ático512Gauss cuadráticofigure.caption.6 úbicoubico613Gauss cúbicofigure.caption.7 áticoatico714Cholesky cuadráticofigure.caption.8 úbicoubico814Cholesky cúbicofigure.caption.9

DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 1

"(No) Todo Pasa"

Metodos numericos
Primer Cuatrimestre de 2016

Integrante	LU	Correo electrónico
Leonardo Raed	579/04	leo_raed@yahoo.com
Ricardo Colombo	156/08	ricardogcolombo@gmail.com
Diego Santos	874/03	diego.h.santos@gmail.com
Luis Badell	246/13	luisbadell@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. Introducción teórica

Las Competencias deportivas , de cualquier indole requieren la comparacion de equipos mediante la confeccion de las tablas de Posiciones y rankings en basea los resultados obtenidos durante un cierto periodo de tiempo.

En este trabajo practico intentaremos modelar y resolver el problema de generar un ranking de equipos a partir de los resultados con la condicilon de que no haya empates. Para confenccionar dicho ranking haremos uso de 2 metodos diferentes. El primeroVes el Winning Porcentage y el 2 es el Colley Matrix Method (CMM). Con estos metodos es posible obtener 2 rankings y nos proponemos hallar la similitudes y diferencia entre ellos.

¿Por que es importante el ranking ?

Es importante porque determina quienes fueron los mejores y peores equipos al final de la temporada , quienes avanzan a la siguiente ronda ,en ciertas ligas como la liga de basquet y de baseball de los estados unidos determina la prioridad para los drafts. Ademas de eso tenemos la justicia deportiva y una gran cantidad de dinero en juego

¿Por que es importante el analisis de datos ?

Es importanta porque permite en las manos adecuadas, optimizar el rendimiento de los jugadores y ademas decidir si los jugadores son buenos en base a ciertos indicadores.

2. Desarrollo

2.1. Entrada y salida de los algoritmos

Dados los requirimientos de la catedra el programa toma como parametros 3 argumentos, el primero es el archivo de entrada, luego el archivo de salida y por ultimo el modo. La catedra solicitaba 3 modos el modo 0,1 y2 para los 3 metodos solicitados, luego agregamos 3 modos mas utilizados durante la etapa de expermentacion.

1. Eliminacion Gaussiana(EG)
2. Factorizacion de Cholesky(CL)
3. WP
4. Cholesky con modificacion de partidos jugados
5. Cholesky haciendo ganar al ultimo con el siguiente reiteradas veces hasta quedar primero
6. Cholesky haciendo ganar al ultimo con el primero del ranking hasta quedar primero

El modo 3 modo corre cholesky y luego busca 2 equipos que hayan jugado previamente para cambiar su resultado y luego volver a ejecutar cholesky, para este modo se puede ingresar un parametro mas donde se indica la cantidad de partidos que se deben jugar de nuevo.
Para el modo 4 corre cholesky y luego ejecuta un ciclo donde el objetivo es lograr que el que haya salido ultimo luego de obtener el ranking de cholesky llegue al primer puesto ganandole al que tiene por arriba inmediato en el ranking. En cada paso agrega un partido mas y vuelve a calcular cholesky para la nueva matriz, asi obtenemos un nuevo ranking y continuamos iterando hasta quedar en la primera posicion, siempre utilizando al que salio ultimo en la primera utilizacion del metodo de cholesky. Una vez finalizado por stdout devuelve la cantidad de partidos jugados, ademas de que en el archivo rankingSTEPS_4.out dentro de la carpeta test se encuentran los rankings en cada iteracion y como es la evolucion. El modo 5 es similar al anterior con la sutil diferencia que el participante que salio ultimo la primera vez que corrio cholesky llegue al primer puesto jugandole al que se encuentra en el primer puesto en cada iteracion.
Tanto el formato de entrada y de salida del programa son los solicitados por la catedra. Para todos los metodos el archivo de entrada es el mismo, que contiene el siguiente formato:

$$\begin{pmatrix} (n) & (k) & & & \\ (x1) & (e1) & (r1) & (t1) & (s1) \\ (x2) & (e2) & (r2) & (t2) & (s1) \\ \dots & & & & \\ (xk) & (ek) & (rk) & (tk) & (s1) \end{pmatrix}$$

la primer linea tiene 2 valores n representa la cantidad de equipos y k representa la cantidad de partidos, seguido k lineas que representa cada partido, donde $x1$ representa una fecha que en nuestro caso no utilizamos, luego ei y ti representan los numeros de los equipos, y por ultimo ri y $s1$ representan las anotaciones de cada equipo respectivamente.
en nuestra experimentacion con fines de no complejizar aum mas el problema no utilizamos en campo que representa la fecha si no que asumimos que dado el orden que venian los resultados era el orden de los partidos.

Luego una vez se ejecutan los metodos 0-4 devuelven un archivo de n lineas donde en la linea se obtiene el ranking del equipo i.

Para los metodos 4 y 5 se devuelve el ranking para cada iteracion antes de jugar el partido, estas se repiten hasta que el que comenzo ultimo termine primero utilizando dos metodos descriptos anteriormente, en la ultima linea se obtiene la cantidad total de partidos, estos dos metodos ademas devuelven por stdout la cantidad de partidos jugados hasta que termino en la primer posicion el que comenzo ultimo.

2.2. Sistema a resolver

$$C_{i,j} = \begin{cases} n_{i,j} & \text{si i} \neq \text{j} \\ 2 + n_i & \text{si i j} \end{cases}$$

Esta matriz es simetrica y definida positiva y ademas es estrictamente diagonal domimante. Por lo tanto es posible hallar la factorizacion de Cholesky , que no es mas que un caso de factorizacion LU , esto nos permite asegurar que a la hora de hacer eliminacion gaussiana no vamos a hallar 0 en la diagonal , con lo cual no hace faltar considerar pivoteo parcial en la eliminacion Gaussiana.

2.3. Método Matriz de Colley

Para la implementación de esta técnica nos basamos en el paper **The Colley Matrix Explained**. La cual consiste en plantear un sistema de ecuaciones

La principal fortaleza de este método es que es útil para obtener rankings en torneos donde los participantes no juegan la misma cantidad de partidos. Además de que al armar el sistema en base a los resultados pasados, da relevancia al calendario de juegos de cada participante. Como se intentará demostrar en la sección de experimentos, utilizando esta tecnica importa contra quien se gana y contra quien se pierde. Además tiene el atractivo de que da una idea sobre la posibilidad de victoria en el siguiente partido, considerando los partidos anteriores.

La principal desventaja y que hace que no sea aplicable a muchos de los deportes es que los empates no pueden ser modelados.

Una vez obtenida la **Matriz de Colley** vamos a presentar dos técnicas de resolucion del sistema de ecuaciones solicitado.

2.3.1. Eliminación Gaussiana

La implementación del algorimo de **Eliminación Gaussiana** que elegimos es la que se encuentra en el libro **Burden[1]**.Con el agregado de backgward substitution para obtener el vector de la ecuacion Cr=b.

Para este algoritmo como se puede observar es de complejidad O(n^3) en el peor caso, ya que en ciclo interno de las posiciones 4 a 6 se ejecuta n veces y el ciclo de las lineas 2 a 6 se ejecuta n veces por lo tanto ya tendriamos n^2 iteraciones en el peor caso y finalizando con el ciclo de las lineas 1 a 6 que se ejecuta otras n veces. Luego en las lineas 7 a 10 se realiza el backgward substitution que tiene en el peor caso se ejecuta n^2 veces.

TP1 1 vector Gauss(matriz A, vector b)

1: Para k=1...$n-1$
2: Para i=1...$n-1$
3: Tomo el elemento $a_{k,k}$ como pivot
4: Para $j = i+1, ...n$
5: $a_{i,j} = a_{i,j} - a_{i,j} * (a_{i,k}/a_{k,k})$
6: $b_i = b_i - a_{i,j} * (a_{i,k}/a_{k,k})$
7: $x_n = a_{n,n+1}/a_{n,n}$
8: para $i = n-1.,1$
9: para $j = i+1..n$
10: $sum+ = a_{i,j} * x_j$
11: devolver x

2.3.2. Cholesky

La implementación del algorimo de **Eliminación Gaussiana** que elegimos es la que se encuentra en el libro **Burden**. Agregandole los pasos que mensiona en el libro (pagina 420) para resolver el sistema Cr=b.

TP1 2 vector Cholesky(matriz A, vector b)

1: $l_{1,1} = \sqrt{a_{1,1}}$
2: Para $j = 2, ...n$
3: $l_{j,1} = a_{j,1}/l_{1,1}$
4: Para $i = 2, ...n-1$
5: $l_{i,i} = (a_{i,i} - \sum_{k=1}^{i-1} {l_{i,k}^2}^{1/2})$
6: Para $j = i+1, ...n1$
7: $l_{j,i} = (a_{j,i} - \sum_{k=1}^{i-1} l_{j,k} l_{i,k}/l_{i,i})$
8: $l_{n,n} = (a_{n,n} - \sum_{k=1}^{n-1} l_{n,k}^2$
9: $y_1 = b_1 l_{1,1}$
10: Para i=2..n
11: Para j=1..i-1
12: sum=$l_{i,j} * y_j$
13: $y_i = (b_i - sum) l_{i,i}$
14: $x_n = y_n l_{n,n}$
15: Para i=n-1..1
16: Para j=i+1..n
17: sum= $l_{i,j} * x_j$
18: $x_i = (y_i - sum) l_{i,i}$
19: devolver x

2.4. Porcentaje de Victorias

La primer técnica es **Porcentaje de Victorias** que a lo largo del análisis denominaremos **WP** que consiste en tomar el promedio de partidos ganados / partidos jugados. Esta técnica basicamente analiza la performance de un equipo participante en los partidos jugados en base a partidos ganados.

En este caso el score de un equipo no es afectado por la cantidad de partidos y resultados obtenidos de los demás participantes, pero esto si afecta su posición final en el ranking.

Esta técnica a priori no aporta mucha informacion respectoa la posibilidad de victoria en el siguiente encuentro y tampoco considera el ranking del rival enfrentado. Ya que todos los partidos valen lo mismo.

La implementación consiste en calcular: $\sum_{i=1}^{n} \frac{G_i}{T}$

Sonde **n** es la cantidad de partidos jugados, **G** corresponde a partidos ganados y **T** al total de partidos jugados.

3. Experimentación

Para analizar la efectividad y ecuanimidad de esta nueva forma de calcular el ranking vamos a realizar una serie de test a fin de obtener un analisis cuantitativo y cualitativo que nos permita compararlo con el clásico método de **WP**.
Con los test esperamos encontrar ventajas y desventajas de esta forma de medición, particularmente en escenarios donde no todos los participantes juegen la misma cantidad de partidos.

Además realizaremos una comparación de los métodos de **Eliminación Gaussiana** y **Cholesky** para ver cual de los dos computa los rankings de manera mas eficiente.

En esta sección solo presentaremos los experimentos realizados y los resultados obtenidos. Las conclusiones de cada experimento las presentaremos en la seguiente sección.

3.1. Ranking

Vamos a comparar la tabla de ranking obtenida a partir de un set de datos de la **ATP 2007**. Es decir calculamos el ranking a partir de la técnica **WP**, considerando partidos ganado / partidos jugados, a pesar de que no todos los jugadores hayan participado de la misma cantidad de partidos. Comparandolo con el **CMM** implementado con **Eliminación Gaussiana** y **Cholesky**.

WP- CMM todos.png

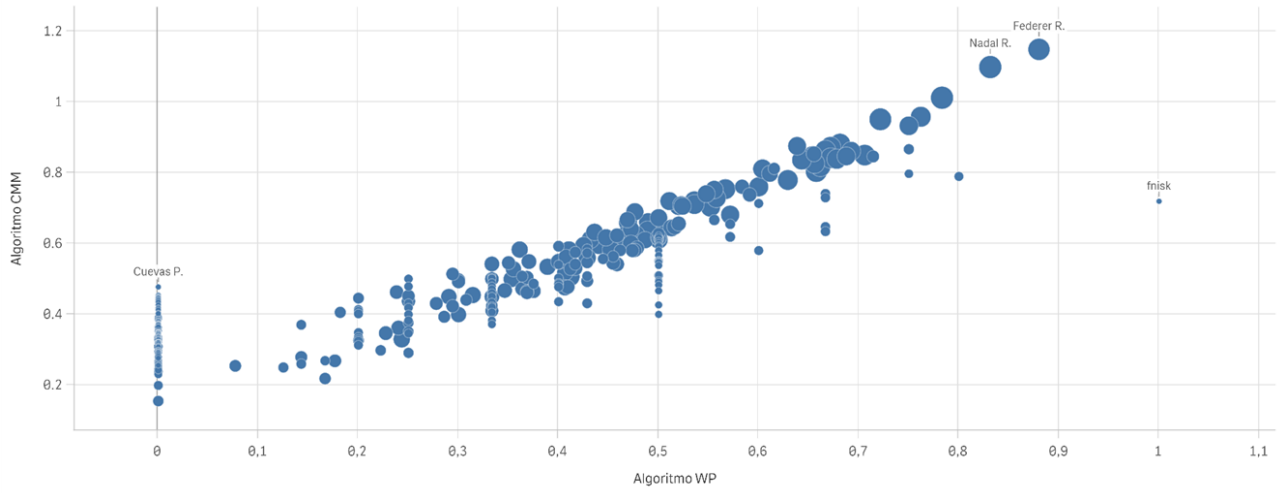

Figura 1: Rankings Calculados con las 3 tecnicas

Hemos realizado un zoom dentro del grafico para corroborar el nombre y posicion de ambos rankings.

WP - CMM zoom.png

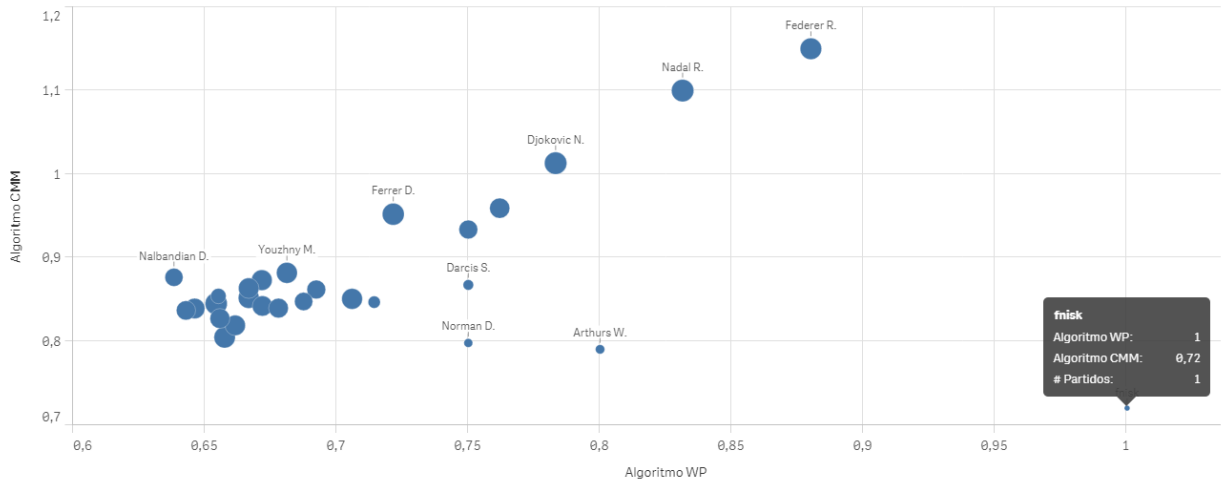

Figura 2: Zoom de las primeras posiciones

3.2. ¿Importa a quien se le gana?

En el escenario que se utilize **WP** realmente no importa a que equipo se le gane, ya que todos los partidos tienen la misma importancia y se les asigna el mismo puntaje. Pero en el caso de **CMM** resulta mas interesante plantearse esta pregunta.

La hipótesis que tenemos es que tomando un equipo de mitad de tabla, que denominamos **medio** el hecho de que le gane al lider de la tabla va a mejorar mucho mas el ranking que derrotando al que ocupe la última posición.

Realizamos un test tomando al equipo **medio**, y agregando un partido victorioso contra el puntero y analizamos como se modifica su ranking. Luego tomamos la tabla inicial, es decir sin ganarle al puntero, y repetimos el experimento esta vez derrotando al último.

Presentamos los resultados obtenidos.

******Aca van los graficos de: tabla inicial, perder contra el primero y perder contra el ultimo (sacando el partido con el primero)

3.3. ¿Importa contra quien se pierde?

Para verificar si importa contra que equipo se juega, proponemos el test de tomar un equipo que se encuentra por la mitad de la tabla. Por notación denominamos a este equipo como **medio**.

Para lograrlo calculamos un ranking a partir de un set de datos. Y luego generamos una nueva instancia enfrentando a **medio**contra el actual puntero y calculamos el nuevo ranking. Volvemos a tomar la primer instancia y lo enfrentamos contra el último, calculamos nuevamente el ranking y luego comparamos los tres rankings obtenidos. La premisa que tenemos del experimento es que el ranking del equipo **medio** no debería ser afectado por el rival contra el que perdió.

Este experimento lo calculamos usando la técnica de **CMM**, ya que considerar **WP** no afecta el resultado.

A continuación presentamos los graficos obtenidos.

******Aca van los graficos de: tabla inicial, perder contra el primero y perder contra el ultimo (sacando el partido con el primero)

3.4. Racha ganadora

Para estudiar la ecuanimidad del **CMM** realizamos un experimento tomando al participante del **ATP 2007** que se encontraba en el último puesto y le asignamos una racha ganadora contra los primeros diez jugadores del ranking.

Además este test nos permite observar como la racha de un jugador afecta al ranking global y si ganandole a los mejores realmente escala una considerable cantidad de posiciones en el ranking.

A continuación presentamos el ranking calculado construido de la siguiente manera:

- Eje X el valor obtenido al ejecutar CMM implementado con Cholesky.
- Eje Y el valor obtenido al ejecutar CMM implementado con Eliminación Gaussiana.
- El tamaño de la burbuja es la cantidad de partidos jugados.

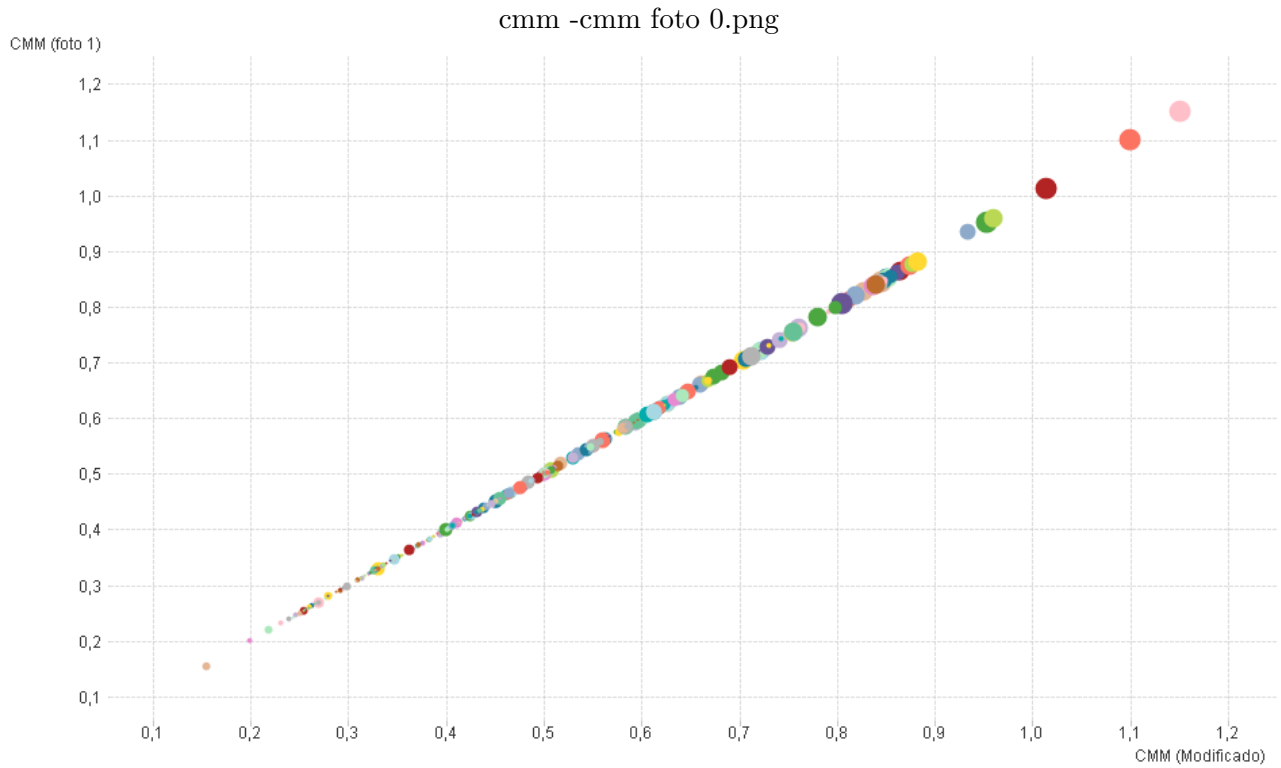

Figura 3: Zoom de las primeras posiciones

Y a continuación el ranking después de los 10 partidos:

En el gráfico se observa mediante una flecha cual es la ganancia en Ranking que tiene el ultimo jugador al ganarle a los top 10.

cmm -cmm foto 10.png

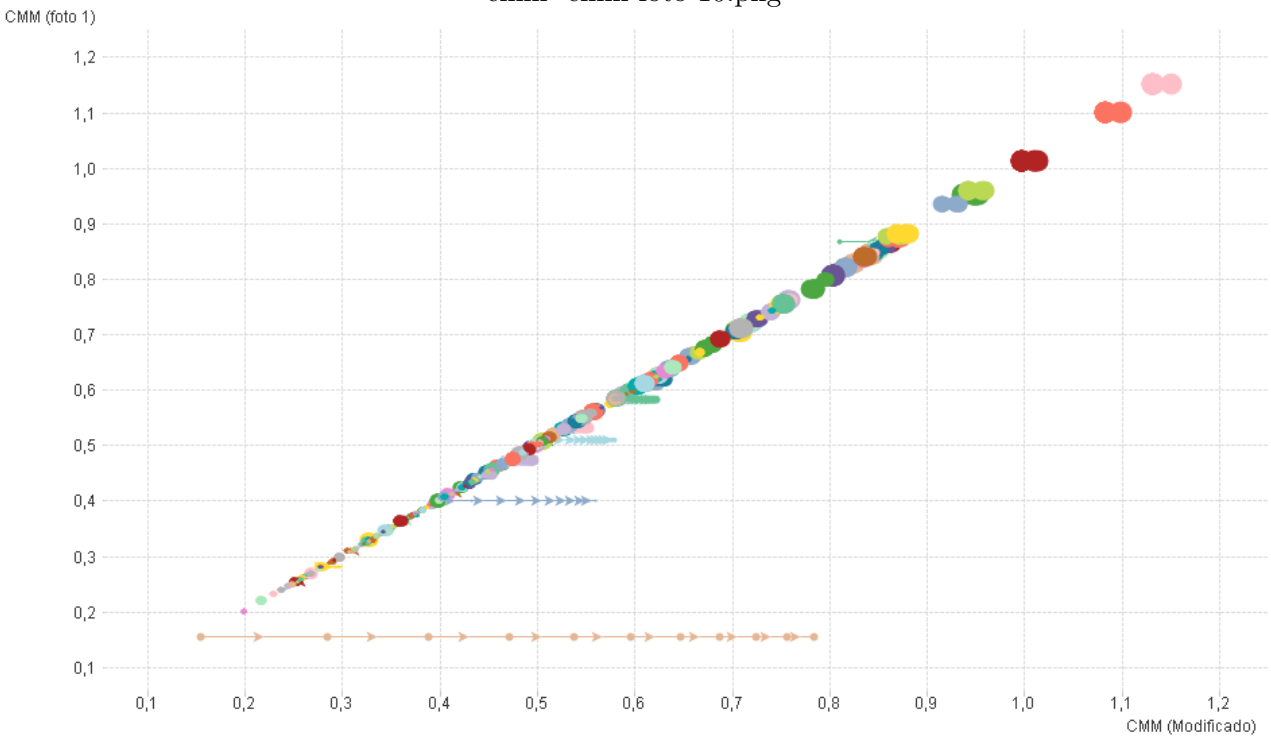

Figura 4: Zoom de las primeras posiciones

3.5. Escalando Posiciones

Una de las consignas del trabajo era encontrar una tecnica para hacer escalar en el ranking a un **equipo**, para lograr esto tenemos dos alternativas.

3.5.1. El torneo ya finalizo

En este escenario el torneo se encuentra finalizado y los resultados no pueden modificarse. Lo que proponemos es ver si modificando el orden de los partidos podemos influir en el ranking de un equipo. Para esto vamos a modificar el orden de sus victorias de forma tal de encontrar una que resulte en una mejoría de su ranking.

3.5.2. Agregando partidos

En este caso vamoa a analizar si podemos influir positivamente en el ranking a favor de un equipo, minimizando la cantidad de partidos ganados.

Nuestra teoría es que tomando el conjunto de equipos que perdio contra el seleccionado y haciendolos jugar y ganar a los principales del ranking, vamos a lograr que nuestro equipo mejore en la tabla de posiciones.

3.6. Análisis Cuantitativo

Vamos a estudiar la eficiencia de ambas tecnicas incrementando y variando los volúmenes de datos. La idea es repetir el cómputo de los rankings para la misma instancia de datos al azar, y posteriormente ir incrementando la cantidad de datos.

Nuestra hipótesis es el que método de basado en **WP** va a tardar lo mismo para instancias de datos iguales, y se irá incrementando de forma casi lineal a medida que incrementemos los datos. En cambio con **CMM** basando en **Eliminación Gaussiana** y **Cholesky** esperamos que difieran en para las mismas intancias. Nuestra hipótesis sobre esto es que la implementación de **Cholesky** va a demorar menos tiempo.

Para esta prueba se generaron schedules variando la cantidad de equipos en 6, 50, 100, 200, 300, 500, 700, 1000 y 2000.

Adicionalmente se varió la cantidad de partidos jugados por cada equipo. Dado el análisis de complejidad de los algoritmos implementados, solo varían el tiempo de calculo en funcion del tamaño de la matriz definida por la cantidad de equipos, por lo cual al variar la cantidad de partidos no esperamos encontrarnos con variaciones en el tiempo.

Ejecutamos los test para nuestra implementación de Eliminación Gaussiana.A continuación se muestran los resultados de tiempos de ejecucion dependiendo la cantidad de equipos. Para evidenciar la complejidad cubica del algoritmo lo hemos encerrado entre 2 funciones cuadraticas que evidencian que no pueden contener la curva de tiempos de nuestro algoritmo.

cuadrativo.png

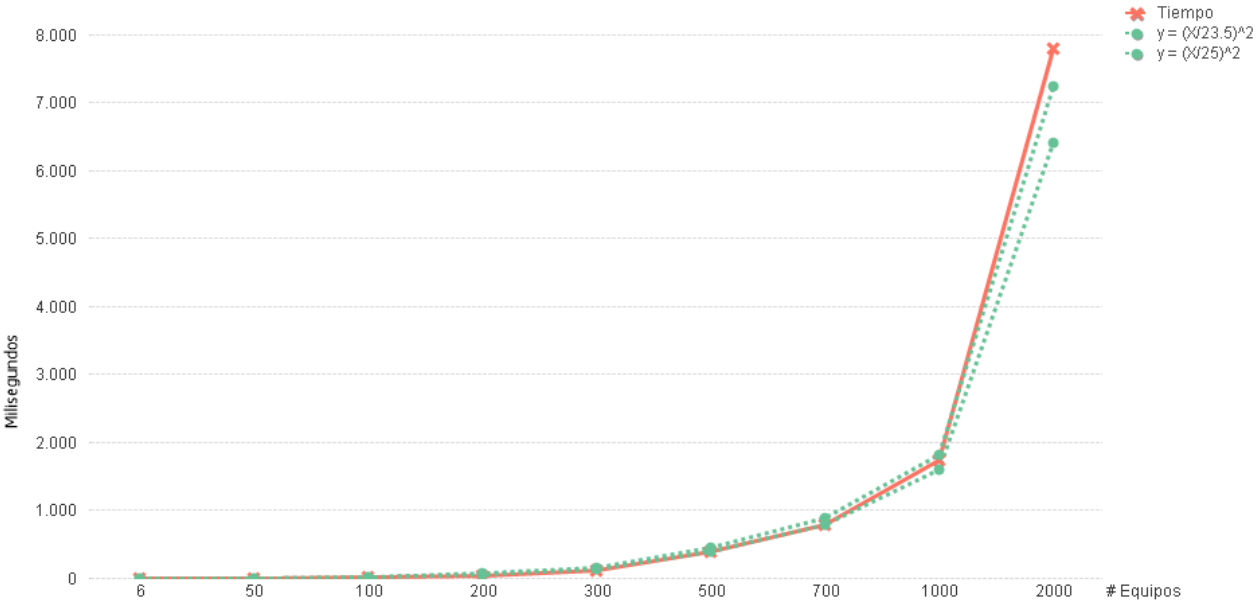

Figura 5: Gauss cuadrático

Luego observamos la misma gráfica pero con lineas de referencia de 2 funciones cubicas.

cubico.png

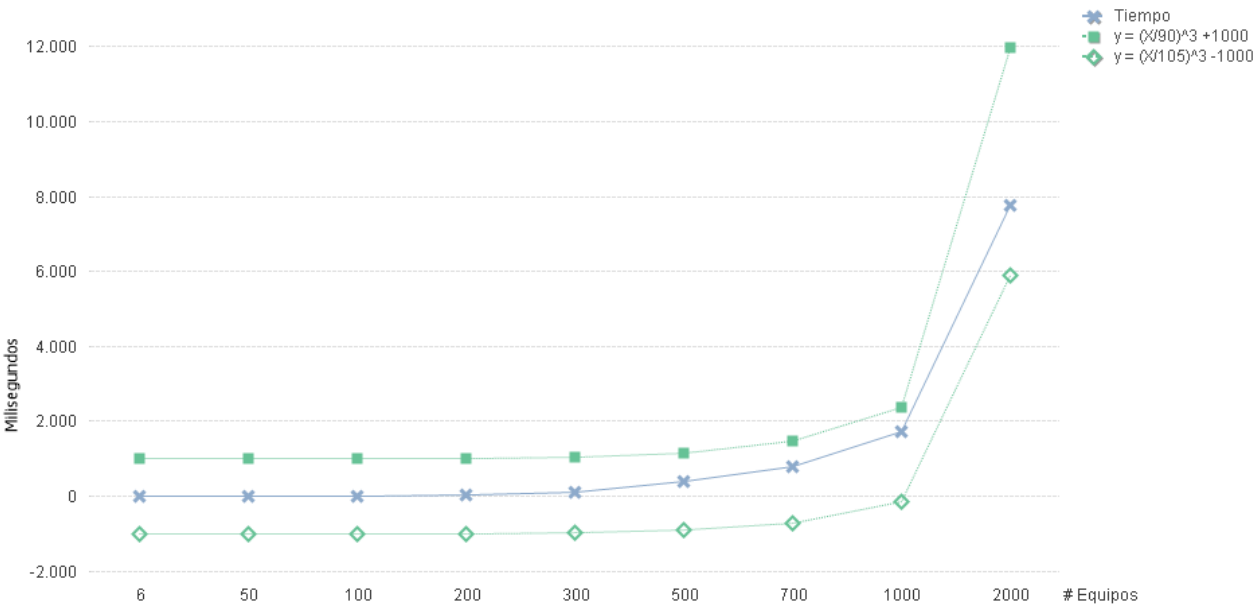

Figura 6: Gauss cúbico

Ejecutamos los test para nuestra implementacion de Cholesky.A continuación se muestran los resultados de tiempos de ejecucion dependiendo la cantidad de equipos. Para mostrar la complejidad cúbica del algoritmo lo hemos encerrado entre 2 funciones cuadráticas que evidencian que no pueden contener la curva de tiempos de nuestro algoritmo.

cuadratico.png

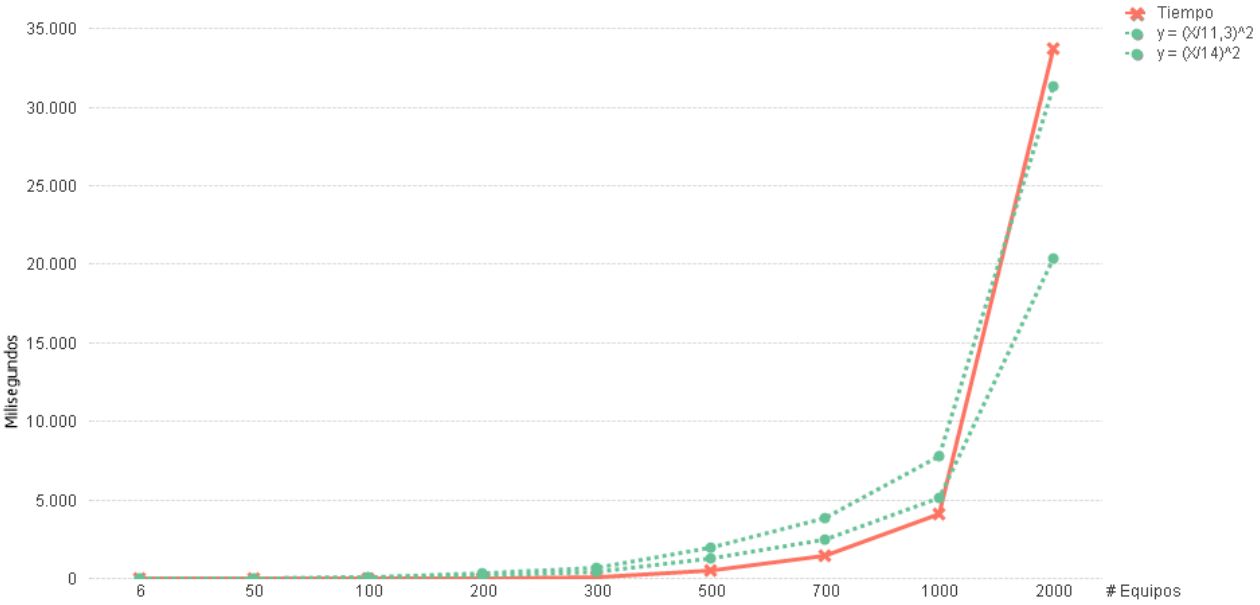

Figura 7: Cholesky cuadrático

Luego observamos la misma grafica pero con lineas de referencia de 2 funciones cúbicas.

cubicos.png

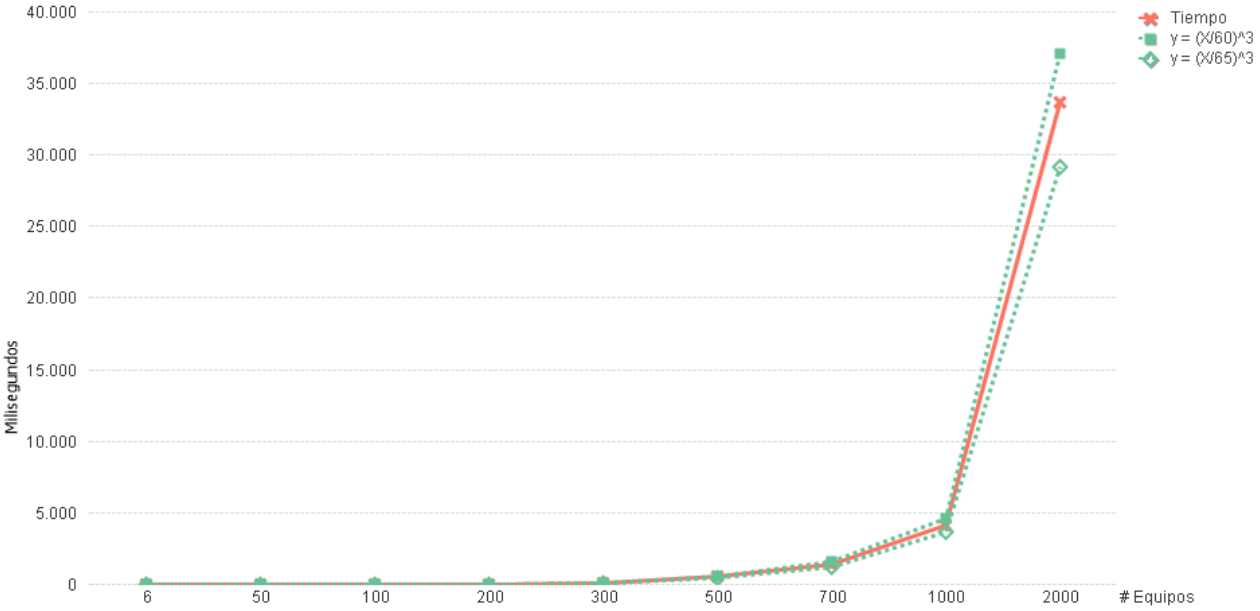

Figura 8: Cholesky cúbico

Ejecutamos los test para nuestra implementacion de WP. A continuación se muestran los resultados de tiempos de ejecucion dependiendo la cantidad de equipos:

lineal.png

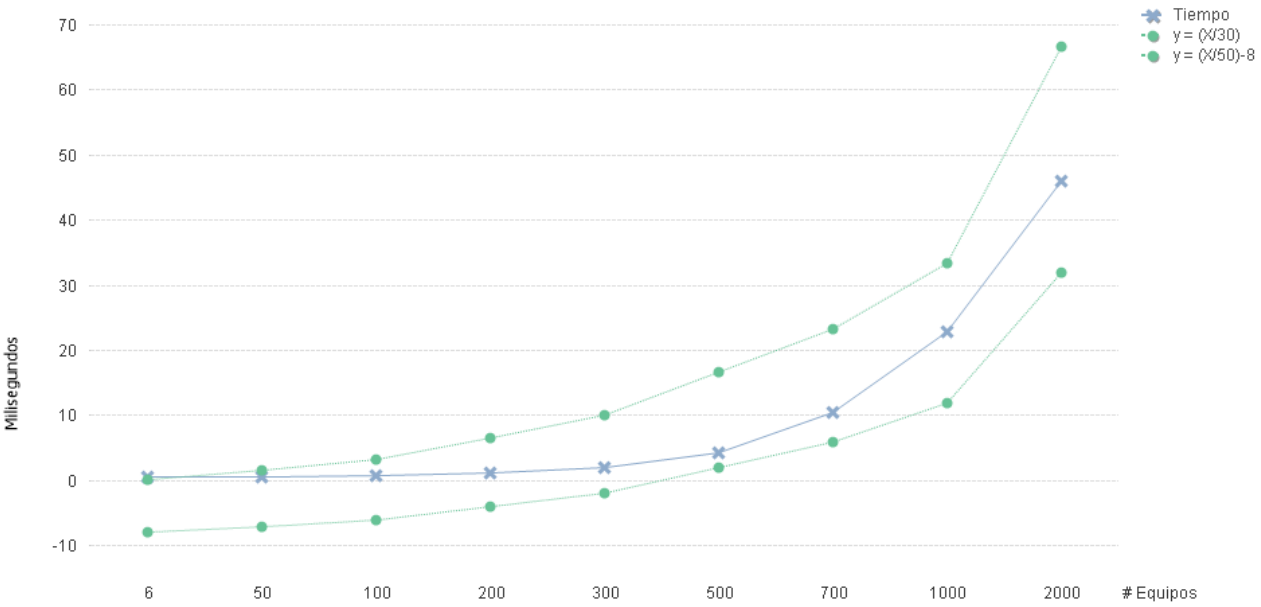

Figura 9: WP lineal

Cuando leimos el enunciado encontramos una frase que nos llamo la atención y era la siguiente:

"Se pide comparar, para distintos tamaños de matrices, el tiempo de computo requerido para cada metodo en el contexto donde la información de la matriz del sistema (C) se mantiene invariante, pero varia el termino independiente (b)"

Entendimos que nuestros tiempos de calculo no debian variar con la modificacion del termino independiente, pero para resolver esta incognita decidimos realizar la prueba.

Lo que hicimos fue tomar una matriz C, calcularle CMM, luego modificar algunos partidos de la matriz y cambiarlos (esto significa cambiar el resultado de ganados/perdidos) y calculamos nuevamente CMM.

Nuestra experimentación intenta demostrar que el tiempo de cálculo no cambia, aunque se varie el termino independiente. A continuación se grafican ambos tiempos.

con otro termino independiente.png

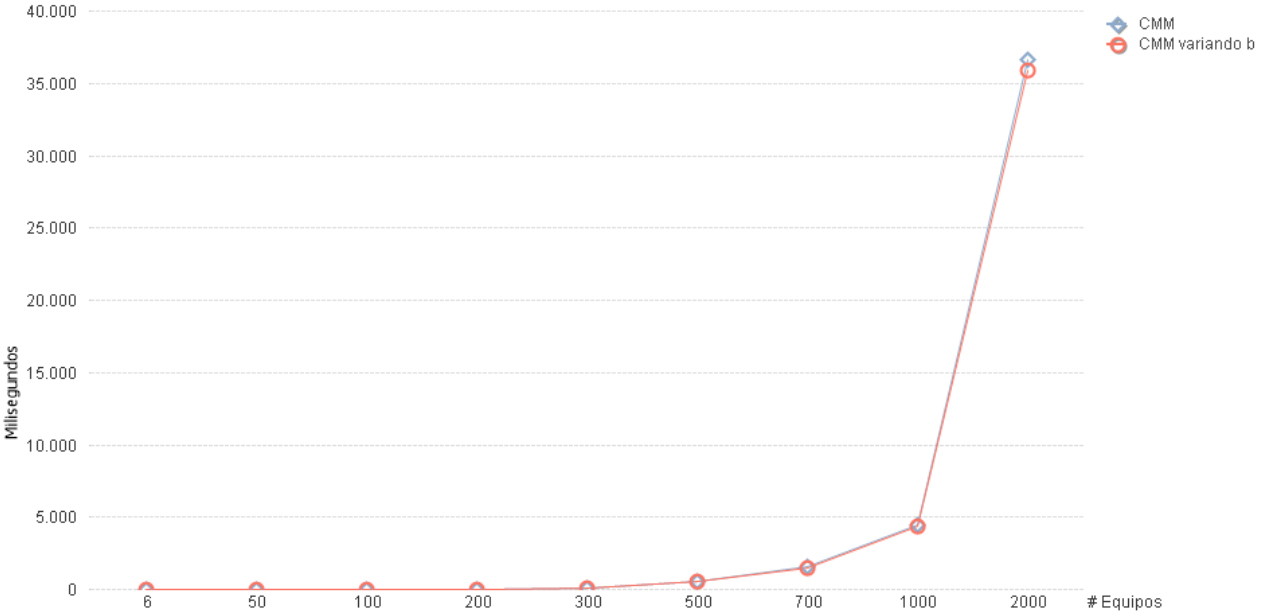

Figura 10: Cholesky con otro termino independiente

4. Discusión

En esta sección presentamos nuestras conclusiones sobre los resultados obtenidos de los experimentos del punto anterior.

4.1. Ranking

De los resultados obtenidos podemos ver que el ranking obtenido con **WP** no es muy realista, ya que la primer posición es ocupada por un participante que jugo y gano un solo partido.

El ranking obtenido por **CMM** refleja de forma mucho mas realista el desempeño de cada jugador en el torneo.

En un escenario donde tenemos participantes que jugaron una cantidad distinta de partidos pensamos que refleja mejor la realidad del torneo el metodo de **CMM**.

4.2. ¿Importa contra quien se pierde?

Como podemos observar realmente importa contra quien se pierde, del experimento realizado observamos que perder contra el participante último afecta mas el puntaje del ranking que perdiendo contra el primero.

La hipótesis con la que calculamos el experimento resulto ser falsa. Analizando más ejecuciones llegamos a la conclusión de que lo resultados obtenidos son lógicos, ya que con esta técnica es mas esperable que un equipo de mitad de tabla tenga un resultado adverso contra los primeros, por lo cual la perdida de ranking es menor.

4.3. Racha ganadora

Por lo visto el jugador escalo rapidamente desde el último puesto en la tabla de posiciones a casi la mitad de tabla. Si bien mejoro su posición, no alcanzo el top ten, y en las últimas victorias su ascenso fue mas lento. Esto nos hace concluir que solo haciendo jugar y ganar a un participante, la capacidad que tiene para crecer en el ranking esta limitada por la falta de juego de sus rivales.

Además sus victorias representaron una mejoría en el ranking de los participantes que lo vencieron a el. De esta forma podemos comprobar que la racha de un jugador afecta al ranking global.

4.4. Escalando Posiciones

4.4.1. El torneo ya finalizo

Poner resultados

4.4.2. Agregando partidos

Poner resultados

4.5. Análisis Cuantitativo

Como era de esperar en el caso de **WP** para instancias el tiempo de ejecución fue el mismo, y el tiempo demorado a medida que crecian los datos de la instancia fue lineal.

En cambio en el caso de **CMM** la implementación de **Cholesky** fue mas eficiente para las mismas instancias, y relativamente mejor a medida que se incrementaban los datos. Esto es esperado ya que nuestras implementaciones se basaron en las propuestas por el libro **Burden**, que afirma que **Cholesky** consume $1/3\ n^3$ flops y **Eliminación Gaussiana** $2/3\ n^3$ flops.

4.6. La aritmética importa

De los experimentos realizados notamos que es importante el tipo de datos utilizados. Principalmente cuando se utiliza **CMM**.

Los errores de redondeo pueden derivar en un mal cálculo del ranking. Es decir, no considerar los suficientes decimales puede derivar en que un participante con un ranking decimalmente menor quede mejor rankeado que otro con mayor puntaje.

Por ejemplo: El participante A con ranking 0,5819 y el participante B con ranking 0,5816 si se consideran solo dos decimales ambos tienen 0,58 y esto podria afectar su orden en el ranking global.

Para evitar esta situacion nuestra implementación usa el tipo de datos float con con 5 decimales despues de la coma.

4.7. Empates

Encontramos que los empates pueden modelarse en el caso de **WP**, asignando un puntaje al partido empatado y continuando con el procedimiento normal.

En el caso de **CMM** nos resulto muy díficil tratar de modelarlo, como alternativa a este resultado proponemos modelarlo como si ambos equipos perdieran. Esto nos permite reutilizar el método y de alguna forma penar a ambos equipos por no haber ganado su partido.

5. Conclusiones

Luego de la experimentación y análisis de los resultados, concluimos el método de calculo basado en **CMM** es mas justo en el caso de torneos donde los equipos no juegan la misma cantidad de partidos y donde el empate no es una opción. Ya que asigna un puntaje en base no solo a los resultados obtenidos, sino contra quien fueron obtenidos. Obteniendo un ranking basado en la meritocracia del resultado.

Para el caso de torneos donde cada equipo juegue la misma cantidad de partidos el método de **WP** a nuestro criterio resulta mas justo. Debido a que todos se pusieron a prueba la misma cantidad de veces.

Respecto a que implementación de **CMM** resulta mas eficiente. La conclusion es que depende. Ambas obtienen el mismo resultado, la principal ventaja de **Cholesky** es que realiza menos computos, mientras que la de **Eliminación Gaussiana** es que es mas sencilla su implementación.

Por último sobre **La utilización de técnicas avanzadas de análisis de datos son imprescindibles para mejorar cualquier deporte**, consideramos que la frase no es del todo cierta. Afortunadamente la frialdad de los números no es aplicable a la pasión de todos los deportes. Mientras en contados deportes el resultado puede predecirse de antemano, debido a las caracteristicas de los rivales, como en el caso del Polo, esta analogía no puede aplicarse a deportes como el Fútbol donde en innumerables ocasiones el equipo menos favorito termina llevandose el partido.

6. Apéndice

6.1. Archivos de test usados