UNIVERSITÀ DEGLI STUDI DI MILANO-BICOCCA

DECISION MODELS FINAL PROJECT

SANTA GIFT MATCHING PROBLEM

Authors:

Abd El Kader Shady - 838487 - s.abdelkader@campus.unimib.it

Calani Massimiliano - 838723 - m.calani@campus.unimib.it

July 4, 2019

Abstract

Immaginando di essere il centro di elaborazione dati di Babbo Natale, ci troviamo alle prese con un annoso problema. È la notte della vigilia e Babbo Natale deve decidere quale regalo consegnare per ogni bambino; dato che possono esistere coppie e triplette di gemelli si deve cercare di non fare disparità nella distribuzione dei loro regali e quindi ad ogni tripletta o coppia bisogna consegnare lo stesso regalo. L'obiettivo finale sarà quello di massimizzare la felicità di bambini e dei regali.
Per arrivare al massimo livello di felicità complessiva, sarà necessario soddisfare entrambi, o basterà massimizzare la felicità di uno dei due gruppi ? Attraverso prove sperimentali e improvement del modello, passeremo da una felicità complessiva di 0,70 a 0,754 ed infine a 0,778.

1 Introduzione

Dal problema proposto si sa di avere a disposizione 1000 unità di 1000 regali diversi e 1 milione di bambini ai quali consegnare un regalo ciascuno. Di questi, si sa che i primi 5001 bambini sono triplette di gemelli, mentre i 40000 bambini successivi sono coppie di gemelli; ad ogni coppia o tripletta di bambini deve essere consegnato lo stesso regalo per tutti i bambini che la compongono. Ogni regalo ha una lista di 1000 bambini preferiti ai quali desidera essere assegnato, mentre i bambini hanno 100 regali preferiti che vorrebbero ricevere. Per ogni bambino e ogni regalo abbiamo un indice di felicità, calcolato come

$$ChildHappiness = \begin{cases} 2 \times GiftOrder, & \text{if the gift is in the child's wish list} \\ -1, & \text{if the gift is out of the child's wish list} \end{cases}$$

$$GiftHappiness = \begin{cases} 2 \times ChildOrder, & \text{if the child is in in the good kids list of the gift} \\ -1, & \text{if the child is out of the gift's good kids list} \end{cases}$$

Vengono quindi creati 2 indici ANCH e ANSH calcolati come:

$$\text{ANCH} = \frac{1}{n_c} \sum_{i=0}^{n_c} \frac{2*(ChildHappiness)}{2*(MaxChildHappiness)}$$

$$\text{ANSH} = \frac{1}{n_g} \sum_{i=0}^{n_g} \frac{2*(GiftHappiness)}{2*(MaxGiftHappiness)}$$

La felicità massima per i bambini è 100, mentre per i regali è 1000

ANCH e ANSH contribuiscono a massimizzare l'indice ANH calcolato come:
ANH = $ANCH^3 + ANSH^3$

(ANSH= Average Normalized Santa Happiness, per i regali)
(ANCH= Average Normalized Child Happiness, per i bambini)
(ANH= Average Normalized Happiness)

Essendo l'obiettivo finale quello di massimizzare il valore ANH ed essendo questo costituito dalla somma di $ANCH^3$ e $ANSH^3$, si va ad analizzare in maniera esplorativa quali combinazioni dei due possibili valori possano portare al risultato migliore.
Da questa analisi risulta che i risultati migliori si ottengono principalmente se uno dei due fattori ha un valore elevato (0,8) a discapito del secondo (0) rispetto ad un valore discreto da entrambe le parti (0,6 e 0,6)
Si è voluto quindi cercare di ottimizzare il punteggio che viene ottenuto dai bambini e successivamente quello ottenuto dai regali; il procedimento di risoluzione scelto risulta essere una tecnica greedy, che risulta comunque ottenere dei buoni risultati.
Questo criterio di ottimizzazione è stato guidato dal fatto che il peso della felicità dei regali risulta essere minore rispetto al peso dei bambini; infatti non soddisfando un regalo, si ottiene un punteggio di -0.005, mentre la non soddisfazione di un bambino porta ad un punteggio di -0.05, che risulta essere 10 volte superiore e quindi è ciò che si vuole evitare.
Una volta ottenuto il valore ottimale tramite la metodologia greedy si è usata questa combinazione bambini/regalo come base di partenza per spostarsi in un risultato migliore attraverso tecniche che prendono spunto dal genetic algorithm; in particolare si è cercato di scambiare la posizione dei regali in maniera casuale per un certo numero di iterazioni mantenendo gli scambi per quei valori che aumentano la felicità totale del problema

2 Dataset

Per risolvere il problema, sono stati forniti due dataset, uno contenente l'elenco dei bambini con le loro preferenze, e l'altro contenente l'elenco dei regali con i loro bambini preferiti.
E' stata inoltre fornita una possibile soluzione come esempio. Questo, pur potendo essere usato come punto di partenza per ulteriore miglioramenti, è stato utilizzato in questo cosa solamente come esempio per capire il tipo di risultato, poiché durante l'elaborazione ne è stato costruito uno nuovo con il metodo di risoluzione creato partendo da zero.

	0	1	2	3	4
0	0	377750	248196	299067	180786
1	1	18884	986611	97454	533768
2	2	320640	212052	763863	942751
3	3	290226	821783	32328	258030
4	4	315731	453346	267048	109022

Figura 1: Dataset Regali

	0	1	2	3	4	5	6	7	8	9	...	91	92	93	94	95	96	97	98	99	100
0	0	61	808	121	167	127	289	840	678	575	...	529	120	283	40	18	4	693	909	491	495
1	1	378	247	299	174	809	108	316	863	241	...	255	682	115	881	96	428	856	596	511	93
2	2	19	981	95	538	262	952	193	426	340	...	278	604	48	608	785	990	838	779	204	84
3	3	322	210	755	930	986	421	941	72	505	...	708	891	108	168	157	83	714	119	207	333
4	4	292	810	33	256	374	863	170	866	360	...	720	338	156	565	184	785	923	352	7	661

Figura 2: Dataset Bambini

	bambino	regalo
0	0	979
1	1	979
2	2	979
3	3	220
4	4	220
5	5	220
6	6	37
7	7	37
8	8	37
9	9	904
10	10	904
11	11	904

Figura 3: Vettore fornito

	bambino	regalo
0	0	707
1	1	707
2	2	707
3	3	360
4	4	360
5	5	360
6	6	915
7	7	915
8	8	915
9	9	748
10	10	748
11	11	748

Figura 4: Vettore creato

3 The Methodological Approach

Avendo fatto delle considerazioni riguardo le specifiche del problema, ad esempio quanto si poteva perdere in punteggio senza soddisfare un regalo o un bambino, oppure andando a considerare le possibili intersezioni dei regali che potevamo prendere in considerazione per le triplette, si decide di sviluppare un modello decisionale greedy, suddiviso in cinque fasi di assegnamento del regalo ad ogni bambino:

Fase 1
Come prima cosa, si è cercato di assegnare i regali alle triplette e coppie di gemelli, andando ad assegnare uno stesso regalo per ogni tripletta e per ogni coppia di gemelli, che allo stesso tempo massimizzava la somma ANCH + ANSH. Ad ogni assegnazione, si teneva anche conto della felicità di regalo e bambino.

Fase 2
Nella fase 2 si è provato ad assegnare un regalo ad ognuno dei bambini, massimizzando la felicità dei bambini. Inoltre si è tenuto conto dei bambini per i quali non ci fosse alcun regalo disponibile tra i loro preferiti, la felicità di questi bambini denominati “insoddisfatti” sarà oggetto di ottimizzazione nella prossima fase

Fase 3
Non essendoci più regali disponibili tra i preferiti dai bambini, per i bambini insoddisfatti si è cercato di ottimizzare le preferenze dei regali e si è passato ad assegnare ai regali, il bambino insoddisfatto che può massimizzare il loro valore di felicità.

Fase 4
Infine, non essendoci più bambini né regali che abbiano delle preferenze tra regali o bambini rispettivamente, si assegnano casualmente i regali restanti ai bambini che ne sono rimasti senza.

Fase 5
Come ultima fase, si passa alla valutazione delle associazioni che abbiamo ottenuto, andando a fare la somma delle felicità dei bambini e quelle dei regali e arrivando ad un valore per la quantità ANH ricercata.

Fase 6
Una volta ottenuto un primo risultato con le precedenti 5 fasi, si passa a cercare di migliorare il risultato con un ulteriore algoritmo basato su uno step del Genetic Algorithm, si sceglie infatti di estrarre casualmente due vettori contenti quindi due bambini (non gemelli o triplette) e i due rispettivi regali, si tenta di scambiare i regali ai bambini ed osservarne la nuova soluzione.
Se la nuova soluzione è migliore della precedente si fissa questo scambio nel risultato finale, altrimenti non si completa lo scambio. Questo step viene svolto n volte, con n deciso in base al miglioramento ricercato e al tempo di esecuzione voluto. (nel progetto in questione si è scelto n=1.000.000)

3.1 Sviluppo del Modello

Le varie fasi del modello seguono le fasi di sviluppo che si sono susseguite:
Dapprima si è cercato di capire cosa bisognasse rispettare come vincolo e in che modo trarne il massimo beneficio, andando ad unire l'utile al necessario.
Poi ci si è concentrati sulla massimizzazione del punteggio per ogni bambino, guardando le sue preferenze e che il limite dei regali già assegnati non superasse il limite di regali disponibili.
Come terzo step si è andati ad assegnare ai bambini un regalo, cercando di massimizzare il punteggio dei regali, poiché per i bambini non erano più disponibili regali che li soddisfassero.
Come quarto passo, ci si affida alla casualità per assegnare un regalo ad ogni bambino che ne era sprovvisto, al fine di consegnare un regalo ciascuno anche se non risultava essere uno dei preferiti.
Come ultimo passo si svolge uno scambio casuale tra coppie di bambini singoli andando a mantenere solo quegli scambi che aumentano il valore finale

Il modello è stato sviluppato cercando di ottimizzare ogni fase del processo; tra queste risultava essere particolarmente onerosa in termini di tempo la fase 2; questa infatti deve valutare per tutti i bambini se il primo regalo è disponibile o meno; nel caso non lo sia, va al regalo successivo, se invece è disponibile si tiene conto del punteggio di regalo e bambino e si aggiorna il contatore dei regali già usati. Se anche l'ultimo regalo della lista del bambino risulta non essere più disponibile allora si procede al suo inserimento nella lista dei bambini insoddisfatti e si passa al bambino successivo.

3.2 Punti salienti

All'interno della fase 1, con un metodo di valutazione precedente, si andava a vedere quali regali in comune fossero presenti nelle righe dei gemelli e si massimizzava in quella maniera il punteggio. Tuttavia, si raggiungeva un risultato peggiore, anche se la risoluzione risultava essere più veloce, circa 10 minuti contro i 30 della soluzione che ha portato al risultato migliore.

All’interno della fase 2, il tempo di computazione senza la giusta funzione risultava essere molto lungo. L’utilizzo di iterrows() ha contribuito alla risoluzione di questo problema.

4 Risultati e Valutazione

Il risultato ottenuto con la prima versione del modello, è stato di 0.7 che risulta essere un buon risultato visti i punteggi ottenuti da altre soluzioni sulla pagina Kaggle della competizione. Avendo verificato la presenza di un errore all’interno del codice, si procede con le migliorie necessarie, portando il modello ad ottenere un punteggio di 0.755 circa (figura 5).

```
1  ANH = ((sum(felicitab)/1000000)**3)+((sum(felicitar)/1000000)**3)
2  ANH
```

```
0.7547137242837073
```

Figura 5: Risultato ottenuto

Successivamente, utilizzando il rimescolamento per i soli bambini successivi al 45000 esimo (descritto nella fase 6 paragrafo 3) , dopo 1.000.000 è stato ottenuto il seguente risultato.

```
1  ANH=(b/1000000)**3+(f/1000000)**3
2  ANH
```

```
0.7779368823591611
```

Figura 6: Risultati dopo mescolamento

5 Discussione

Il risultato complessivo ottenuto è dato dalla somma dei cubi delle due grandezze ANCH e ANSH; basandoci su questo ragionamento e sui singoli valori ottenuti, si può vedere che questi confermano il criterio di scelta di

associazione ipotizzato inizialmente, infatti il valore di ANCH risulta essere molto maggiore rispetto a ANSH.
SIGNIFICATO DEL NOSTRO RISULTATO OTTENUTO
Ragionando in maniera inversa si può verificare che un valore di 1 per la quantità ANH può essere ottenuto se andiamo ad avere un valore di ANCH^3 e ANSH^3 uguali entrambi a 0,5. Questo risultato può essere ottenuto, avendo un valore di ANCH e ANSH pari a 0.79, che vorrebbe dire che il regalo o bambino preso in considerazione sono tra i primi 21 o 210. Questa soluzione dovendo fare un check sulla posizione per tutto il milione di bambini risulterebbe tuttavia computazionalmente molto complessa, anche se facile da sviluppare a livello di programmazione.

Altro possibile miglioramento che può essere fatto è andare a vedere il punteggio che ottiene ogni bambino con l'assegnamento ad esso di un determinato regalo e nel caso che sia al di sotto di una certa soglia, lasciare il bambino da parte. Poi, seguendo lo stesso criterio di scelta si può ed andare ad ottimizzare il valore secondo il regalo e lasciare da parte quelli che non raggiungono un certo punteggio.
Solo successivamente si va ad usare un criterio che massimizza il punteggio dei bambini e poi dei regali.

Uno dei punti deboli riscontrati è sicuramente il tempo di compilazione di tutto il codice, poiché impiega circa un'ora e mezza per tutta la compilazione .
Sviluppi futuri potrebbero quindi essere una migliore implementazione del codice, migliorando nel caso la sintassi e le funzioni usate, aumentare il numero di scambi nella funzione che scambia i regali cercando però prima di diminuire il tempo di esecuzione, oppure procedere con l'implementazione di un'euristica che migliori i risultati anche in termini di tempo.

6 Conclusione

Quindi, considerando lo scopo del progetto e il metodo di risoluzione proposto, si può affermare che questo è riuscito a trovare una soluzione corretta al

problema. Tutte le triplette e coppie di gemelli hanno lo stesso regalo, e tutti i bambini hanno un regalo assegnato. Inoltre il risultato ottenuto rispecchia la policy di ottimizzazione ipotizzata; tuttavia alcune modifiche potrebbero essere apportate per snellire il codice o cambiare punto di vista, in cerca di un metodo di risoluzione che può essere più efficace e possa aumentare l'indice ANH in quanto il risultato offre ampi margini di miglioramento dato che nella competizione kaggle il primo classificato ha ottenuto un punteggio di 0.9363 .

Referenze

[1]

[2] Sito kaggle: https://www.kaggle.com/c/santa-gift-matching/