1.5em 0pt

DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

El Gran RTP - Sports Analytics

Sistemas de Ecuaciones, Eliminación Gaussiana, Factorización Cholescky

Métodos Numéricos

Primer Cuatrimestre de 2017

Integrante	LU	Correo electrónico
Len, Julián	467/14	julianlen@gmail.com
Len, Nicolás	819/11	nicolaslen@gmail.com
Mascitti, Augusto	954/11	mascittija@gmail.com

Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

http://www.fcen.uba.ar

Índice

1. Resumen

Las competencias deportivas, en todas sus variantes y disciplinas, requieren casi inevitablemente la comparación entre competidores mediante la confección de Tablas de Posiciones y Rankings en base a resultados obtenidos en un perıodo de tiempo determinado. En este trabajo nos proponemos estudiar el comportamiento de distintos enfoques, para la confección de rankings de una competencia en base a resultados de distintos partidos. Los distintos métodos que estudiaremos son, Colley Matrix Method (CMM) y Winning Percentage (WP). Además de comparar tanto la justicia, como los distintos Rankings devueltos por cada método, en el trabajo se estudian distintas implementaciones y resoluciones de CMM. Aprovechando propiedades de las matrices resultantes, para optimizar el tiempo de cómputo de los cálculos.

En este trabajo, llegamos a distintas conclusiones.

En términos cuantitativos, encontramos que la resolución de un sistema de ecuaciones, donde el término de independiente varía, es conveniente utilizar Cholesky. Si las matrices son siempre diferentes, es mínima la diferencia entre resolverla utilizando Eliminación Gaussiana o Factorización Cholesky.

Pudimos verificar cualitativamente que CMM refleja fielmente la transitividad en la disputa de distintos encuentros. Con lo cual es conveniente considerarlo en aquellas competencias en la cual los participantes no juegan todos contra todos o no juegan la misma cantidad de partidos.

2. Introducción

2.1. Objetivos Generales

En este trabajo práctico, vamos a ver distintas formas de resolver sistemas de ecuaciones aplicados a un problema de la vida real. En particular vamos a trabajar con *Sports Analytics*, específicamente con el problema de la confección de *Rankings*, para analizar los rendimientos de participantes en distintos tipos de competencias.

La confección de un ranking, es la asignación de un valor (en relación a algún estándar de calidad definido) que representa el orden relativo de cada participante con respecto al resto en un determinado momento de la competencia. Este tipo de problemas es muy común en competencias deportivas como basket ball y tenis.

Dependiendo del tipo de competencia, tiene más o menos sentido, algunas formas de calcular los rankings, ya sea, la utilización de la locaía como dato relevante a la hora de asignar un puntaje, o puntos bonus por realizar determinadas acciones a lo largo del encuentro. De esta manera, es una discusión aparte la *justicia* del ranking elegido para calificar a los participantes, por lo tanto se plantean circunstancias para las cuales la simple evaluación de la cantidad de partidos ganados sobre el total puede no ser suficiente.

Principalmente nos vamos a enfocar en el *Colley Matrix Method*, el cual elabora un ranking basándose en la cantidad de partidos ganados, perdidos y disputados entre pares de equipos. Desde un principio podemos apreciar que con este método no es relavante ni la diferencias de puntos obtenidos, ni la localía, ni ningún otro atributo que no tenga que ver con el resultado de los encuentros entre los participantes. Este método no solo tiene en cuenta la cantidad de partidos ganados sino también la historia de partidos ganados, para lo cual tiene relevancia la posición en el ranking del resto de los equipos contra los que se jugó.

Para estudiar este método estudiaremos también el método tradicional de contar la cantidad de partidos ganados sobre el total de partidos jugados (Winning Percentage). De esta forma podremos contrastar ambos métodos y conocerlos en mas detalle.

Dado que el método de Colley plantea la resolución de un sistema de ecuaciones lineales, estudiaremos dos métodos que permiten resolver estos sistemas en forma exacta. El primero de ellos es el método de Eliminación Gaussiana que permite resolver el sistema con costo O(n^3). El segundo es el método de factorización de Cholesky, que tiene el mismo costo computacional en primera instancia, pero que presenta una mejora a la hora de calcular sistemas parecidos, en el cual varia el tèrmino independiente.

Analizaremos ambos métodos en el contexto de la resolución del problema de armar un ranking de equipos y nos enfocaremos en dos aspectos. El primero es un análisis cualitativo que permita conocer las características del método de Colley para distintos tipos de torneos con distinta cantidad de equipos. El segundo es un análisis cuantitativo enfocado al estudio de las características, en particular de la evolución del tiempo de ejecución, de los dos métodos que utilizaremos para resolver los sistemas de ecuaciones.

Para realizar el análisis de los métodos mencionados desarrollamos una herramienta en C++ que implementa los algoritmos de Colley (CMM) y WP, y generan como salida el ranking según el método seleccionado.

2.2. Colley Matrix Method

Una de las principales ventajas de este método es que es bastante simple, realiza los análisis solo en función de las victorias y derrotas del participante. De esta manera no se ve afectado el ranking cuando un participante *muy bueno* le gana a un participante *muy malo* por una diferencia de puntos considerables.

Es bastante efectivo a la hora de realizar rankings en competencias en donde los participantes no se enfrentan todos entre sí, y hay que establecer algun tipo de transitividad a la hora de decir que participante es mejor que otro sin siquiera haberse enfrentado.

Sea T la cantidad de equipos participantes en el torneo. El método CMM, propone construir una matriz $C \in \mathbb{R}^{TxT}$ y un vector $b \in \mathbb{R}^{T}$, tal que el ranking buscado $r \in \mathbb{R}^{T}$ es la soluciòn del sistema $Cr = b$. Para el armado del sistema, se definen

$$C_{ij} = \begin{cases} -n_{ij} & si \quad i \neq j \\ \\ 2 + n_i & si \quad i = j \end{cases}$$

y

$$b_i = 1 + (w_i - l_i)/2$$

Siendo n_i la cantidad de partidos jugados por el equipo i, w_i ca cantida de partidos ganados por el equipo i, y analogamente, l_i la cantida de partidos perdidos por el equipo i. Y por último $n_{\hat{i}j}$ la cantidad de partidos disputados entre el equipo i y el equipo j.

2.3. Eliminación Gaussiana

La eliminación Gaussiana es un algoritmo que sirve para determinar las soluciones de un sistema de ecuaciones lineales. Un sistema de ecuaciones se resuelve por el método de Gauss cuando se obtienen sus soluciones mediante la reducción del sistema dado a otro equivalente en el que cada ecuación tiene una incógnita menos que la anterior. El método de Gauss transforma la matriz de coeficientes en una matriz triangular superior, para luego resolver el sistema mediante *backward substitution*.

2.4. Factorización Cholesky

Sabemos que una matriz simétrica definida positiva puede ser descompuesta como el producto de una matriz triangular inferior y su traspuesta. Esto nos favorece ya que, por enunciado, las matrices con las cuales se resolverán sistemas, son simétricas definidas positivas. La matriz triangular inferior es el triángulo de Cholesky de la matriz original definida positiva. La Factorización Cholesky es una factorización de la matriz original, en la multiplicación de una matriz diagonal inferior y su traspuesta. De esta manera sirve para resolver dos sistemas de ecuaciones, que es equivalente a resolver el sistema original.

3. Desarrollo

3.1. Implementación

En esta parte del trabajo práctico, pensamos cuál era la mejor opción para desarrollar los métodos deseados, y poder analizarlos y compararlos.
Para empezar, decidimos utilizar la librería Eigen[1] que nos facilita la creación, edición y las operaciones elementales entre matrices y vectores.

Se implementaron los algoritmos de la *Eliminación Gaussiana* y de la *Factorización de Cholesky*.
Ambos algoritmos reciben como parámetro una matriz A cuadrada y un vector b que será el término independiente del sistema de ecuaciones que se desea resolver. Y devuelve, el resultado del sistema de ecuaciones.
Se tomaron ciertas decisiones al separar el código de cada algoritmo, de manera que ciertas partes del código puedan ser reutilizadas (como por ejemplo, la función backward sustitution).
La complejidad computacional de ambos algoritmos para una matriz de $M \in \mathbb{R}^{nxn}$ es de $O(n^3)$.

3.1.1. Algoritmo de Eliminación Gaussiana

Comenzamos desarrollando el algoritmo de eliminación gaussiana con pivoteo parcial. Pero luego nos dimos cuenta que para este trabajo no es usado, ya que nos asegurábamos que las matrices sean Estrictamente Diagonal Dominante (EDD), o sea, siempre encontraba números distintos de cero en la diagonal. Aunque el pivoteo parcial sirve para reducir el error numérico, la implementación del algoritmo solo pivotea cuando se encuentra con un 0.

Algorithm 1 Eliminación Gaussiana

```
function eliminacionGaussiana(Matriz A, Vector b)
    v = 0
    for k ∈ 0..n − 2 do
        for i ∈ k + 1..n − 1 do
            v = A(i, k)/A(k, k)
            A(i, k) = 0
            for j ∈ k + 1..n − 1 do
                A(i, j) = A(i, j) − v ∗ A(k, j)
            end for
            b(i) = b(i) − v ∗ b(k)
        end for
    end for
    return backwardSubstitution(A, b)
end function
```

3.1.2. Algoritmo de Factorización de Cholesky

Para la factorización de Cholesky, nos aseguramos que las matrices de entrada sean Simétricas Definidas Positivas, para que exista factorización. Una vez asegurado esto, el algoritmo completo se puede dividir en dos partes (las cuales para la sección de experimentación serán usadas de forma separadas). Por un lado la encargada de encontrar la factorización de Cholesky, es decir dado una matriz A, descomponerla en $A = LL^t$, con L triangular inferior y L^t triangular superior. Una vez factorizada la matriz A, el sistema a resolver es $L{*}L^tx = b$. Es decir, primero $L{*}y = b$ y luego $L^t{*}x = y$. Pero como mencionamos antes L y L^t son triangulares. Como L es triangular inferior, L^t es triangular superior. Por lo tanto, $L * y = b$

se resuelve con *fordwardSubstitution*, mientras que $L^t * x = y$ se calcula con *backwardSubstitution*. El coste de esta factorización es de $O(n^3)$ mientras que *backwardSubstitution* y *fordwardSubstitution* es de $O(n^2)$. Luego, puedo tomar un término independiente b_2 distinto, y resolver el sistema de ecuaciones es resolver el sistema $L * y = b_2$, usando únicamente *fordwardSubstitution*.

Algorithm 2 Factorización de Cholesky

```
function ResolucionCholesky(Matriz A, Vector b)
    L, Lt = dameLLt(A)
    y = forwardSubstitution(L, b)
    x = backwardSubstitution(Lt, y)
    return x
end function
```

Se desarrolló el main del proyecto adecuado, de manera que podamos realizar las pruebas y los experimentos desarrollados, con sólo ingresar ciertos parámetros de entrada al ejecutar el programa.

3.2. Experimentación

En esta sección del trabajo práctico, vamos a realizar distintos tipos de experimentos que van a atacar al problema en base a dos focos.

Los cuantitativos, que tratan de evidenciar la diferencia a la hora de elegir un algoritmo de resolución de sistemas de ecuaciones determinado (Eliminación Gaussiana, Factorización Cholesky), con respecto al tiempo de procesamiento. Tanto manteniendo el termino independiente constante, como variándolo.

Y los cualitativos, que están pensados para entender la naturaleza de los distintos métodos de rankeo propuestos (WP, CMM).

3.2.1. Cuantitativo

En la siguiente sección, utilizaremos dos métodos numéricos para la resolución de sistemas de ecuaciones planteados en el trabajo. En particular, realizaremos una comparación de los tiempos de cómputo entre ambos en diferentes escenarios.

En primer lugar, buscamos comparar el tiempo de cómputo total de los métodos, Eliminación Gaussiana y Factorización de Cholesky. Es decir, el tiempo desde que ambos métodos *triangulan* la matriz, hasta que resuelven el sistema. Y poder concluir que ambos, para una matriz, tienen complejidad $O(n^3)$ en resolver un sistema.

Para el cálculo con ambos métodos se crearon matrices cuadradas *random* utilizando la librería *Random* de la *STL* de *C++*. Lo que se buscó en este análisis fue el tiempo de cómputo con respecto al tamaño de la matriz. Para esto, las matrices generadas, iteraban su tamaño de a 100 elementos desde 100 hasta 2900. Por cada tamaño se generaron 10 matrices diferentes, y por cada matriz se corrió el mismo algoritmo otras diez veces.

En particular, para probar la Factorización de Cholesky, nos aseguramos que las matrices sean Simétricas Definidas Positivas (SDP), es decir, sea $A \in \mathbb{R}^{nxn}$,

$$A \text{ es simétrica}$$
$$\text{y}$$
$$x^t A x > 0 \forall x \neq 0$$

Para esto, el siguiente código a partir de una matriz random, la convertía en una matriz random simétrica definida positiva.

```
void simDefPosGenerator(Matriz A){
    A = 0.5*A*(A.inversa());
    #columnas = A.rows();
    matrizAux = matrizAux.llenarconZero(col_rows, col_rows);
    matrizAux.volverlaIdentidad();
    A = A + A.filas()*(matrizAux);
}
```

fuente: [2]

Luego en ambos métodos, se generaron términos independientes aleatorios del tamaño adecuado.

Una vez tomados los tiempos, para evitar outliers, por cada matriz corrida diez veces, se podaron cuatro resultados, dos con el mayor tiempo y dos con el menor. Después se saco un promedio de los resultados de tiempos restantes, quedando así, diez resultados diferentes por cada tamaño. Luego se repitió el procedimiento para estas diez matrices del mismo tamaño, dejando así, un resultado diferente por tamaño de matriz.

En segundo lugar, se experimentó para comprobar nuestra hipótesis de la mejora en cuanto a tiempos de cómputo de la factorización de Cholesky contra la Eliminación Gaussiana al alterar el término independiente. O sea, el objetivo, es probar que al tener factorizada la matriz con Cholesky, al cambiar el término independiente, la matriz queda *"triangulada"*, a diferencia de la Eliminación Gaussiana que debe triangular la matriz cada vez que resuelve un sistema. Para probar esto, se generaron de la misma manera que en el anterior experimento, matrices random (SDP, para asegurarnos que tenga Factorización Cholesky), con tamaños de 100 a 1600 con intervalos de 100 elementos. Luego, por cada matriz, se cambio el término independiente diez veces. Y luego por cada matriz con su respectivo término independiente se corrió diez veces con cada método calculando el tiempo de cómputo. La particularidad de este experimento fue, que a la hora de resolver un sistema, con distintos términos independientes, con Cholesky, se guardaba la factorización de la matriz original. Luego para calcular los tiempos y evitar outliers, se usó el mismo método que el experimento anterior. Con la diferencia que en vez de tener 10 tamaños diferentes por matriz, se tienen 10 términos independientes distintos. Además, el tiempo de cómputo de la factorización de Cholesky, se suma una única vez por tamaño de matriz.

3.2.2. Cualitativo

En este apartado especifico vamos a comparar los distintos Rankings entre ellos. Primero vamos a ver que efectivamente hay diferencias sustantivas entre los mismos. Luego ir viendo como se comporta cada ranking en determinadas situaciones.

3.2.2.1. Variación de los Distintos Rankings

Proponemos medir la variación de los distintos métodos de Rankeo en base a la cantidad de Swaps de las posiciones de cada equipo en cada Ranking. Utilizamos uno de los ranking como patrón de comparación, y calculamos la diferencia absoluta de la posición de cada equipo en el otro ranking. Esta medida de la *diferencia* (dispersión) la vamos a tomar tanto para el ranking completo, como para *los mejores* equipos (las primeras posiciones del Ranking).

La fórmula para calcular la dispersión es la siguiente.

$$\Delta = \frac{\sum_{x \in Equipos} |CMM_x - WP_x|}{\#Equipos}$$

Es razonable utilizarla como medida de dispersión, ya que si los rankings son iguales, entonces $\Delta = 0$, y a medida que la cantidad de Swaps aumentan, esto se aleja del 0. Rápidamente podemos darnos cuenta, que mientras más alejado del origen se encuentra, más Swaps hay entre los rankings. Lo dividimos por la cantidad de equipos que tiene la competencia, para tener una idea del promedio de distancia de cada equipo entre los distintos rankings.

Calculamos los respectivos rankings tanto para los datos del circuito de tenis de 2015 y el de la NBA en 2016.

3.2.2.2. *Justicia* de los Rankins

En este experimento queremos evidenciar que CMM es una forma mas *justa*, que WP, de rankear equipos en competencias en las cuales no todos juegan contra todos, ni tienen la misma cantidad de partidos.

Para esto proponemos generar una fixture, en el cual un equipo le gane reiteradas veces a un mismo competidor, y otro equipo que gane la misma cantidad de encuentros que el primero, pero distribuido en varios equipos distintos.

Sabemos que WP solo se fija en la cantidad de partidos jugados y ganados, por lo cual asumimos que en el ranking, estos dos equipos van a tener el mismo puntaje.
Por otro lado, suponemos que en CMM, aquel equipo que tuvo un fixture más complicado (le ganó partidos a distintos equipos) se tiene que ver reflejado en una diferencia entre los rankings de estos dos equipos.

Para llevar acabo esto, propusimos que un equipo le gane 20 encuentros a uno que perdió todos los que disputó, y un segundo equipo que ganó 20 encuentros, pero todos estos fueron disputados contra 4 equipos distintos, y estos 4 equipos, a su vez tuvieron encuentros entre sí que fueron ganados y perdidos eventualmente, para darle algún tipo de relevancia a estos 4 equipos.

Por último queremos verificar una hipótesis un poco más fuerte, suponemos que un tercer equipo ganó 19 encuentros, también disputados con varios equipos distintos, y queremos ver que con CMM este tiene un mejor ranking que aquel que le ganó 20 veces al mismo equipo. Sabemos que según WP este equipo tiene un ranking peor que dicho equipo.

Equipo	Jugados	Ganados	Perdidos
A	20	20	0
B	25	0	25
C	20	20	0
D	14	4	10
E	19	14	5
F	27	4	23
G	27	4	23
H	19	19	0

Cuadro 1: Encuentros

A le ganó a B 20 veces	E le ganó a F 7 veces	D le ganó a G 2 veces
C le ganó a D 5 veces	E le ganó a G 7 veces	H le ganó a D 4 veces
C le ganó a E 5 veces	F le ganó a G 4 veces	H le ganó a B 5 veces
C le ganó a F 5 veces	G le ganó a F 4 veces	H le ganó a F 5 veces
C le ganó a G 5 veces	D le ganó a F 2 veces	H le ganó a G 5 veces

Cuadro 2: Resultados

3.2.2.3. Orden relativo

En este segundo experimento queremos probar que CMM mantiene un orden similar a WP cuando los equipos no compiten entre sí.

Para esto realizamos un fixture en el cual los 3 primeros equipos de la competencia ganaron sus respectivos partidos al resto de los equipos, pero nunca disputaron encuentros entre ellos.

Inferimos que tienen que mantener el mismo orden que WP.

Equipo	Jugados	Ganados	Perdidos
A	50	50	0
B	54	52	2
C	55	52	3
D	32	1	31
E	32	1	31
F	32	1	31
G	32	1	31
H	31	1	30

Cuadro 3: Encuentros

A le ganó a D 10 veces	B le ganó a D 11 veces	C le ganó a D 10 veces
A le ganó a E 10 veces	B le ganó a E 11 veces	C le ganó a E 10 veces
A le ganó a F 10 veces	B le ganó a F 10 veces	C le ganó a F 11 veces
A le ganó a G 10 veces	B le ganó a G 10 veces	C le ganó a G 11 veces
A le ganó a H 10 veces	B le ganó a H 10 veces	C le ganó a H 10 veces
D le ganó a B 1 vez	E le ganó a B 1 vez	F le ganó a C 1 vez
G le ganó a C 1 vez	H le ganó a C 1 vez	

Cuadro 4: Resultados

3.2.2.4. Ganarle a los mejores

Queremos verificar ahora que CMM es más *justo* agregando un equipo al fixture anterior que ganó una cantidad similar pero menor de encuentros a los equipos perdedores, y a su vez, ganó los partidos disputados contra los primeros de la tabla, y este tendría que tener el mejor ranking de todos. No así según WP.

Equipo	Jugados	Ganados	Perdidos
A	51	50	1
B	55	52	3
C	56	52	4
D	41	1	40
E	41	1	40
F	42	2	40
G	42	2	40
H	41	2	39
I	51	48	3

Cuadro 5: Encuentros

A le ganó a D 10 veces	B le ganó a D 11 veces	C le ganó a D 10 veces	I le ganó a D 9 veces
A le ganó a E 10 veces	B le ganó a E 11 veces	C le ganó a E 10 veces	I le ganó a E 9 veces
A le ganó a F 10 veces	B le ganó a F 10 veces	C le ganó a F 11 veces	I le ganó a F 9 veces
A le ganó a G 10 veces	B le ganó a G 10 veces	C le ganó a G 11 veces	I le ganó a G 9 veces
A le ganó a H 10 veces	B le ganó a H 10 veces	C le ganó a H 10 veces	I le ganó a H 9 veces
D le ganó a B 1 vez	E le ganó a B 1 vez	F le ganó a C 1 vez	H le ganó a I 1 vez
G le ganó a C 1 vez	H le ganó a C 1 vez	F le ganó a I 1 vez	G le ganó a I 1 vez
I le ganó a A 1 vez	I le ganó a B 1 vez	I le ganó a C 1 vez	

Cuadro 6: Resultados

3.2.2.5. Relación del desarrollo de un encuentro con respecto al ranking

Sabemos que con WP el desarrollo de un encuentro en cualquier punto de la competencia, solo afecta los ratings de los equipos en disputa.

Queremos verificar que con CMM, el desarrollo de un encuentro, afecta el rating de todos los equipos que participan en la competencia. Y aventurandonos un poco más, inferimos que el desarrollo de un partido *difícil* (entre equipos que se encuentran primeros en la tabla) genera variaciones más pronunciadas en los ratings, que disputando un encuentro intrascendente (entre equipos que se encuentran al final de la tabla).

3.2.2.6. Estrategia para subir el rating

Suponemos que un equipo que quiere aumentar su rating y por lo tanto su posicionamiento en el ranking, dependiendo el método de rankeo puede optar por distintas estrategias.

Según WP no importa a quien le gane, si gana un encuentro, inmediatamente su rating se ve incrementado. Con lo cual ganarle a los peores equipos debería ser más fácil que ganarle a los mejores, pero el rating aumenta en la misma proporción.

Suponemos que es todo lo contrario en CMM. Al ganarle a un equipo que es *bueno* el rating de un equipo aumenta considerablemente más que si le gana a un equipo *malo*.

4. Resultados, Discusión y Conclusiones

4.1. Experimentación

En esta sección vamos a ver los resultados de todos los experimentos realizados y los análisis pertinentes.

4.1.1. Cuantitativo

4.1.1.1. Experimento A: Tiempo de cómputo total

Como fue explicado anteriormente, este experimento busca acotar el tiempo de cómputo de cada método desde que triangula la matriz hasta que resuelve el sistema. Los resultados de ambos son,

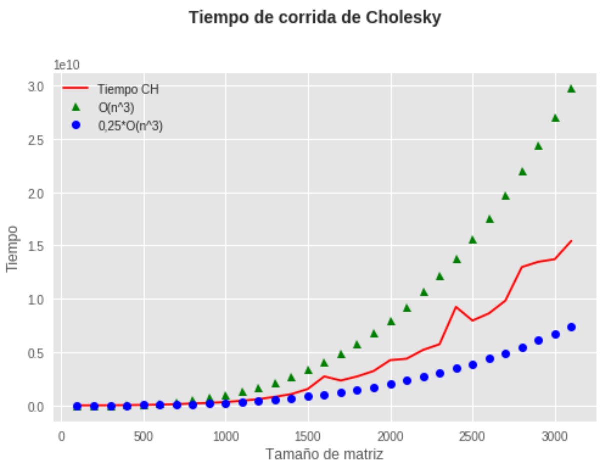

Figura 1: Complejidad Cholesky

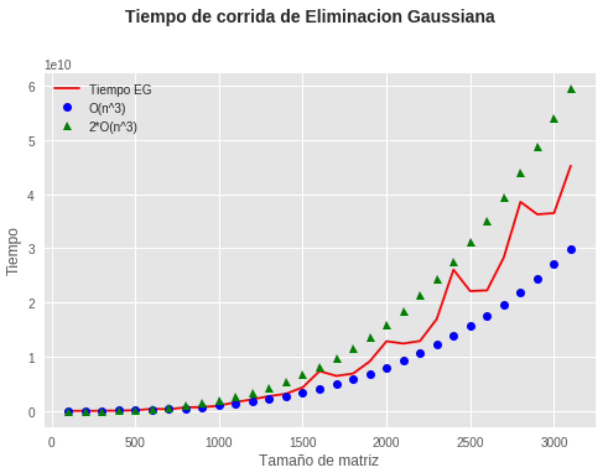

Figura 2: Complejidad Eliminación Gaussiana

4.1.1.2. Conclusiones y discusión

Finalmente como se puede observar, ambas cotas de complejidad son $O(n^3)$. Además, se pueden ver ciertas irregularidades en los ejes tanto del tiempo de Cholesky como de Gauss. No podemos deducir a que se deben estas anomalías, ya que se producen para los dos algoritmos de resolución de sistemas de ecuaciones, y en el desarrollo llegamos a realizar las implementaciones con dos librerías de matrices distintas para C++ y los resultados siguieron siendo iguales, quedaría para futuras experimentaciones encontrar si se debe al entorno donde el experimento fue corrido o a la implementación de alguno de los métodos o alguna particularidad del tamaño de las matrices y la utilización de la memoria.

Por otro lado, mencionar que si debemos comparar las distintas constantes que multiplican a $O(n^3)$, es el tiempo de cómputo de Cholesky menor al de Eliminación Gaussiana.

4.1.1.3. Experimento B: Fijando matriz y cambiando el término independiente

Para este experimento, comparamos los resultados de la resolución de un sistema utilizando Cholesky y Eliminación Gaussiana alterando el término independiente. Los resultados fueron,

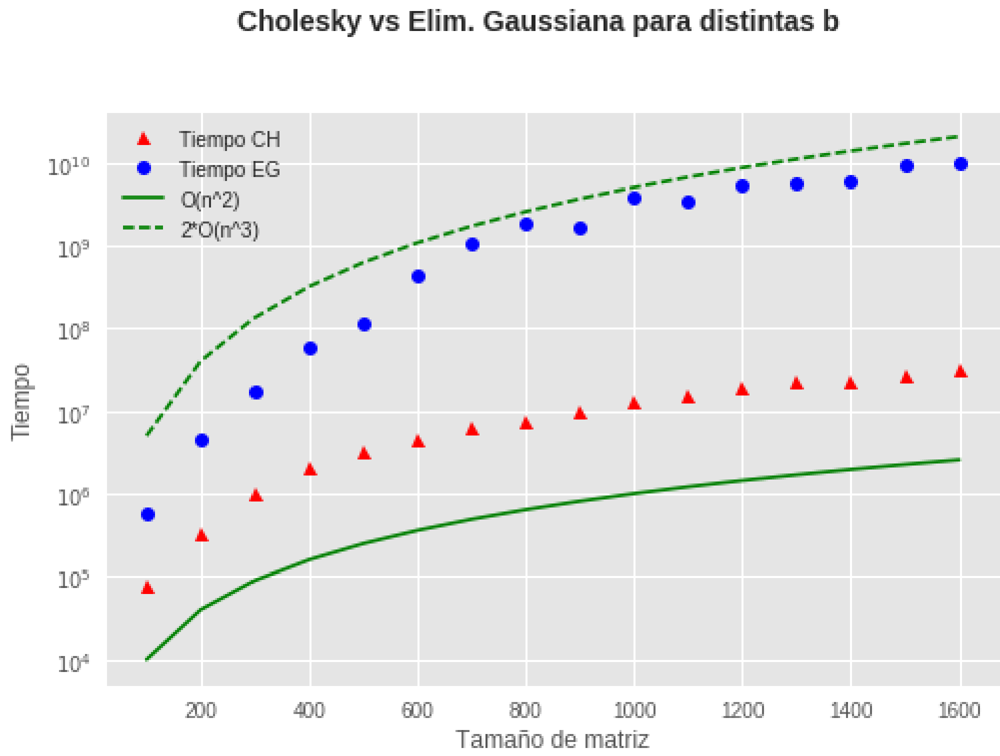

Figura 3: Tiempos de cómputo para distintos térm. independientes

4.1.1.4. Conclusiones y discusión

Luego nuestra hipótesis de que, al tener factorizada la matriz principal con Cholesky, si cambian el término independiente tiene mejor tiempo de cómputo que la Eliminación Gaussiana queda confirmada para las instancias de la experimentación. Pues como fue explicado en secciones anteriores, una vez factorizada la matriz con Cholesky, es independiente el término al que se le iguale, pues el costo de factorizar Cholesky se tiene una sola vez por matriz, mientras que la Eliminación Gaussiana, por cada matriz, debe triangular y resolver en su totalidad. O sea el costo de $O(n^3)$ Cholesky lo tiene la primera vez que factoriza una matriz, luego para distintos términos independientes el costo es $O(n^2)$. Mientras que la Eliminación Gaussiana es siempre $O(n^3)$.

4.1.2. Cualitativo

4.1.2.1. Variación de los Distintos Rankings

Para el tenis, los resultados tanto para el *TOP 10*, como para el ranking en su totalidad nos dieron valores lejos del 0. Para ser mas exactos:

$$\Delta_{tot} = 54$$

$$\Delta_{top_{10}} = 67$$

Podemos ver como los dos sistemas de rankeo son distintos y más aún la diferencia es considerablemente mayor en el *TOP 10*.

Luego corriendo el mismo experimento para la NBA vemos que los rankings coinciden posición a posición por lo tanto, no hay swaps y el delta es igual a cero. Por lo tanto la utilización de cualquiera de los dos rankings no tiene ninguna relevancia.

$$\Delta_{tot} = 0$$

$$\Delta_{top_{10}} = 0$$

No sabemos bien como se desarrolla la NBA para poder llegar a una conclusión interesante sobre este resultado. Hay partidos de todos contra todos por *conferencia*, más algunos partidos *inter-conferencia*, luego se hacen *playoffs* también por conferencia, y por último una final de toda la NBA, que la juegan el campeón de cada conferencia.
Pero por lo menos en las primeros equipos del ranking, son aquellos que ganaron la mayor cantidad de partidos, y al mismo tiempo a los mejores, dado la eliminación de los *playoffs*.

4.1.2.2. *Justicia* de los Rankings

A continuación ponemos el resultado de los rankings para dicho experimento.

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	1	C	0.829651
2º	C	1	H	0.741727
3º	H	1	A	0.727694
4º	E	0.736842	E	0.599595
5º	D	0.307692	D	0.430406
6º	F	0.148148	B	0.250464
7º	G	0.148148	G	0.210232
8º	B	0	F	0.210232

Cuadro 7: Rankings

Pudimos verificar que CMM se comportaba (casi) como esperábamos. No solo aquel que tuvo la misma cantidad de victorias contra distintos participantes (C) tuvo un mejor rating, sino aquel que tuvo menos victorias, pero a distintos equipos (H), también tuvo un mejor rating que el que perdió todos sus encuentros.

También podemos apreciar pequeñas distinciones entre los primeros equipos de la tabla, la cual WP no realiza, con lo cual consideramos que CMM tiene más márgen de *diferenciabilidad*, el cual depende de la *calidad* de partidos que ganó cada participante.

Lo que nos parece extraño, es la posición de B, que perdió todos los encuentros, pero según CMM es mejor que F y G que ganaron algunos. Esto puede ser producto de que B perdió partidos ante A y H que están en el *TOP 3* de la tabla. Esto, puede ser que le asigne algunos puntos de rating que no teníamos contemplado.

Esperamos comprender un poco más la *diferenciabilidad* y la asignación de puntos por parte de CMM con el resto de los experimentos.

4.1.2.3. Orden relativo

A continuación ponemos el resultado de los rankings para dicho experimento.

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	1	A	0.815409
2º	B	0.962963	B	0.780611
3º	C	0.945455	C	0.76385
4º	H	0.0322581	H	0.329168
5º	D	0.03125	D	0.328233
6º	E	0.03125	E	0.328233
7º	F	0.03125	F	0.327247
8º	G	0.03125	G	0.327247

Cuadro 8: Rankings

Verificamos que se cumple nuestra hipótesis, y el orden entre los dos rankings coincide posición a posición. Se puede ver que cada uno calcula su propio rating, pero el orden final de los dos es el mismo.

4.1.2.4. Ganarle a los mejores

A continuación ponemos el resultado de los rankings para dicho experimento.

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	0.980392	A	0.777389
2º	B	0.945455	I	0.757014
3º	I	0.941176	B	0.742883
4º	C	0.928571	C	0.728256
5º	H	0.0487805	H	0.308923
6º	F	0.047619	G	0.307089
7º	G	0.047619	F	0.307089
8º	D	0.0243902	D	0.285678
9º	E	0.0243902	E	0.285678

Cuadro 9: Rankings

Se cumple a medias nuestra hipótesis. Esperábamos que se vea reflejado que el equipo I era el mejor de todo el certamen.
CMM nos muestra un mejor rendimiento, pero I terminó en la segunda posición y esperábamos que termine en la primera.

Esto puede ser producto de que hicimos que pierda solo tres partidos, y no contemplamos que estos partidos que perdió fueron contra equipos a los que le ganaron los tres primeros. Por lo tanto por transitividad se empiezan a mezclar las cosas y no es tan claro quien es el mejor de todos.

De esta manera nos surjen una nueva experimentación para corroborar que eliminando estos partidos que representan algún tipo de transitividad, el rating de I va a mejorar considerablemente y también su posición relativa en el ranking generado por CMM.

4.1.2.5. Ganarle a los primeros, sin perder transitivamente

Vamos a hacer que aquel equipo le gane a los tres primeros (I), pierda contra equipos que *NO* disputaron encuentros contra equipos que haya participado contra los tres primeros.

Para esto, agregamos dos nuevos equipos que compitieron entre ellos y luego le ganaron a I. Esto produce que el WP de I se reduzca, pero al haberle ganado a los tres primeros tendría que demostrar nuestra hipótesis de que en CMM tiene un mejor ranking que en el experimento anterior por transitividad.

Equipo	Jugados	Ganados	Perdidos
A	51	50	1
B	55	52	3
C	56	52	4
D	41	1	40
E	41	1	40
F	41	1	40
G	41	1	40
H	41	2	39
I	54	48	6
J	43	23	20
K	43	23	20

Cuadro 10: Encuentros

A le ganó a D 10 veces	B le ganó a D 11 veces	C le ganó a D 10 veces	I le ganó a D 9 veces
A le ganó a E 10 veces	B le ganó a E 11 veces	C le ganó a E 10 veces	I le ganó a E 9 veces
A le ganó a F 10 veces	B le ganó a F 10 veces	C le ganó a F 11 veces	I le ganó a F 9 veces
A le ganó a G 10 veces	B le ganó a G 10 veces	C le ganó a G 11 veces	I le ganó a G 9 veces
A le ganó a H 10 veces	B le ganó a H 10 veces	C le ganó a H 10 veces	I le ganó a H 9 veces
D le ganó a B 1 vez	E le ganó a B 1 vez	F le ganó a C 1 vez	H le ganó a I 1 vez
G le ganó a C 1 vez	H le ganó a C 1 vez	F le ganó a I 1 vez	G le ganó a I 1 vez
I le ganó a A 1 vez	I le ganó a B 1 vez	I le ganó a C 1 vez	
J le ganó a K 20 veces	K le ganó a J 20 veces	J le ganó a I 3 veces	K le ganó a I 3 veces

Cuadro 11: Resultados

4.1.2.6. Resultado del experimento de Transitividad

A continuación ponemos el resultado de los rankings para dicho experimento.

Posición	WP		CMM	
	Equipo	Rating	Equipo	Rating
1º	A	0.980392	J	0.913238
2º	B	0.945455	K	0.913238
3º	C	0.928571	I	0.68873
4º	I	0.888889	A	0.685275
5º	J	0.534884	B	0.651414
6º	K	0.534884	C	0.63502
7º	H	0.025	H	0.203588
8º	D	0.0243902	D	0.202756
9º	E	0.0243902	E	0.202756
10º	F	0.0243902	F	0.201993
11º	G	0.0243902	G	0.201993

Cuadro 12: Rankings

Pudimos corroborar nuestra hipótesis rotundamente. Pensábamos que el equipo I, bajo estas nuevas condiciones, iba a superar tanto a A, B y C, pero podemos ver que tanto J como K, no solo superaron a A, B y C, sino que también a I.

Por lo tanto la transitividad queda bastante reflejada en el método de CMM. Esto fue producto de que por más que J y K hayan ganado la mitad de sus partidos, estos le ganaron por *transitividad* a los mejores de todos.

No podemos cuantificar bien que relevancia tiene ganarle 3 o 1 partido a aquel equipo que le ganó a la mayoría, pero esperamos seguir entendiendo esta dinámica con los siguientes experimentos.

4.1.2.7. Relación del desarrollo de un encuentro con respecto al ranking

Para realizar este experimento vamos a tomar el desarrollo de una competencia y agregar resultados de un único partido para ver la evolución de los ratings de los equipos participantes y calcular sus variaciones.

Vamos a tomar el siguiente fixture.

Equipo	Jugados	Ganados	Perdidos
A	51	50	1
B	55	52	3
C	56	52	4
D	41	1	40
E	41	1	40
F	41	1	40
G	41	1	40
H	41	2	39
I	54	48	6
J	43	23	20
K	43	23	20

Cuadro 13: Encuentros

4.1.2.8. Partido Fácil

Consideramos que F y G son los peores equipos para los dos rankings propuestos y el desarrollo del fixture. Por lo tanto vamos a ver como evolucionan los mismos simulando un encuentro en el cual G le gana un partido a F.

Antes de disputar el partido los rankings son los siguientes:

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	0.980392	J	0.913238
2º	B	0.945455	K	0.913238
3º	C	0.928571	I	0.68873
4º	I	0.888889	A	0.685275
5º	J	0.534884	B	0.651414
6º	K	0.534884	C	0.63502
7º	H	0.025	H	0.203588
8º	D	0.0243902	D	0.202756
9º	E	0.0243902	E	0.202756
10º	F	0.0243902	F	0.201993
11º	G	0.0243902	G	0.201993

Cuadro 14: Rankings sin partido disputado entre F y G

Luego del partido disputado los rankings son los siguientes:

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	0.980392	J	0.913238
2º	B	0.945455	K	0.913238
3º	C	0.928571	I	0.68873
4º	I	0.888889	A	0.685275
5º	J	0.534884	B	0.651414
6º	K	0.534884	C	0.63502
7º	G	0.047619	G	0.213104
8º	H	0.025	H	0.203588
9º	D	0.0243902	D	0.202756
10º	E	0.0243902	E	0.202756
11º	F	0.0238095	F	0.190882

Cuadro 15: Rankings con partido disputado entre F y G

No podemos corroborar nuestra teoría y empezamos a pensar que no importa la dificultad del partido, tanto WP como CMM se modifica los rating de los equipos que disputan algún encuentro entre ellos.

Podemos ver que la victoria se vio reflejada subiendo el rating y la posición de G con respecto a F y reduciendo el rating de F. Pero esperábamos ver modificado, aunque sea minimamente, los ratings del resto los equipos.

Podemos apreciar una evolución del equipo ganador de un 5.5 % de aumento en su ranking y una disminución en los mismos términos del equipo perdedor.

4.1.2.9. Partido Difícil

Consideramos que K y J son los mejores equipos para los dos rankings propuestos y el desarrollo del fixture. Por lo tanto vamos a ver como evolucionan los mismos simulando un encuentro en el cual K le gana un partido a J.

Antes de disputar el partido los rankings son los siguientes:

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	0.980392	J	0.913238
2º	B	0.945455	K	0.913238
3º	C	0.928571	I	0.68873
4º	I	0.888889	A	0.685275
5º	J	0.534884	B	0.651414
6º	K	0.534884	C	0.63502
7º	H	0.025	H	0.203588
8º	D	0.0243902	D	0.202756
9º	E	0.0243902	E	0.202756
10º	F	0.0243902	F	0.201993
11º	G	0.0243902	G	0.201993

Cuadro 16: Rankings sin partido disputado entre J y K

Luego del partido disputado los rankings son los siguientes:

	WP		CMM	
Posición	Equipo	Rating	Equipo	Rating
1º	A	0.980392	K	0.918985
2º	B	0.945455	J	0.907491
3º	C	0.928571	I	0.68873
4º	I	0.888889	A	0.685275
5º	K	0.545455	B	0.651414
6º	J	0.522727	C	0.63502
7º	H	0.025	H	0.203588
8º	D	0.0243902	D	0.202756
9º	E	0.0243902	E	0.202756
10º	F	0.0243902	F	0.201993
11º	G	0.0243902	G	0.201993

Cuadro 17: Rankings con partido disputado entre J y K

Nuevamente nos encontramos con que tanto en CMM como WP, se ven modificados solos los ratings de los participantes involucrados en el partido disputado.

Con este segundo experimento nos hace pensar más fuertemente que nuestra hipótesis es falsa. Y tanto en WP, como en CMM, el desarrollo de un partido modifica el rating solo de los equipos involucrados.

Podemos apreciar una evolución del equipo ganador de un 0.63 % de aumento en su rating y una disminución en los mismos términos del equipo perdedor.

4.1.2.10. Estrategia para subir el rating

Pudimos ver con los experimentos anteriores que una victoria de un equipo *muy malo*, implica un buen incremento de su rating. Ganándole al peor de los equipos del torneo, el incremento de su rating es de alrededor de un 5.5 %. Verifiquemos que al ganarle a un *buen* equipo, este incremento es aún mayor.

Para esto volvimos a realizar el mismo experimento pero ganándole un partido a el mejor equipo del certamen.

Hagamos que el equipo G le gane al equipo K.

Antes de disputar el partido:

	CMM	
Posición	Equipo	Rating
2º	K	0.913238
7º	G	0.201993

Cuadro 18: Ranking

Luego de disputar el partido:

	CMM	
Posición	Equipo	Rating
2º	K	0.806878
7º	G	0.245962

Cuadro 19: Ranking

Podemos verificar que el incremento del rating de G se ve incrementado en un 21.76 % y el de K se ve reducido en un 11.65 %.

Por lo tanto efectivamente es una buena política intentar ganar pocos partidos a los mejores del torneo o muchos partidos a los mas débiles.

5. Introducción del Empate en CMM

5.1. Introducción del empate en CMM

Luego de comprender un poco más el funcionamiento CMM, entendemos que en la matriz que propone el método, se encuentra reflejado el fixture y los partidos disputados entre los distintos equipos. Luego en el termino independiente, se ve reflejado la *potencia* de cada equipo en el certamen.

Por lo tanto una idea razonable es que tanto las victorias, como los empates se vean reflejados en la potencia de un equipo. Por lo que proponemos el siguiente cambio en el termino independiente que se utiliza para resolver el sistema de ecuaciones.

$$b_i = 1 + \frac{(3 * w_i + d_i - l_i)}{2}$$

Donde d_i se refiere a los empates del equipo i-ésimo.

De esta manera las victorias se ve multiplicada por 3 y se le agrega los empates al calculo de la formula. Con lo cual logro diferenciar y darle una relevancia más importante a las victorias sobre los empates y también estoy considerando los empates dentro de la formula para asignarle la potencia a los equipos.

De esta manera el método funciona exactamente igual pero ahora empatar un encuentro incrementa un poco el rendimiento del equipo, siempre siendo mejor ganar un encuentro.

6. Referencias

[1] http://eigen.tuxfamily.org/index.php?title=Main_Page [2] https://math.stackexchange.com/questions/357980/ho
to-generate-random-symmetric-positive-definite-matrices-using-matlab