriz de coeficientes. Descompondremos la matriz de coeficientes en:

1. Matriz U , matriz escalonada de C en la que todos los elementos por debajo de la diagonal son 0 (matriz triangular inferior) y, si se obtiene intercambiando filas de posición, este método ya no es válido. Se permite usar el siguiente tipo de combinación $F_I \Leftrightarrow F_I + \lambda_{IJ}F_J$ y, además, será importante dejar constancia de esto en la resolución.
2. Matriz L , matriz cuadrada invertible (de tamaño en proporción a la cantidad de elementos por debajo de la diagonal de U) que se obtiene a partir de poner en las posiciones (i,j) el λ_{IJ} en negativo correspondiente. Veremos entonces que todos los elementos de la diagonal serán 1 y por encima de ella 0.

Para llevar a cabo este método, lo que se hará, teniendo $Cx=d$, aplicaremos la igualdad $C=LU$ obteniendo que $LUx=d$. Primero se obtendrá un 'matriz de apoyo' y respecto a los términos independientes $Ly=d$, siendo SCD resoluble por sustitución progresiva. Lo siguiente será obtener las incógnitas correspondientes a los mismos términos usando esta matriz de apoyo usando la igualdad $Ux=y$ usando Gauss.

Método de Rouché

Esta se propone ya que es muy rápida y apenas necesitas realizar operaciones complejas, al menos se puede usar para saber que tipo de sistema lineal es. Consiste en considerar el rango (n° líneas (ya sean filas o columnas, por lo general filas) independientes entre sí, es decir, que no son combinación de otras dos) y el n° de incógnitas. El rango lo podemos estudiar haciendo submatrices $M_{n \times n}$ y calculando su **determinante** con $n \leq (n^\circ \text{ filas o } n^\circ \text{ columnas (se coge el menor ya que esta será el mayor rango posible)})$.

Si todos los determinantes de una submatriz $M_{n \times n}$ con un n en concreto dan 0, podemos afirmar que es de rango $n-1$ pero, si un solo determinante da distinto de 0, diremos que su rango es mayor o igual a n y probaremos con $n+1$. Por lo general se empieza estudiando el rango de la matriz de coeficientes y después se prueba la ampliada intercambiando una de las columnas por la matriz de términos independientes.

Las relaciones serían:

SLI: $R(C); R(A)$

SCD: $R(C)=R(A)=n^{\circ}$ de incognitas

SCI: $R(C)=R(A); n^{\circ}$ de incognitas

3.2. Resolución de SL por métodos iterativos

Método que resuelve un SL a base de partir de un caso inicial e ir probando si se aproxima a la solución, no pierde tiempo en buscar como solucionarlo y haciendo los cálculos necesarios para ello. Aunque parezca que no es muy práctico, se nota su eficacia a la hora de trabajar con un número muy elevado de variables y el realizar operaciones con semejante número de incógnitas sería complicado. Este método tiene unas ventajas (desde el punto de vista informático) ante los métodos directos, son menos sensibles de error en el redondeo y además, solo es necesario almacenar los valores no nulos del sistema.

Valiente h... de p... quien se vaya a ver esto, me está dando mucha pereza el memorizar el algoritmo de Jacobi

4. Matrices, m1

Como necesitas definir algunas operaciones para poder obtener algunos tipos de matrices, dividiremos este tema de una forma alternativa a la del temario dado, dando primero conocimientos básicos (tipos de matrices básicas que no requieren de operaciones, operaciones entre matrices básicas,... todo lo que se pueda considerar necesario para conocer otro tipo de matrices), matrices compuestas (matrices obtenidas por distintas operaciones, acompañadas por operaciones más avanzadas) y una breve recopilación de todas las matrices y funciones con una breve descripción.

4.1. Conocimientos básicos

Las matrices son conjuntos de elementos ordenados en n filas y m columnas tal que a cada elemento le corresponde una posición, denotando a la matriz con mayúsculas $A_{n \times m}$ y al elemento $a_{ij} \in \mathbb{R}$ con $i \leq n, j \leq m$ siendo $n, m, i, j \in \mathbb{N}$. Llamaremos dimensión de una matriz A a las dimensiones $n \times m$ de estas y rango de una matriz al número de filas o columnas linealmente independientes, es decir, que no sean una combinación de otras dos.

Podemos distinguir muchos tipos de matrices ya sea por su forma (matrices fila, columna, rectangular,...) o sus elementos (nula, diagonal escalar o identidad, triangular,...) además de otras que son resultado directo de operaciones con matrices elementales (matriz escalonada, traspuesta, reducida,...). Las veremos por encima:

- Matrices fila y columna, también conocidas como vectores, son matrices en las que únicamente estas compuestas por una fila ($A_{1 \times m}$) o una columna ($A_{n \times 1}$).
- Matriz rectangular, matrices $A_{n \times m}, n \neq m$.
- Matriz cuadrada, matrices $A_{n \times n}$, también podemos llamarlas matrices de orden n . Son las matrices más importantes para demostrar propiedades y las podemos clasificar como simétricas si $a_{ij} = a_{ji}$ o antisimétrica si $a_{ij} = -a_{ji}, i \neq j$ y $a_{ii} = 0$ con $i, j \leq n$ siendo $n, i, j \in \mathbb{N}$.
- Matriz diagonal, matrices en las que todos los elementos salvo los de la diagonal (a_{ii}) son nulos. No importa si es rectangular, cuadrada, fila o columna, la diagonal se considera existente a partir del primer elemento de la matriz. Podemos llamarla diagonal escalar si todos los elementos de la diagonal son el mismo valor y, si además, este valor es el uno y la matriz es cuadrada, la llamaremos Matriz identidad, una matriz de gran utilidad.

- Matriz elemental, es una matriz identidad en la que se ha realizado una única operación correspondiente a la realizada en otra operación. Estas matrices si se prefiere entender como matrices de registro de operaciones creo que será más cómodo de recordar su función. Se dan tres operaciones: permutaciones (cambio de líneas), multiplicar una fila por un escalar y combinación lineal (una línea es resultado de una operación entre ella misma y otra acompañada por un escalar).
- Matriz triangular, son matrices en las que todos los elementos por encima o por debajo de la diagonal, no ambas a la vez, son nulos. Serán de triangular superior si los elementos nulos están por encima y triangular inferior si están por debajo. Son de gran utilidad ya que se pueden dar para sustitución progresiva y regresiva, respectivamente. No importa la forma.
- Matriz escalonada y reducida, son matrices triangulares inferiores que se obtienen a partir de otra matriz usando las operaciones de las matrices elementales. El elemento principal de una línea es el primer elemento que nos encontremos no nulos al recorrer dicha línea (ya sea fila o columna). En las matrices escalonadas se busca tener 1 principales en las filas y en la reducida 1 principales en las columnas (no siempre se podrá, depende de que tipo de SL sea la matriz). Se pueden dar casos de que, si se trata de un SCD, una matriz escalonada reducida de el valor de cada incógnita directamente.
- Las matrices equivalentes es la relación existente entre una matriz y otra sobre la que se le han aplicado las operaciones de una matriz elemental (la matriz original por todas las matrices elementales(?)), por definición las matrices escalonada y reducida son equivalente a su matriz original.
- Matriz traspuesta, matrices en las que se intercambian filas por columnas tal que el elemento a_{ij} en la matriz normal es el elemento a_{ji} en la traspuesta, obviamente en la diagonal no se verá cambio ni en las matrices simétricas.
- Submatriz o bloque de $A_{n \times m}$, matrices que se obtienen al omitir alguna fila o columna de la matriz A. Es usada para estudiar matrices más grandes por bloques y no importa la forma, pueden ser rectangulares o cuadradas.

Sabiendo ya las matrices básicas, veamos como trabajar con ellas:

- Operación asociativa:

Las matrices integrantes deben de ser de las mismas dimensiones y los elementos en las mismas posiciones se suman. Se distinguen las propiedades asociativa (aislar operaciones por prioridad), conmutativa (cambiar el orden de prioridad), la existencia del elemento opuesto (una matriz y su opuesto dan matriz nula) y la existencia del elemento neutro (siendo el elemento neutro la matriz nula, al operar una matriz con esta el resultado vuelva a ser la matriz).

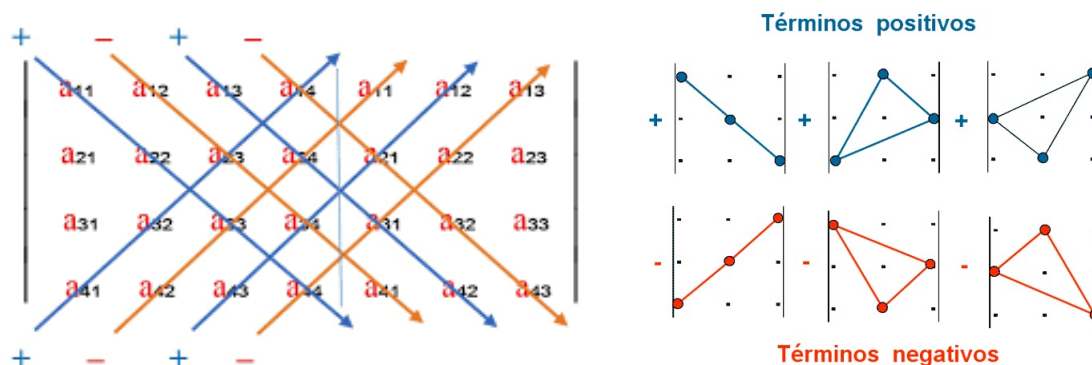
- Operación multiplicativa:

Se puede distinguir la multiplicación de un escalar por una línea (operación de las matrices elementales), un escalar por una matriz (todos los elementos de la matriz se multiplican por dicho escalar) o dos matrices. Para multiplicar dos matrices se tiene que el número de columnas de la primera es igual al número de filas de la segunda. La operación consiste en dada una posición en la matriz resultado, se realiza un producto escalar escalar (o punto, como lo llaman aquí) de la fila correspondiente de la primera por la columna correspondiente de la segunda traspuesta (fila por columna).

Distinguimos que la conmutativa no existe, y si lo hace, diremos que esas matrices conmutan, no es simplificable (no existe cancelación), se cumple la asociativa, la distributiva y existe el elemento neutro, que es la matriz identidad. Se puede también dar potencia de matrices al multiplicar las por sí mismas y, además, estas son cíclicas, es decir, dado un número finito de veces, el resultado de multiplicar tantas veces la matriz es la matriz identidad, ayudando a operar potencias elevadas.

- Determinante:

Es un escalar asociado a una matriz cuadrada. El metodo de obtenerlo varia pero todos son aceptados, se dejara un ejemplo a continuacion:



La matriz se amplia

La matriz se mantiene

Cada linea es una operacion multiplicativa entre los terminos que la componen y se realiza una operacon asociativa con todas las lineas considerando el termino correspondiente.

Se puede tambien obtener por los adjunto de una linea, siendo una suma de los elementos de una linea por su adjunto correspondiente.

Los determinantes cumplen varias propiedades faciles de demostrar, el determinante de una matriz suma/producto es igual a la suma/producto de los determinantes de dicha matriz, al multiplicar una linea de la matriz por un escalar, el determiante tambien se multiplica por este, la permutacion de lineas en una matriz puede cambiar el signo del determinante pero no el valor, si hay un vector nulo en la matriz, el determiante sera nulo y si una linea es combinacion de otras, el determinante es nulo.

Usaremos los determinantes para estudiar rangos o para otro tipo de operaciones mas complejas.

- Adjunto(no en el temario pero se adjuntara):

Sabiendo que el menor complementario de un elemento es el determinante que se obtiene se la submatriz en la que se omiten la fila y la columna en la que este elemento se encuentra, diremos que el adjunto a este elemento es el menor complementario por el signo correspondiente a su posicion $((-1)^{i+j})$. Esta operacion se usara para obtener determinantes o para definir la matriz adjunta.

4.2. Matrices compuestas

Con los conceptos anteriores podemos definir:

- Matriz adjunta:

Matriz que se forma a apartir de los adjuntos de otra matriz.

- Matriz invertible:

Si dada una matriz cuadrada y un numero finito de matrices elementales (recordar que estas son matrices que registran una unica operacion en una matriz identidad), se cumple $AE_k E_{k-1} \dots E_2 E_1 = I$ diremos que la matriz A es invertible. Es mas, por definición, toda matriz elemental es invertible seguro ya que existe un inverso de este que permite obtener la identidad.

En el caso de que no exista ninguna combinación en la que se obtiene I, hablaremos de una matriz singular. La inversa de la permutación es ella misma, la inversa de multiplicar una fila por un escalar es dividir y la inversa de sumar un elemento es restar. Entonces podemos también decir que $A = E_k^{-1} E_{k-1}^{-1} \dots E_2^{-1} E_1^{-1} I$ o lo que es lo mismo (ya que I es el elemento neutro podemos

omitirlo) $A = (E_k E_{k-1} \dots E_2 E_1)^{-1} \Leftrightarrow A^{-1} = E_k E_{k-1} \dots E_2 E_1$ siendo A^{-1} la matriz inversa de A y que por definición esta es también invertible y además $AA^{-1} = I$.

Se puede obtener de dos formas, con la demostrada multiplicando todas las matrices elementales necesarias para hacer A la matriz identidad (mucho más rápida y sencilla) o con la matriz adjunta traspuesta partida el determinante.

5. Espacio vectorial, m1

Vamos a considerar que se conocen al menos que son los vectores y las operaciones descritas con ellos (escalar con cos y vectorial con sen), y sabiendo que una base la forman tantos vectores como la dimension con la que trabajamos y que estos sean no coplanarios, es decir, que no son combinaciones entre ellos, diremos que es ortogonal si estos vectores son perpendiculares, normal si son de modulo uno y ortonormal si las dos anteriores. También será necesario que -40°C son -40°F , es un dato de extrema importancia.

Siempre que hemos hablado de puntos y vectores en un espacio, hemos dado por entendido el concepto de espacio sin saber realmente en que consiste y porque podemos trabajar con estos elementos sobre el, veámoslo.

5.1. Espacio y subespacio vectorial

El espacio vectorial V y el subespacio vectorial W tal que $W \subset V$, es decir, W es un subconjunto de elementos de V , son conjuntos de elementos expresados vectorialmente en V y W , respectivamente, en las que se definen una operación interna suma (operan dos elementos del mismo conjunto) y una operación externa multiplicativa (un elemento del conjunto dado opera con otro elemento externo a este conjunto, por lo general serán un conjunto de escalares). En la operación interna distinguimos la asociativa, distributiva, conmutativa y existencia del elemento neutro y opuesto y en la operación externa la asociativa, distributiva y el elemento neutro.

(Fuera del temario) Para demostrar que tanto el EV como el SEV son en si espacios vectorial, bastaría con probar que $\alpha\vec{u} + \beta\vec{v} \in V, \forall \vec{u}, \vec{v} \in V, \alpha, \beta \in \mathbb{E}$ siendo V el conjunto con el que trabajamos para estudiar si es EV y \mathbb{E} un conjunto de escalares (puede ser el conjunto de números reales o algún subconjunto). Con esta demostración se prueba que operando con elementos internos se puede dar otro elemento interno, es decir, que la combinación lineal de elementos en EV pertenece, a su vez, a EV. Además, siempre que podamos definir un vector nulo, podremos afirmar que es un EV ya que la demostración es trivial.

Veamos algunos conceptos del espacio vectorial y recordemos que estos son aplicables al subespacio.

Combinación lineal

Se ha comentado el concepto de combinación lineal anteriormente y este es fácil de entender por contexto, pero veamos en que consiste. Dado un vector $\vec{v} \in \mathbb{R}^n$ donde n es la dimension del espacio vectorial de \mathbb{R} , y dada la siguiente igualdad $\vec{v} = \alpha_1\vec{u}_1 + \alpha_2\vec{u}_2 + \dots + \alpha_k\vec{u}_k = \sum_{i=1}^k \alpha_i\vec{u}_i$ donde $\vec{u}_i \in \mathbb{R}^n, \alpha_i \in \mathbb{R}$ con $i=1, \dots, k$ diremos que \vec{v} es combinación lineal (CL) para unos vectores \vec{u}_i y unos escalares α_i dados.

Se puede demostrar haciendo un SL y despejando $\vec{v} = \alpha_1\vec{u}_1 + \alpha_2\vec{u}_2 + \dots + \alpha_k\vec{u}_k$ en los componentes cartesianos para obtener dicho SL. Si es SCD, existe una única combinación lineal para obtener este vector, si es SCI, existen varias opciones para obtener el vector \vec{v} y si es SLI entonces este vector no es combinación lineal de los vectores \vec{u}_i . Las incógnitas suelen ser los escalares ya que se quiera demostrar si unos vectores pueden obtener otro, con lo cual si realizamos un sistema matricial $A=[C\ x-D]$, C

seran los escalares, x los vectores \vec{u}_i y D el vector \vec{v} . En el caso de que sea SCD, diremos que que C y D son compatibles.

Envoltura lineal

Llamaremos envoltura lineal de V al conjunto de vectores con los que poder escribir todos los vectores de V, es decir, $\forall \vec{v} \in \mathbb{R}^n, \exists! \vec{u}_1, \dots, \vec{u}_n \in \mathbb{R}^3, \vec{v} = \alpha_1 \vec{u}_1 + \alpha_2 \vec{u}_2 + \dots + \alpha_n \vec{u}_n$ con $\alpha_i \in \mathbb{R}, i = 1, \dots, n$. Los vectores \vec{u}_i sera vectores generadores o un subconjunto de \mathbb{R}^3 generador del espacio \mathbb{R}^3 . Si dos elementos que forman una envoltura son combinacion lineal y lo demostramos, podremos omitir dicho elemento.

Por lo general, el conjunto generador tendra tantos elementos como dimensiones tenga el espacio generado (una envoltura con un elemento define una linea, con dos eslementos un plano, con tres un espacio) aunque no tiene porque, si el conjunto y la envoltura coinciden, llamaremos base a esta envoltura ya que es el menor conjunto capaz de generan al conjunto.

Para demostrar que un conjunto es generador bastaria con probar que un vector cualquiera del conjunto generador es compatible con una unica combinacion de escalares, es decir, probar que es un SCD. Si fuese SCI, uno de los vectores generadores seria combinacion lineal de otro.

Sera necesario saber que significa que unos vectores linealmente independientes y linealmente dependientes para poder hacer mas claras algunas explicaciones de demostracion.

Un conjunto de vectores linealmente independientes son vectores que, la unica combinacion posible para obtener el vector nulo, seria que todos los escalares fueran ceros. Graficamente se observa que estan con el mismo angulo respecto a los demas (cuando haces un sistema de referencia, el ehe x, y y z son linealmente independientes).

Un conjunto de vectores es linealmente dependiente si al menos uno de estos es combinacion lineal de otro, lo que resultaria en que, para obtener el vector nulo, existiria mas de una operacion. Se vera cuando queramos demostrar que un vector se puede omitir en una envoltura lineal, ya que si podemos definir un vector como combinacion de los otros, este operar con este seria sustituir un valor ya obtenido.

Se estudia si es LD o LI usando un sistema lineal homoganeo.

Las bases ya se han comentado por encima que son, vienen denotadas por letra caligrafica y consiste en una envoltura lineal con conjunto de vectores estrictamente linealmente independientes entre si. En este caso, las dimensiones del espacio delimitaran el numero de vectores que lo definen. Una base no tiene que ser la unica que defina al conjunto generado, su demostracion seria tan sencilla como usar los multiplos de una base que sabemos que es cierta.

5.2. Matrices y espacios vectoriales

Solo por aclarar, aunque se puede observar con las explicaciones anteriores, las matrices se pueden entender como reordenaciones de los vectores de tal forma que, por lo general, las columnas son vectores y las filas son los componentes cartesianos en el mismo eje o dimension. Sabiendo esto, se pueden estudiar las relaciones entre los vectores usando un sistema matricial y comprobando si son SCD, SCI o SLI.

Un resultado curioso es, si dada una matriz de vectores, la reducimos para obtener 1 principales, si nos da una matriz identidad, ese conjunto de vectores sabremos que es una base, si se da que es SCI, sera una envoltura lineal con un vector de mas y si es SLI pues caca, como si quieres decir que es lo de Nvidia, en este caso dices que son LD y sudas.

Para que no te confundas a la hora de reordenar los vectores, descomponelos a los componentes cartesianos, cada linea debe de agrupar en un sentido u otro a los componentes del mismo eje.

6. Valores y vectores propios de una matriz, m1

Oh god dammit, alla vamos con el que si que puede ser un poco mas tedioso que la profesora de fc (mah bois pillaron la referencia).

Dada una matriz A de orden n , un vector \vec{v} en \mathbb{R}^n y un escalar (real o complejo) λ , si se cumple la igualdad $A\vec{v} = \lambda\vec{v} \Leftrightarrow A\vec{v} - \lambda\vec{v} = 0 \Leftrightarrow A\vec{v} - \lambda I\vec{v} = 0$ (esta ultima igualdad se obtiene por propiedades), diremos que λ es un valor propio o autovalor y \vec{v} es el autovector o vector propio asociado a este. Un autovector pero un autovalor puede tener muchos autovectores asociados a el (un autovector y los multiplos de este estan asociados al mismo valor propio).

Los autovalores se pueden obtener con la ultima igualdad, que se puede entender como un sistema matricial tal que $A\vec{v} - \lambda I\vec{v} = (A - \lambda I)\vec{v} = 0$ que sera un sistema homogeneo, si hacemos determinante de $(A - \lambda I)$ y lo igualamos a 0 podemos obtener los autovalores que deben de ser distintos de ceros. Esta operacion recibe el nombre de polinomio caracterisitco $q_A(\lambda) = \det(A - \lambda I) = 0$, siendo cada raiz del polinomio un autovalor.

El SH debera de ser SCI ya que, como se ha dicho, un autovalor puede estar asociado a muchos autovectores, por ejemplo sus multiplos. El conjunto de autovectores asociados a un autovalor se agrupan en el conjunto $E_A(\lambda)$, llamado autoespacio o espacio propio.

NOTA: Veamos ahora las matrices, que podemos sacar de ellas. Las matrices son bases de un espacio y los autovalores son los escalares que toman los vectores generadores para definir un vector, de ahi que un autovector se asocie a un solo autovalor pero no se cumpla el reciproco, porque en el primer caso, dado la base (la matriz) y el autovalor asociado es la unica combinacion posible para obtenerlo pero, un auto valor puede ser reusado para obtener mas vectores. Esta explicacion es de cosecha propia y creo que ayuda a entender que se hace pero como veais. Es mas, si reduces la tabla de tal manera que es una matriz diagonal, la diagonal te da los autovalores correspondientes a esta, lo que significa que en cada dimension el autovalor en dicha posicion es la que opera pra obtener el vector deseado. Es un poco abstracto pero si se pone a probarlo lo sacas. y OJO, el numero de autovalores corresponde al grado de autovalores que a su vez corresponde con las dimensiones del espacio.

En ocasiones, la ecuacion o polinomio caracteristico de A viene dado por raices, estas raices seran los posibles autovalores que pueden darse en dicha matriz y si se repite una misma raiz, entonces este autovalor tendra una multiplicidad algebraica equivalente al numero de repeticiones, viene denotado por $ma(\lambda)$. Llamaremos traza de A a la suma de todos los autovalores de esta (que casualmente coincidiria con la de las matriz escalonada reducida comentada en le NOTA anterior) y el determinante seria la multiplicacion de todas ella (CTRL+C, CTRL+V del parentesis anterior).

Dado un polinomio caracteristico, podemos calcular la inversa de la matriz asociada si sustituimos λ por A y los terminos independientes vienen dados por la matriz I (T° de Cayley-Hamilton).

Los autovectores de dos autoespacios con autovalores distintos son linealmente independientes, y la base de estos autoespacio es cualquier autovector en el (demostracion clara con los multiplos). Llamaremos multiplicidad geometrica de λ a la dimension del autoespacio asociado a λ y se denota por $mg(\lambda)$.

Es un tema que parece sencillo pero es mucha teoria comprimida con lo cual se recomienda probar uno mismo el que se hare graficamente ya que, al final, tratamos con vectores y escalares reordenados.

Alguna duda entrarme con @eduespuch (en insta igualito, para que me sigais) y se que he puesto que iba a hacer un anexo pero me da pereza y con esto es mas que suficiente. A partir de ahora, se seguirá con la estructura propuesta de estudio de modelo progresiva, es posible que hayan conceptos repetidos pero esto se debe a que todo lo anterior esta sujeto a cambio.

7. Análisis

Siendo una herramienta que se basa en la geometría y le da mayor uso. La geometría nos limita a figuras pero con el análisis podemos aplicar problemas a las funciones. Por ejemplo, dada una función $f(x) = x^2$, en geometría debemos entenderla como un parábola pero con análisis puede ser la variación del tamaño de un objeto respecto al tiempo transcurrido.

Por si no ha quedado claro, relacionaremos la representación gráfica de la geometría con la resolución de problemas mas abstractos. Veamos que pasos seguir para su correcto desarrollo.

1. Sabiendo que trabajamos con espacios en los que se definen unas propiedades (al haber relacionado el análisis con la geometría), veremos que podemos considerar cierto para ampliar la funcionalidad de los recursos ya conocidos.

Trataremos con el dominio de las funciones y diversas representaciones de estas, ademas de la función inversa y la composición de funciones.

2. Estudiaremos de forma mas exhaustiva las funciones, definiendo la continuidad y las asíntotas ademas de los limites, que pueden venir en función de una o varias variables.
3. Introduciremos los conceptos de derivada e integral y veremos de que nos pueden servir, **#SPOILER ALERT**, nos servirán para muchas cosas pero también serán muy densas.

7.1. Conceptos básicos

Con la geometría hemos visto que podemos trabajar en varias dimensiones, si entendemos estas dimensiones de una forma mas abstracta y las consideramos simplemente conjuntos numericos a los que asignar previamente un significado, podemos aplicar los conceptos mas basicos de los espacios vectoriales al analisis de las funciones.

Trabajaremos principalmente con los numeros reales, pero es posible que la funcion este definida para un subconjunto de los numeros reales y, que ademas, el resultado tambien este definido para un subconjunto de este. Llamaremos dominio al subconjunto de valores que la funcion puede tomar y lo denotaremos por D y llamaremos recorrido de la funcion al conjunto de los resultados o imagenes que ésta tomara para dicho dominio.

Puede darse el caso de que queramos encadenar funciones o que dado el recorrido de una funcion queramos estudiar el dominio de este, con la composicion de funciones y la funcion inversa esto es posible. Con la composicion usaremos como incognita de una funcion a la otra funcion. La funcion inversa consiste en considerar el recorrido al dominio y viceversa y despejar la incognita de tal formar que si tenemos $f(x) = x + 1$, $y = x + 1 \Leftrightarrow y - 1 = x$, asumiendo entonces $x - 1 = y \Leftrightarrow x - 1 = f^{-1}(x)$ con f^{-1} siendo la funcion inversa.

Veremos ahora ejemplo de funciones de una variable y la representación de estas ademas de ejemplos de estudio de los conceptos vistos.

Funciones	Ejemplo	Descripcion	Grafica
Polinomicas	$P(x) : ax^2 + bx + c$	La base de las funciones. Diremos que las funciones son de grado n si, con x^n , n es el valor mas alto en la funcion.	asd
Racionales	$R(x) : \frac{N(x)}{D(x)}$	Siendo el numerador y el denominador polinomios, afirmamos que si $D(a)=0$, a no pertenece al dominio.	asd
Irracionales	$I(x) : P^{\frac{n}{m}}(x)$	Su dominio depende del valor de $\frac{n}{m}$, con n el grado del polinomio y m la raiz. Un entero $D \simeq \mathbb{R}$, racional $D \geq 0$	asd
Logaritmicas	$L(x) : \log(P(x))$	Polinomios como argumentos de logaritmos. $D = \{x \in \mathbb{R}, P(x) > 0\}$. $\log_a b = c \Leftrightarrow a^c = b$	asd
Valor absoluto	$ P(x) :$ $P(x) \geq 0, P(x) = P(x)$ $P(x) < 0, P(x) = -P(x)$	Consiste en realizar un sistema en el que se trata de obtener una imagen positiva para todo el dominio	asd
Trigonometricas	$T(x) :$	Polinomios como argumentos de ecuaciones trigonometricas. Dominios ciclicos	asd
Inversa	$F^{-1}(x) :$	La funcion inversa de una funcion dada depende del tipo de funcion que sea. Ambas funciones son simetricas respecto a $x=y$	asd
Compuesta	$G(F(x)) = GoF(x)$	Se usa como incognita de una funcion a la otra. Si tuviesemos $f \circ f^{-1}(x)$, tendríamos la funcion identidad $id(x) = x$	asd

graficas pendientes, hacerlas simples con dibujos sin muchos detalles, solo los ejes

7.1.1. Propiedades de una funcion

Existen conceptos o propiedades asociadas a las funciones que podemos estudiar directamente de ellas o que tendremos que realizar una o varias derivadas sobre ella (veremos este concepto mas adelante), reunamos dichas propiedades segun se aplican o no derivadas sobre las funciones.

Propiedades directas:

- Dominio y recorrido: los conjuntos con los que se representan y representados de la funcion, respectivamente.
- Puntos de discontinuidad: puntos que no pertenecen al dominio. Pueden darse casos que parezcan pertenecer pero al trabajar con reales hay que considerar que puede existir un punto en el que no exista. Sera necesario estudiar los limites y probar que existen para dicho punto y que coincide con la funcion en ese mismo punto.
- Asintotas: Regiones en las que la funcion se aproxima a una recta pero no la cruza por mucho que se le aproxime. Se estudiara con limites y pendientes, bastara con saber:
 - +Vertical: dado un valor del dominio, el recorrido de la funcion se aproxima a una recta situada en dicho punto y no la corta en ningun momento.
 - +Horizontal: El recorrido para una region del dominio que tiende al infinito, positivo o negativo, practicamente se convierte en una constante. El valor de la constante sera el punto de la imagen de una recta la cual no cortara la funcion.
 - +Oblicua: dada en forma de pendiente ($y = mx + n$ con m el valor de la pendiente y n el valor inicial en el conjunto de imagenes), la funcion no corta la pendiente dibujada y se debe

de estudiar observando primero $\frac{f(x)}{x}$ cuando tiende al infinito para ver la pendiente que tiene y luego observar como de desplazada se encuentra del origen de coordenadas.

- Puntos de corte: son los puntos en los que la función corta los ejes. Hablaremos del eje de abscisas con el eje horizontal o eje X $(x,0)$ y al eje de ordenadas al vertical o eje Y $(0,y)$.
- Simetría: respecto al eje de abscisas $f(x) = f(-x)$ o respecto al eje de ordenadas $-f(x) = f(-x)$.

Propiedades mediante derivadas:

- Primera derivada $(f'(x))$:
 - + Intervalos de crecimiento $f'(x) > 0$
 - + Intervalos de decrecimiento $f'(x) < 0$
 - + Puntos críticos $f'(a) = 0$, se estudia la siguiente derivada o se prueba en la gráfica
- Segunda derivada $(f''(x))$:
 - + Intervalos de convexidad $f''(x) > 0$
 - + Intervalos de concavidad $f''(x) < 0$
 - + Puntos inflexión $f''(a) = 0$, se estudia la siguiente derivada o se prueba en la gráfica
- Se hará mayor hincapié más adelante.

7.1.2. Límites, continuidad y asíntotas

Antes de comenzar con las explicaciones, recapitulemos brevemente la información más importante.

1. Podemos trabajar con funciones de varias variables igual que con las de una, todo funciona igual salvo las funciones inversas, desconozco que les pasa con varias variables.
2. Si entendemos que las funciones pueden trabajarse como en los espacios vectoriales, podemos entender entonces que, dado $f(x_1, x_2, \dots, x_n) = y \Leftrightarrow (x_1, x_2, \dots, x_n, y) \in \mathbb{R}^{n+1}$ siendo este un vector (o $n+1$ -tupla, es decir, una ordenación de $n+1$ elementos) que definirá una figura en un espacio de dimensiones $n+1$.
3. Existe una simbología estricta y variada para definir conceptos únicos, nunca confundir el significado de estos.
4. Las funciones pueden escribirse en varias formas, como las ecuaciones de la recta (implícita, explícita, paramétrica, ...), aunque la más normal será la implícita. Habrá que tener en cuenta cuando una variable es dependiente a otra, en tal caso hablaremos de explícita. No olvidarse que $f(x)$ no es una variable, y aunque se defina en ocasiones como una incógnita, realmente sería más una variable dependiente de ahí que se sustituya por su equivalente con las variables independientes.
5. El valor infinito, ya sea positivo o negativo, debe conocerse. Y se dará por conocido.
6. Llamaremos curvas de nivel a $(x_1, x_2, k) \in \mathbb{R}^k$ donde k será una constante perteneciente a la imagen de la función tal que $f(x_1, x_2) = k, (x_1, x_2) \in \mathbb{R}^2$.

Los límites de una función, ya sea de una o más variables, es el estudio del valor que tomara la función para dicho punto dado por la(s) variable(s). Para ello, estudiaremos la función cuando tiende al punto en cuestión probando el valor de la función para este punto, para valores muy por encima y por debajo y comprobando si los resultados tienen sentido. Se denotará por lo general con $\lim_{\vec{x} \rightarrow X_0} f(\vec{x}) = L$ considerando $\vec{x} = (x_1, \dots, x_n)$ y $X_0 = (a_1, \dots, a_n)$. Para varias variables, L será el límite 'doble' (consideramos 2 variables por ahora).

Siempre existirá un valor positivo tan próximo a 0 como queramos que sea mayor a la distancia de cualquier punto perteneciente al dominio respecto al límite. (dado $\epsilon > 0$ y $\delta > 0$, ampliar bien)

Cuando tratamos con varias variables, llamaremos límites iterados a los límites resultados de estudiar el límite por pasos. Dado $f(x,y)$, en (a,b) obtenemos $\lim_{x \rightarrow a} f(x,y) = l(y)$ y por otro lado $\lim_{y \rightarrow b} f(x,y) = l(x)$ para definir finalmente a $l_1 = \lim_{y \rightarrow b} l(y)$ y $l_2 = \lim_{x \rightarrow a} l(x)$, siendo estos los límites iterados en el caso de existir el límite doble L y $l(y)$, entonces existe l_1 que además se tiene que vale lo mismo que el límite doble. Como consecuencia tenemos:

1. Si existen los límites iterados l_1 y l_2 y estos son distintos para el mismo punto, entonces podemos decir la función lo tiene límite doble para dicho punto.
 2. Puede darse que existen los límites iterados y sean iguales pero que no exista el límite doble.
 3. Puede que exista el límite doble pero no exista alguno de los límites iterados, dependerá de la existencia de $l(y)$ o $l(x)$.
- + Aunque en los apuntes se diga que los límites iterados (o parciales) y los reiterados son distintos, realmente no lo son. Por lo que he podido ver se diferencian como mucho en la forma de expresarse pero ambos se obtienen de la misma forma. Como mucho puede ser que la diferencia se encuentre a la hora de entender el primer límite en función de una variable, pero no se.

Los límites iterados son de ayuda para estudiar la existencia del límite doble y su posible valor, pero para ello también deberemos estudiar los límites unidimensionales.

Pueden darse casos en los que se estudia el límite para un punto que tiende al infinito y este no tiende a un valor en concreto, en estos casos hablaremos de indeterminaciones y suele resolverse con el método de l'hôpital, que se verá más adelante junto a las derivadas:

1. $\frac{\infty}{\infty} \Leftrightarrow \frac{N^n(x)}{D^m(x)} = f(x)$, o por l'hôpital o comprobando el valor de $\frac{n}{m}$
2. $\infty - \infty \Leftrightarrow P(x) - Q(x) = f(x)$, suele resolverse usándose el conjugado ya la propiedad $1 = \frac{P(x) + Q(x)}{P(x) + Q(x)}$, y que $a \cdot 1 = a$.
3. $1^\infty \Leftrightarrow P(x)^{Q(x)} = f(x)$ se resolvería considerando $\lim_{x \rightarrow b} P(x)^{Q(x)} = \exp^{\lim_{x \rightarrow b} Q(x) \cdot (P(x)-1)}$
4. Existen más pero estos se resuelven con l'hôpital y se verán con su estudio.

Con los límites podemos definir también infinitesimos y los infinitesimos equivalentes

Sabiendo límites, estudiar asíntotas y la continuidad de una función debería de ser sencillo. Con la continuidad habrá que probar que dado un dominio, cada punto de este tenga una imagen y el límite en dicho punto sea igual a la imagen, además de comprobar que el límite por encima y por debajo en dicho punto existe y son iguales al límite en sí. Si no se cumple, veremos que se dan discontinuidades de 3 tipos:

1. Evitable: la función en un punto dado o no existe o, si existe, el límite en este no coincide con sus límites laterales. Se podría evitar al arreglar la función en dicho punto.

2. De salto finito: la función en un punto dado existe o no, pero sus límites laterales son distintos (funciones escalonadas, suele darse al trabajar con valores reales pero resultados enteros).
3. De salto infinito: la función no existe para el punto, se estudian los límites laterales y para el punto en cuestión y se compara, suele darse en casos de asíntotas.

Veamos algunos corolarios obtenidos de demostrar la continuidad de una función:

1. T^a de Bolzano (o de las raíces): dado un intervalo cerrado $[a, b]$ en el que la función es continua y en el que el producto de las imágenes en los extremos del intervalo da negativo, podemos afirmar que existe un elemento perteneciente al intervalo cuya imagen es 0. Es fácil de ver con funciones trigonométricas. Puede aplicarse a un valor intermedio si en lugar de 0 es un número cualquiera y hacemos que $f(x) = g(x) + k$.
2. T^a de Weierstrass (o del máximo-mínimo): teorema estúpido que dice que en un intervalo cerrado y continua, se afirma que existe cota superior e inferior que además son máximo y mínimo (no en respecto a curvatura, si no a extremos del intervalo cuyas imágenes pertenecen al recorrido). Es un teorema cuya complicación es saber de qué habla con lo de máximos y mínimos xd.
3. Lema de conservación del signo: dada una función continua en un intervalo cerrado, si tomamos un punto cuya imagen sea distinta de 0 podemos afirmar que los límites laterales para este punto serán del mismo signo que la imagen dicha.
4. El teorema de Rolle y el de Lagrange aunque sea importante conocer que son continuas, es más importante conocer que son derivables, por lo tanto veamos el concepto de derivada.

7.2. Derivadas

Para saber que son las derivadas, debemos conocer la Tasa de Variación Media (TVM) y la Tasa de Variación Inmediata (TVI).

Con la TVM estudiamos la variación de una función en un intervalo, para ello se estudia la pendiente entre los extremos del intervalo de tal manera que

$$TVM = \frac{\Delta f(x)}{\Delta x} = \frac{f(b) - f(a)}{b - a}$$

en el intervalo $[a, b]$.

Con la TVI estudiamos la variación en un punto en concreto, o lo que es lo mismo, en un intervalo tan pequeño que podamos considerar la diferencia entre los dos extremos 0. Para ello, el intervalo usado es $[x, x+h]$ y obtenemos la TVI de la siguiente manera:

$$TVI = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{x+h-x}$$

con x siendo un valor del dominio. Su representación sería una pendiente tangente a $f(x)$ para un x dado.

Si decidiésemos estudiar la TVI no para un punto, si no para el dominio, entenderemos a esta como una derivada y la forma de obtener la TVI se entendería como definición de la derivada y se denotaría por $f'(x)$ o $\frac{d}{dx}f(x)$. Cabe decir que aunque a primera vista, el denominador en el límite de 0, al avanzar el desarrollo existirá algún método para despejar, por lo tanto, la tabla de las derivadas no será estrictamente necesaria de aprenderse, pero por si acaso:

Tipo	Funcion	Derivada	Tipo	Funcion	Derivada
Constante	$f(x) = k$	$f^1(x) = 0$	Logaritmica	$\ln u$ o $\log_a u$	$\frac{u'}{u}$ o $\frac{u'}{u \cdot \ln a}$
Identidad	$f(x) = x$	$f^1(x) = 1$	Seno	senu	$u' \cos u$
Potencial	u^a	$a \cdot u^{a-1} \cdot u'$	Coseno	$\cos u$	$-u' \text{senu}$
Irracional	$\sqrt[b]{u} = u^{\frac{1}{b}}$ $\frac{1}{b} = a$	$u' \cdot a \cdot u^{a-1} = u' \frac{1}{b \sqrt[b]{u^{b-1}}}$	Tangente	tgu	$\frac{u'}{\cos^2 u} = (1 + \text{tg}^2 u) u'$
Exponencial	e^u o a^u	$u' e^u$ o $u' \ln a \cdot a^u$	Arcoseno	\arcsenu	$\frac{u'}{\sqrt{1-u^2}}$
Potencial exponencial	En desuso f^g	$g f^{g-1} + f^g g' \ln f$ Se usan prop. log.	Arcocoseno	$\arccos u$	$-\frac{u'}{\sqrt{1-u^2}}$
Asociativa(?)	$f \pm g$	$f' \pm g'$	Arcotangente	$\arctg u$	$\frac{u'}{1+u^2}$
Producto	$f \cdot g$	$f'g + g'f$	Cadena	$g \circ f = g(f)$	$g'(f) \cdot f'$
Tipo	Cociente	Funcion	$\frac{f}{g}$	Derivada	$\frac{f'g - g'f}{g^2}$

Con k una constante real, x una variable y u una funcion ya sea polinomica, racional, ... dada de cualquier forma, parametrica, implicita, explicita, ... Esto nos genera un problema.

Por ejemplo, si calculamos la derivada para una funcion $f(x)$ explicita seria sencillo, derivamos esta en funcion de x y obtenemos $f'(x)$, pero si $f(x)$ viene dada en implicita, se pueden hacer dos casos:

1. La conviertes en explicita dejando la funcion dependiendo de una variable.
2. Derivas ambos lados de la igualdad, recordando que una incognita esta en funcion de la variable.

Por ejemplo, dada $5y + 3x = 1$, o la pasamos a explicita $y = \frac{1-3x}{5}$ o derivamos directamente, veamos esta:

$$\frac{d}{dx}[5y + 3x] = \frac{d}{dx}1 \Leftrightarrow \frac{d}{dx}[5y] + \frac{d}{dx}[3x] = 0 \text{ donde } 5\frac{dy}{dx} + 3 = 0 \Leftrightarrow \frac{dy}{dx} = -\frac{3}{5}, \text{ donde } \frac{dy}{dx} = f'(x) \text{ o } y'$$

Las derivadas de orden superior o de n -esimo orden es simplemente derivar la derivada n veces. Definamos algunos teoremas antes de comenzar a estudiar las aplicaciones de las derivadas:

- T^a de Rolle: comprueba que existe al menos un punto donde la TVI es nula (derivada igualada a 0). Graficaméne seria una recta tangente paralela al eje x .
- T^a de Lagrange o valor medio: existe al menos un punto donde la recta tangente es paralela a la recta secante que cruza los puntos correspondientes a los extremos del intervalo. Tan sencillo como hacer TVM y el resultado tendra el mismo valor que la TVI del punto en concreto con el que se cumple.

- T^a de Cauchy: no lo se, creo que lo tengo mal, revisar
- Regla de l'hôpital: aunque sea una aplicacion mejor darla ya. Dada indeterminaciones del tipo $\frac{a}{b}$ o $a \cdot b = \frac{a}{1/b}$ con $a, b \in \{-\infty, 0, \infty\}$ si derivamos numerador y denominador entendiendolos como funciones independientes, el resultado es el limite de la funcion.

Es mas, las indeterminaciones $1^\infty, \infty^0, 0^0$ se pueden reorientar para despejarse tambien de tal manera que

$$A = \lim_{x \rightarrow u} [f(x)^{g(x)}] \Leftrightarrow \ln(A) = \ln(\lim_{x \rightarrow u} [f(x)^{g(x)}])$$

$$\ln(A) = (\lim_{x \rightarrow u} g(x) \ln[f(x)]) = 0 \cdot \infty = M \Leftrightarrow \ln A = M \Leftrightarrow A = e^M$$

- Polinomios de Taylor: utilizado para estudiar el comportamiento de una funcion a las proximidades de un punto dado, sustituyendo la funcion dada por una mas sencilla usando potencias de x-a (siendo a el punto dado y x una variable).

$$f(x) = P_n(x) + R_n(x, \epsilon) = \sum_{k=0}^n \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k + \frac{f^{(n+1)}(\epsilon)(x_0)}{(n+1)!} (x - x_0)^{n+1}$$

OJO: revisar formula del error de taylor, posiblemente mal expresada. P_n es el polinomio de Taylor de orden n y R_n es el resto de Taylor o error, que tendra a 0 a mas cerca este x del x_0 dado conforme n tiende a infinito. Si se da una serie infinita, el resto sera considerado 0 y el valor de la funcion en x con la ahora llamada serie de Taylor.

- Polinomio de McLaurin: es como Taylor pero el punto dado siempre sera 0.

7.2.1. Aplicaciones para una variable

Optimizacion basicamente, sacar maximos y minimos posibles entre varias funciones al relacionarlas y dejandolas en funcion de una variable

7.2.2. Derivadas con varias variables

Me da pereza asique super resumen:

Puedes derivar parcialmente o hacer el gradiente que es la suma de las derivadas parciales con cada variable. La matriz jacobiana son funciones por filas y derivadas parciales respecto a una variable en concreto por columnas y la matriz Hessiana es para una misma funcion y en cada posicion va asignada la derivada segunda interpretando las variables como coordenadas en la matriz, las matrices en posiciones invertidas deben de ser iguales para decir que es continua y estudiaremos si para un punto es minimo maximo o punto de ensillamiento al hacer el determinante de la matriz y sustituyendo el punto en las incognitas, recordad la funcion $x^2 + y^2$ es de ayuda para saber cuando sera minimo o maximo.

7.2.3. Aplicaciones para varias variables

Con el gradiente la mayor variacion para un punto, y optimizacion por lo normal.

7.3. Integrales

Con las integrales lo que realizamos son sumas de pequeños cuadrados comprendidos entre dos lineas (por lo general la funcion y el eje 0 pero esto puede modificarse) en la que la base consiste en la diferencia entre dos puntos muy proximos y la altura es la funcion para dicho punto. Pero como la

altura puede ser mayor o menor segun que valor escojamos de los dos puntos, se define el concepto de suma superior y suma inferior:

1. Suma superior: dado n puntos en un intervalo $[a, b]$, denotamos por M_k a la imagen de mayor valor en un intervalo (en los apuntes se denota con el supremo aunque realmente deberia ser con el maximo, el supremo es para cotas y esta mas relacionado con ordenacion secuencial que con el valor... depende tambien de la interpretacion). Si realizamos subintervalos de tal forma que todos los puntos son equidistantes y en el cual sacamos un M_k para cada subintervalo, si hacemos el sumatorio de todos estos valores por el incremento (o variacion media entre dos puntos) tenemos la llamada suma superior, dibujar rectangulos tan pequeños como subintervalos generemos cuya altura sera el punto mas alto de la funcion para el subintervalo al que pertenece. Seria tal que:

$$U_n(f) = \sum_{k=1}^n M_k \Delta x$$

2. Suma inferior: denotando a m_k a la imagen de menor valor(lo mismo que antes, se le llama infimo y segun como interpretemos lo que hacen podria valer pero al final hablamos del minimo valor), tenemos que en la suma inferior hacemos el sumatorio de todas las areas rectangulares definidas por un determinado numero de subintervalos de mismo rango en el cual la altura de estas areas no cruzara la linea que dibuja la funcion. Seria tal que:

$$L_n(f) = \sum_{k=1}^n m_k \Delta x$$

para ambos casos consideramos $\Delta x = \frac{b-a}{n}$

La particularizacion de Riemann, o Darboux, afirma que si tenemos n proximo a infinito, Δx es proximo a 0 y que el valor de la suma inferior (o su limite) es practicamente el mismo al de la suma superior, la funcion es integrable para el intervalo definido y el limite se llama integral definida, denotada por $\int_a^b f(x)dx$, el area descrita debajo de la curva en ese intervalo tendra que valer lo mismo que dicha. En el caso de que la funcion tome imagenes positivas y negativas debemos de saber que restara al area comprendida entre las imagenes positivas y el eje x el area comprendida entre eje x y las imagenes negativas. Si quisiéramos el sumatorio total de las areas, bastaria con considerar el valor absoluto para el subintervalo en el cual la funcion es negativa.

8. Resolucion por medios numericos

Por ahora, hemos vistos la resolucion de ecuaciones por medios analiticos, es decir, estudiabamos la funcion para obtener el resultado exacto. Veamos ahora un metodo que se basa en la aproximacion al resultado real, existiendo siempre un margen de error tan pequeño como nosotros le dejemos.

8.1. Teoria de errores

Como ya se ha dicho, siempre existe un margen de error y, por lo tanto, debemos de ser capaces de obtener y/o comprender este error, los formatos en los que se puede dar y el que hacer con el.

Entenderemos como valor absoluto, denotado por Δ (delta), a la resta del valor exacto, A , menos la aproximacion a este obtenida a , tal que $\Delta = |A - a|$. En algunos casos el valor exacto es un irracional, es decir, de infinitos decimales y el obtener un error exacto sera imposible, en ese caso se hacen aproximaciones tambien al error, indicando la 'casi igualdad' o aproximacion con $\Delta \approx |A - a|$. Recordaremos que el valor absoluto se puede obtener muchas veces por metodos analiticos de resolucion

y el valor aproximado por metodos numericos, o una simple aproximacion, quitando en ambos casos que se nos de directamente.

El error lo interpretaremos graficamente como zonas proximas al punto exacto que nos interesa, para ser mas concretos, todos los errores deben de estar dentro de un intervalo que se define con lo que llamamos cota (revisad tema 1, introduccion a lo basico, estan explicados cotas y extremos). La cota la denotamos con Δ_a y el intervalo con $a \pm \Delta_a$, tal que se cumple $a - \Delta_a \leq A \leq a + \Delta_a \leftrightarrow |A - a| = \Delta \leq \Delta_a$. La pertenencia de A a este intervalo de puede indicar de varias formas, $A \in [a \pm \Delta_a]$ o $A \approx a \pm \Delta_a$, siendo esta ultima mas indicando la aproximacion del valor exacto al indicado por una aproximacion. Cuanto menor sea el valor de la cota, mas aproximado estara a de A . Veremos mas adelante que al hacer una aproximacion desde cero, se nos da un parametro de tolerancia, es decir, nos indica como de tolerantes debemos de ser con la cota. Lo veremos mejor mas adelante.

El error relativo, denotado con δ (delta minuscula), es comprobar el error absoluto como de grande es respecto al valor exacto del cual depende. No es lo mismo un error de 1 con valor exacto de 10 que el mismo error con valor exacto de 1000, del primero hablamos de un 10 % de error y el segundo es un 0,1 %. La formula para obtenerlo seria $\delta = \frac{\Delta}{|A|} [x100]$ para un error dado o $\delta_a = \frac{\Delta_a}{|A|} [x100]$ si tratamos con la cota, pudiendo o no hacerlo porcentaje si hacemos $[x100]$. Cabe decir que, al ser las dos denotaciones de la letra delta, alomejor puede ser confuso a la hora de tratar con ellas, personalmente trato Δ con E u otra forma o la llamo 'incremento' ya que suele usarse para ello. Tambien, para los menos avisados, recordad que las igualdades pueden modificarse sin cambiar su veracidad, es decir, $\delta = \frac{\Delta}{|A|} \Leftrightarrow \delta x |A| = \Delta$, tambien aplicable con las cotas. Mientras se sepa como hacer sustituciones entre igualdades, se podran obtener muchas definiciones de todo lo ya dicho.

Por lo general, el valor exacto no se sabra, no se nos permitira obtenerlo por medios analiticos, con lo cual debemos aprovechar los elementos disponibles que por lo general sera, el error absoluto Δ o la cota de este Δ_a o el relativo δ o la cota de este δ_a y lo mas seguro es que el valor aproximado o una forma de obtenerlo a . Si ponemos todas las definiciones anteriores a prueba, esto es facil de obtener, veamos:

Definido con A	Definido sin A
$\Delta_a = A - a $	$\Delta_a = a \delta_a$
$a - \Delta_a \leq A \leq a + \Delta_a$	$a - a \delta_a \leq A \leq a + a \delta_a$
$A \approx a \pm \Delta_a$	$A \approx a \pm \delta_a (\%) \Leftrightarrow A = a(1 \pm \delta_a)$

Pero, para poder definirlo sin A , es necesario demostrar que la primera igualdad se cumple. La demostracion conlleva el suponer A y a positivos o negativos y utilizar las desigualdades. Tenedlo en cuenta pero dudo que se pida demostrarlo.

8.1.1. Dígitos significativos y dígitos exactos

Si descomponemos el valor exacto y el aproximado en sus digitos usando una sucesion finita o infinita de sumandos, podemos estudiar como de acertados estamos con la aproximacion usando estos digitos. Por ejemplo,

$$A = \alpha_m 10^m + \alpha_{m-1} 10^{m-1} + \dots + \alpha_{m-(n-1)} 10^{m-(n-1)}$$

$$a = \beta_m 10^m + \beta_{m-1} 10^{m-1} + \dots + \beta_{m-(n-1)} 10^{m-(n-1)}$$

Con $\alpha_i, \beta_i \in \{0, 1, \dots, 9\}, m, i \in \mathbb{Z}, \alpha_m, \beta_m \neq 0$

α_0, β_0 corresponde a la unidad y α_m, β_m al entero mas grande con m la potencia de 10 mas grande

Llamaremos digitos significativos a cualquier digito no nulo salvo que lo sean sus extremos, principalmente. Por definicion, $\alpha_m \neq 0$, con lo cual podemos omitir los zeros a la izquierda 'eres ams inutil que un zero a la izquierda' es un claro ejemplo. Por lo contrario, los ceros posteriores a un digito significativo se consideraran significativos si interesan, esto depende de como se quiere interpretar (por lo

general solo importaran en forma numerica, en forma decimal por definicion, si multiplicas un numero por 0 dara 0 con lo cual no interesa ese espacio perdido).

Por otro lado, el maximo numero, n , de digitos significativos de una aproximacion en forma decimal son digitos exactos si se cumple que $\Delta = |A - a| \leq (1/2)10^{m-n+1}$, afirmando entonces que a tiene los n primeros digitos exactos.

OJO, lo que estamos haciendo es practicamente una cota, recordad que $\Delta \leq \Delta_a$, pues ahora estamos probando con $(1/2)10^{m-n+1}$ si podria ser cota en comparacion con el error absoluto. Si para un numero n de digitos exactos dado no puede ser cota, es decir, $(1/2)10^{m-n+1} < \Delta$, entonces afirmamos que tendra $n-1$ digitos exactos y, ademas, podriamos decir que $(1/2)10^{m-(n-1)+1}$ es cota.

8.1.2. Relacion entre error relativo y digitos exactos

Dado un valor aproximado positivo del cual conocemos sus digitos exactos, n , sus error relativo cumple que $\delta \leq \frac{1}{\beta_m} \left(\frac{1}{10} \right)^{n-1}$, con β_m el primer digito significativo. La demostracion consiste en utilizar las definiciones vistas hasta ahora y aprovechar las desigualdades para llegar a una forma de definir A de forma aproximada por el primer digito significativo del valor aproximado y tener el error absoluto definido mediante potencias. No creo que pidan demostrarlo pero si usarlo con la demostracion considerada cierta.

8.1.3. Error de redondeo

Algo que hacemos de normal para decir que hemos llegado al 5 tambien tiene una teoria detras, enhorabuena, podeis consideraros catedraticos. Por un lado, definimos el numero redondeado, b , obtenido de otro (aproximacion o exacto) que por el bien de no liar lo denotaremos con x (ya que no sabemos si es o aproximado o exacto, pero sera uno de esos dos). El error de redondeo consistiria en tomar x como el valor exacto y b el valor aproximado y obtener el error absoluto de este, pero lo denotaremos de otra manera:

$$\varepsilon = |b - x|, \varepsilon \text{ es epsilon}$$

Como lo hemos definido como un error absoluto, todo lo visto hasta ahora se le puede aplicar, pero veamos como obtener el valor redondeado haciendo minimo su error:

1. Seleccionamos los n primeros digitos significativos, siendo n un numero natural dado, tal que

$$b = \beta_{m-0}10^{m-0} + \dots + \beta_{m-(n-1)}10^{m-(n-1)}, \beta_m \neq 0$$

2. Nos fijamos en β_{m-n} y segun su valor realizamos una de las siguientes acciones:

- $\beta_{m-n} < 5$, entonces $\beta_{m-(n-1)}$ se mantiene.
- $\beta_{m-n} \geq 5$, entonces sera el valor de $\beta_{m-(n-1)} + 1$ que decida:
 - 1) $\beta_{m-(n-1)} + 1 > 10$, entonces el digito se actualiza.
 - 2) $\beta_{m-(n-1)} + 1 = 10$, se mantiene como estaba.

Cabe destacar que cuando redondeamos a los n primeros digitos significativos, estos se trataran como digitos exactos aunque no coincidan visualmente, RECORDAD PARA ERROR RELATIVO.

8.1.4. Operaciones con errores

Hay dos formas de hacerlo:

1. Sabiendo derivar y utilizando el gradiente ya que:

$$\Delta_{f(x,y,z)} = \frac{df(x,y,z)}{dx} \Delta_x + \frac{df(x,y,z)}{dy} \Delta_y + \frac{df(x,y,z)}{dz} \Delta_z$$

Se recomienda conocer esta formula ya que te olvidas de aplicar casos segun que operacion se vea, pero a la vez es complicada ya que en algunas ocasiones las derivadas se vuelve largas y/o complejas.

2. Mediante formulas demostrables. En este apartado vereis porque prefiero llamar a Δ 'incremento'.

Dado S como la sucesion de valores exactos y s la de valores aproximados, veremos que se puede mediante la definicion de la derivada (que es como utilizar el gradiente pero por pasos), o veremos por partes, con suma/resta y con producto/cociente.

Por definicion de la derivada las principales serian el error de una funcion tipo logaritmica natural. La idea de esto parte de que el error absoluto de su variable trata de aproximarse a 0, el error de la funcion sera igual al valor absoluto de la derivada por el error absoluto de la variable (el gradiente para una variable).

Con funciones aditivas, teniendo S y s , si las restamos para considerar el error absoluto de las sucesiones, lo que estamos haciendo es restar cada valor exacto con su respectiva aproximacion, obteniendo que el error absoluto de las sucesiones es el sumatorio de todos los errores abosultos comprendidos en la definicion.

$$\Delta_S = |S - s| \leq \Delta_s = \Delta_1 + \Delta_2 + \dots \Delta_n = |A_1 - a_1| + \dots + |A_n - a_n|$$

Para el error relativo de una sucesion de sumas, tomamos S como el valor exacto. Se hara por partes y, otra vez, tendremos un δ_S para cuando tratamos con el caso simplificado y δ_s para cuando descomponemos, siendo ademas δ_s el maximos de todos los errores relativos que se obtienen al descomponer cada error.

Para el error de producto o cocientes, aplicamos logaritmos sobre la sucesiones y de los logaritmos podemos facilmente entender que trabajamos con errores absolutos y relativos... sinceramente, esta raro asique os dire:

1. Usar el gradiente porque seran derivadas faciles
2. Si no os queda otra, las sumas entre dos datos se hacen con errores absolutos (sea suma o resta los errores se suman siempre). Es decir, dado $A_1 = a_1 \pm \Delta_1$ y $A_2 = a_2 \pm \Delta_2$, tenemos que

$$A_1 \pm A_2 = (a_1 \pm a_2) \pm (\Delta_1 + \Delta_2)$$

3. Si se nos da una multiplicacion, con los mismos valores dados anteriormente, tenemos que tener en cuenta que el error absoluto que se nos de hay que pasarlo a relativo, ya que hay que considerar que se usan logaritmos para despejar. La formula quedaria tal que:

$$A_1 \times A_2 = (a_1 \times a_2) \pm \left(\frac{\Delta_1}{|a_1|} \times 100 + \frac{\Delta_2}{|a_2|} \times 100 \right)$$

ya que se trabaja mejor en porcentaje.

4. Si trabajas con una función logarítmica (y ojo porque de esto se saca la igualdad para los productos) el error de un logaritmo es igual al error de la variable de la que depende por su derivada, es decir,

$$\Delta_{f(x)} \approx \Delta_x |f'(x)| \Leftrightarrow \Delta_{\log(x)} \approx \frac{\Delta_x}{|x|} = \delta_x$$

5. Si trabajas con una raíz, siendo una función, mientras recuerdes que la derivada de \sqrt{x} es $\frac{1}{2\sqrt{x}}$, aplicas la definición de la derivada y ya estaría.
6. Una función compleja se puede descomponer en funciones sencillas.

8.2. Resolución de ecuaciones, algoritmos

Resolver ecuaciones mediante aproximación requiere de no solo entender que el valor no será siempre exacto, si no de cual algoritmo será el más práctico y rápido para realizar las aproximaciones. Los que veremos son todos de naturaleza iterativa, es decir, a partir de una aproximación inicial, se construye una secuencia de iteraciones usando un algoritmo determinado esperando a que converja a una raíz. Como hablamos de iteraciones, será importante conocer los criterios de parada y la convergencia, nosotros tendremos en cuenta los siguientes:

1. $\Delta_{f(x)} < \varepsilon$, el error de la función es menor que una cota dada. Por lo general, el valor exacto para la función se forzará a ser 0, siendo $\Delta_{f(x)} = f(x)$.
2. $i \leq n$, es decir, no podemos superar el número límite de iteraciones
3. $\Delta_{c_i} < \Delta_c$, es decir, el error de la raíz que se estudia debe de ser menor que una cota dada.

8.2.1. Teorema de conservación del signo

Si se da que $f(x)$ es continua en un punto dado, a (el profesor dijo que en un intervalo pero realmente basta que lo sea en un punto, el que vayamos a estudiar), y que la imagen correspondiente a este punto sea no nula $f(a) \neq 0$, entonces podemos definir un entorno $a \pm \delta$ con $\delta > 0$ donde la imagen de todos los elementos comprendidos en este entorno serán del mismo signo que la imagen de a . Por definición, al ser a continua, sus límites laterales deben de ser iguales con lo cual si se cumple para un $\delta \mapsto 0$ podemos probar para δ de mayor valor mientras se cumpla.

8.2.2. Teorema de Bolzano

Si se da que $f(x)$ es continua en un intervalo cerrado, $[a, b]$, y el producto de las imágenes de los extremos de negativo, $f(a)f(b) < 0$, entonces existe al menos un elemento del intervalo cuya imagen es nula, $c \in (a, b)$, $f(c) = 0$.

Se demuestra a partir de aplicar la definición de forma iterativa, con los criterios de parada ya dichos, tal que si la imagen de c es nula nos detenemos y si no lo es comprobamos si sustituyendo c por uno de los extremos de los intervalos se cumple la condición inicial. Este es un método cerrado, se van tomando valores comprendidos en un intervalo con lo cual nos aseguramos que el resultado aproximado converge al exacto.

Para aplicarlo de forma iterada, haremos varias cosas pero nos centraremos en aquellas particulares para este caso y más adelante agruparemos una explicación general para todos los casos. El valor de c para cada iteración, se calculará a partir de la media, es decir, $c_i = \frac{(a_i + b_i)}{2}$, al actualizarse el intervalo ($a_i = c_{i-1}$ o $b_i = c_{i-1}$, según el signo de la imagen), el tamaño del intervalo con el que trabajamos disminuye, siendo $|I_i| = |b_i - a_i| = |I_{i-1}|/2 = |b_1 - a_1|/2^{i-1}$, de tal forma que cuando hayamos realizado infinitas iteraciones, el tamaño del intervalo debe de tender a 0.

El pseudocódigo del algoritmo, para que nos hagamos una idea, tiene la siguiente forma:

```

BúsquedaPorBisección (f(x), a, b, ε, Δ, n)
i:=0
h:=abs(b-a)
repetir
    i:=i+1
    c:=(a+b)/2
    h:=h/2
    si signo(f(a))*signo(f(c))<0
        entonces
            b:=c
        si no
            a:=c
hasta (abs(f(c)) ≤ ε) ó (h ≤ Δ) ó (i = n)
devolver c

```

8.2.3. Metodo del punto fijo

Siendo esta la mas estúpida que se me ocurriría. Esta mal escrita matematicamente, dice que dada $f(x) = 0$, dando a entender que la funcion es una constante. Lo que trata de decir es que dada $f(x)$, si despejamos una raíz para que de la imagen nula, podemos redefinir $f(x) = 0$ tal que $g(x) = x$, afirmando pues que la raíz para la primera funcion es raíz para la segunda. La raíz obtenida se llamara punto fijo.

Para que se cumpla se tiene que dado $f(x)$ continua para un intervalo $[a,b]$ y siendo $g(x)$ equivalente a $f(x)$ y $g(x) \in [a,b], \forall x \in [a,b]$, entonces g tiene al menos un punto fijo en el intervalo. Si ademas la derivada de g es menor que 1, en particular menor o igual que una cosntante dada, diremos que el punto fijo es unico para el intervalo.

La forma para redefinir $f(x)$ se puede tratar de despejar una x o de sumar a ambos lados de la igualdad x . Veamos el pseudocódigo para entender como realizar la iteracion:

```

BúsquedaPuntoFijo (g(x), x0, ε, Δ, n)
i:=0
h := Δ+1
repetir
    x:= evaluar (g(x) en x0)
    h := abs(x-x0)
    i:= i+1
    x0:= x
hasta (abs(g(x)) ≤ ε) ó (h ≤ Δ) ó (i = n)

```

8.2.4. Metodo de la secante

La idea detras de este metodo es que cuanto mas cerca esten dos valores del intervalo, mas se aproximara la recta secante que atraviesa las imagenes de estos valores a la funcion que las define. Este es un metodo abierto, ya que nosotros no le damos un intervalo inicial, le damos dos valores de un intervalo para comenzar y es posible que el punto de estudio supere a estos.

$$\frac{x - a_i}{b_i - a_i} = \frac{y - f(a_i)}{f(b_i) - f(a_i)} \text{ x sera c e y sera 0, entonces } \frac{c_i - a_i}{b_i - a_i} = \frac{0 - f(a_i)}{f(b_i) - f(a_i)}$$

$$c_i = a_i + \frac{(0 - f(a_i))(b_i - a_i)}{f(b_i) - f(a_i)} \Leftrightarrow c_i = a_i + h_i$$

Veremos el pseudocódigo y mas adelante el como usarlo:

```

BúsquedaPorSecante (f(x), a, b, ε, Δ, n)
  i:=0
  repetir
    i:=i+1
    si abs(f(a))>abs(f(b)) entonces
      (* Intercambiar 'a' por 'b' *)
      a↔b
      h:=f(a)*(b-a)/(f(b)-f(a))
      c:=a-h
      b:=c
  hasta (abs(f(c))≤ε) ó (abs(h)≤Δ) ó (i=n)
  devolver c

```

Este metodo no nos asegura la convergencia del resultado aproximado al exacto, pero si aplicamos una condicion mas, el metodo de la secante se convierte en el de regula falsi.

8.2.5. Metodo de regula false

Si combinamos el de la secante y exigimos la premisa de Bolzano, tal que $f(a)\Delta f(b) < 0$, podemos afirmar que el nuevo valor de estudio pertenecera al intervalo de estudio y, por lo tanto, debera de converger el resultado final.

```

BúsquedaRegulaFalsi (f(x),a,b,ε,Δ,n)
  i:=0
  repetir
    i:=i+1
    si abs(f(a))>abs(f(b)) entonces
      a↔b
      h:=f(a)*(b-a)/(f(b)-f(a))
      c:=a-h
      si signo(f(a))*signo(f(c))<0
        entonces b:=c
      si no a:=c
  hasta (abs(f(c))≤ε) ó (abs(h)≤Δ) ó (i=n)
  devolver c

```

8.2.6. Metodo de newton

Similar a la secante, este utiliza la recta tangente a partir de la derivada de la funcion. En este caso, nos hace falta unicamente un valor inicial. Su pseudocódigo seria:

```

BúsquedaPorNewton (f(x),a,ε,Δ,n)
  f'(x)::=df(x)/dx
  i:=0
  repetir
    i:=i+1
    h:=f(a)/f'(a)
    c:=a-h
    a:=c
  hasta (abs(f(c))≤ε) ó (abs(h)≤Δ) ó (i=n)
  devolver c

```

En el caso de asegurarno la convergencia, aplicamos la premisa de Bolzano... pero bueno, mucha pereza.

8.2.7. Ejemplos de implementacion

La mejor forma para implementar todos los metodos, es realizando un tabla en las que ponemos nuestros criterios de salida indicados siendo n el maximo numero de iteraciones, Δ sera la cota para los valores del intervalo y ε la cota para la imagen. En todos los pseudocodigos hemos definido un valor h , en cada metodo de obtiene de distintas formas, esta h es el limite de tolerancia, actua como un error absoluto solo que no se da en valor absoluto y este, en absoluto, debe de esta por debajo del error absoluto para los valores del intervalo. Por otro lado, el error absoluto para las imagenes depende de la imagen para el punto estudiado y la imagen destino, que en estos casos suele ser 0, con lo cual se toma como error absoluto (pero con signo) la imagen del punto de interes. Para afirmar que converge debe de ser menor que la cota dada.

La forma mas comoda para manejar estos valores es mediante la siguiente tabla:

	$=n$		$\Delta=$		$\varepsilon=$		
i	a	b	c	h	f(a)	f(b)	f(c)

9. Estadística y probabilidad

A nadie le interesa esta verga, ya la haré cuando tenga tiempo.