



UNIVERSITÀ DI PISA

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Triennale in Ingegneria Informatica

APPLICAZIONE DELLA STIGMERGIA COMPUTAZIONALE
ALLA CARATTERIZZAZIONE DI HOTSPOT
ATTRAVERSO DATI FORNITI
DA LOCATION-BASED SOCIAL NETWORKS

Candidato:

Lorenzo Mannocci

Relatori:

Prof. Mario G. C. A. Cimino

Prof. Gigliola Vaglini

Ing. Antonio Luca Alfeo

Febbraio 22, 2019

Indice

Introduzione.....	5
1.1 Analisi dei dati in un contesto di smart city	6
1.2 Hotspots.....	7
1.3 Stigmergia.....	8
1.4 Descrizione framework.....	9
Analisi geografica del social network tramite istogrammi bivariabili.....	11
2.1 Importazione del dataset e formattazione dei dati	11
2.2 Stoccolma, Dallas, San Antonio, Austin e ‘South by Southwest’ festival	15
Estrazione dati di Austin (Texas)	16
3.1 Estrazione dati e analisi	16
Discretizzazione delle coordinate	19
4.1 Codice per la discretizzazione delle coordinate geografiche.....	19
Disposizione dei dati in base all’input previsto dal framework.....	21
5.1 Codice MatLab per formattazione dati.....	21
Prima analisi output algoritmo stigmergico.....	22
6.1 Analisi settimanale	22
6.2 Zona geografica ristretta.....	23
6.3 Necessità di un’analisi più dettagliata	25
6.4 Analisi weekend e weekdays.....	25
Seconda analisi output algoritmo stigmergico.....	27
7.1 Analisi weekend	29
7.2 Analisi weekdays.....	30
7.3 Analisi festival ‘South by Southwest’ festival.....	31
Conclusioni.....	35
Bibliografia.....	36
Ringraziamenti	37

Elenco figure

<i>Figura 1: Esempio di dati presenti nel dataset</i>	7
<i>Figura 2: Differenti tipi di hotspots</i>	8
<i>Figura 3: Architettura framework</i>	9
<i>Figura 4: Istanza di funzione S-Shape, usando $\alpha = 1, \beta = 8$.</i>	10
<i>Figura 5: Codice Java per la formattazione dei dati</i>	12
<i>Figura 6: Codice MatLab per la realizzazione dei grafici con istogrammi</i>	13
<i>Figura 7: Grafico di dicembre. Si può vedere sulla sinistra l'istogramma relativo a Stoccolma, mentre quelli sulla destra quelli relativi a Dallas, San Antonio, Austin e Sacramento.</i>	14
<i>Figura 8: Codice Java per selezionare i dati relativi ad un'area geografica, che sia un quadrilatero.</i>	16
<i>Figura 9: Frequenza di login nella zona di Dallas, Austin, San Antonio nel mese di maggio</i>	17
<i>Figura 10: Frequenza di login nella zona di Dallas, Austin, San Antonio nel mese di marzo</i>	18
<i>Figura 11: Area prelevata per l'analisi dei dati. Misure dei lati: 40 km</i>	18
<i>Figura 12: Codice in Java per poter discretizzare le coordinate geografiche in 400 * 400 bin da 100 * 100 metri ciascuno</i>	20
<i>Figura 13: Codice che per ogni mese, per ogni giorno del mese selezionato, per ogni deposito orario del giorno del mese selezionato, controlla quali dati ricadono in tale deposito.</i>	21
<i>Figura 14: Codice di Figura 13, modificato per disporre i dati per un'analisi settimanale</i>	22
<i>Figura 15: Parametri in ingresso per il framework</i>	23
<i>Figura 16: Frequenza di Austin, mese di marzo. Si può notare come la maggior parte degli accessi sia concentrata in pochi bin.</i>	24
<i>Figura 17: Frequenza di Austin, mese di maggio. Si può notare come la maggior parte degli accessi sia concentrata in pochi bin.</i>	24
<i>Figura 18: Funzione che viene invocata su ogni riga letta dal file .txt. Ritorna true, qualora il giorno considerato appartenga al weekend.</i>	25
<i>Figura 19: Grafico 3D relativo ai weekend, output dell'algoritmo stigmergico</i>	27
<i>Figura 20: Grafico 3D relativo ai weekdays, output dell'algoritmo stigmergico</i>	27
<i>Figura 21: Grafico 2D relativo ai weekend, output dell'algoritmo stigmergico</i>	28
<i>Figura 22: Grafico 2D relativo ai weekdays, output dell'algoritmo stigmergico</i>	28
<i>Figura 23: Codice Java, utilizzato per la riconversione delle coordinate discretizzate in coordinate reali.</i> ..	29
<i>Figura 24: Secondo hotspot, relativo ai weekdays, si trova in corrispondenza di questa zona.</i>	30
<i>Figura 25: 15 marzo, grafico 3D</i>	31
<i>Figura 26: 19 marzo, grafico 3D</i>	31
<i>Figura 27: 21 marzo, grafico 3D</i>	32
<i>Figura 28: 15 marzo, grafico 2D</i>	32
<i>Figura 29: 19 marzo, grafico 2D</i>	33

Figura 30: 21 marzo, garfico 2D 33
Figura 31: Zona in cui si svolge il 'South by Southwest' festival 34

Capitolo 1

Introduzione

Nel **capitolo 2**, l'obiettivo principale è quello di fare una mappatura mese per mese, dei luoghi in cui il social network è utilizzato maggiormente: per fare questo viene utilizzata una funzione di MatLab, che genera istogrammi.

Vengono descritti i problemi riscontrati nell'importare il dataset in MatLab e come sia stato necessario formattare diversamente i dati, tramite codice scritto in linguaggio Java, per poter far sì che fossero accettati ed elaborati correttamente (in particolar modo il formato DateTime).

Visti i risultati trovati nel capitolo 2, grazie agli istogrammi è risultato che principalmente ci si sarebbe potuti concentrare su tre diverse città, caratterizzate da un'alta concentrazione di login: Stoccolma, Austin, San Antonio.

Si è deciso di focalizzare il resto dell'analisi su Austin, in quanto certi che nella terza settimana di marzo, questa città ospitasse un festival internazionale della musica, 'South by Southwest' festival. Il codice sviluppato potrà comunque essere applicato ad altre aree geografiche.

Nel **capitolo 3** vengono estratti i dati relativi alla città di Austin: questo è avvenuto, selezionando un'area quadrata, per semplificare le fasi successive, in particolare modo la discretizzazione delle coordinate. Comunque, la conformazione della città lo permetteva agilmente, senza l'introduzione di eccessive approssimazioni.

Si è fatta un'ulteriore analisi, per confermare l'incremento dei login in concomitanza del festival sopra citato.

Nel **capitolo 4**, si è scritto il codice necessario alla discretizzazione delle coordinate: visto la dimensione del quadrato (40 km), si è ritenuto sufficiente una dimensione di bin di 100 x 100 metri. Ma come vedremo, successivamente l'area scelta per l'analisi finale è più ridotta rispetto a quella su cui è avvenuta la prima parte del lavoro. Le funzioni sono state scritte comunque in modo tale da poter variare i parametri in input (ad esempio successivamente, visto la scelta di un'area più ridotta, la dimensione dei bin è diventata 25 x 25 metri).

Nel **capitolo 5**, è stato necessario disporre i dati in base all'input previsto dal framework [1] che implementa l'algoritmo che utilizza tecniche stigmergiche. Questo passaggio è stato fatto tramite un codice scritto in MatLab.

Nel **capitolo 6** è stato valutato l'output del framework: sono stati prodotti 2 grafici per ogni giorno dei mesi passati in input (marzo, perché è il mese del festival, e maggio, un mese canonico, per poter fare un confronto).

Tuttavia, la numerosità dei grafici non permette di fare un'analisi semplice, dunque si è deciso di ripetere l'analisi, disponendo i dati per settimana. Dunque, in questo modo i grafici prodotti sono 2 x (4 settimane) x 2 mesi.

Con questo livello di risoluzione di analisi, non sono state però notate nei grafici notevoli differenze e questo ci ha indotto a pensare che la zona geografica considerata fosse troppo grande e che i check-in si concentrassero maggiormente in una zona più ristretta.

Sono state quindi svolte nuovamente le operazioni presenti dal capitolo 3 in poi, ovviamente riutilizzando il codice già scritto e cambiando i parametri in input.

Dunque, nel **capitolo 7**, su questo nuovo dataset relativo a una zona di Austin più ristretta, è stata fatta un'analisi non più settimanale, che è stata ritenuta troppo grossolana, ma un'analisi per weekend e per weekdays aggregando indistintamente marzo e maggio. Sono stati prodotti anche in output i grafici relativi ai 10 giorni in cui si tiene il festival ad Austin.

Quindi si è passati all'analisi dei dati, ma solo dopo ad un'operazione di debug, in cui è stato trovato un errore nel codice che formatta i dati nel corretto input per il framework. Dunque, è stato necessario rielaborare i dati e a questo punto fare un'analisi sui grafici prodotti, che hanno mostrato alcune peculiarità di Austin, di alcuni suoi edifici fondamentali, come l'Austin Convention Center.

Di seguito presenteremo il problema dell'analisi dei dati in un contesto di smart city e il framework utilizzato per l'analisi con tecniche stigmergiche. Tali questioni vengono affrontate, anche grazie all'analisi fatta nella tesi [1] nella parte introduttiva.

1.1 Analisi dei dati in un contesto di smart city

Il framework utilizzato in questa tesi e descritto nel paragrafo successivo è nato per compiere un'analisi spazio-temporale delle transazioni bancarie effettuate in una città.

Infatti, oggi con l'aumentare dei pagamenti tramite carte di credito, ma soprattutto smartphone, è diventato molto più semplice tracciare i movimenti del denaro, avendo quindi a disposizione una grande quantità di dati da memorizzare e da poter analizzare. È estremamente importante conoscere "Come si muove il denaro" all'interno di una grande città: esso può infatti fornire supporto agli economisti e ai sociologi per delineare comportamenti e benessere della popolazione (diversità, lealtà, regolarità) e creare statistiche, considerando che ogni volta che un individuo esegue una transazione, rilascia un'impronta digitale. Le informazioni che possiamo estrarre dall'impronta sono di tipo spazio-temporale.

La posizione, la data, l'ora e l'importo di una transazione sono esempi di dati rilasciati. Combinando i dati relativi non solo alle transazioni, ma anche ai dati demografici della popolazione, è possibile ampliare l'analisi cercando possibili cause legate alla sfera lavorativa degli individui, o per identificare gruppi di individui simili tra loro. Tutto questo può essere usato per identificare aree molto attraenti della città.

Nel nostro caso di studio, le tracce spazio-temporali sono più facili da memorizzare, in quanto l'accesso ad un social network è di per sé, un'azione necessariamente collegata al mondo tecnologico e più facilmente tracciabile, rispetto ai pagamenti o alle transazioni bancarie.

Il nostro obiettivo è identificare gli hotspots, ovvero le aree della città analizzata con maggior concentrazione di login (come vedremo la città scelta sarà Austin). Possiamo fare un'analisi suddividendo i dati fra quelli appartenenti al weekend e ai weekdays, così da poter fare un confronto tra gli hotspots di questi due tipi di giorni così diversi.

Useremo i dati relativi all'anno 2010, prelevati dal sito web Gowalla, un social network basato su posizione: ovvero per ogni post pubblicato, è necessario inserire una posizione.

Tali dati presentano le seguenti informazioni:

[user]	[check-in time]	[latitude]	[longitude]	[location id]
196514	2010-07-24T13:45:06Z	53.3648119	-2.2723465833	145064
196514	2010-07-24T13:44:58Z	53.360511233	-2.276369017	1275991
196514	2010-07-24T13:44:46Z	53.3653895945	-2.2754087046	376497
196514	2010-07-24T13:44:38Z	53.3663709833	-2.2700764333	98503
196514	2010-07-24T13:44:26Z	53.3674087524	-2.2783813477	1043431
196514	2010-07-24T13:44:08Z	53.3675663377	-2.278631763	881734
196514	2010-07-24T13:43:18Z	53.3679640626	-2.2792943689	207763
196514	2010-07-24T13:41:10Z	53.364905	-2.270824	1042822

Figura 1: Esempio di dati presenti nel dataset

Ulteriori analisi, che non vengono affrontate in questa tesi per motivi di tempo, riguardano i login al social network durante le feste comandate, come il Natale, Pasqua ecc...: durante queste giornate particolari infatti è molto probabile che gli hotspots si modifichino.

Inoltre, sarebbe possibile e fruttuoso unire i risultati di questa tesi, andando a integrarli con i dati relativi alle amicizie presenti sul social network. Andare dunque ad analizzare come gli spostamenti di un individuo siano correlati alle amicizie più strette.

Quindi come auspicava l'autore della tesi [1] e del framework utilizzato, quest'ultimo viene utilizzato in un settore diverso da quello delle transazioni bancarie.

1.2 Hotspots

Secondo il dizionario Merriam-Webster, un hotspot è un luogo con un interesse anormale. Ad esempio, nell'applicazione dell'analisi di transazioni bancarie, gli hotspots sono i luoghi in cui si verificano maggiormente, cioè i luoghi con alta densità di transazioni. Allo stesso modo nel caso del social network, gli hotspots sono i luoghi in cui si verificano maggiormente gli accessi allo stesso.

Data una finestra temporale a lungo termine composta da n finestre a breve termine e uno spazio definito da due attributi (cioè LAT, LON), possiamo definire tre tipi di hotspot (Figura 2):

- Intermittente
- Semipermanente
- Permanente

Gli hotspots intermittenti (H4) appaiono sporadicamente nello spazio degli attributi e non coprono interamente una finestra a lungo termine. Gli hotspots semipermanenti (H1) appaiono nello stesso spazio degli attributi per finestre "sufficientemente" a breve termine, ma non interamente in quella a lungo termine. Ad esempio, il "sufficientemente" può essere definito da una soglia per uno specifico spazio degli attributi. Hotspot semi permanente potrebbe non essere contiguo (H2). Gli hotspots permanenti (H3) vengono visualizzati nell'intera finestra a lungo termine nello stesso attributo spazio bin.

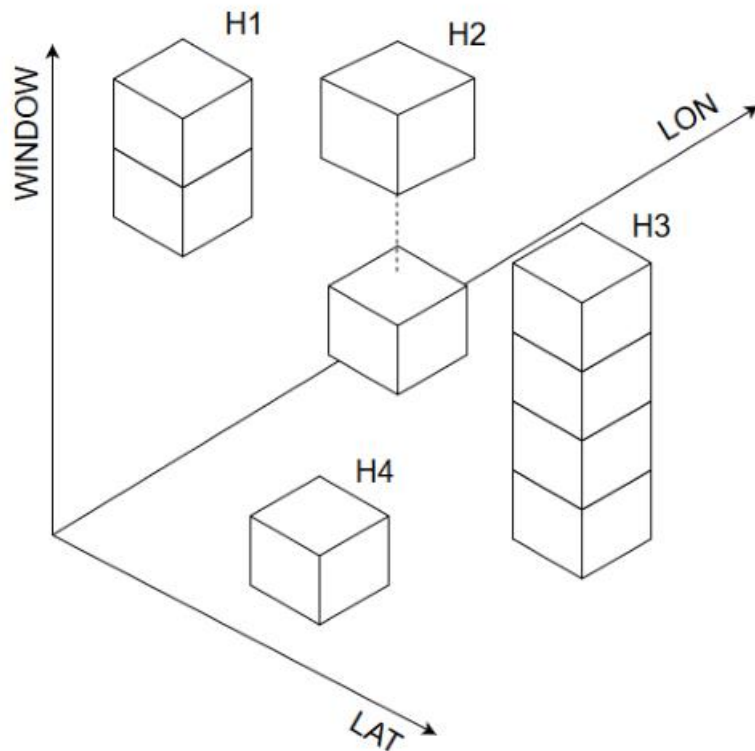


Figura 2: Differenti tipi di hotspots

1.3 Stigmergia

“La stigmergia è un metodo di comunicazione utilizzato nei sistemi decentralizzati col quale gli individui del sistema comunicano fra loro modificando l'ambiente circostante. La stigmergia è stata osservata inizialmente in natura; per esempio, le formiche comunicano le une con le altre lasciando una traccia di feromoni quindi una colonia di formiche è un esempio di sistema stigmergico. Un altro esempio comune è quello della costruzione dei nidi delle termiti. Anche le termiti usano i feromoni per costruire strutture molto complesse seguendo un semplice insieme di regole decentralizzato. Ogni insetto scava una pallina di fango dal suo ambiente, la copre di feromoni e la lascia sul terreno. Le termiti sono attratte dai feromoni degli individui dello stesso termitaio e quindi depositano le loro palline di fango vicino a quelle già depositate. Con il tempo questo comportamento porta a costruire pilastri, archi, gallerie e camere.” [5]

La traccia del feromone è un segnale indiretto lasciato nell'ambiente per le altre formiche, così che altri possono seguirlo per trovare una risorsa. La traccia del feromone si dissipa con il tempo, scomparendo dopo molto tempo. Analizzando il comportamento di una singola formica il suo contributo alla comunità non è chiaro, perché agisce e prende decisioni in modo indipendente, e apparentemente senza interesse per le altre formiche. Ma analizzando l'insieme delle formiche nel suo comportamento collettivo, si osserva un'intelligenza emergente evidente.

Lo studio del comportamento delle colonie di formiche e delle loro capacità di auto-organizzazione è interessante per gli informatici perché fornisce modelli di organizzazione distribuita che sono utili per risolvere problemi di ottimizzazione e controllo distribuito, sottolineando il ruolo svolto dalla stigmergia come paradigma di comunicazione distribuita.

1.4 Descrizione framework

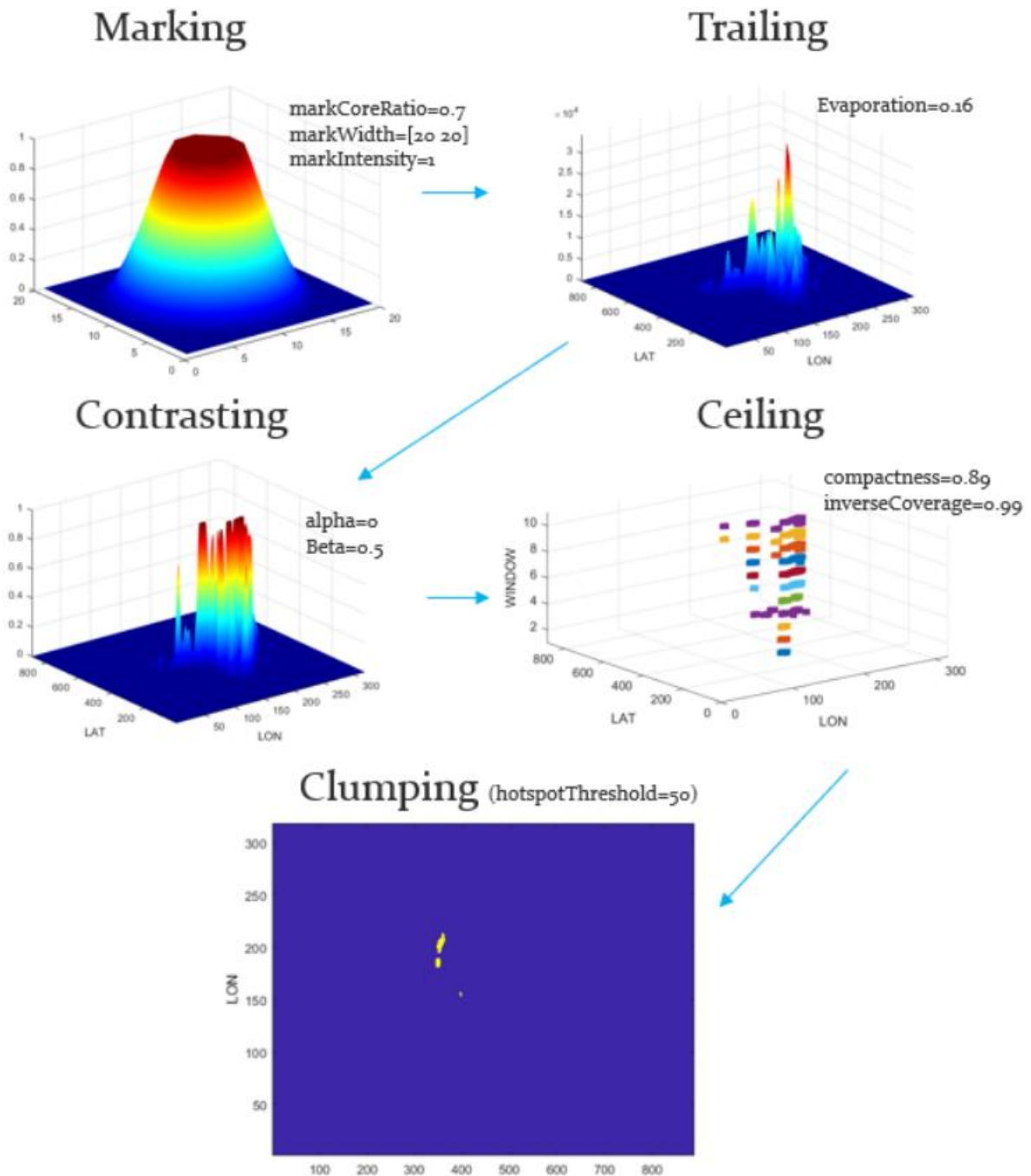


Figura 3: Architettura framework

L'architettura del framework è composta da 5 componenti fondamentali: marking, trailing, contrasting, ceiling, clumping (Figura 3). Dato un intervallo di tempo a lungo termine e le coppie latitudine – longitudine relative a finestre temporali a breve termine, la struttura identifica e restituisce come output la più alta densità e le aree più significative, note come hotspot.

1) Per ogni finestra temporale di breve termine $[t \pm \Delta t]$, il framework prende in input i vettori di triple $[t, x, y]$ ovvero le coordinate di un insieme di samples nello spazio (opportunamente discretizzato) di due feature

per gli istanti dentro la finestra temporale di breve termine. Le finestre di breve termine sono semisovrapposte al fine di mitigare il rischio che una determinata dinamica sia “tagliata” da una divisione in finestre nette.

2) Per ogni tripla, il modulo **Mark** crea una impronta a tronco di cono di coordinate centrali x_t, y_t intensità 1, diametro maggiore $widht$, e rapporto tra le basi $ratio_top/bottom$;

3) Per ogni Mark rilasciato, il modulo **Trail**, deposita l’impronta nello spazio x,y alle corrispondenti coordinate, e fa evaporare una quantità e , minore di uno.

4) Per ogni finestra temporale il modulo **Contrast**, prende il Trail, deamplifica le zone con minore intensità e amplifica quelle a maggiore intensità, producendo un **DenseTrail**. Il grado di deformazione è parametrizzabile tramite α, β .

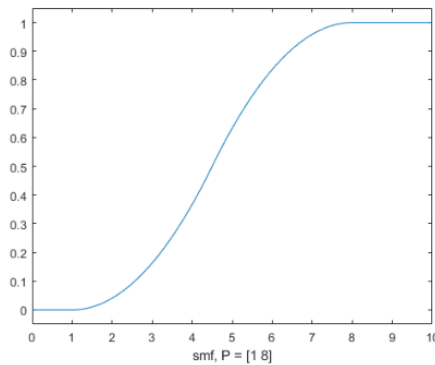


Figura 4: Istanza di funzione S-Shape, usando $\alpha = 1, \beta = 8$

5) Per ogni finestra temporale, il modulo **Ceiling** prende il **DenseTrail** e produce in uscita le aree con intensità massima.

6) Per ogni finestra temporale di lungo termine, il modulo di **Clump** addensa tutti i cubi che hanno una intersezione lungo la dimensione del tempo, rilevando l’area dello spazio $[x,y]$ in cui il *cluster* è presente per almeno una data percentuale (*consistency*) di finestre di breve termine.

7) Differential evolution (DE) è un metodo che ottimizza un problema cercando iterativamente di migliorare una soluzione possibile per quanto riguarda una data misura di qualità.

È un algoritmo di ottimizzazione implementato dal framework utilizzato in questa tesi. Sarà necessario quindi prima di qualsiasi esecuzione del framework, eseguire il modulo DE, così che i parametri immessi input successivamente siano corretti. In questa fase, i dati in input devono appartenere a giorni e mesi il più possibili “normali”: ovvero non le feste comandate, non i weekend ecc..

Nel nostro caso di studio, sono utilizzati i giorni infrasettimanali di maggio: è stato verificato che ad Austin (città dell’analisi) in questo periodo non vi fossero nessun tipo di eventi o feste.

Differential evolution adatta la ricerca dei cluster variando i parametri presentati sopra, usando due metriche di qualità e una condizione di esistenza. Tali metriche mirano a specializzare la ricerca del cluster corrispondente alla minima zona più spazio-temporalmente densa, elaborando i samples passati in input come training.

Capitolo 2

Analisi geografica del social network tramite istogrammi bivariabili

Obiettivi di questa prima fase sono:

Studiare l'ambiente di sviluppo di MatLab e l'omonimo linguaggio, strumenti necessari per la successiva analisi dei dati;

Prendere confidenza e capire il framework per l'analisi con tecniche stigmergiche, necessario in una seconda fase allo studio dei dati;

Analizzare il dataset con istogrammi bivariabili, allo scopo di fare una mappatura mese per mese, dei luoghi in cui il social network è stato utilizzato maggiormente.

2.1 Importazione del dataset e formattazione dei dati

Il primo punto da affrontare per poter condurre l'analisi del dataset, è ovviamente l'importazione di quest'ultimo nell'ambiente di lavoro (MatLab). In questa fase di importazione dei dati, sono stati riscontrati problemi nell'interpretazione corretta da parte di MatLab della colonna contenente il timestamp (seconda colonna, vedi *Figura 1*). Infatti sia tramite importazione dei dati con GUI, messa a disposizione da MatLab, sia tramite l'utilizzo delle funzioni, come 'importdata()' o 'load()', MatLab non riesce ad accettare il formato ISO 8601, con cui invece si presentava il timestamp nei dati.

Per questo motivo è stato scritto il codice in *Figura 5*, in linguaggio Java, per poter formattare i dati con un formato accettato da MatLab, e che allo stesso tempo permettesse di memorizzare i timestamp direttamente come Date e non come String, così da evitare una successiva conversione di tipo.

A questo punto, con i dati così formattati, è stato possibile fare un'analisi degli stessi, suddividendoli mese per mese. Il codice, scritto in MatLab, ha quindi come output 12 grafici (uno per mese): essi mappano a livello globale, i login effettuati in base alla posizione, determinate dalle due coordinate latitudine (LAT) e longitudine (LON), che sono proprio le variabili presenti sull'asse X e Y. Sull'asse Z è invece presente l'intensità, ovvero la frequenza con cui la coppia latitudine-longitudine compare nei dati forniti.

```

import java.io.*;

public class FormattatoreDati {

    public static void main(String[] args) throws FileNotFoundException, IOException {
        FileReader f;
        f = new FileReader("dati.txt");
        BufferedReader b;
        b = new BufferedReader(f);

        FileWriter w;
        w = new FileWriter("datiFormattati.txt");
        BufferedWriter bw;
        bw = new BufferedWriter (w);

        String s;

        while(true) {

            s = b.readLine();
            //System.out.println(s);
            if (s == null)
                break;
            String[] arrOfStr = s.split("\t");

            //System.out.println(arrOfStr[1]);

            String informazioniData[] = arrOfStr[1].split("-");
            String giorno[] = informazioniData[2].split("T");
            String time [] = giorno[1].split("Z");
            //utente          anno          mese
            String finale = arrOfStr[0] + "\t" + informazioniData[0] + "-" + informazioniData[1] + "-"
                //| giorno          time          latitudine          longintuidine          idLoc
                + giorno[0] + "," + time[0] + "\t" + arrOfStr[2] + "\t" + arrOfStr[3] + "\t" + arrOfStr[4];

            //System.out.println(finale);

            bw.write (finale);
            bw.write ("\n");

            bw.flush();

        }
    }
}

```

Figura 5: Codice Java per la formattazione dei dati

Lo scopo di questa analisi è trovare e quindi selezionare i luoghi, che con più probabilità coincideranno ovviamente con delle città, in cui il social network viene maggiormente utilizzato, così da poter concentrare lo studio successivo solamente sui dati relativi ai luoghi con una più alta concentrazione di login.

Una piccola nota da fare a questo punto riguarda la risoluzione che il grafico deve avere. Ovvero nell'utilizzo della funzione di libreria 'hist3()', per la realizzazione dei grafici con istogrammi bivariabili, è possibile, e nel nostro caso obbligatorio come vedremo, specificare la risoluzione del grafico.

Con risoluzione, intendo in realtà la dimensione dei bin utilizzati dalla 'hist3()', necessari per una discretizzazione delle variabili. Nel caso di latitudine e longitudine, è stato possibile verificare che con bin di dimensione maggiore di 1, l'errore finale risulta essere troppo grande. Infatti, se ci pensiamo bene, soprattutto nella zona centrale del globo terrestre, già 1 grado di differenza in coordinate (sia nella latitudine che nella longitudine, quindi un doppio errore), può significare un grande spostamento in termini chilometrici. Quindi fissati bin di dimensione 1, l'errore nella maggior parte dei casi è accettabile per un'analisi soddisfacente e corretta.

La riga 1, 2, 3 sono necessarie per importare i dati. Il codice utilizzato è il seguente:

```
creaGraficiMensiliDati.m x +
1 - fileID = fopen('datiFormattati.txt');
2 - A = textscan(fileID, '%f %{\yyyy-MM-dd,HH:mm:ss}D %f %f %f');
3 - fclose(fileID);
4
5 - s = 12;
6 - months = {'Gennaio', 'Febbraio', 'Marzo', 'Aprile', 'Maggio', 'Giugno', ...
7           'Luglio', 'Agosto', 'Settembre', 'Ottobre', 'Novembre', 'Dicembre'};
8 - for c = 1:s
9 -     B = [month(A(:, 2)), A(:, 3), A(:,4)];
10 -    C = B(B(:, 1) == c, (2:3));
11 -    figure;
12 -    hist3(C, 'Ctrs', {-180:1:180 -180:1:180});
13 -    xlabel('LAT')
14 -    ylabel('LON')
15 -    %set(gcf, 'renderer', 'opengl');
16 -    %set(get(gca, 'child'), 'FaceColor', 'interp', 'CDataMode', 'auto');
17
18 -    title(strcat("Frequenza dati mese di ", months(c)));
19 - end
```

Figura 6: Codice MatLab per la realizzazione dei grafici con istogrammi

I risultati provenienti dai 12 grafici mensili sono i seguenti (latitudine, longitudine), in ordine di intensità frequenza:

- Gennaio:
 1. 58.5, 18.5 -> Stoccolma
 2. 38.5, -121.5 -> Sacramento (frequenza simile a Stoccolma)
- Febbraio:
 1. Stoccolma
 2. Sacramento (molto inferiore a Stoccolma)
- Marzo:
 1. San Antonio
 2. Altre città, ma frequenza nettamente inferiore
- Aprile:
 1. Stoccolma (superiore)
 2. San Antonio
 3. Dallas (simile a San Antonio)
- Maggio:
 1. Stoccolma
 2. Sacramento
 3. Dallas
 4. San Antonio (Sacramento, Dallas, San Antonio simili)
- Giugno:
 1. Stoccolma
 2. San Antonio
 3. Austin (simile a San Antonio)

Capitolo 2. Analisi geografica del social network tramite istogrammi bivariabili

- Luglio:
 1. Stoccolma
 2. Dallas
 3. San Antonio (simile a Dallas)
- Agosto:
 1. Stoccolma
 2. Dallas
 3. San Antonio (simili
 4. Austin
- Settembre
 1. Stoccolma
 2. Dallas
 3. San Antonio (simili)
- Ottobre
 1. Stoccolma
 2. Dallas
 3. San Antonio (simili)
- Novembre
 1. Stoccolma
 2. Sacramento
- Dicembre:
 1. 38.5, -121.5 -> Sacramento (11200)
 2. 58.5, 18.5 -> Stoccolma (9500)
 3. 32.5, -97.5 -> Dallas
 4. 29.5, -98.5 -> San Antonio (simile a Dallas)

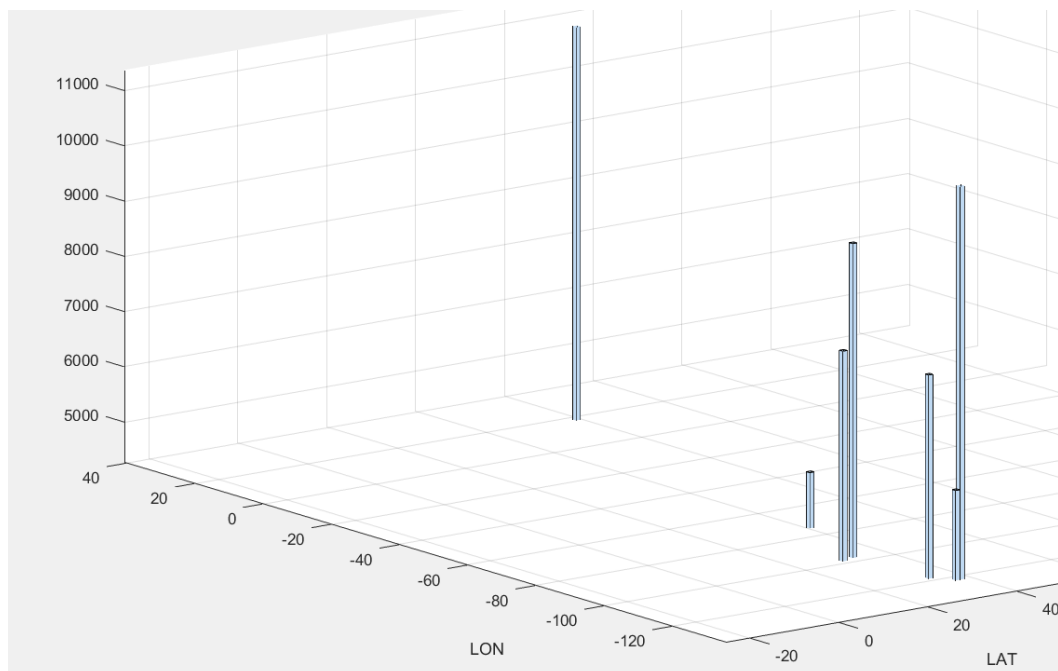


Figura 7: Grafico di dicembre. Si può vedere sulla sinistra l'istogramma relativo a Stoccolma, mentre quelli sulla destra quelli relativi a Dallas, San Antonio, Austin e Sacramento.

In Figura 7 un esempio di grafico prodotto. Un maggior zoom di esso, avrebbe implicato di concentrarsi solamente su un singolo istogramma, perdendo il senso generale del risultato, mentre in questo modo è possibile avere una visione di insieme.

2.2 Stoccolma, Dallas, San Antonio, Austin e ‘South by Southwest’ festival

Alla luce dei risultati trovati, le città da prendere in considerazione per l’analisi futura, sono:

- Stoccolma. Risultato confermato dal fatto che una ricerca a posteriori su internet, ha messo in evidenza che Stoccolma è stata scelta da Gowalla, per una fase di test del medesimo;
- Dallas, Austin, San Antonio e Austin. Sono 3 città molto vicine fra loro, per questo vengono raggruppate in unico risultato. Quest’ultimo è confermato a posteriori dal fatto che Austin è la sede centrale del social network, dunque dove è più probabile un suo maggiore utilizzo;
- Sacramento. Città che dista circa due ore in auto dalla Silicon Valley, dove la spinta tecnologica è molto forte, per ovvi motivi.

Ultima osservazione riguarda l’impennata di login che si ha nel mese di marzo, nell’area Dallas – Austin – San Antonio. Il motivo è da ritrovare nel ‘South by Southwest’ festival, uno dei più importanti festival del cinema e della musica, non solo degli Stati Uniti, ma anche nel mondo (ricerca a posteriori, compiuta sul web tramite le keyword “Austin march”).

Capitolo 3

Estrazione dati di Austin (Texas)

Avendo notato, come riportato alla fine del capitolo 2, che ad Austin e nelle zone limitrofe, vi è un incremento dei login al social network, nel mese di marzo, si è deciso di concentrarsi su questo primo fenomeno, avendo certezza dell'esistenza di un evento che probabilmente modifica gli hotspots.

Gli obiettivi di questa seconda fase, sono quindi:

- Estrarre i dati relativi ad Austin, nel mese di marzo e nel mese di maggio (un mese scelto per le sue caratteristiche piatte, privo di eventi particolari o feste comandate);
- Oltre ai dati relativi ad Austin, verranno estratti i dati relativi all'area geografica ricoperta dalle città di Austin, Dallas e San Antonio. In quest'area si è verificato infatti un aumento dei login nel mese di marzo, ma si vuole dimostrare come l'aumento nella città di Austin sia nettamente maggiore e che sia proprio Austin la causa dell'aumento in tutta la zona.

3.1 Estrazione dati e analisi

Il seguente codice è stato utilizzato per selezionare i dati relativi ad Austin e alle tre città Dallas-Austin-San Antonio nel mese di marzo e nel mese di maggio.

```
double a = Double.parseDouble(arrOfStr[2]);
double b = Double.parseDouble(arrOfStr[3]);

if ((Integer.parseInt(informazioniData[1]) == mese) && (
    f(a, b, x1, y1, x2, y2) <= 0 &&
    f(a, b, x2, y2, x3, y3) <= 0 &&
    f(a, b, x3, y3, x4, y4) <= 0 &&
    f(a, b, x4, y4, x1, y1) <= 0)) {

    //utente          anno          mese
    String finale = arrOfStr[0] + "\t" + informazioniData[0] + "-" + informazioniData[1] + "-"
    // giorno          time          latitudine          longintuidine          idLoc
    + giorno[0] + "," + time[0] + "\t" + arrOfStr[2] + "\t" + arrOfStr[3] + "\t" + arrOfStr[4];

    //System.out.println(finale);

    bw.write(finale);
    bw.write("\n");

    bw.flush();
    i++;
}

}
System.out.println(i);
}

public static double f(double a, double b, double x1, double y1, double x2, double y2) {
    return ((a-x1)*(y2-y1)) - ((b-y1)*(x2-x1));
}
```

Figura 8: Codice Java per selezionare i dati relativi ad un'area geografica, che sia un quadrilatero.

Capitolo 3. Estrazione dati di Austin (Texas)

Seguono il numero di login per area geografica e mese. Le coordinate dell'area effettivamente utilizzata nelle fasi successive variano, in quanto si è deciso per semplicità di utilizzare una forma quadrata. Inoltre, come vedremo, verrà ridotta notevolmente la zona di Austin considerata e che viene poi sottoposta alle analisi future. Questo per evidenziare il fatto che i numeri che seguono sono indicativi, visto che possono variare (di poco), al variare dell'area effettivamente estratta. Infine, anche la funzione di *Figura 8*, può ovviamente essere semplificato nel caso di un'area di forma quadrata.

Austin (numero di login relativi alla sola area geografica di Austin):

- Marzo: 74'641 login.
- Maggio: 25'366 login

Dallas-Austin-San Antonio:

- Marzo: 100'906 login
- Maggio: 50'649 login

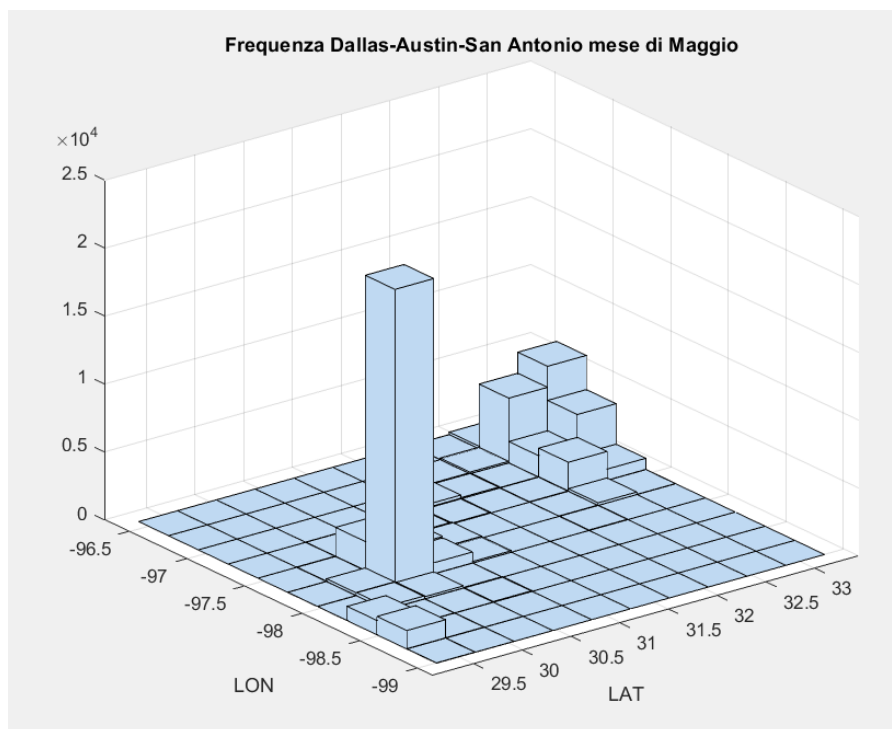


Figura 9: Frequenza di login nella zona di Dallas, Austin, San Antonio nel mese di maggio

Capitolo 3. Estrazione dati di Austin (Texas)

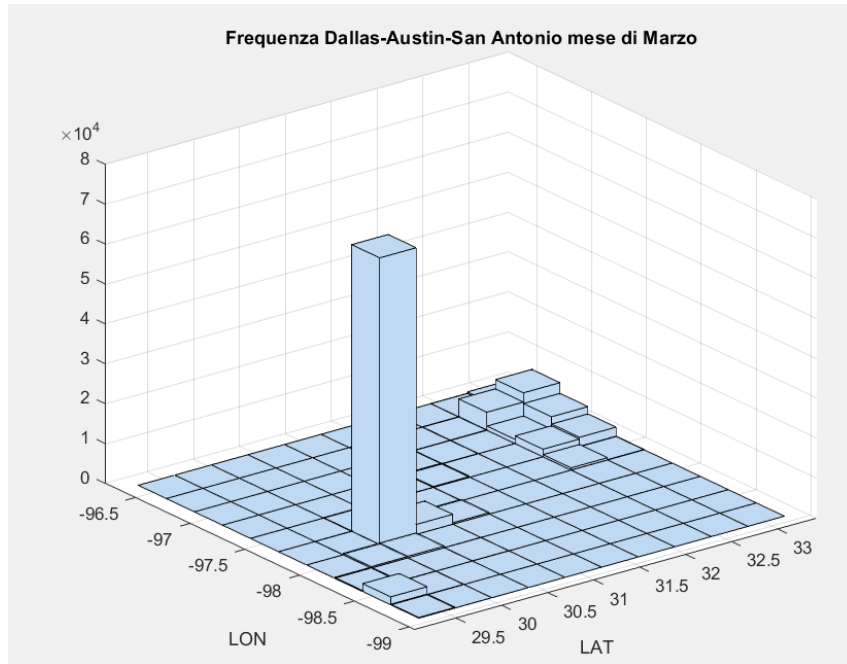


Figura 10: Frequenza di login nella zona di Dallas, Austin, San Antonio nel mese di marzo

In Figura 9 si può notare in alto a sinistra i bin relativi a Dallas; quello più alto, centrale è il bin relativo ad Austin. Mentre i due bin più piccoli in basso a destra sono quelli relativi alla posizione di San Antonio. Dunque, Austin presenta comunque un numero di login maggiore rispetto alle città vicine, anche nel mese di maggio.

Dalla Figura 10 si osserva che i login ad Austin triplicano, mentre quelli di Dallas e San Antonio rimangono pressoché identici (circa 0.5×10^4). Questo incremento relativo ad Austin è da imputare al 'South by Southwest' festival, che si tiene ad Austin a marzo.

Visti i risultati della zona Dallas-Austin San Antonio, si decide dunque di procedere ad un'analisi dei dati relativi ad Austin. L'area prelevata come accennato in precedenza è a forma di quadrato, per semplificare la fase di discretizzazione delle coordinate.

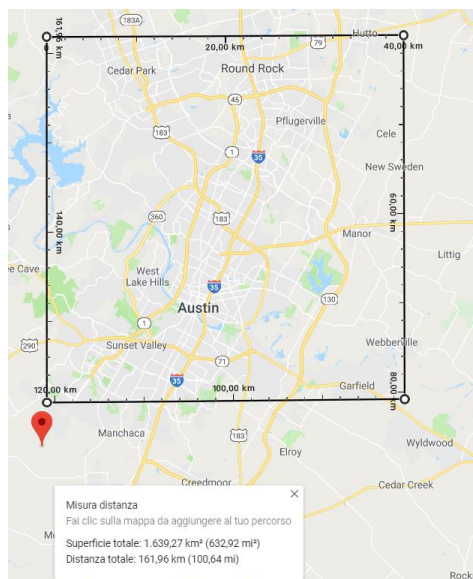


Figura 11: Area prelevata per l'analisi dei dati. Misure dei lati: 40 km

Capitolo 4

Discretizzazione delle coordinate

In questa fase si ha l'obiettivo di discretizzare le coordinate geografiche, per poi poter, una volta disposti i dati nel corretto formato di input, darli in ingresso all'algoritmo che utilizza tecniche stigmergiche, implementato dal codice del framework.

4.1 Codice per la discretizzazione delle coordinate geografiche

Prima osservazione da fare, riguarda il fatto che una medesima variazione in gradi in latitudine e longitudine ha una diversa corrispondenza in chilometri (tale relazione dipende anche dalla posizione sul globo terrestre).

Visto inoltre la superficie quadrata dell'area in esame, si è deciso di utilizzare una risoluzione di bin da 100 x 100 metri per la discretizzazione.

Le coordinate dei vertici del quadrato che racchiude Austin sono le seguenti:

- 1) 30.541532, -97.919929
- 2) 30.541532, -97.5023702
- 3) 30.1819047, -97.919929
- 4) 30.1819047, -97.5023702

Avendo queste informazioni, è stato calcolato la variazione in gradi, tra il massimo e il minimo, sia per la latitudine che per la longitudine: $8.9906825 \cdot 10^{-4}$ (latitudine) e $2.2405675 \cdot 10^{-3}$ (longitudine).

Dunque:

$$40'000 : 0.3586273 = 100 : y \rightarrow y = : 8.9906825 \cdot 10^{-4}$$

$$40'000 : 0.896227 = 100 : x \rightarrow x = 2.2405675 \cdot 10^{-3}$$

Queste due proporzioni mettono in relazione la misura del lato dell'area scelta (40 km, ovvero 40'000 m) e la lunghezza in gradi decimali dello stesso, con la misura scelta per il bin, ovvero 100 m. Il risultato è la dimensione in gradi del bin. Nel primo caso per la latitudine e nel secondo per la longitudine.

Capitolo 4. Discretizzazione delle coordinate

```
1 public static void discretizza(int n, int m, double x, double y, double latMin, double latMax,
2     double longMin, double longMax, String filename) throws IOException {
3     FileReader f;
4     f = new FileReader(filename);
5     BufferedReader br;
6     br = new BufferedReader(f);
7
8
9     FileWriter w;
10    String[] p = filename.split("\\.");
11    w = new FileWriter(p[0] + "_discretizzato.txt");
12    BufferedWriter bw;
13    bw = new BufferedWriter(w);
14
15    String s;
16    while(true) {
17
18        s = br.readLine();
19        //System.out.println(s);
20        if (s == null)
21            break;
22        String[] arrOfStr = s.split("\t");
23
24        //System.out.println(arrOfStr[1]);
25
26        double a = Double.parseDouble(arrOfStr[2]);
27        double b = Double.parseDouble(arrOfStr[3]);
28
29
30        int i, j;
31
32        for (i = 0; i < n; i++) {
33            for (j = 0; j < m; j++) {
34                if (((a >= latMin + i * y) && (a <= latMin + (i+1) * y))
35                    && ((b >= longMin + j * x) && (b <= longMin + (j+1) * x))) {
36                    //utente //data
37                    String finale = arrOfStr[0] + "\t" + arrOfStr[1] + "\t" +
38                        //lat discretizzata long discretizzata idLoc
39                        Integer.toString(i+1) + "\t" + Integer.toString(j+1) + "\t" + arrOfStr[4];
40
41                    //System.out.println(finale);
42
43                    bw.write(finale);
44                    bw.write("\n");
45
46                    bw.flush();
47                }
48            }
49        }
50    }
51 }
```

Figura 12: Codice in Java per poter discretizzare le coordinate geografiche in 400 * 400 bin da 100 * 100 metri ciascuno

I passaggi più importanti di questo codice sono:

- Le variabili y,x che memorizzano la dimensione in gradi di ogni bin, rispettivamente in latitudine e in longitudine;
- Il controllo, nei due cicli for annidati, che permette di discretizzare il dato, facendolo ricadere in un bin.

La funzione è completamente parametrizzata: questo ci sarà utile successivamente.

Arrivati a questo punto, la discretizzazione delle coordinate permette di passare alla fase successiva: preparare i dati con il formato previsto dall'algorithm stigmergico.

Capitolo 5

Disposizione dei dati in base all'input previsto dal framework

Il codice che implementa l'algoritmo stigmergico, prevede una struttura dati in ingresso con le seguenti caratteristiche:

- Un array di celle 12x1: ogni cella corrisponde a un mese;
- Per ogni mese, sono presenti array di celle che corrispondono ai giorni;
- Per ogni giorno, array di celle che corrispondono ai depositi: i depositi orari sono 4 (uno ogni 15 minuti) per un totale di 96 depositi giornalieri;
- La selezione dei mesi e dei giorni da selezionare per l'analisi avviene tramite l'array coordinates, che specifica le coppie giorno-mese.

Da notare però che non è necessario in realtà formattare l'input in questo modo, se ovviamente modifichiamo la riga del codice che preleva i dati. Questa notazione è fondamentale per una fase di analisi successiva.

5.1 Codice MatLab per formattazione dati

```
1  dati = cell(12,1);
2
3  for m = 1:12
4      dati{m} = cell(31,1);
5
6      for d = 1:31
7          dati{m}{d} = cell(96,1);
8
9          for dep = 1:96
10             ora = ceil(dep/4) - 1;
11             col = dep - ora * 4;
12             if col == 1
13                 dati{m}{d}{dep} = B(B(:,1) == m & B(:,2) == d & B(:,3) == ora ...
14                     & (B(:,4) >= 0 & B(:,4) <= 14), (5:6));
15             elseif col == 2
16                 dati{m}{d}{dep} = B(B(:,1) == m & B(:,2) == d & B(:,3) == ora ...
17                     & (B(:,4) >= 15 & B(:,4) <= 29), (5:6));
18             elseif col == 3
19                 dati{m}{d}{dep} = B(B(:,1) == m & B(:,2) == d & B(:,3) == ora ...
20                     & (B(:,4) >= 30 & B(:,4) <= 44), (5:6));
21             elseif col == 4
22                 dati{m}{d}{dep} = B(B(:,1) == m & B(:,2) == d & B(:,3) == ora ...
23                     & (B(:,4) >= 45 & B(:,4) <= 59), (5:6));
24             end
25         end
26     end
27 end
```

Figura 13: Codice che per ogni mese, per ogni giorno del mese selezionato, per ogni deposito orario del giorno del mese selezionato, controlla quali dati ricadono in tale deposito

Questo codice permette di scorrere e inserire nel corretto deposito i dati appartenenti a tutti i mesi.

Noi creeremo due input separati, uno per il mese di marzo e uno per il mese di maggio e quindi sarà necessario eseguire il framework due volte, con due coordinates diversi.

Capitolo 6

Prima analisi output algoritmo stigmergico

Formattati i dati con il corretto input previsto dal codice che implementa l'algoritmo stigmergico, è possibile passare alla fase di analisi.

Inizialmente si è scelto un'analisi giornaliera, ovvero in output si aveva 31x2 grafici per mese, che sono 124 grafici da confrontare nel nostro caso, essendo i mesi considerati 2: troppo laborioso.

6.1 Analisi settimanale

Si è deciso quindi di fare un'analisi settimanale. Per settimana si intende un periodo di 7 giorni che va dal lunedì alla domenica. Quindi, qualora fosse necessario, dovrebbero essere inclusi anche i giorni appartenenti al mese precedente, che ricadono nella prima settimana del mese analizzato, e i giorni appartenenti al mese successivo che rientrano nell'ultima settimana del mese. Con questo tipo di analisi si ottengono 4x2 grafici per mese. Nel nostro caso, analizzando marzo e maggio si hanno 8 grafici.

```
13 - training = cell(12,1);
14
15 - for m = 1:12
16 -     training{m} = cell(4,1);
17
18 -     for s = 1:4
19 -         training{m}{s} = cell(672,1);
20
21 -         for dep = 1:672
22 -             giorno = ceil(dep/96);
23 -             ora = ceil((dep - 96*(giorno-1))/4) -1;
24 -             col = (dep - 96*(giorno - 1)) - ora * 4;
25 -             if col == 1
26 -                 training{m}{s}{dep} = B(B(:,1) == m & B(:,2) == (giorno+2 + (s-1) *7) & B(:, 3) == ora &
27 -
28 -             elseif col == 2
29 -                 training{m}{s}{dep} = B(B(:,1) == m & B(:,2) == (giorno+2 + (s-1) *7) & B(:, 3) == ora &
30 -
31 -             elseif col == 3
32 -                 training{m}{s}{dep} = B(B(:,1) == m & B(:,2) == (giorno+2 + (s-1) *7) & B(:, 3) == ora &
33 -
34 -             elseif col == 4
35 -                 training{m}{s}{dep} = B(B(:,1) == m & B(:,2) == (giorno+2 + (s-1) *7) & B(:, 3) == ora &
36 -
37 -             end
38 -         end
39 -     end
40 - end
```

Figura 14: Codice di Figura 13, modificato per disporre i dati per un'analisi settimanale

```
2 - par.resolution = [800 800]; % risoluzione dello spazio stigmergico
3 - par.markCoreRatio = 0.7; % rapporto basi del mark per tronco di cono
4 - par.markIntensity = 1; % intensità del mark
5 - par.maxIntensity = 1; % massima intensità modulo Ceiling
6 - par.timeWindow = 4; % larghezza finestra temporale semisovrapposta
7 - par.ltwStart = 1; % inizio finestra temporale a breve termine
8 - par.ltwEnd = 4224; % fine finestra temporale a breve termine
9 - par.hotspotThreshold = 0.5; % soglia di permanenza minima per Extraction
10
11 % parametri in DE
12 - par.markWidth = [12 12]; % larghezza del mark
13 - par.evaporation = 0.01; % evaporazione della traccia
14 - par.alpha = 0.1; % parametro smf per Contrasting
15 - par.beta = 0.6; % parametro smf per Contrasting
16
```

Figura 15: Parametri in ingresso per il framework

Come è possibile osservare in *Figura 14*, il codice è molto simile a quello utilizzato in *Figura 13*, per la disposizione mensile dei dati. La principale differenza è che ciascuna cella (rappresentante un mese), contiene non più 31 celle (rappresentanti i giorni del mese), ma 4 celle, che rappresentano le 4 settimane presenti in un mese. Una modifica riguarda invece la riga 8 del file Main.m del framework (*Figura 15*), in cui è necessario inserire il numero massimo di depositi: nel caso di un'analisi giornaliera-mensile, il numero massimo è 96 (un deposito ogni 15 minuti), mentre nel caso settimanale il numero di depositi è $96 * 7 = 672$. Dunque:

```
par.ltwEnd = 672;
```

È quindi sempre possibile utilizzare il framework senza ulteriori modifiche, inserendo in questo caso nell'array coordinates, la coppia settimana-mese.

Questo tipo di analisi non ha tuttavia riportato risultati utili: l'obiettivo era quello di andare a confrontare la settimana in cui avveniva il festival (la terza di marzo e parte della seconda, che quindi risulta essere una settimana "ibrida", da non prendere in considerazione), con le altre settimane dello stesso mese e del mese di maggio.

Non sono state osservate però dai grafici notevoli differenze. Questo ci ha indotto a pensare che la zona geografica considerata fosse troppo grande e che i check-in si concentrassero maggiormente in una zona più ristretta. Inoltre, un'analisi settimanale è risultata troppo grossolana: non permette cioè di mettere in evidenza le differenze tra giorni diversi.

6.2 Zona geografica ristretta

Sono stati prodotti i seguenti grafici dunque, tramite lo stesso codice, utilizzato inizialmente, per verificare la frequenza dei login a livello globale. In questo caso l'analisi è ristretta alla sola zona di Austin precedentemente considerata.

Capitolo 6. Prima analisi output algoritmo stigmergico

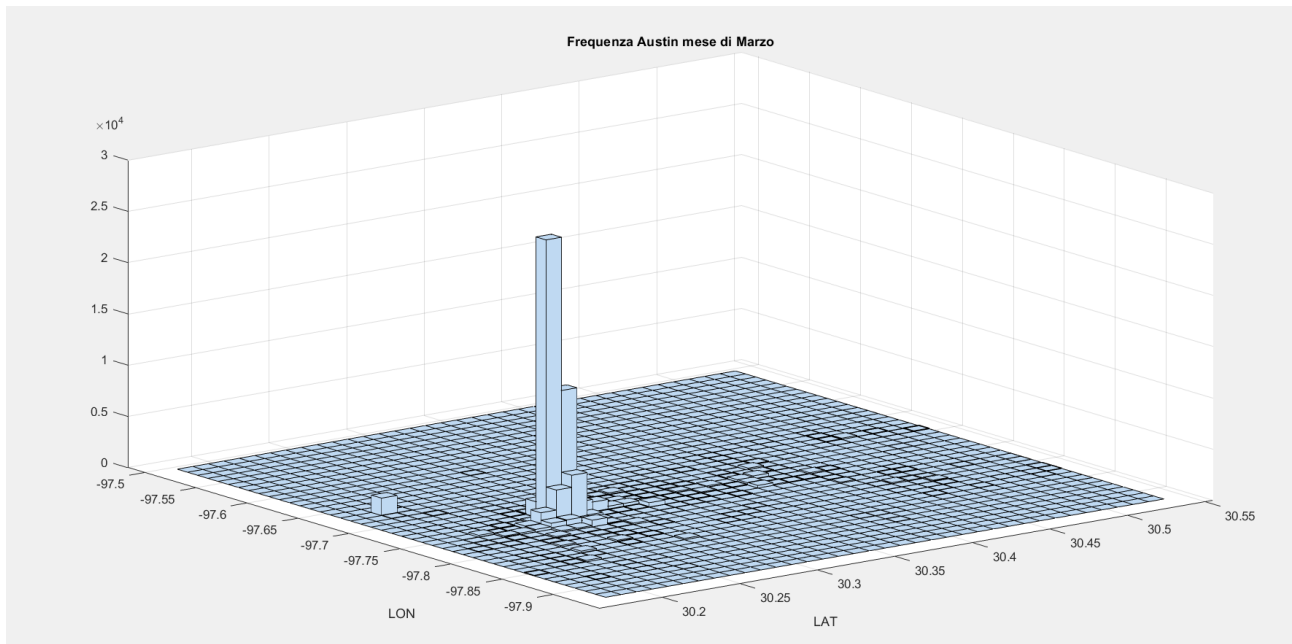


Figura 16: Frequenza di Austin, mese di marzo. Si può notare come la maggior parte degli accessi sia concentrata in pochi bin

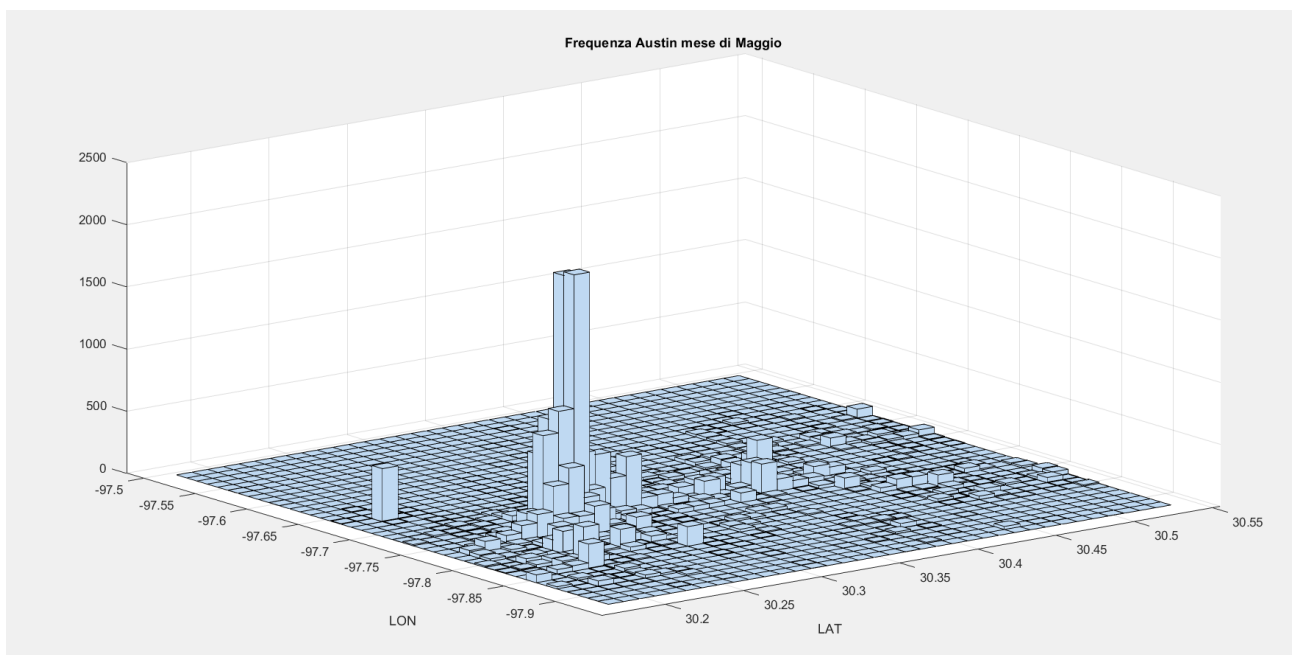


Figura 17: Frequenza di Austin, mese di maggio. Si può notare come la maggior parte degli accessi sia concentrata in pochi bin

Si può notare da questa analisi inoltre che gli accessi al social network nel mese di marzo sono maggiormente concentrati in una zona (il centro di Austin), rispetto al mese di maggio, che sono più distribuiti in tutta la città: in realtà uno zoom del grafico di marzo, evidenzia sempre una presenza costante in numero di login nella zona periferica di Austin, ma l'enorme aumento nel centro di Austin, fa sì che la scala del grafico consideri il numero di login nella zona periferica, tendenti a zero.

Capitolo 6. Prima analisi output algoritmo stigmergico

Si è deciso dunque di estrarre i dati relativi solamente alla seguente zona, più ristretta rispetto alla precedente (coordinate dei vertici del quadrato):

- 1) 30.45, -97.85
- 2) 30.45, -97.65
- 3) 30.15, -97.85
- 4) 30.15, -97.65

Il lato di questa nuova area è di 20 km. Si è deciso quindi di utilizzare una dimensione pari a 25 x 25 metri per i bin.

La variazione in gradi decimali è la seguente:

$$0.3 : 20'000 = y : 25 \rightarrow y = 3.75 * 10^{-4}$$

$$0.2 : 20'000 = x : 25 \rightarrow x = 2.5 * 10^{-4}$$

6.3 Necessità di un'analisi più dettagliata

Come accennato precedentemente alla fine del paragrafo 6.1, si è ritenuto, visti i risultati non concludenti di un'analisi settimanale, di fare un'analisi maggiormente dettagliata, suddividendo i giorni fra quelli appartenenti al weekend e ai weekdays, unendo quelli di marzo e di maggio, senza fare distinzione fra i due mesi.

Inoltre, è prevista un'analisi dei soli giorni in cui è avvenuto il festival (venerdì 12 marzo – domenica 21 marzo). Questa analisi non ha nulla di diverso dalla normale analisi fatta inizialmente nel capitolo 6, ovvero mensile e giornaliera. Dunque, per i passi preliminari all'output del framework, ci concentreremo solamente su quella relativa ai weekend e ai weekdays.

6.4 Analisi weekend e weekdays

```
214 public static boolean isWeekend(String ts) {
215     //2009-03-17,02
216     String[] data = ts.split(",");
217     String[] info = data[0].split("-");
218     int year = Integer.parseInt(info[0]);
219     int month = Integer.parseInt(info[1]);
220     int day = Integer.parseInt(info[2]);
221
222     Calendar cal = new GregorianCalendar(year, month - 1, day);
223     int dayOfWeek = cal.get(Calendar.DAY_OF_WEEK);
224     return (Calendar.SUNDAY == dayOfWeek || Calendar.SATURDAY == dayOfWeek);
225 }
```

Figura 18: Funzione che viene invocata su ogni riga letta dal file .txt. Ritorna true, qualora il giorno considerato appartenga al weekend

Con un codice simile a quello utilizzato in *Figura 14*, con l'ausilio di un codice scritto in Java che suddivide i dati fra quelli appartenenti al weekend e quelli appartenenti ai weekdays (*Figura 18*), si è prodotto due cell array, uno con i soli dati del weekend e uno con i soli dati dei weekdays.

Capitolo 6. Prima analisi output algoritmo stigmergico

Un input così fatto, ha reso però non più necessario l'utilizzo dell'array *coordinates* nel file *Main.m* del framework. In quanto le due coordinate nel framework specificano (nel caso di un'analisi giornaliera), la coppia giorno – mese, mentre nel caso dell'analisi settimanale la coppia settimana – mese.

In questo caso è sufficiente prendere tutto il contenuto dell'array, che contiene già tutti i depositi.

È stato dunque necessario modificare, oltre al parametro *par.ltwEnd* di *Figura 15*, la riga di codice 40 del file *RunFramework.m*:

```
runResult.hotspotSpace{i,1} = sf.compute(training{month,1}{day,1}(par.ltwStart:par.ltwEnd,1));
```

nella seguente riga:

```
runResult.hotspotSpace{i,1} = sf.compute(training par.ltwStart:par.ltwEnd,1));
```

Si può notare come le due coordinate *month*, *day* non siano più utilizzate, per selezionare i dati all'interno dell'array.

Oltre a questa modifica è stato necessario modificare, come già accennato, il parametro *par.ltwEnd*, che indica il numero massimo di depositi. Già nell'analisi settimanale era stato necessario modificare questo parametro, settandolo a $96 * 7$. In questo caso il numero di weekend in marzo e maggio è 18 e il numero di weekdays è 44. Dunque, il numero massimo di depositi nell'esecuzione dell'algoritmo stigmergico per weekend è $96 * 18$, e per weekdays è $96 * 44$.

Quindi, gli output con questo tipo di analisi sono:

- Due grafici relativi ai weekend
- Due grafici relativi ai weekdays

Oltre a questi, $10 * 2$ grafi sono prodotti relativamente ai 10 giorni del festival, così da poter osservare i cambiamenti dovuti a questo evento internazionale.

Capitolo 7

Seconda analisi output algoritmo stigmergico

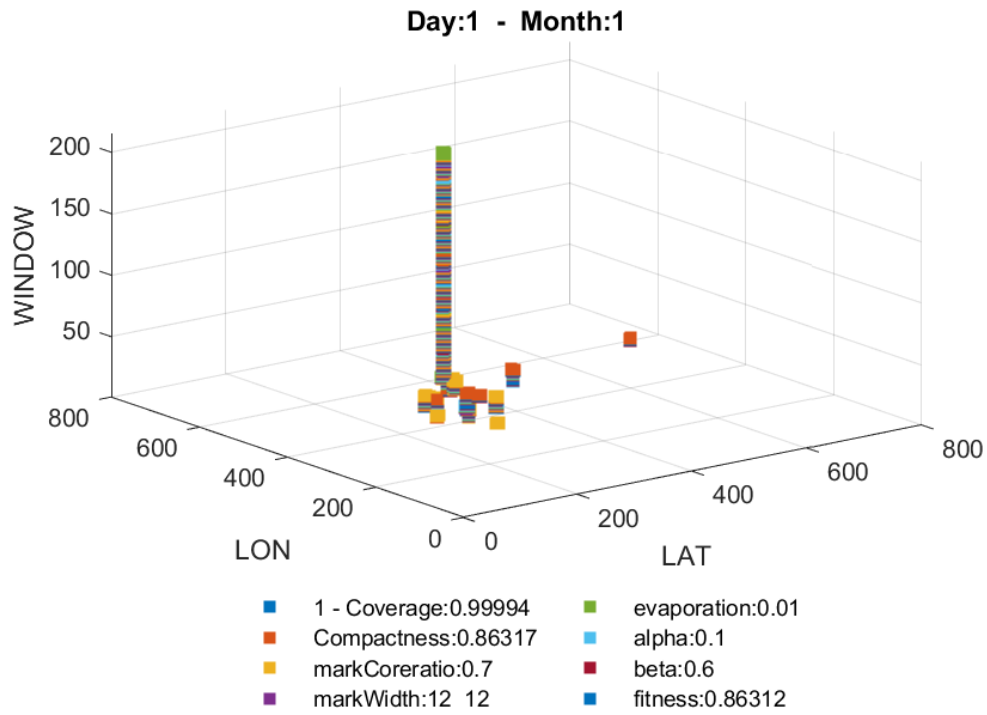


Figura 19: Grafico 3D relativo ai weekend, output dell'algoritmo stigmergico

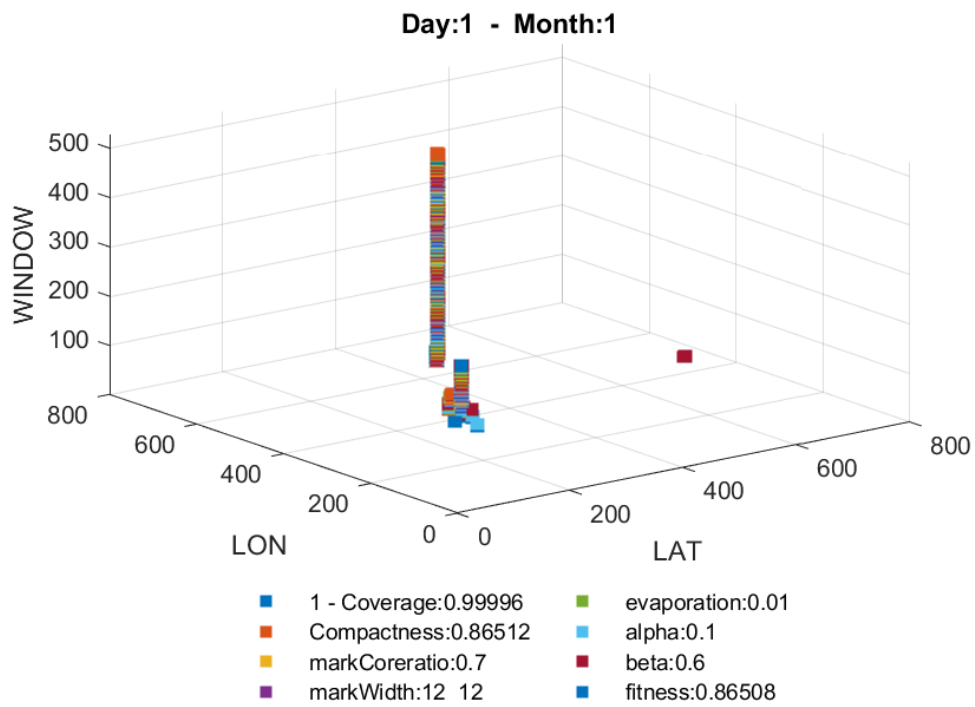


Figura 20: Grafico 3D relativo ai weekdays, output dell'algoritmo stigmergico

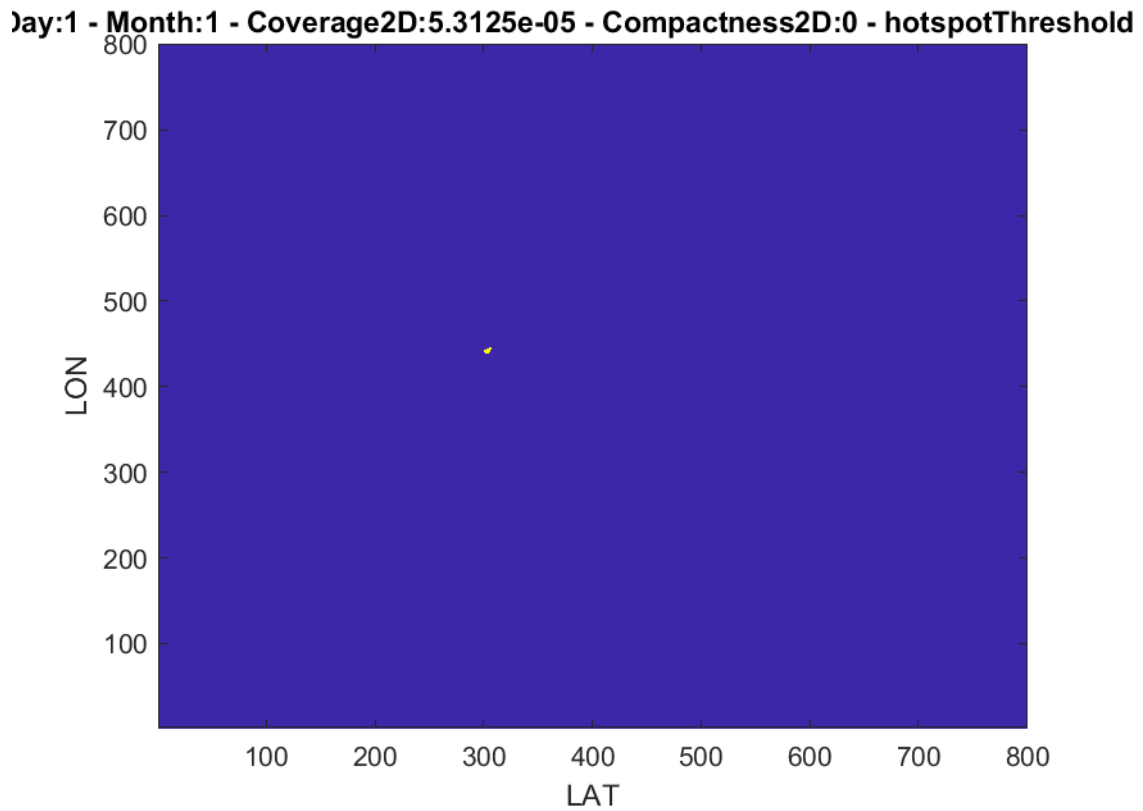


Figura 21: Grafico 2D relativo ai weekend, output dell'algoritmo stigmergico

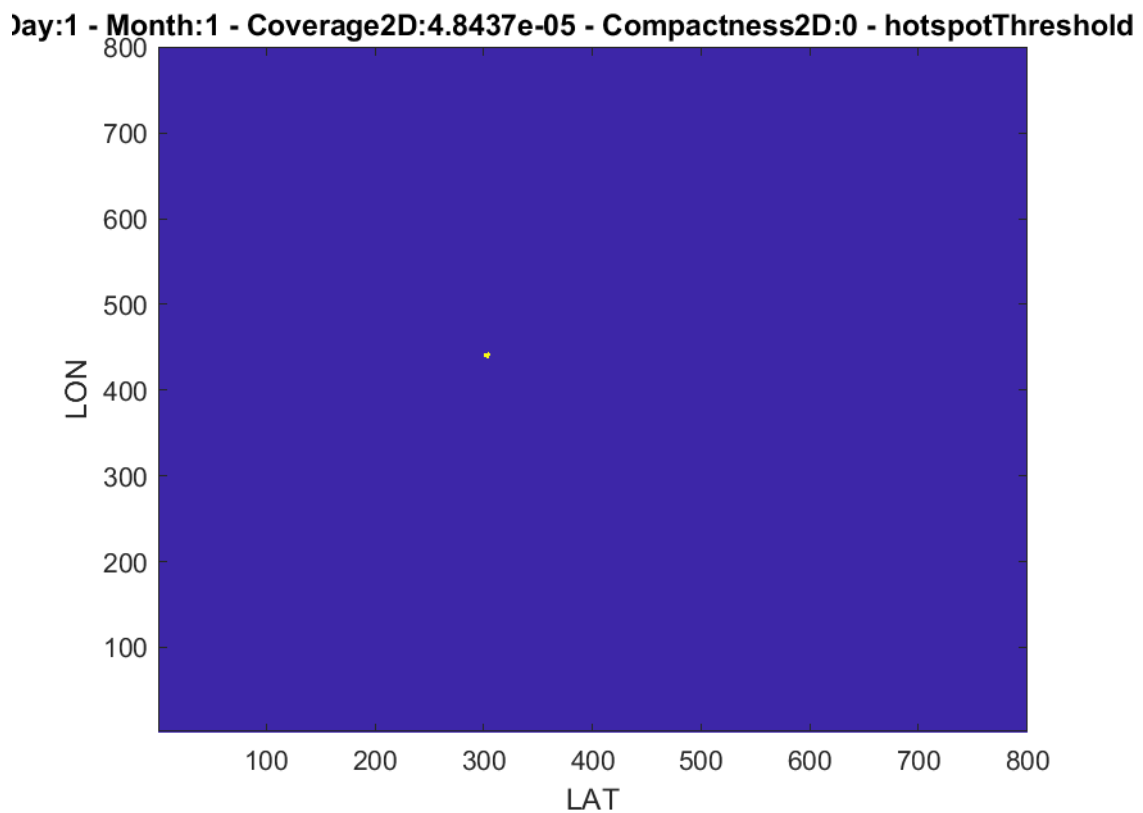


Figura 22: Grafico 2D relativo ai weekdays, output dell'algoritmo stigmergico

Sono stati prodotti dei grafici simili a questi precedentemente, ma una fase di debugging del codice ha evidenziato un errore nel codice di disposizione dei dati in base all'input previsto dal codice del framework. Dunque, è stato necessario, una volta corretto l'errore eseguire nuovamente il framework sui nuovi dati. L'errore aveva come conseguenza, quella di scartare dei dati, senza inserirli in nessuna cella dell'array di input. I dati effettivamente inseriti però, erano inseriti nella corretta posizione nell'array. Un'osservazione interessante quindi è che comunque l'analisi fatta precedentemente sulla base dell'output fornito dai dati immessi con la presenza dell'errore, era identica all'analisi che segue. La presenza dell'errore aveva infatti solamente fatto sì che alcuni dati venissero scartati: dunque la differenza la si trova nel fatto che i fenomeni osservati nel secondo caso (quello con assenza dell'errore), sono identici a quelli del primo caso (con presenza dell'errore), ma più marcati, in quanto accentuati e "giustificati" da un maggior numero di dati.

Infine, un'importante osservazione da fare è come i grafici in output dal framework, non siano dei semplici grafici con istogrammi, che misurano l'intensità di login nello spazio, come può essere per esempio per esempio per il grafico in *Figura 7*. Infatti, quello che il framework esegue, è un'analisi spazio-temporale della presenza dei login: quindi non è solo una mappatura geografica, ma anche appunto temporale.

7.1 Analisi weekend

La maggior concentrazione si ha nelle coordinate coincidenti con l'Austin Convention Center (*Figura 21*). Questa struttura ha un perimetro rettangolare di circa 300 x 200 metri ed ospita eventi sia nel weekend che nei weekdays (come vedremo sarà il primo hotspot anche per i weekdays). La sua grande struttura fa sì che sia uno, se non il principale punto di ritrovo di Austin per qualsiasi tipo di evento, sia lavorativo che di svago. Essa coincide con i bin 303, 444 (e adiacenti, vista la sua grande dimensione) e grazie al codice in *Figura 23* è possibile risalire alle coordinate reali.

```
202 public static void getCoordinate (double x, double y, double latMin, double longMin, int latDisc, int longDisc) {
203     double supLat = latMin + y * (latDisc);
204     double infLat = latMin + y * (latDisc - 1);
205
206     double supLong = longMin + x * (longDisc);
207     double infLong = longMin + x * (longDisc - 1);
208
209     //System.out.println("Latitudine - sup: " + supLat + " / inf: " + infLat);
210     //System.out.println("Longitudine - sup: " + supLong + " / inf: " + infLong);
211     System.out.println(supLat + ", " + supLong);
212 }
```

Figura 23: Codice Java, utilizzato per la riconversione delle coordinate discretizzate in coordinate reali

Anche dalla *Figura 19* si può notare (anche se la prospettiva, falsa il valore delle coordinate), la presenza nel tempo di login nella zona del Austin Convention Center.

L'hotspot si estende (in minor misura però) anche nelle strade vicino a questa struttura, come si può notare sempre dalla *Figura 19*, grafico in 3D.

7.2 Analisi weekdays

Il grafico mostra un hotspot coincidente, come sopra accennato, con l'Austin Convention Center (Figura 22). I motivi per cui questa struttura attrae persone, e dunque login al social network, sia nei weekend, che nei weekdays è da ritrovarsi nella presenza di eventi, anche di contenuto diverso, in entrambi i periodi.

Questo hotspot corrisponde ai bin 303, 444 e per ulteriore chiarezza nel grafico di Figura 20 è rappresentato dall'hotspot "più lontano" per intendersi.

Troviamo però anche un hotspot (Figura 20), quello più vicino all'origine, con coordinate 317, 405, in corrispondenza di una zona con attività prettamente inerenti al mondo del lavoro (Figura 24):

- Horizon Bank
- Agenzia di assicurazioni
- Peak Capital Funding
- The Hay Legal Group
- BB&T (bank)

Tutto questo è in linea con il fatto che l'analisi si concentra sui giorni infrasettimanali, ovvero quelli lavorativi.

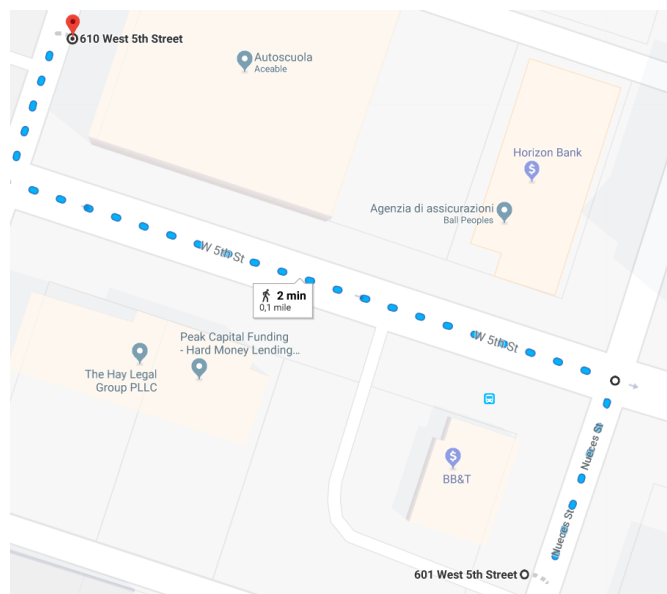


Figura 24: Secondo hotspot, relativo ai weekdays, si trova in corrispondenza di questa zona

7.3 Analisi festival ‘South by Southwest’ festival

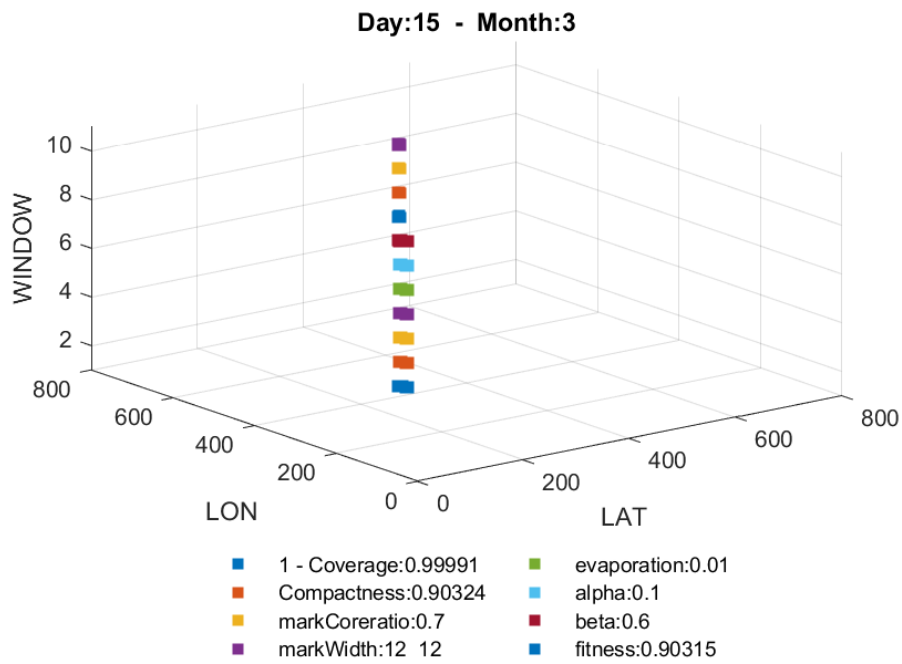


Figura 25: 15 marzo, grafico 3D

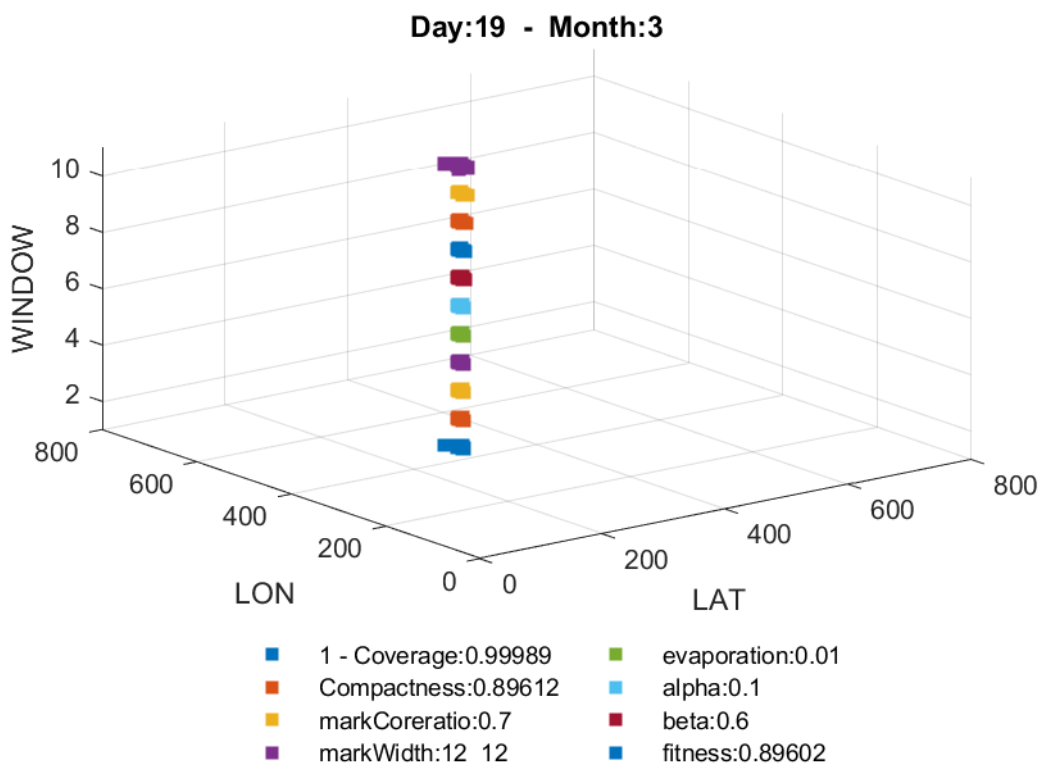


Figura 26: 19 marzo, grafico 3D

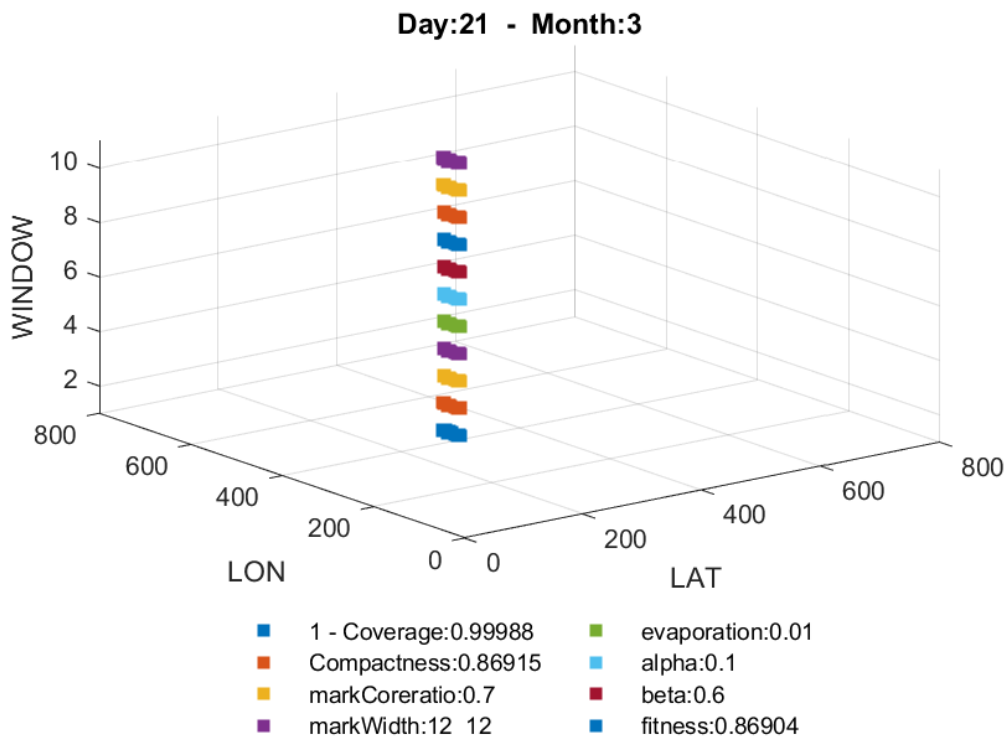


Figura 27: 21 marzo, grafico 3D

ay:15 - Month:3 - Coverage2D:0.00012187 - Compactness2D:0 - hotspotThreshol

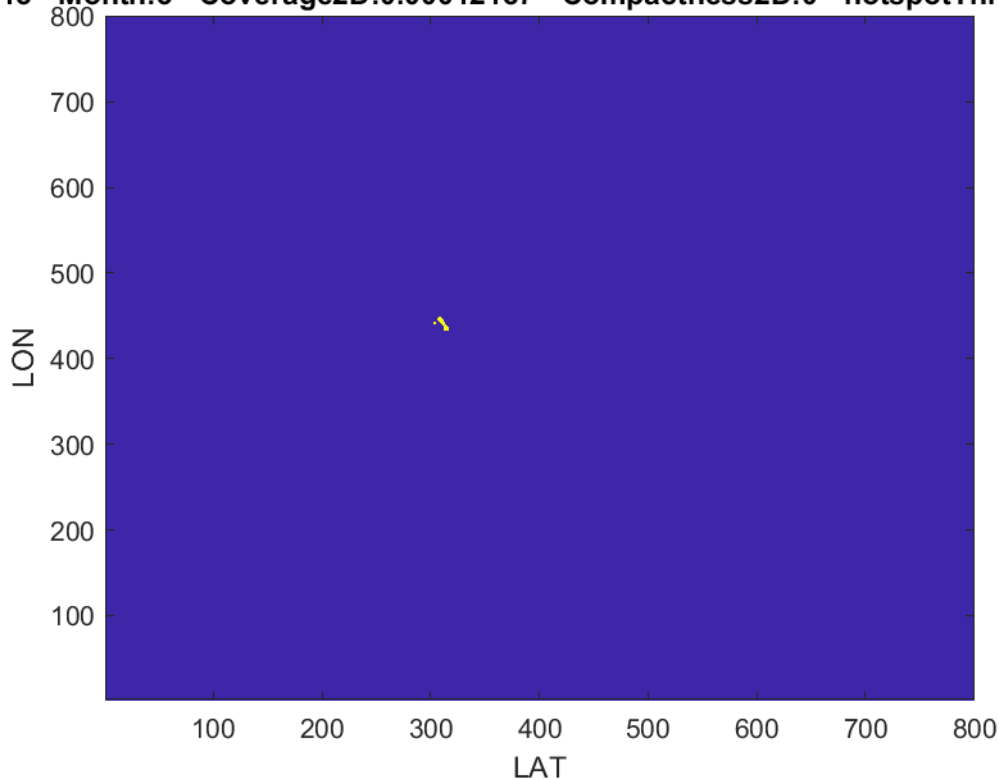


Figura 28: 15 marzo, grafico 2D

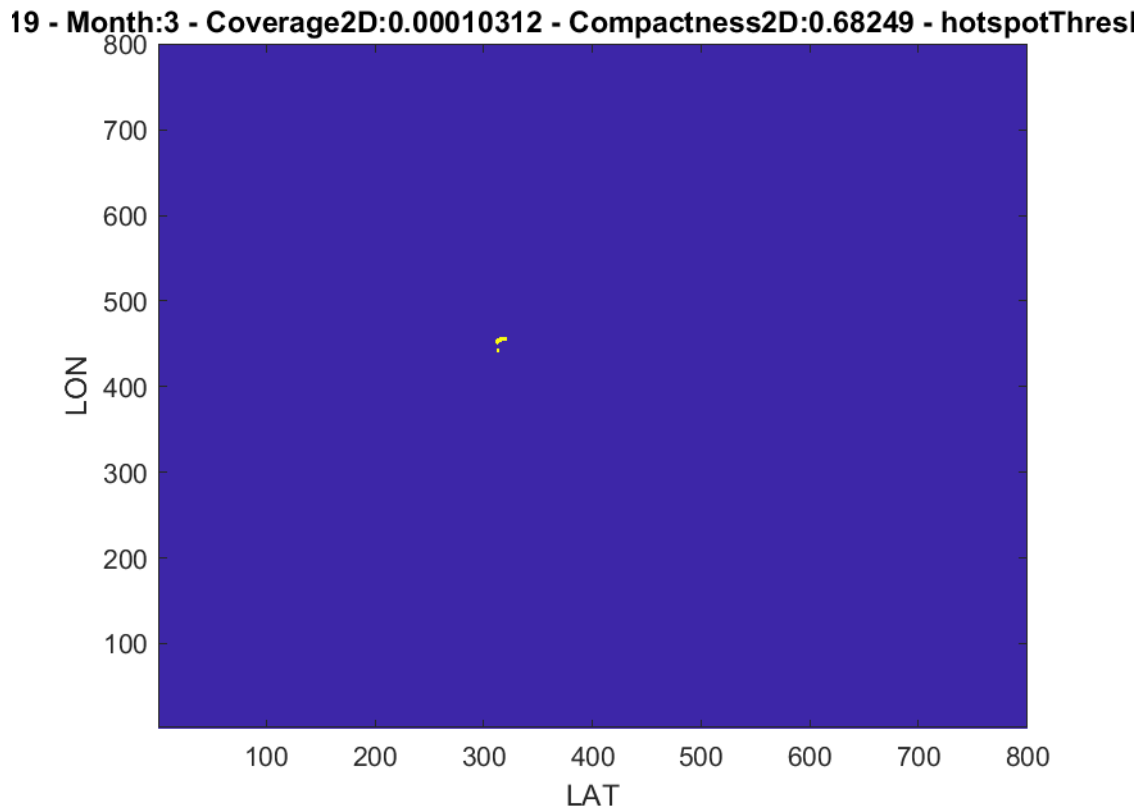


Figura 29: 19 marzo, grafico 2D

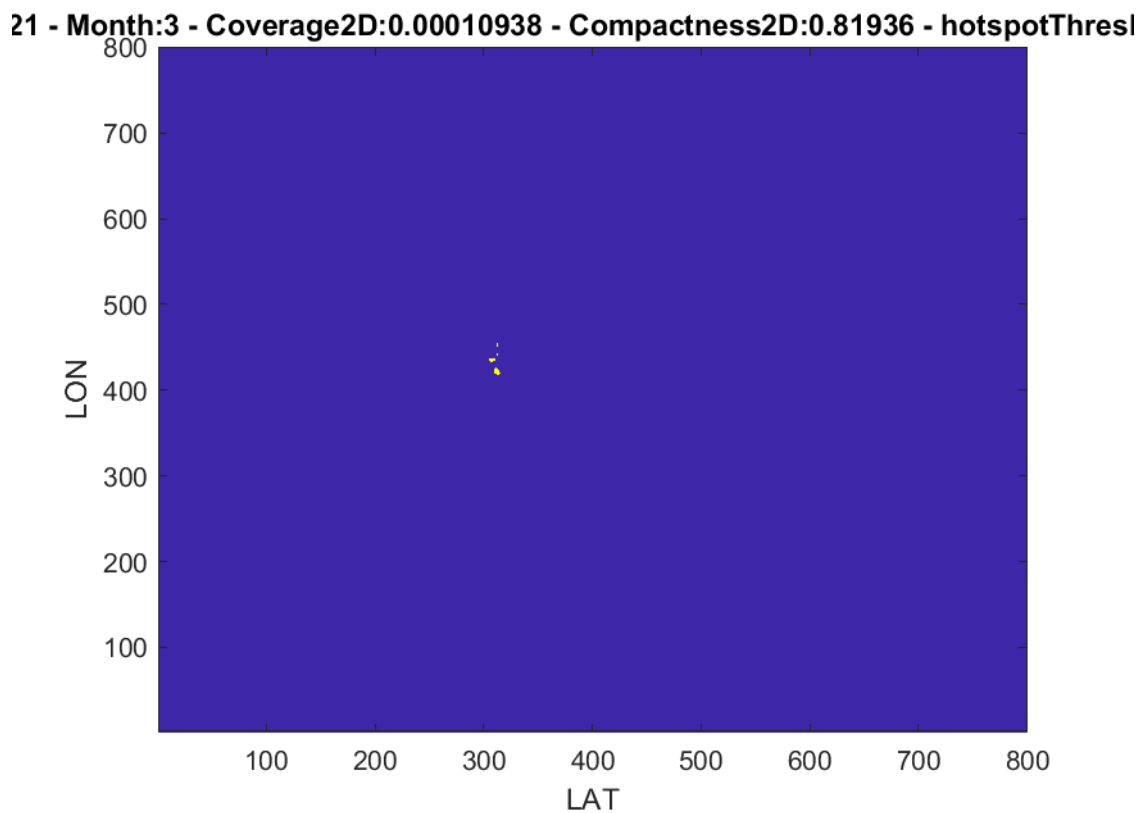


Figura 30: 21 marzo, grafico 2D

Capitolo 7. Seconda analisi output algoritmo stigmergico

Ricordiamo che il festival si svolge nei giorni da venerdì 12 marzo a domenica 21 marzo 2010.

Sono state riportate *Figura 25 – 30*, a titolo esemplificativo dei 10 giorni del festival, che risultano essere tutti identici o quasi nei grafici.

Il festival, come si può immaginare visto l'enorme struttura che mette a disposizione, si concentra soprattutto nell'Austin Convention Center (unico hotspot presente nei grafici).

Questo però non è l'unico luogo dove si svolge, è infatti un festival internazionale, che quindi necessariamente coinvolge altre strutture di Austin, che però si trovano sempre nei pressi e nelle vicinanze dell'Austin Convention Center: dunque fanno parte del solito hotspot nel grafico, che si estende per una decina di bin, che nella realtà corrispondono ad un centinaio di metri.

Essendo il festival composto da molti eventi, sono necessarie molti tipi di struttura diverse: nella maggior parte dei casi, queste sono hotel, che mettono a disposizione le loro sale, ma anche musei e locali.

In *Figura 31* sono stati segnati 3 esempi di questi luoghi:

- Mexic-Arte Museum (in alto a sinistra)
- The Westin Austin Downtown (in alto centrale)
- Hilton Austin (in basso a destra)

Un'osservazione che si può fare è che in basso a destra si trova l'Austin Convention Center, più volte nominato nell'analisi e che i luoghi sopra citati sono tutti nel raggio di poche centinaia di metri da questa struttura.

Questi luoghi in cui si svolgono gli eventi del festival, sono stati trovati grazie al sito web ufficiale [2] del festival, in cui sono elencati i luoghi in cui si tengono le conferenze.

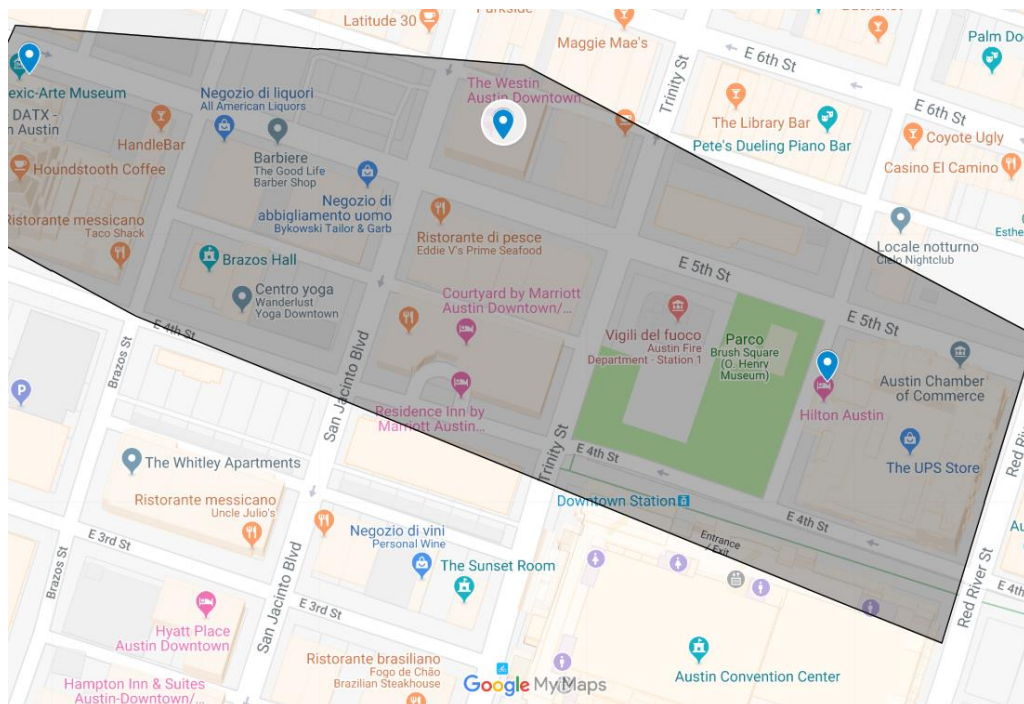


Figura 31: Zona in cui si svolge il 'South by Southwest' festival

Ci sono molti altri luoghi, sempre appartenenti all'hotspot che i grafici mostrano, luoghi che comunque si trovano in prossimità del centro di Austin e in particolar modo dell'Austin Convention Center.

Conclusioni

È stato possibile quindi arrivare ad un'analisi, benché non troppo approfondita a livello urbanistico della città di Austin, grazie ai grafici prodotti dal framework, che utilizza tecniche stigmergiche. Un'ulteriore analisi dei grafici, per una maggior comprensione dell'utilizzo delle aree di Austin, andrebbe svolta con la collaborazione di esperti di urbanistica. Per questo motivo non ci siamo addentrati maggiormente in un'analisi di questo tipo.

Una naturale continuazione dello studio fatto fin qui, porterebbe a un'analisi più dettagliata della zona in cui si svolge il festival preso in esame, restringendo ulteriormente la zona da cui estrarre i dati.

È inoltre possibile riutilizzare il codice scritto, per l'analisi di altri dati: ad esempio ricordo, come un'altra città con un alto utilizzo del social network Gowalla, fosse Stoccolma.

Bibliografia

- [1] Simone Musetti, *Using evolutionary optimization and computational stigmergy to detect purchase hotspots from spatio-temporal credit card transactions*. Master Thesis in Computer Engineering – 2019.
- [2] <https://www.sxsw.com/> , sito web ufficiale del ‘South by Southwest’ festival.
- [3] <https://www.sunearthtools.com/it/tools/distance.php>, sito web con tool di supporto nella scelta dell’area da cui prelevare i dati.
- [4] <https://it.mathworks.com/help/matlab/> , documentazione ufficiale MatLab.
- [5] <https://it.wikipedia.org/wiki/Stigmergia> , definizione di stigmergia in natura.

Ringraziamenti

Alla mia famiglia per essermi stata vicino durante tutto il percorso, per avermi supportato nei momenti di difficoltà incontrati durante la mia vita personale e universitaria.

Ad alcuni dei miei amici, per avermi strappato una risata nei momenti più stressanti, durante i periodi di esame.

Ringrazio me stesso, per essermi riuscito a rialzare, per essermi ricordato sempre in ogni difficoltà incontrata, che “passerà anche questa stazione senza far male”.