به نام خدا

دانشگاه شهید بهشتی

دانشکده مهندسی و علوم کامپیوتر

تحلیل مه داده‌ها

مجموعه تمرین سوم

استاد درس:

دکتر ملک

دانشجو:

پارسا جعفرقلی

۴۰۱۴۲۲۰۴۸

پاییز ۱۴۰۲

سوال اول (توجه کنید که تمام راه حل مرحله به مرحله آورده شود) **(۲۰ نمره)**

دوازده سبد شامل سه آیتم وجود دارد:

{1,2,3} {2,3,4} {3,4,5} {4,5,6} {1,3,5} {2,4,6}

{1,3,4} {2,4,5} {3,5,6} {1,2,4} {2,3,5} {3,4,6}

در صورتی که support برابر ۴ باشد و تابع hash برابر (i*j)%9 باشد:

الف) با استفاده از الگوریتم PCY کدام جفت‌ها در Pass2 شمارش می‌شوند.

ب) این بار از الگوریتم multistage برای حل این مسأله استفاده کنید. تابع hash دوم برابر (i+j)%11 است. کدام جفت‌ها در pass3 شمارش می‌شوند؟

ج) آیا انتخاب تابع hash (i+j)%11 به عنوان دومین تابع hash انتخاب مناسبی است؟

Freq ۱: {۱} , {۲} , {۳} , {۴} , {۵} , {۶}

h(۱,۲) = ۲ h(۱,۳)= ۳ h(۱,۴)=۴ h(۱,۵)=۵ h(۱,۶)=۶ h(۲,۳)=۶

h(۲,۴)=۸ h(۲,۵)=۱ h(۲,۶)=۳ h(۳,۴)=۳ h(۳,۵)=۶ h(۳,۶)=۰

h(۴,۵)=۲ h(۴,۶)=۵ h(۵,۶)=۳

الف) h(۱,۳) و h(۲,۶) و h(۳,۴) و h(۵,۶) وارد قسمت pass۲ می شوند.

$h(۵,۶) = ۰$ $h(۳,۴) = ۷$ $h(۲,۶) = ۸$ $h(۱,۳) = ۴$

ب) هیچ کدام داخل یک باکت نیوفتادند.

ج) ارزیابی تابع hash دوم

- تابع hash خوب تابعی است که توزیع یکنواخت داشته باشد تا collision کمتری داشته باشیم.

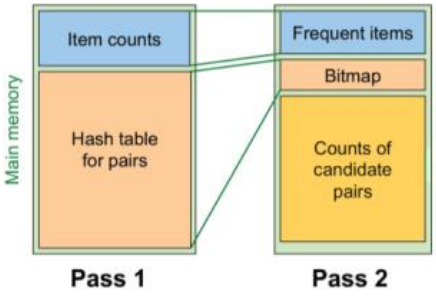
- $h(i, j) = (i + j) \%۱۱$) ممکن است خوب نباشد اگر توزیع یکنواخت نداشته باشیم، به خصوص وقتی که فضای ممکن برای سطل‌ها به اندازه مقدار مد نظر نباشد.

- باید توزیع جفت‌ها در این تابع hash مورد آزمایش و بررسی قرار گیرد تا اینکه بتوان نتیجه گرفت آیا انتخاب مناسبی است یا خیر.

بدیهی است که برای رسیدن به پاسخ‌های دقیق‌تر، نیازمند اجرا کردن فرآیندهای توضیح داده شده بر روی ماشین‌حساب یا کدنویسی برای شمارش دقیق و تعیین bucketهای hash فرکوئنت هستیم. این کار نیازمند جمع‌آوری و تجزیه‌وتحلیل داده‌های خام است که در این متن امکان‌پذیر نیست.

سوال دوم (برای گزینه درست راه حل آورده شود. اعداد صحیح با ۴ بایت نمایش داده می‌شوند.) **(۱۰ نمره)**

در الگوریتم PCY اگر فضای مورد نیاز برای شمارش آیتم‌ها 4 MB باشد و تعداد باکت‌ها برابر $8 * 10^9$ باشد، به طور تقریبی چه فضایی از Main Memory برای شمارش جفت آیتم‌ها در Pass 2 قرار می‌گیرد (قسمت زرد رنگ در شکل)؟



الف) اطلاعات مسأله ناقص است ب) 32 GB ج) 28 GB د) 31 GB

هر کدام از هش تیبل ها برای محاسبه به یک فضای ۴ بایتی نیاز دارند. پس فضایی که کل هش تیبل می گیرد برابر $۴ * ۸ * ۱۰^۹$ است که ۳۲ گیگابایت می باشد. از طرفی فضای بردار Bitmap یک فضای ۱/۳۲ از فضای هش تیبل را پر می کنند بنابر این یک بردار ۱ گیگابایتی است. با توجه به این که اندازه شمارش ایتم ۴ مگابایت است و فرکونت ایتم نیز کوچک تر از آن است (تنها مقادیر فرکونت را بر می گرداند) برای همین زیاد در محاسبات اهمیت ندارند. پس مقدار باقی مانده می شود هش مپ قسمت اول منهای بیت مپ قسمت دوم که می شود ۳۱ گیگابایت.

سوال سوم (۲۰ نمره)

مجموعه داده‌ای را در فضای دو بعدی با توجه به نقاط A(1,2), B(3,4), C(5,6), D(7,8), E(9,10) در نظر بگیرید. الگوریتم k-means را با فرض ۲ خوشه (k=2) و نقاط ابتدایی A(1,2) و D(7,8) برای یک بار روی مجموعه اعمال کرده و مقادیر مراکز خوشه‌ها را به روزرسانی کنید. در نهایت مراکز خوشه جدید و نقاط اختصاص داده شده به هر کلاستر را نمایش دهید.

بیایید الگوریتم k-means را برای یک تکرار با k=۲ خوشه و نقاط شروع A(۱,۲) و D(۷,۸) پیاده کنیم. این مراحل به صورت زیر است:

مرحله ۱: تعیین مراکز اولیه خوشه

خوشه ۱: C۱ = A(۱,۲)

خوشه ۲: C۲ = D(۷,۸)

مرحله ۲: اختصاص دادن نقاط به نزدیکترین مرکز خوشه

نقاط B، C، و E را در نظر گرفته و فاصله هر کدام را از دو مرکز خوشه اندازه‌گیری می‌کنیم. برای فاصله از فاصله اقلیدسی استفاده می‌کنیم:

فاصله بین دو نقطه (x۱, y۱) و (x۲, y۲) از رابطه زیر محاسبه می‌شود:

فاصله = sqrt{(x۲ - x۱)^۲ + (y۲ - y۱)^۲}

مرحله ۳: محاسبه فاصله‌ها و اختصاص به خوشه‌ها

با استفاده از رابطه بالا، فاصله هر نقطه (B, C, E) را از هر دو مرکز خوشه (C۱ و C۲) محاسبه و مقایسه می‌کنیم:

- فاصله B از C۱ : sqrt{(۳-۱)^۲ + (۴-۲)^۲} = sqrt۸

- فاصله B ازC۲ : sqrt{(۷-۳)^۲ + (۸-۴)^۲} = sqrt۳۲

B به C۱ نزدیک‌تر است.

- فاصله C از C۱ : sqrt{(۵-۱)^۲ + (۶-۲)^۲} = sqrt۳۲

- فاصله C از C۲ : sqrt{(۷-۵)^۲ + (۸-۶)^۲} = sqrt۸

C به C۲ نزدیک‌تر است.

- فاصله E ازC۱ : sqrt{(۹-۱)^۲ + (۱۰-۲)^۲} = sqrt۱۲۸

- فاصله E ازC۲: sqrt{(۷-۹)^۲ + (۸-۱۰)^۲} = sqrt{۸}

E به C۲ نزدیک‌تر است.

جدول اختصاص نقاط به خوشه‌ها:

خوشه ۱ فاصله از C۱ خوشه ۲ فاصله از C۲

--- --- --- ---

A(۱,۲) ۰

B(۳,۴) √۸

C(۵,۶) √۳۲

D(۷,۸) ۰

E(۹,۱۰ √۸

با توجه به جدول بالا، B به C۱، در حالی که C و E به C۲ اختصاص داده شده‌اند.

گام چهارم: به روزرسانی مراکز خوشه

میانگین مختصات نقاط اختصاص داده شده به هر خوشه محاسبه می‌شود تا مختصات مراکز جدید خوشه‌ها تعیین شوند.

برای خوشه ۱ که شامل نقاط A(۱,۲) و B(۳,۴) می‌شود:

میانگین مختصات x: (۳+۱)/۲ = ۲

میانگین مختصات y: (۴+۲)/۲ = ۳

پس، مرکز جدید خوشه C۱،۱_{جدید} می‌شود: (۲, ۳)

برای خوشه ۲ که شامل نقاط C(۵,۶)، D(۷,۸) و E(۹,۱۰) می‌شود:

میانگین مختصات x: (۹+۷+۵)/۳ = ۷

میانگین مختصات y: (۱۰+۸+۶)/۳ = ۸

پس، مرکز جدید خوشه ۲، C۲_{جدید} می‌شود: (۷, ۸)

در نتیجه، مراکز جدید خوشه‌ها به شکل زیر است:

- مرکز جدید خوشه ۱: C۱_{جدید} = (۲,۳)

- مرکز جدید خوشه ۲: C۲_{جدید} = (۷,۸)

سوال چهارم (۵ نمره)

فرض کنید می‌خواهیم الگوریتم Balance را برای تخصیص تبلیغات به دو تبلیغ کننده A و B اجرا کنیم. تبلیغ کننده A روی کوئری‌های x و y قیمت می‌دهد و تبلیغ دهنده B روی کوئری‌های x و z. هر دو نیز بودجه‌ای برابر ۲ دلار دارند. آیا الگوریتم می‌تواند برای کوئری zxyy تخصیص بهینه را انجام دهد؟

برای حل این مثال با استفاده از الگوریتم Balance، ما باید نحوه توزیع تبلیغات بر روی کوئری ها را بررسی کنیم. ما بودجه مساوی برای تبلیغ‌کننده‌های A و B داریم (هر کدام ۲ دلار) و می‌خواهیم برای سلسله‌ای از کوئری‌ها تبلیغ‌ها را تخصیص دهیم که به ترتیب zxyy هستند. برای این کار، هر تبلیغ‌دهنده به طور عادلانه باید چهار تبلیغ از سری کوئری‌ها را تقسیم کند. با این فرض که هر کلیک دقیقاً ۱ دلار هزینه داشته باشد.

نحوه اجرا به این صورت خواهد بود:

۱. کوئری z فقط توسط B پوشش داده می‌شود، بنابراین اولین تبلیغ z به B تخصیص داده می‌شود. حالا B دقیقاً ۱ دلار از بودجه‌اش را خرج کرده است.

۲. کوئری x هم توسط A و هم توسط B پوشش داده می‌شود. از آنجا که ما می‌خواهیم بودجه را متعادل نگه داریم، این کوئری باید به A داده شود چون B در حال حاضر ۱ دلار هزینه کرده و A هنوز هزینه‌ای نداشته است. حالا A و B هر کدام ۱ دلار از بودجه خود را خرج کرده‌اند.

۳. دو کوئری y بعدی فقط توسط A پوشش داده می‌شود، بنابراین هر دو را به A تخصیص می‌دهیم. حالا A کل بودجه ۲ دلاری خود را خرج کرده است.

نتیجه تخصیص بهینه به شکل زیر خواهد بود:

- کوئری z -> تبلیغ‌کننده B (۱ دلار باقی مانده برای B)

- کوئری x -> تبلیغ‌کننده A (۱ دلار باقی مانده برای A)

- کوئری y -> تبلیغ‌کننده A (بودجه A تمام شد)

- کوئری y -> تبلیغ‌کننده A (بودجه A تمام شد)

در این نقطه، تبلیغ‌کننده A دیگر بودجه‌ای ندارد، ولی هنوز یک کوئری x باقی مانده است که هر دو تبلیغ‌کننده می‌توانند پاسخ دهند و ما همچنان ۱ دلار بودجه از تبلیغ‌کننده B باقی مانده داریم که می‌توانیم خرج کنیم. پس، آخرین کوئری x به تبلیغ‌کننده B داده می‌شود. این تبلیغ باعث می‌شود که بودجه‌ی B هم تمام شود.

نتیجه نهایی تخصیص بودجه تبلیغات به شکل زیر خواهد بود:

- تبلیغ‌کننده A: کوئری x و دو بار کوئری y تخصیص داده شد (بودجه صرف شده = ۲ دلار)

- تبلیغ‌کننده B: کوئری z و کوئری x تخصیص داده شد (بودجه صرف شده = ۲ دلار)

بنابراین، الگوریتم Balance بهینه این B,B,A,A است که تولید نشد.

سوال پنجم (۲۰ نمره)

یک ناشر آگهی، سه آگهی را برای قرار دادن در هر صفحه به ترتیب از بالا انتخاب می‌کند. CTR در هر موقعیت برای هر تبلیغ کننده متفاوت است و هر تبلیغ کننده برای هر موقعیت CTR متفاوتی دارد. هر تبلیغ‌کننده برای تعداد کلیک‌ها پیشنهاد قیمت می‌دهد و یک بودجه روزانه دارد. هنگامی که یک کلیک رخ می‌دهد، تبلیغ کننده مبلغی را که پیشنهاد داده است پرداخت می‌کند. در زیر جدولی از پیشنهادها، CTR برای موقعیت‌های ۱، ۲، ۳ و بودجه برای هر تبلیغ کننده آمده است.

Advertiser	Bid	CTR1	CTR2	CTR3	Budget
A	$.10	.015	.010	.005	$1
B	$.09	.016	.012	.006	$2
C	$.08	.017	.014	.007	$3
D	$.07	.018	.015	.008	$4
E	$.06	.019	.016	.010	$5

برای پیدا کردن تعداد کلیک‌های هر تبلیغ دهنده باید میزان درآمد مورد انتظار برای هر موقعیت از هر تبلیغ‌کننده محاسبه کنیم و موقعیت‌های تبلیغاتی را بر اساس بالاترین درآمد مورد انتظار تخصیص دهیم. فرمول انتظار میزان درآمد برای هر موقعیت به صورت CTR مربوط به آن موقعیت ضربدر پیشنهاد قیمت (bid) برای هر کلیک است.

ابتدا برای هر تبلیغ‌کننده، میزان درآمد مورد انتظار برای هر موقعیت را محاسبه کنیم:

قرار است ناشر آگهی‌ها را بر اساس بالاترین میزان درآمد مورد انتظار انتخاب کند و هر تبلیغ‌کننده فقط یکبار می‌تواند برای یک موقعیت انتخاب شود.

۱. موقعیت ۱:

- A: $۰.۱۰ * ۰.۰۱۵ = $۰.۰۰۱۵

- B: $۰.۰۹ * ۰.۰۱۶ = $۰.۰۰۱۴۴

- C: $۰.۰۸ * ۰.۰۱۷ = $۰.۰۰۱۳۶

- D: $۰.۰۷ * ۰.۰۱۸ = $۰.۰۰۱۲۶

- E: $۰.۰۶ * ۰.۰۱۹ = $۰.۰۰۱۱۴

تبلیغ‌کننده A بالاترین میزان درآمد مورد انتظار را برای موقعیت ۱ دارد.

۲. موقعیت ۲:

- B: $۰.۰۹ * ۰.۰۱۲ = $۰.۰۰۱۰۸

- C: $۰.۰۸ * ۰.۰۱۴ = $۰.۰۰۱۱۲

- D: $۰.۰۷ * ۰.۰۱۵ = $۰.۰۰۱۰۵

- E: $۰.۰۶ * ۰.۰۱۶ = $۰.۰۰۰۹۶ (A را حذف می‌کنیم چرا که برای موقعیت ۱ انتخاب شده)

تبلیغ‌کننده C بالاترین میزان درآمد مورد انتظار را برای موقعیت ۲ دارد.

۳. موقعیت ۳:

- B: $۰.۰۹ * ۰.۰۰۶ = $۰.۰۰۰۵۴

- D: $۰.۰۷ * ۰.۰۰۸ = $۰.۰۰۰۵۶

- E: $۰.۰۶ * ۰.۰۱۰ = $۰.۰۰۰۶ (A و C را حذف می‌کنیم چون برای موقعیت‌های قبل انتخاب شده‌اند)

تبلیغ‌کننده E بالاترین میزان درآمد مورد انتظار را برای موقعیت ۳ دارد.

حالا، باید تعداد کلیک‌هایی که هر تبلیغ دهنده در یک روز بدست می‌آورد محاسبه کنیم. هر تبلیغ‌دهنده برای تعداد کلیک‌های بدست آمده مبلغ پیشنهادی خود را پرداخت می‌کند تا زمانی که بودجه‌شان تمام شود.

برای مثال، اگر فرض کنیم که در یک روز ۱۰۰ کلیک وجود دارد:

- تبلیغ‌کننده A برای هر کلیک $۰.۱۰ پرداخت می‌کند و بودجه‌اش $۱ است، پس تبلیغ‌کننده A می‌تواند ۱۰ کلیک (۱ / ۰.۱۰) بدست آورد.

- تبلیغ‌کننده C برای هر کلیک $۰.۰۸ پرداخت می‌کند و بودجه‌اش $۳ است، پس تبلیغ‌کننده C می‌تواند ۳۷.۵ کلیک (۳ / ۰.۰۸) بدست آورد، اما چون کلیک‌ها نمی‌توانند نیمه باشند، ۳۷ کلیک تمام کلیک‌هایی خواهد بود که تبلیغ‌کننده C می‌تواند بدست آورد.

- تبلیغ‌کننده E برای هر کلیک $۰.۰۶ پرداخت می‌کند و بودجه‌اش $۵ است، پس تبلیغ‌کننده E می‌تواند ۸۳.۳۳ کلیک (۵ / ۰.۰۶) بدست آورد، اما باز هم باید به تعداد تمام کلیک‌ها گرد شود پس ۸۳ کلیک بدست خواهد آمد.