UNIVERSITÀ
DI PARMA

Dipartimento di Ingegneria e Architettura
Corso di Laurea in Ingegneria Informatica

Approssimazione di funzioni di distribuzione cumulativa discreta

ANNO ACCADEMICO 2022-2023

Indice

Capitolo 1

Modello Matematico

1.1 Descrizione del problema

Il progetto si pone l'obiettivo di definire una funzione di distribuzione cumulativa G che approssima una funzione F, definite su un insieme {1..N}; in particolare G é vincolata da assumere al massimo n valori distinti, con n < N. Inoltre la funzione F é vincolata dall'uguaglianza F(i) = y(i) e con F(i)≥F(i-1), per ogni i definito in {2..N}. Poste queste condizione si vuole scegliere G in modo che l'errore assoluto della somma delle differenza tra F e G sia il piú piccolo possibile; ci si trova quindi in un problema di minimo.

1.2 Componenti del modello

Le componenti del modello:

1.2.1 Insiemi

In questo problema si hanno tre differenti insiemi:

- {1..N} insieme di F;
- {2..N} insieme per il vincolo di F;
- {1..n} insieme per la definizione di G;

1.2.2 Variabili

La variabile di questo problema risultano essere:

- la funzione G, che essendo una funzione cumulativa puó assumere solo valori tra 0 e 1;
- y_r, variabile utilizzata per la formulazione dell'obiettivo;
- salto, variabile binaria per decidere la posizione dei salti;

1.2.3 Parametri

In questo problema si hanno diversi parametri:

- n, il numero di valori distinti che G puó assumere;
- N, la dimensione degli insiemi su cui sono definite F e G;
- y(i), i valori che assume la funzione F(i);
- F(i), la funzione di partenza che é approssimata da G e puó assumere valori solo tra 0 e 1;

1.2.4 Vincoli

In questo problema si hanno diversi vincoli:

- $0 \leq F(i) \leq 1$;
- $0 \leq G(i) \leq 1$;
- $F(i) \geq F(i-1)$;
- $G(i) \geq G(i-1)$;
- $n < N$;

1.2.5 Funzione obiettivo

Si vuole scegliere G in modo che l'errore assoluto della differenza tra F e G sia il piú piccolo possibile quindi min $|F(i) - G(i)|$

Capitolo 2

Modello AMPL

2.1 File .mod

Si riporta il codice del file .mod

```
param n; #Numero di valori distinti che G può assumere
param N; #Dimensione degli insiemi su cui sono definite F e G


set T:= 1..N; #Insieme su cui vengono definite F e G
set R:= 2..N; #Insieme su cui vengono definiti i vincoli
set A:= 1..n; #Insieme per l' approssimazione

param y_i{T}; #Valore che assume la funzione F
param F{i in T} = y_i[i]; #Assegnazione dei valori di y a F


var G{T} <= 1 >= 0; #Definizione della variabile G
var y_r{T}; #Variabile per l' obiettivo
var salto{i in T} binary; # variabile binaria per la posizione del salto

subject to vincolo_FS {i in R} : F[i] >= F[i-1]; #vincolo di continuità del lim dell' intorno destro della funzione F
subject to vincolo_GS {i in R} : G[i] >= G[i-1]; #vincolo di continuità del lim dell' intorno destro della funzione G


subject to valSalto : sum{i in T} (salto[i]) = n-1; #vincolo che assegna il numero di salti da fare
subject to vincoloBinZero{i in T : i > 1} : G[i] <= G[i-1] + salto[i]; #vincolo secondo cui nel caso in cui salto[i] è 1 allora G[i]<=1
                                                                       #e nel caso in cui sia uguale a zero G[i]<= G[i-1]

# Vincoli che permettono di rendere la funzione obiettivo lineare
subject to abs1 {i in T} : y_r[i] >= (F[i] - G[i]); #Vincolo che impone che y_r sia maggiore o uguale della differenza
subject to abs2 {i in T} : y_r[i] >= -(F[i] - G[i]); #Vincolo che impone che y_r sia maggiore o uguale della differenza negata

#minimize minimo_salto :
minimize errore_totale : sum{i in T} (y_r[i]);
```

Figura 2.1: File. mod

Capitolo 3

Implementazione e Considerazioni

Posto l'obiettivo del problema, si é proceduto impostando i vincoli di continuitá del limite dell'intorno destro per la funzione F e G cioé vincolo_FS e vincolo_GS. Successivamente si é proceduto definendo i due vincoli per l'approssimazione di F:

- valSalto, secondo cui il numero di salti da eseguire é pari a n-1;
- vincoloBinZero, che impone che la funzione G sia minore o uguale a 1 quando salto é pari a 1 e che invece sia minore o uguale a G(i-1) quando salto é uguale a 0;

Per quanto riguarda l'obiettivo si é proceduto impostando due vincoli:

- abs1, secondo cui il valore di y_r deve essere maggiore o uguale della differenza tra F e G;
- abs2, secondo cui il valore di y_r deve essere maggiore o uguale della differenza negata tra F e G ;

Infine si é calcolato l'errore totale come la somma di y_r.

Capitolo 4

Risultati

Si mostrano i risultati ottenuti con i diversi file .dat:

- Risultati con Cumulativa.dat.

```
Gurobi 4.0.1: optimal solution; objective 0.4
49 simplex iterations
plus 21 simplex iterations for intbasis
salto [*] :=
1  0
2  1
3  0
4  1
5  0
6  1
7  0
8  1
9  0
;

G [*] :=
1  0.1
2  0.2
3  0.2
4  0.5
5  0.5
6  0.6
7  0.6
8  1
9  1
;

errore_totale = 0.4
```

Figura 4.1: Risultati con Cumulativa.dat

- Risultati con Cumulativa1.dat.

```
Gurobi 4.0.1: optimal solution; objective 0.37
5821 simplex iterations
2318 branch-and-cut nodes
plus 78 simplex iterations for intbasis
absmipgap = 5.55e-17, relmipgap = 1.5e-16
salto [*] :=
 1 0    4 0    7 0   10 0   13 0   16 0   19 0   22 0   25 0   28 0
 2 1    5 0    8 1   11 1   14 0   17 1   20 1   23 0   26 1   29 0
 3 1    6 1    9 0   12 1   15 1   18 0   21 0   24 1   27 0   30 1
;

G [*] :=
 1 0.1     5 0.23    9 0.4    13 0.5    17 0.61   21 0.7    25 0.76   29 0.89
 2 0.15    6 0.3    10 0.4    14 0.5    18 0.61   22 0.7    26 0.89   30 1
 3 0.23    7 0.3    11 0.45   15 0.54   19 0.61   23 0.7    27 0.89
 4 0.23    8 0.4    12 0.5    16 0.54   20 0.7    24 0.76   28 0.89
;

errore_totale = 0.37
```

Figura 4.2: Risultati con Cumulativa1.dat

- Risultati con Cumulativa2.dat.

```
Gurobi 4.0.1: optimal solution; objective 0.37
125 simplex iterations
15 branch-and-cut nodes
plus 33 simplex iterations for intbasis
absmipgap = 5.55e-17, relmipgap = 1.5e-16
salto [*] :=
 1  0
 2  1
 3  1
 4  0
 5  0
 6  0
 7  1
 8  0
 9  0
10  0
11  1
12  0
;

G [*] :=
 1  0.01
 2  0.28
 3  0.48
 4  0.48
 5  0.48
 6  0.48
 7  0.79
 8  0.79
 9  0.79
10  0.79
11  1
12  1
;

errore_totale = 0.37
```

Figura 4.3: Risultati con Cumulativa2.dat