Trabajo Práctico Nº 1 - Recuperatorio

“(No) Todo Pasa”

Primer cuatrimestre de 2016 — Métodos Numéricos

Integrante	LU	Correo electrónico
Gonzalez, Juan Alberto	324/14	gonzalezjuan.ab@gmail.com
Rodriguez, Santiago	094/14	santi_rodri_94@hotmail.com
Sticco, Patricio Bernardo	337/14	pbsticco@hotmail.com
Walter, Nicolás	272/14	nicowalter25@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

En el presente trabajo práctico nos proponemos estudiar distintos métodos de confección de Rankings en competencias deportivas.

Los métodos más usados en la actualidad están basados en calcular el porcentaje de partidos ganados sobre los partidos jugados o asignar un puntaje fijo por cada victoria.

Éstos métodos no siempre son precisos ni eficaces, por ejemplo, cuando difiere la cantidad de partidos jugados por los equipos. Frente a esta problemática, compararemos los rankings generados por el método de porcentaje de victorias (WP) frente al *CMM (Colley Matrix Method)*.

A la hora de implementar dichos métodos, el problema de confección de rankings se reduce a la resolución de un sistema de ecuaciones lineales los cuales resolveremos mediante los métodos de Eliminacion Gaussiana (EG) y Factorización de Cholesky.

Realizaremos distintos experimentos para comparar CMM contra WP. Entre estas comparaciones se encuentran comparar los tiempos de cómputo al variar ciertos datos como así también los rankings generados.

Palabras clave

Confección de Rankings, Resolución de sistemas lineales, Eliminación Gaussiana, Factorización de Cholesky.

2. Introducción Teórica

El objetivo del presente informe es resolver un problema práctico mediante el modelado matemático del mismo. Este problema consiste en comparar dos métodos de determinación de rankings de equipos de una competencia en base al resultado de un conjunto de partidos.

Para generar los distintos rankings, utilizaremos datos reales de ciertas competencias, como el circuito ATP de tenis o la liga de la NBA.

El método WP hará un ranking mediante el cálculo : $\frac{\#partidos\ ganados}{\#partidos\ jugados}$[1] Se puede observar que si se quiere subir de posición en el ranking lo importante es tener una tasa alta de victorias, no teniendo importancia quién es el oponente ni cuan significativa fue la victoria en términos de diferencia de puntaje en un partido.

Por el otro lado, el *Colley Matrix Method*[2] se basará en la Regla de Laplace de sucesos. Esta regla permite aproximar las probabilidades de eventos booleanos, en nuestro caso que un equipo gane o pierda un partido. En particular, si sobre k eventos observamos s casos exitosos, la regla establece que $\frac{s+1}{k+2}$ es un mejor estimador que el porcentaje estándar, $\frac{s}{k}$.

El método CMM propone construir una matriz C $\in \mathbb{R}^{TxT}$[3] y un vector b $\in \mathbb{R}^T$ tal que el ranking buscado r $\in \mathbb{R}^T$ se reduce a la resolución del sistema lineal Cr=b.

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

y $b_i = 1 + \frac{w_i - l_i}{2}$, i pertenece al conjunto de equipos.

Para resolver estos sistemas utilizaremos el método de Eliminación Gaussiana y la factorización de Cholesky dado que la matriz resultante del sistema planteado por el CMM es simétrica y definida positiva asegurando de este modo la existencia de dicha factorización. Con la finalidad de lograr mejoras de eficiencia también es importante notar que la matriz no posee 0's en la diagonal permitiendo de este modo utilizar la factorización Gaussiana sin pivoteo.

[1] Utilizamos el símbolo # para hablar de cantidad
[2] http://colleyrankings.com/method.html
[3] T es el cardinal del conjunto de equipos

3. Desarrollo

3.1. Métodos numéricos usados

A partir del sistema de ecuaciones lineales generado por el CMM, el problema de generar el ranking se reduce a encontrar la solución a ese sistema. Para resolverlo, haremos uso de Eliminación Gaussiana sin pivoteo y de la factorización de Cholesky.

Además, contrastaremos la eficiencia de estos métodos ante distintas situaciones. El hecho de que siempre se pueda aplicar eliminación Gaussiana a la matriz del sistema sin necesidad de hacer pivoteo (es decir, que nunca aparezca un 0 en la diagonal al triangular) se debe a que la matriz generada por el método de Colley es estrictamente diagonal dominante.

3.1.1. Eliminación Gaussiana

Algoritmo 1: Eliminación Gaussiana

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$

Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$, considerando a $\mathbf{A}$ como una matriz triangular inferior.

para $i \leftarrow 1$ **a** $n-1$ **hacer**
  para $j \leftarrow i+1$ **a** n **hacer**
    $m_{ji} \leftarrow \frac{a_{ji}}{a_{ii}}$
    if ($\mathrm{m}_{ji} \neq 0$) **then**
      $A_j \leftarrow A_j - m_{ji} \cdot A_i$
      $b_j \leftarrow b_j - m_{ji} \cdot b_i$
    endif
  fin
fin
$s \leftarrow$ resolverTS(A,b)

Algoritmo 2: resolver Triangular Superior

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$

Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$.

$s_{n1} \leftarrow \frac{b_{n1}}{a_{nn}}$
para $i \leftarrow n-1$ **a** 1 **hacer**
  $s_{i1} \leftarrow \frac{b_{i1} - \sum_{j=i+1}^{n} a_{ij} \cdot s_{j1}}{a_{ii}}$
fin

3.1.2. Factorización de Cholesky

Algoritmo 3: Factorización de Cholesky

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}$
Resultado: $s \in \mathbb{R}^{n \times n}$, solución del algoritmo, considerando a $\mathbf{A}$ como una matriz simétrica y definida positiva. Al finalizar el algoritmo s es triangular inferior

$s_{11} \leftarrow \sqrt{a_{11}}$
para $i \leftarrow 2$ **a** n **hacer**
  $s_{j1} \leftarrow \frac{a_{j1}}{s_{11}}$
fin
para $j \leftarrow 2$ **a** $n-1$ **hacer**
  $s_{jj} \leftarrow \sqrt{a_{jj} - \sum_{k=1}^{j-1} s_{jk}^{2}}$
  para $i \leftarrow j+1$ **a** n **hacer**
    $s_{ij} \leftarrow \frac{a_{ij} - \sum_{k=1}^{j-1} s_{ik} \cdot s_{jk}}{s_{jj}}$
  fin
fin
$s_{nn} \leftarrow \sqrt{a_{nn} - \sum_{k=1}^{n-1} s_{nk}^{2}}$

Algoritmo 4: Resolver sistema utilizando Cholesky

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$
Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$, considerando a $\mathbf{A}$ como una matriz triangular inferior.

$B \leftarrow$ Cholesky(A)
Matriz L= resolverTI(B,b)
$L \leftarrow L^t$
$s \leftarrow$resolverTS(B,L)

3.2. Aplicabilidad de los métodos elegidos

Los métodos numéricos utilizados no son aplicables a cualquier sistema de ecuaciones lineales, requieren que el sistema satisfaga ciertas hipótesis. Si bien el hecho de restringir el dominio de los métodos puede parecer algo negativo, trae mejoras importantes en términos de velocidad y complejidad asintótica. Tomamos la decisión de realizar los métodos de esta manera ya que los sistemas de ecuaciones que resolveremos siempre cumplirán las hipótesis necesarias, por lo que el dominio no se ve restringido dentro del marco de este problema.

3.3. Algoritmos optimizados

Realizamos una modificación en el algoritmo de Eliminación Gaussiana aprovechando que la matriz generada por el CMM es simétrica y definida positiva, lo que asegura que no habrá 0's en la diagonal en ningún paso de la Eliminación Gaussiana. Esto nos permite realizar el algoritmo sin tener necesidad de pivotear, reduciendo la cantidad de operaciones. También consideramos, en el algorítmo de EG, el caso en el cual el coeficiente $m_{ji} = 0$ en el cual no realizamos ninguna acción ya que en este punto la fila j ya se encuentra triangulada.

3.4. Estructuración del código

Para modelar el problema diseñamos el módulo Matriz. Fue diseñado utilizando el tipo Vector de la biblioteca estándar de C++. El diseño de la matriz es muy simple, un vector de vectores fila, y dos enteros que representan la cantidad de filas y columnas de la misma.

Utilizamos el mecanismo de clases del lenguaje para mejorar la organización del código, proporcionado métodos que implementan los algoritmos principales : Eliminación Gaussiana, Factorización de Cholesky.

También implementamos funciones que nos permitieron modificar los valores de la matriz, resolver sistemas con matrices triangulares superiores o inferiores, trasponer matrices y demás.

Como el sistema tenía características particulares, realizamos optimizaciones ya aclaradas.

4. Experimentos cuantitativos

En esta sección del informe no vamos a concentrarnos en estudiar la correctitud o la calidad de los distintos métodos de rankeo utilizados, sino que vamos a centrarnos en el estudio de los costos temporales que conlleva la resolución de los distintos sistemas de ecuaciones lineales utilizando el Método de eliminación Gaussiana y el Método de Cholesky.

Para comparar la eficiencia entre algoritmos, resulta una buena medida comparar sus complejidades asintóticas. No es la única manera ya que un algoritmo puede tener mejor rendimiento que otro aunque tengan la misma complejidad.

Esto motiva a realizar un análisis detallado y bajo el marco del contexto de uso de nuestro algoritmos.

Dividiremos la experimentación cuantitativa en dos partes: Por un lado registraremos el tiempo que demoran los dos métodos en resolver distintos sistemas de ecuaciones variando el tamaño de la matriz. Por el otro lado, variaremos el término independiente dejando sin alterar la matriz y de nuevo compararemos los tiempos de cómputo.

4.1. Tiempo en función del tamaño de la matriz

Para este experimento evaluaremos la variación del tiempo de cómputo en la resolución del sistema caracterizado por una matriz simétrica definida positiva (sdp) con los dos métodos de resolución de sistemas de ecuaciones lineales que utilizamos, Eliminación Gaussiana y el método de Cholesky.

Para realizar las mediciones creamos matrices de Colley aleatorias de distintos tamaños y calculamos el tiempo de resolución del sistema lineal Ax=b.

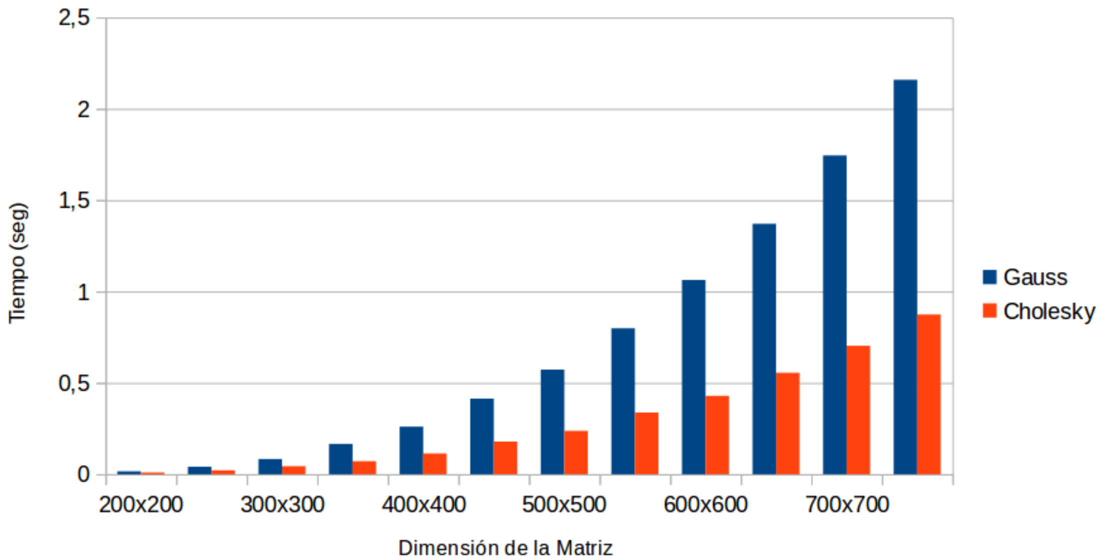
La forma de medir estos tiempos de cómputo son mediante los ticks de reloj que pasaron entre que empezó a resolver el sistema lineal y terminó de resolverlo. Este valor lo transformamos a milisegundos y obtenemos el tiempo que tardó el programa.

Los ticks del reloj pueden verse alterados fácilmente si, en el momento de la ejecución de nuestros algoritmos, el sistema operativo está realizando otras tareas. Estas tareas pueden añadir más ticks de clock a nuestras mediciones generando mediciones atípicas. Estas alteraciones son más notorias en matrices chicas donde el tiempo de cómputo es pequeño y la varianza de las mediciones es muy alta. Teniendo en cuenta esta consideración vamos a realizar las mediciones en matrices de un tamaño mayor en donde los casos atípicos son menos influyentes. A su vez vamos a repetir varias veces los experimentos promediando los resultados para obtener aproximaciones más precisas.

La forma de crear las matrices aleatorias de modo que sean SDP es creando competencias artificiales mediante el script llamado $randomfile.py$. La forma de utilizar este script es la siguiente: $pythomrandomfile.py\ e\ p$ donde e es la cantidad de equipos del torneo y p es la cantidad de partidos a jugar en dicho torneo.

Una vez creado la competencia consideramos la matriz de Colley que queda definida para así obtener nuestra matriz SDP utilizados para los casos de prueba. Para el siguiente gráfico utilizamos los archivos ubicados en la ruta $./Experimento/matricesRandom/MuchosPartidos/..$: utilizando el archivo $mainMismaA.cpp$

Hechas las mediciones obtenemos los siguientes resultados



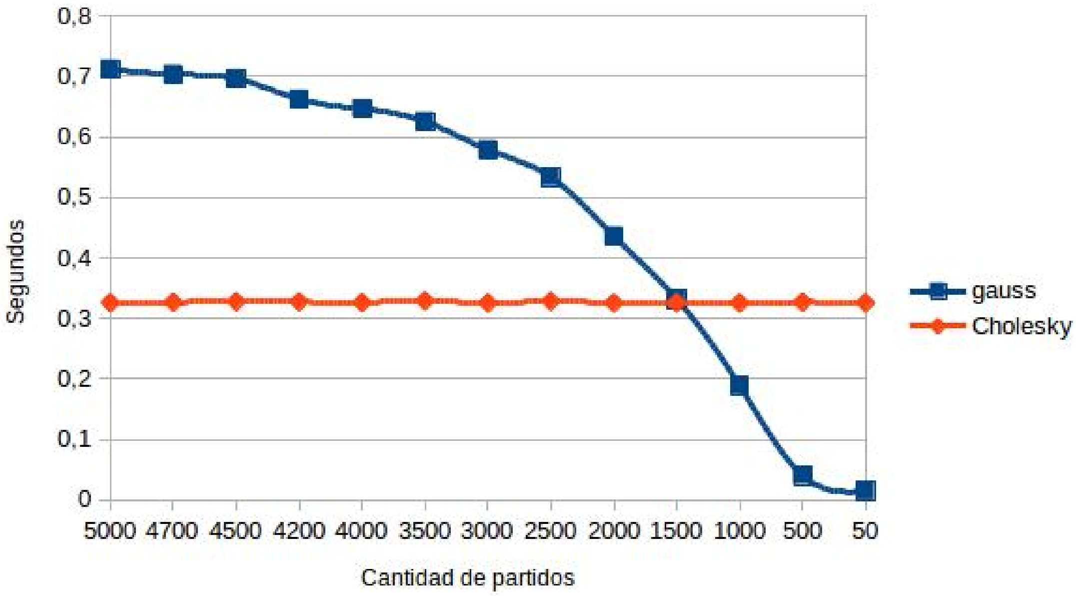
En el gráfico anterior podemos observar claramente una amplia diferencia a favor de la factorización de Cholesky.Esto se debe a que, si bien ambos son algoritmos de orden cúbico, el método de Cholesky es mucho más eficiente porque utiliza que la matriz es SDP y resuelve un sistema de ecuación de la forma $LL^t = A$ donde L es una matriz triangular superior.

Por otro lado, el método de eliminación Gaussiana no explota este resultado y previamente triangula la matriz (lo cual tiene un orden cúbico) y después resuelve una ecuación de la forma $A'x = b'$ donde A' es una matriz triangular superior equivalente a la A original agregando una complejidad cuadrática al tiempo de cómputo.

En este punto de la experimentación nos encontramos tentados a concluir que el método Cholesky es superior al método de eliminación Gaussiana en cuanto a complejidad temporal y espacial. Sin embargo mediante la experimentación detectamos que existen casos bordes en los cuales Gauss supera ampliamente a Cholesky en tiempo de ejecución. Posteriormente descubrimos, comparando los resultados con otros equipos que esta mejora se debe a nuestra implementación del método de eliminación Gaussiana.En nuestro código se aprovechan las matrices con muchos ceros pues si la posición $A_{ji} = 0$ entonces el multiplicador $M_{ji} = -\frac{A_{ji}}{A_{ii}} = \frac{0}{A_{ii}} = 0$ entonces la fila $j-ésima$ de A no se ve modificada (pues ya está triangulada) y se avanza a la siguiente fila evitando de este modo realizar operaciones innecesarias.

Para reflejar el resultado observado anteriormente generamos un torneo en donde participen 500 equipos y vamos decrementando la cantidad de partidos que se juegan en total en el torneo. Esto va a provocar que, eventualmente, aparezcan $A_{ii} = 0$ pues va a existir un caso en el cual el equipo i no va a poder jugar con el equipo j generando de este modo ceros en nuestra matriz y consecuentemente aumentando la eficiencia de Gauss. Por otro lado, como el método de Cholesky no depende de la cantidad de ceros sino que depende del tamaño de la matriz, por ende su coste temporal va a ser constante (*+/- errores de medición*).

Hechas las mediciones los resultados observados son los siguientes.



En este gráfico podemos observar que el tiempo de cómputo mejora notablemente en el caso de EG a medida que genero mas 0's en la matriz mientras que el tiempo de cómputo del método de Cholesky se mantiene constante poniendo en evidencia lo mencionado anteriormente.

La matriz utilizada para los experimentos se encuentra en

./Experimento/matricesRandom/MuchosPartidosApocos/500x5000p.dat y el cpp utilizado para las mediciones es ExpGaussCholoVariarPartidos.cpp. Este experimento puede ser fácilmente repetido utilizando el generador de torneos random randomfile.py.

4.2. Tiempo de cómputo variando el término independiente

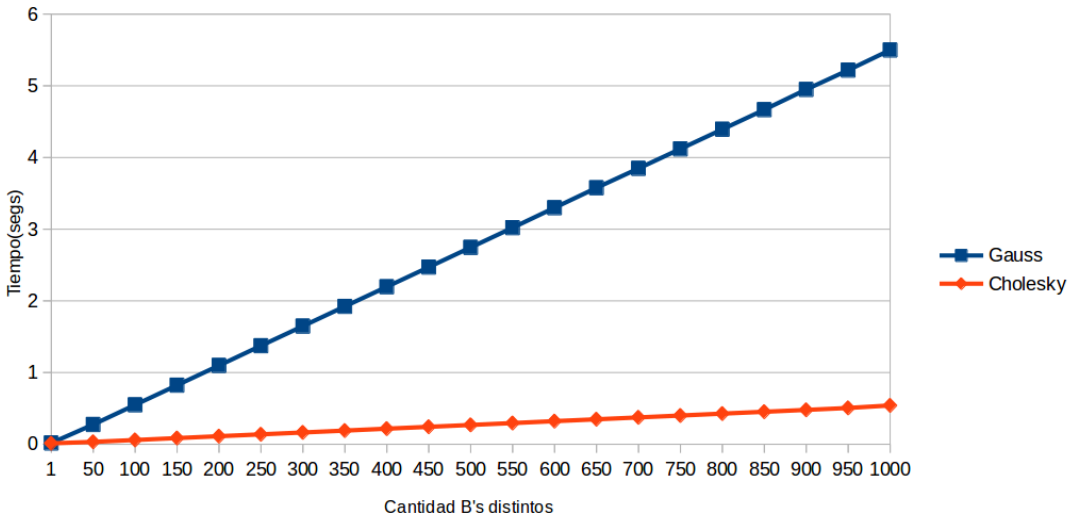
En esta sección del experimento vamos a tratar de observar la diferencia (en caso de exista dicha diferencia) entre el método de EG y el método de Cholesky a la hora de resolver distintos sistemas de ecuaciones lineales en las cuales se fija la matriz A y se varía el término independiente b.

Como en este contexto no nos interesa el valor específico del vector b pues lo que evaluamos es el aspecto temporal, vamos a crear distintos términos independientes que no serán necesariamente consistentes con un vector de Colley y vamos a resolver el sistema lineal $\{Ax_1 = b_1, Ax_2 = b_2, ..., Ax_n = b_n\}$ siendo n va a ser la cantidad de sistemas lineales distintos que generados y resueltos con cada uno de los métodos.

Así como en la sección anterior, para obtener el tiempo de resolución de un $Ax_i = b_i$ no alcanza con realizar una medición y tomar el resultado. Es por eso que vamos a realizar distintas mediciones para resolver cada sistema y vamos a graficar el tiempo promedio utilizado para resolverlos.

Lo que esperamos ver en esta parte del experimento es una amplia mejora a favor del método de Cholesky considerando que el método de EG resuelve siempre el sistema $Ax_i = b_i$ lo cual es cúbico pero con Cholesky calculamos la matriz triangular superior L una vez (orden cúbico) y luego, utilizando backward substitution y forward substitution, $Ax_i = b_i \equiv (LL^t)x_i = b_i \equiv Ly = b \wedge L^t x = y$ y resolver estas dos últimas ecuaciones es solamente cuadrático.

El siguiente gráfico muestra los experimentos realizados para una matriz de 128x128 en donde el término independiente aumenta de tamaño en 50.



El archivo utilizado para esta experimentación es mainMismaA.cpp y la matriz es una genérica de 128x128.

Como muestra el gráfico anterior y como sugiere la intuición, en el caso de variar el término independiente, es mucho más conveniente usar Cholesky. Si bien al principio pagamos un orden cúbico al calcular la matriz L ese costo se ve amortizado a medida que vamos aumentando la cantidad de los términos independientes distintos y los cálculos necesarios para encontrar la solución del sistema se reducen a la resolución de dos sistemas, una triangular superior y otra triangular inferior, que tienen ambos un coste cuadrático. Esta mejora no es compartida por el método de Gauss ya que con cada nuevo término independiente se recalcula el sistema y cada vez tarda un orden cúbico. Este coste puede verse amortizado si la matriz tiene muchos ceros pero en el caso general el coste es siempre cúbico y mucho mayor que el coste de Cholesky.

5. Experimentos Cualitativos

La realización de este trabajo práctico se vio motivada por el hecho de analizar distintos sistemas de rankeo de equipos, intentando encontrar el más justo. En esta sección del informe estudiaremos diferentes métodos con el objetivo de evaluar su comportamiento en distintas situaciones, explicando en qué casos es favorable utilizar uno por sobre otros y en qué casos podría fallar.

5.1. Experimento: Comparación entre el método Colley y WP

Presentación

Para una misma entrada observaremos cómo funcionan ambos métodos a la hora de clasificarlos equipos y compararemos los rankings generados.

El experimento se realizó en tres etapas, primero se experimentó sobre una entrada pequeña, se la procesó registrando ambos rankings. En segundo lugar se modificó la instancia anterior agregando un equipo con sus respectivos partidos y se repitió el procedimiento. Finalmente trabajamos con datos reales y comparamos los datos de salida con los resultados de los dos experimentos anteriores con la finalidad de hallar el comportamiento de cada método.

Datos de entrada

Para los dos primeros casos se generaron las entradas de manera artificial, se creo un torneo con 5 equipos, 10 partidos y en el que la cantidad de partidos jugados no fue el mismo para cada equipo. Luego para el caso 2 se agregó un sexto equipo con un partido solo ganándole al equipo peor rankeado del experimento anterior. Finalmente se tomaron datos del Australian Open 2012, y con éstos se generaron los rankings utilizando los métodos ya mencionados
Los datos de entrada del Ausopen pueden ser encontrados en:
./data/Entradas/ausopen12.dat
Los resultados de los 2 metodos al ausopen12:
./data/rankings/ausopen12Colley
./data/rankings/ausopen12WP
Datos de la primera instancia:
./data/Experimento51_Entrada_instancia_1.dat
Datos de la segunda instancia:
./data/Experimento51_Entrada_instancia_2.dat

Hipótesis

El método de Colley es más justo que el método WP ya que tiene más factores en cuenta, como por ejemplo el rival contra quien jugó el equipo. Mientras que Wp sólo cuenta los partidos ganados y jugados, generando que en ciertos casos borde, el equipo mejor rankeado no sea aquel con mejor desempeño.

Resultados

Primera instancia

Rank	Equipo	Rating
1	5	75
2	2	60
3	1	40
4	4	33.3333
5	3	33.3333

Método Wp

Rank	Equipo	Rating
1	5	65.4751
2	2	55.9266
3	1	47.413
4	4	42.0978
5	3	39.0875

Método Colley

Segunda instancia

Rank	Equipo	Rating
1	6	100
2	5	75
3	2	60
4	1	40
5	4	33.3333
6	3	25

Método de Wp

Rank	Equipo	Rating
1	5	64.1438
2	6	64.0152
3	2	52.6372
4	1	45.0901
5	4	42.0455
6	3	32.0683

Método Colley

Instancia real

Rank	Jugador	Rating
1	Darcis S.	100
2	Djokovic N.	100
3	Nadal R.	85.7143
4	Federer R.	83.3333
5	Murray A.	83.3333
6	Berdych T.	80
7	Ferrer D.	80
8	Nishikori K.	80
9	Del Potro J.M.	80
10	Tomic B.	75

Método de Wp

Rank	Jugador	Rating
1	Djokovic N.	109.942
2	Nadal R.	98.3808
3	Murray A.	92.2056
4	Federer R.	90.405
5	Ferrer D.	85.4608
6	Berdych T.	83.5251
7	Nishikori K.	82.4913
8	Del Potro J.M.	81.497
9	Hewitt L.	78.1207
10	Lopez F.	75.8829

Método Colley

Discusión

Se puede apreciar que en ciertos casos el método Wp no genera un ranking justo ya que sólo considera la cantidad de victorias y la cantidad de partidos, posibilitando así casos borde como se ve en el ejemplo del Atp, en el cual un jugador gana su partido de primera ronda y luego se retira. A causa de la decisión tomada por la ATP de no considerar su segundo encuentro como derrota posibilita que el método WP le asigne un 100 por ciento de efectividad generando así un ranking en el cual dicho jugador se encuentra primero junto con el campeón del torneo (quien es el verdadero merecedor de la cima del ranking). En conclusión el método de Colley genera un ranking más verídico ya que tiene más factores en cuenta a la hora de calcular las posiciones minimizando de este modo la posibilidad de casos bordes similares al estudiado anteriormente.

5.2. Experimento: CMM Vs WP

Presentación

En este experimento volveremos a comparar ambos métodos pero esta vez buscaremos hallar qué factores tienen en consideración para armar los respectivos rankings. En particular estudiaremos si simplemente consideran la cantidad de partidos jugados o si por el contrario se tienen en cuenta factores más complejos como el rendimiento de los otros equipos del torneo.

Modificaremos los resultados de varios partidos de manera tal que se mantenga igual la cantidad de partidos ganados y perdidos por equipo, lo que cambiará serán sus rivales.

Datos de Entrada

Creamos un pequeño escenario que involucra a 4 equipos y la distribución de los partidos esta dada de la siguiente manera:

Equipo	Score	Equipo	Score
1	1	2	0
1	1	2	0
1	1	2	0
1	1	2	0
1	1	2	0
1	1	2	0
1	1	2	0
1	1	2	0
3	1	1	0
3	1	1	0
3	1	1	0
4	1	3	0
4	1	3	0

Hipótesis

Nuestra hipótesis es que al rankear equipos utilizando el método Colley influye más la calidad de las victorias (entendiendo calidad como el rating del rival que fue vencido) que la cantidad total de victorias y derrotas a diferencia del método WP en el cual la relación $\frac{\#ganados}{\#jugados}$ es el único parámetro que toma en consideración a la hora de rankear.

Resultados

Ranking Original de los 4 equipos

Rank	Equipo	Rating
1	4	82.3529
2	3	64.7059
3	1	46.0784
4	2	6.86275

Método Colley

Rank	Equipo	Rating
1	4	100
2	1	72.7273
3	3	60
4	2	0

Método WP

A continuación agregamos el equipo número 5 con un nuevo encuentro victorioso en el cual vamos a ir modificando el contrincante.

Una victoria para el equipo 5 frente al equipo 2 que se encontraba último en la tabla original

Rank	Equipo	Rating
1	4	82.197
2	3	64.3939
3	5	51.9886
4	1	45.4545
5	2	5.96591

Método Colley

Rank	Equipo	Rating
1	5	100
2	4	100
3	1	72.7273
4	3	60
5	2	0

Método WP

Una victoria para el equipo 5 frente al equipo 1 que se encontraba anteúltimo en la tabla original

Rank	Equipo	Rating
1	4	81.0212
2	5	63.5838
3	3	62.0424
4	1	40.7514
5	2	2.60116

Método Colley

Rank	Equipo	Rating
1	5	100
2	4	100
3	1	66.6667
4	3	60
5	2	0

Método WP

Una victoria para el equipo 5 frente al equipo 3 que se encontraba segundo en la tabla original

Rank	Equipo	Rating
1	4	78.2857
2	5	68.8571
3	3	56.5714
4	1	42.381
5	2	3.90476

Método Colley

Rank	Equipo	Rating
1	5	100
2	4	100
3	1	72.7273
4	3	50
5	2	0

Método WP

Una victoria para el equipo 5 frente al equipo 4 que es el puntero de la tabla original

Rank	Equipo	Rating
1	5	72.8261
2	4	68.4783
3	3	59.7826
4	1	43.8406
5	2	5.07246

Método Colley

Rank	Equipo	Rating
1	5	100
2	1	72.7273
3	4	66.6667
4	3	60
5	2	0

Método WP

Discusión

Analizando los rankings generados por ambos métodos es inmediato observar que la nueva tabla generada por método de Colley utiliza en gran medida el nivel del contrincante derrotado a la hora de clasificar un nuevo equipo. Esto se puede apreciar observando que al agregar el equipo número 5 con victorias contra distintos equipos, ubicados en distintos lugares de la tabla, en los casos en los cuales dicha victoria fue contra equipos mejor rankeados, el equipo número 5 se ubica en puestos más favorables de la tabla. Por el contrario, si las victorias fueron contra equipos peor posicionados el equipo número 5 no alcanza posiciones demsiado elevadas en la tabla de posiciones debido al poco peso de estas victorias. Por otro lado, en el método WP en el cual se le asigna un mismo peso a una victoria sin importar si el contrincante estaba primero o último en la tabla de posiciones, se le asigna el primer puesto en todas las tablas (debido a que ganó dicho encuentro) provocando de este modo que pueda ubicarse en la punta del ranking sin realizar ningún mérito.

Si observamos con atención los rankings que devuelven Colley y WP podemos ver también una clara diferencia entre ambas. El equipo 1 (que fue el que jugó más partidos) ganó la gran mayoría permitiendole de este modo alcanzar un puesto alto en el ranking aún cuando esas victorias fueron frente a equipos pésimamente rankeados. Este comportamiento no se observa en el método de Colley en donde esas victorias carecen de valor condenando al equipo 1 a una posición poco favorable en la tabla.

Conclusión

Gracias al análisis realizado previamente podemos concluir que para el método de Colley si bien se tiene en cuenta la cantidad de victorias, se pondera la calidad de dichas victorias por sobre la cantidad permitiendo de este modo crear rankings más precisos e intuitivos que los creados por el método WP en donde se tiene en cuenta solamente el *hit-rate* a la hora de rankear.

5.3. Experimento: Diferencia entre enfrentarse a distintos contrincantes

Presentación

En este experimento volveremos a comparar ambos métodos pero esta vez buscaremos hallar qué factores tienen en consideración para armar los respectivos rankings. En particular estudiaremos si simplemente consideran la cantidad de partidos jugados o si por el contrario se tienen en cuenta factores más complejos como el rendimiento de los otros equipos del torneo.

Modificaremos los resultados de varios partidos de manera tal que se mantenga igual la cantidad de partidos ganados y perdidos por equipo, lo que cambiará serán sus rivales.

Datos de Entrada

Creamos un pequeño escenario que involucra a 5 equipos y la distribución de los partidos está dada de la siguiente manera:

Resultados torneo 1

	Equipo 1		Equipo 2		Equipo 3		Equipo 4		Equipo 5	
Equipo 1			2G	2P	0G	2P	1G	2P	3G	1P
Equipo 2	2G	2P			1G	1P	2G	0P	2G	0P
Equipo 3	2G	0P	1G	1P			3G	0P	1G	1P
Equipo 4	2G	1P	0G	2P	0G	3P			3G	2P
Equipo 5	1G	3P	0G	2P	1G	1P	2G	3P		

Resultados torneo 1 modificado

	Equipo 1		Equipo 2		Equipo 3		Equipo 4		Equipo 5	
Equipo 1			3G	3P	1G	1P	0G	1P	2G	2P
Equipo 2	3G	3P			0G	0P	2G	0P	2G	0P
Equipo 3	1G	1P	0G	0P			5G	0P	1G	1P
Equipo 4	1G	0P	0G	2P	0G	5P			4G	1P
Equipo 5	2G	2P	0G	2P	1G	1P	1G	4P		

Hipótesis

Suponemos que WP generará el mismo ranking para ambos torneos, ya que solo toma en consideración $\frac{\#ganados}{\#jugados}$ a la hora de rankear. Por el otro lado, Colley tiene más cosas en cuenta, tales como el rating del rival vencido, por lo que suponemos que generará distintos rankings.

Resultados

Ranking de los 5 equipos

Rank	Equipo	Rating
1	3	0.653837
2	2	0.616255
3	1	0.483813
4	4	0.410799
5	5	0.335295

Método Colley

Rank	Equipo	Rating
1	3	0.777778
2	2	0.700000
3	1	0.461538
4	4	0.384615
5	5	0.307692

Método WP

Ranking - Torneo modificado

Rank	Equipo	Rating
1	3	0.698456
2	2	0.647193
3	1	0.456756
4	4	0.381473
5	5	0.316121

Método Colley

Rank	Equipo	Rating
1	3	0.777778
2	2	0.700000
3	1	0.461538
4	4	0.384615
5	5	0.307692

Método WP

Discusión de los resultados

Como supusimos, los dos rankings generados por WP son iguales, por la razón previamente explicada. Tambien acertamos en cuanto a Colley, como se puede ver, los ratings de todos los equipos se modificaron, debido a que en la ecuación del CMM, al ganar se suma el rating de todos los oponentes de un equipo, por ende cuanto mayor es dicha suma, mayor es el rating.

5.4. Experimento: Subir al puesto numero uno

Presentación

El objetivo de este experimento es dado una competencia y un equipo de dicha competencia hallar una estrategia para determinar la mayor posición posible, buscando minimizar el número de partidos ganados.

Para cumplir dicho objetivo desarrollamos una estrategia que consiste en quitarle todos los partidos al equipo seleccionado, calcular el ranking sin estos partidos y comenzar a agregarle partidos respetando el fixture original, pero modificando los resultados con la finalidad de alcanzar el primer puesto en el ranking al final de la competencia.

Para decidir el resultado en cada partido calculamos el estado del ranking actual y si nuestro equipo no se encuentra en el primer puesto agregamos una victoria con el contrincante mejor rankeado con el objetivo de escalar rápidamente a puestos más altos. Una vez que el equipo elegido se encuentre en la primera posición del ranking (como se busca minimizar la cantidad de victorias) se busca agregar una derrota contra equipos rankeados en el primer tercio de la tabla (preferentemente que no sea el segundo en la tabla de posiciones) logrando de este modo un descenso menos significativo en la tabla. Si no se puede lograr una derrota en el primer tercio del ranking se agrega una derrota contra el segundo equipo en la tabla en caso de ser posible. Si no es posible dicha derrota, se sigue buscando una derrota pero esta vez en el segundo tercio de la tabla. Si todo este análisis falla solamente agrego una victoria contra el primer equipo con el que se pueda jugar.

Esto se repite hasta que nuestro equipo haya jugado la cantidad de partidos que había jugado originalmente contra cada uno de los otros equipos de la competición.

Datos de entrada

El experimento fue dividido en dos etapas, en la primera se utilizaron datos creados artificialmente de una manera aleatoria[1] con la finalidad de poder verificar rápido si la estrategia era la adecuada. Luego se aplica esta estrategia sobre datos de competencias reales.

Dado que las competencias creadas artificialmente no capturan de una manera precisa el comportamiento de competencias reales (en particular nuestro generador de torneos crea una competencia en la cual las victorias presentan una distribución uniforme) decidimos trabajar sobre instancias reales en donde se podrá sacar un mayor provecho al análisis del comportamiento de nuestro algoritmo.

Hipótesis

La cantidad de partidos que debe ganar un equipo para alcanzar la punta está directamente relacionada con la performance del actual puntero del torneo.

Resultados

En los siguientes cuadros vamos a mostrar los resultados arrojados por nuestro algoritmo al aplicarlo sobre datos de la Nba 2014. Para estudiar el comportamiento del método vamos a aplicar este algoritmo para distintos equipos en distintas posiciones en el ranking capturando de este modo un comportamiento más general del mismo.

[1] Para crear competencias aleatorias usamos el archivo *randomfile.py*

Nba 2014

El puntero en esta competencia tenía 62 victorias					
Equipo	Ranking original	Ganados original	Ranking nuevo	Ganados nuevo	Diferencia ganados
22	28	23	1	65	42
19	20	34	1	65	28
30	15	44	1	74	30
7	10	49	1	74	25
11	4	56	1	74	18

Resultados al aplicar la estrategia sobre distintos equipos participantes

Nba 2014

Al puntero en este caso le agregamos 3 victorias ficticias otorgándole un total de 65 en la competencia					
Equipo	Ranking original	Ganados original	Ranking nuevo	Ganados nuevo	Diferencia ganados
22	28	23	1	74	51
19	20	37	1	74	37
30	15	44	1	74	30
7	10	49	1	74	25
11	4	56	1	74	18

Resultados al aplicar la estrategia sobre distintos equipos participantes

Discusión

Analizando los cuadros anteriores se puede observar que si bien muestran algunos incrementos en la cantidad de partidos ganados (por ejemplo para el caso del equipo 22 y el equipo 19), a medida que agregamos victorias ficticias al puntero con el objetivo de aumentar su desempeño en el torneo, no son lo suficientemente significativos para sacar ninguna conclusión.

En este momento del experimento decidimos seguir aumentando la cantidad de victorias del equipo puntero con la esperanza de observar un incremento más significativo en las victorias pero no observamos mayores cambios. Ante esta situación decidimos no agregar más tablas comparativas y tratar de encontrar el motivo detrás de este fenómeno.

A estas alturas del experimento llegamos a considerar que quizás nuestro algoritmo podría estar funcionando de una manera inesperada arrojando estos resultados pero finalmente logramos descubrir que la razón por la cual no aumentaban las victorias de manera significativa se debe al hecho de que las victorias agregadas son contra equipos ubicados en un lugar muy bajo de la tabla. Al agregar estas victorias en el historial del equipo puntero no afectan su desempeño de una manera muy significativa (dado el poco peso de estas victorias) provocando que la cantidad de victorias que se necesiten para lograr el primer lugar de la tabla no varíen tanto como nosotros esperábamos.

Manera distinta de encarar el problema

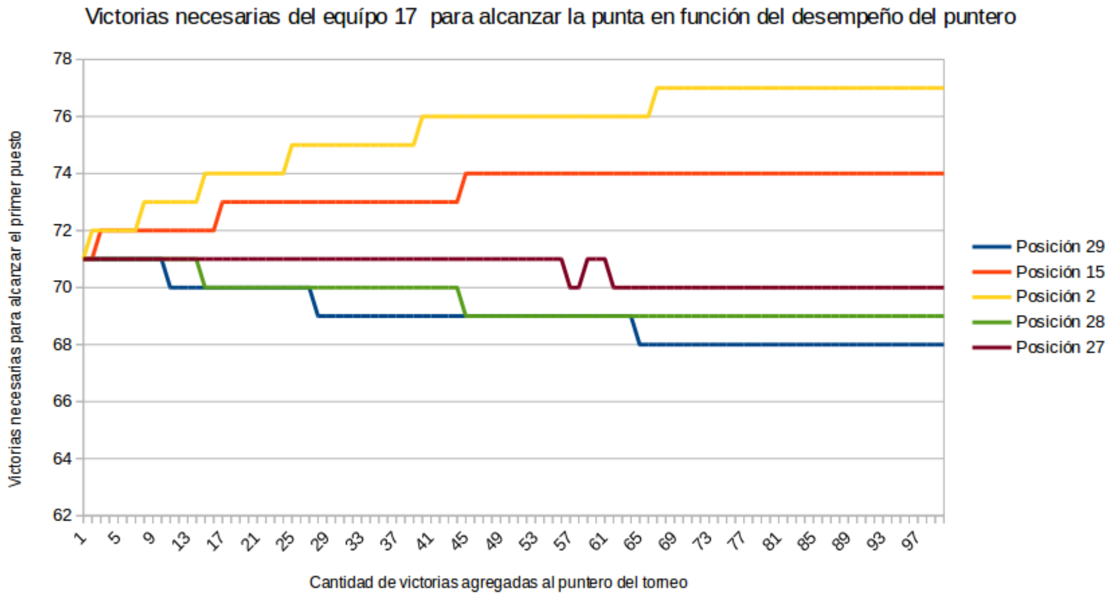
Teniendo en cuenta en análisis realizado previamente decidimos que un cuadro no es una buena manera de representar nuestros hallazgos. También decidimos que tampoco es conveniente agregar victorias contra equipos ubicados en el fondo de la tabla solamente ya que en estos casos el rendimiento del equipo puntero no varía de manera significativa. Es por este motivo que decidimos proceder de la siguiente manera: Trabajaremos con los mismos datos de entrada (en este caso los datos correspondientes a la Nba del año 2014) y para tratar de verificar o refutar nuestra hipótesis original[2] agregaremos victorias ficticias a favor del equipo puntero contra equipos ubicados en distintos lugares de la tabla. Lo que esperamos observar es que la cantidad de victorias necesarias para alcanzar la cima no varíen de manera significativa en el caso de agregar victorias frente a los equipos peor rankeados y en el caso de agregar victorias frente a equipos ubicados en lugares favorables de la tabla esperamos ver un incremento significativo dicha cantidad de victorias.

Nuevos Resultados

Para el siguiente gráfico decidimos tomar al equipo que se encuentra último en la tabla de pociones y llevarlo al primer puesto[3]. Para tratar de observar como varía la cantidad de victorias que se necesita para dicho objetivo agregamos victorias al equipo puntero contra equipos ubicados en distintos lugares de la tabla. Los resultados que se obtuvieron fueron los siguientes.

[2]Nuestra Hipótesis Original consistía en estudiar si el rendimiento del equipo puntero afectaba de manera significativa a la cantidad de partidos que necesita un equipo para alcanzar el primer puesto.

[3]Subimos de puesto a distintos equipos, ubicados en distintos puestos de la tabla pero los resultados eran similares. Presentamos un gráfico pero el análisis para los demás equipos es similar.



Cada color representa a un equipo distinto, en una posición distinta en la tabla. Dichos equipos son los utilizados para agregar las victorias consideradas a la hora de subir el rendimiento del equipo puntero.

5.5. Discusión

Observando el cuadro anterior y teniendo en cuenta el análisis realizado previamente podemos observar dos factores importantes. El primero de ellos es que el desempeño de los equipos está relacionado fuertemente con el de sus oponentes. Las victorias contra equipos muy mal rankeados no aumentan de manera significativa mi liderazgo en el torneo posibilitando de este modo que un equipo (en este caso el último de la tabla) pueda alcanzar el primer lugar sin necesidad de muchas victorias. Por el contrario las victorias frente a equipos bien posicionados ayudan a afianzar mi liderazgo y obligan a los equipos que quieran arrebatar dicho liderazgo a un desempeño mucho mejor en el torneo. Esto último puede observarse en la cantidad de victorias que necesitó el equipo número 17 a la hora de alcanzar el primer puesto.

Se puede observar situaciones atípicas en los cuales la cantidad de victorias que son necesarias en ciertos momentos disminuye por un momento y luego vuelve a subir. Esto podría deberse a la particularidad del torneo y al fixture del mismo.

Conclusiones

Finalmente, luego de este análisis podemos concluir que la cantidad de partidos necesarios a la hora de alcanzar la cima del ranking se encuentra estrechamente relacionada con el desempeño del equipo líder del torneo. A su vez este último desempeño se ve también ligado al rendimiento de los equipos rivales dado que a la hora de afianzar el liderazgo de un equipo en un torneo son más decisivas las victorias frente a equipos mejor ubicados.

6. Conclusiones

La realización de este trabajo práctico nos permitió extraer las siguientes conclusiones sobre el rendimiento de los métodos y el comportamiento de los rankings , observadas a partir de los experimentos.

En cuanto al costo temporal de los métodos en la resolución de sistemas de ecuaciones lineales generados por el CMM, observamos que la factorización de Cholesky es mucho mejor que el método de Eliminación Gaussiana, corroborando nuestras hipótesis. La verdadera diferencia se observa al fijar la Matriz A e ir variando el término independiente, es mucho más rápido, pero no es un caso que vayamos a encontrar en la realidad por lo que ésta mejora en rendimiento no tendrá impacto en los casos reales.

Cabe notar, que como realizamos una modificación al algoritmo de EG será mucho mas conveniente en los torneos que registren muy pocos partidos, ya que esto se traduce en más ceros en la matriz, dónde se ve la mejora en EG, siendo aún más rápido que Cholesky.

Resumiendo, salvo el caso donde se jueguen muy pocos partidos, donde será conveniente utilizar EG, es mejor utilizar el método de Cholesky.

En cuanto a la comparación del método WP y CMM, concluimos que CMM es más justo. Cuando todos los equipos jugaron la misma cantidad de veces, no notamos diferencias sustanciales en la distribución de los equipos a lo largo de los rankings generados por ambos métodos.

Pero CMM resulta útil en el caso que dos equipos registren el mismo porcentaje de victorias para realizar un "desempate", ya que es probable que aunque hayan ganado la misma cantidad de veces, su rating quede distinto. Otro caso donde encontramos diferencias es cuando un equipo registra muchísimos menos partidos que el resto, en el método WP puede terminar primero en el ranking de manera muy fácil, lo cual no es justo, ya que el resto de lo equipos ha jugado muchos partidos más. Ese mismo equipo utilizando el CMM, podría encontrarse en una posición mucho mas baja.

También notamos diferencias al elegir un equipo y hacerlo subir de ranking. En el método WP basta hacerlo ganar suficientes veces para que el cociente entre los partidos ganados y los totales sea mayor que el del resto de los equipos. En cambio, en CMM es muy importante tener en cuenta a quien hay que ganarle para llegar a la punta minimizando la cantidad de partidos ganados.

Para finalizar, queremos destacar que este trabajo práctico fue muy útil para observar que es muy importante analizar bien las características de los datos y el contexto de uso de nuestro problema para optimizar el rendimiento de todos los algoritmos.

Unused "captionsetup[1] on input line AtEndDocument