Stima downloads dell'app Immuni

Max Pierini

*info@maxpierini.it

June 23, 2020

1 Metodo

Quanti hanno scaricato l'app Immuni?

I dati reali sono visibili solo agli sviluppatori, ma è possibile ottenerne una stima dal numero di recensioni ricevute, tramite inferenza bayesiana.

Su App Store (Apple) viene mostrato solo il numero di recensioni ma su Play Store (Google), oltre alle recensioni, è mostrato un numero dei downloads arrontondato per difetto alle potenze di 10 o alle potenze di 10 moltiplicate per 5 (100, 500, 1'000, 5'000, 10'000, 50'000, 100'000, 500'000, ecc). Dunque se su Play Store leggiamo "installazioni 1.000.000+", il dato reale potrebbero essere tra 1'000'000 e 4'999'999.

Per ciascun giorno dalla pubblicazione dell'app (1 Giugno 2020), vogliamo ottenere, secondo il Teorema di Bayes, $P(D|R)$ la probabilità a posteriori di D downloads date R recensioni

$$P(D|R) = \frac{P(R|D)P(D)}{P(R)}$$

dove al numeratore $P(R|D)$ è la verosimiglianza di ricevere R recensioni dati tutti i possibili downloads e $P(D)$ è la probabilità a priori di avere D downloads, mentre al denominatore $P(R)$ è la probabilità marginale a priori di aver ricevuto R recensioni. Limite del metodo è supporre una funzione di verosimiglianza $P(R|D)$ costante nel tempo: il numero di recensioni non dipende solo dai downloads ma anche da nuovi aggiornamenti (che possono portare ad ondate di recensioni positive o negative) e dalla data di pubblicazione dell'app. Per semplicità però supporremo qui che rimanga costante.

Sappiamo dal Ministero della Salute che il 3 giugno 2020 l'app era stata scaricata globalmente da circa un milione di utenti (link) e il 21 Giugno 2020 da circa 3 milioni (link). Questi unici dati riportati ufficialmente serviranno da controllo per la bontà del metodo.

1.1 Priori

Ci interessano i valori di D non superiori all'attuale popolazione italiana pari a circa 60 milioni, quindi

$$0 \leq D \leq 6 \cdot 10^7$$

e supponiamo per $P(D)$ una distribuzione di probabilità a priori non vincolante

$$P(D) \sim \textbf{Gamma}(\omega = D, \sigma = 10^6)$$

ovvero una distribuzione **Gamma** con moda D e un'ampia deviazione standard di 1 milione (figura 1).

Suddivideremo l'intervallo scelto in *steps*, ad esempio 6000, per evitare di processare matrici con un numero eccessivamente elevato di elementi.

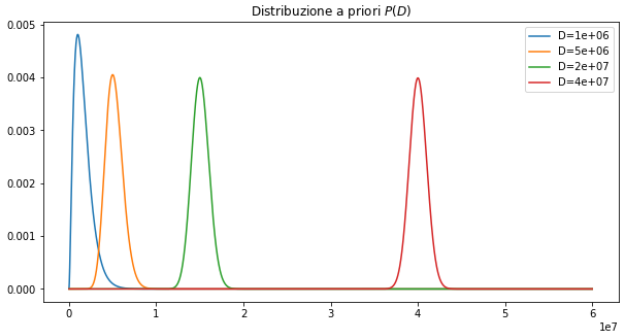

Figure 1: Esempi di distribuzione a priori $P(D) \sim \textbf{Gamma}(\omega = D, \sigma = 10^6)$

Per il primo giorno, non avendo alcuna informazione precedente, supporremo una distribuzione continua uniforme.

1.2 Verosimiglianza

Per ottenere la verosimiglianza $P(R|D)$ deriviamo una funzione empirica dai dati di altre app su Play Store.

Come detto sopra, su Play Store viene riportato un numero di downloads arrotondato per difetto alle potenze di 10 o alle potenze di 10 moltiplicate per 5. Suddivideremo perciò le app sul Play Store in *classi di downloads* in base al valore nominale dichiarato $\mathbb{D}$ e supponendo il valore medio $\overline{D}$ di downloads tra il minimo e il massimo come indicativo di ciascuna classe, ovvero

$$\begin{cases} \overline{D} = [\mathbb{D} + (5 \cdot \mathbb{D} - 1)]/2 & , \lfloor \log_{10} \mathbb{D} \rfloor = \log_{10} \mathbb{D} \\ \overline{D} = [\mathbb{D} + (2 \cdot \mathbb{D} - 1)]/2 & , \lfloor \log_{10} \mathbb{D} \rfloor \neq \log_{10} \mathbb{D} \end{cases}$$

Abbiamo (attualmente) a disposizione i dati di recensioni e valori di downloads nominali $\mathbb{D}$ di 1064 apps dal Play Store (vedi allegato), da cui per ogni classe possiamo ricavare una distribuzione di probabilità delle recensioni.

Presupporremo una distribuzione **Gamma** con moda ω e deviazione standard σ in funzione del valore medio $\overline{D}$ delle classi $\mathbb{D}$ di downloads

$$P(R|\mathbb{D}) \sim \textbf{Gamma}(\omega = a \cdot \overline{D}, \sigma = b \cdot \omega)$$

ed effettueremo test di Kolmogorov-Smirmov (KS) sui valori di a e b maggiormente verosimili, ovvero $a \in \{200...2000\}$ e $b \in \{1...20\}$ (figura 2).

Sceglieremo la distribuzione con p-value del KS test più elevato (tabella 1).

Table 1: Migliori p-values medi dei test di Kolmogorov-Smirnov.

	a b tests	KS p-value
8935	912.751677852349 11.838926174496645	0.264008
13474	1275.1677852348994 16.812080536912752	0.264119
10902	1069.798657718121 14.006711409395974	0.264128
12415	1190.6040268456377 15.664429530201344	0.264216
9389	948.993288590604 12.348993288590606	0.264267
13928	1311.4093959731542 17.322147651006713	0.264355
14382	1347.6510067114093 17.832214765100673	0.264384
11356	1106.0402684563758 14.516778523489934	0.264427
7876	828.1879194630873 10.691275167785236	0.264477
12869	1226.8456375838925 16.174496644295303	0.264494
8784	900.6711409395973 11.711409395973154	0.264512
9843	985.234899328859 12.859060402684564	0.264543
8330	864.4295302013422 11.201342281879196	0.265007

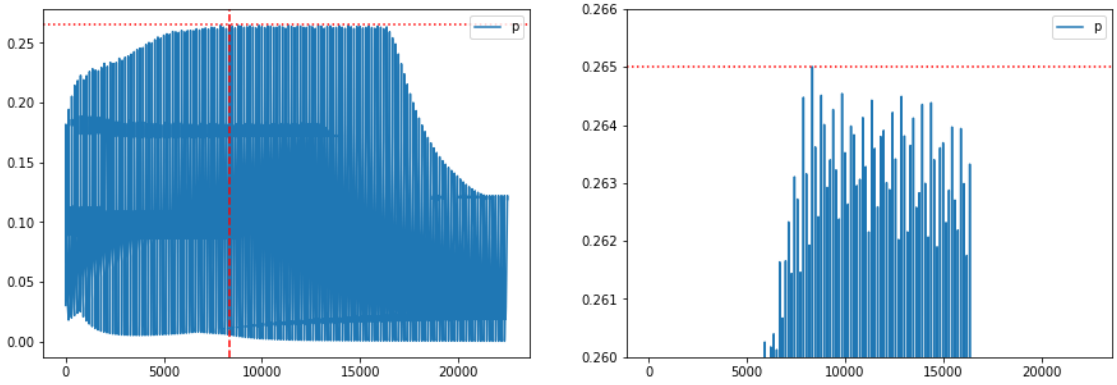

Figure 2: Test di Kolmogorov-Smirnov per i valori di a e b.

Tra i 22500 KS test effettuati, il migliore con media di p-value $= 0.2650$ indica la seguente come migliore distribuzione

$$P(R|\mathbb{D}) \sim \textbf{Gamma}(\omega = \frac{\overline{D}}{864.43}, \sigma = 11.20 \cdot \omega)$$

ovvero per ogni classe $\mathbb{D}$ di downloads, la probabilità di ricevere R recensioni è una distribuzione **Gamma** con moda ω pari al valore medio della classe diviso per 864.43 e deviazione standard σ pari ai 11.20 volte la moda.

Effettueremo un test di Kolmogorov-Smirnov per verificare la bontà della distribuzione presupposta. Definiamo l'ipotesi nulla H_0 che i campioni R misurati per ciascuna classe $\mathbb{D}$ siano stati estratti dalla distribuzione di probabilita $P(R|\mathbb{D})$ e definiamo il livello di significatività $\alpha = 0.05$. Se il p-value del test KS è minore di α rifiuteremo l'ipotesi nulla.

I risultati del test (tabella 2) dimostrano che l'ipotesi nulla è accettabile per la maggior parte delle classi di downloads.

Table 2: Kolmogorov-Smirnov test per i valori scelti di a e b.

CLASSE	Apps	KS p-value	H0 test
100+	2	0.172068	OK
500+	6	0.0794538	OK
1,000+	23	0.729144	OK
5,000+	20	0.0525401	OK
10,000+	63	0.000367931	-
50,000+	35	0.165592	OK
100,000+	105	0.489823	OK
500,000+	88	0.0809824	OK
1,000,000+	227	0.615419	OK
5,000,000+	125	0.1226	OK
10,000,000+	201	0.424157	OK
50,000,000+	68	0.0722562	OK
100,000,000+	66	0.600838	OK
500,000,000+	12	0.179999	OK
1,000,000,000+	17	0.000521866	-
5,000,000,000+	6	0.454343	OK
TOTAL	1064	0.2650067	OK

Pertanto, nell'intervallo scelto, la funzione descrive correttamente la distribuzione di probabilità (figura 3).

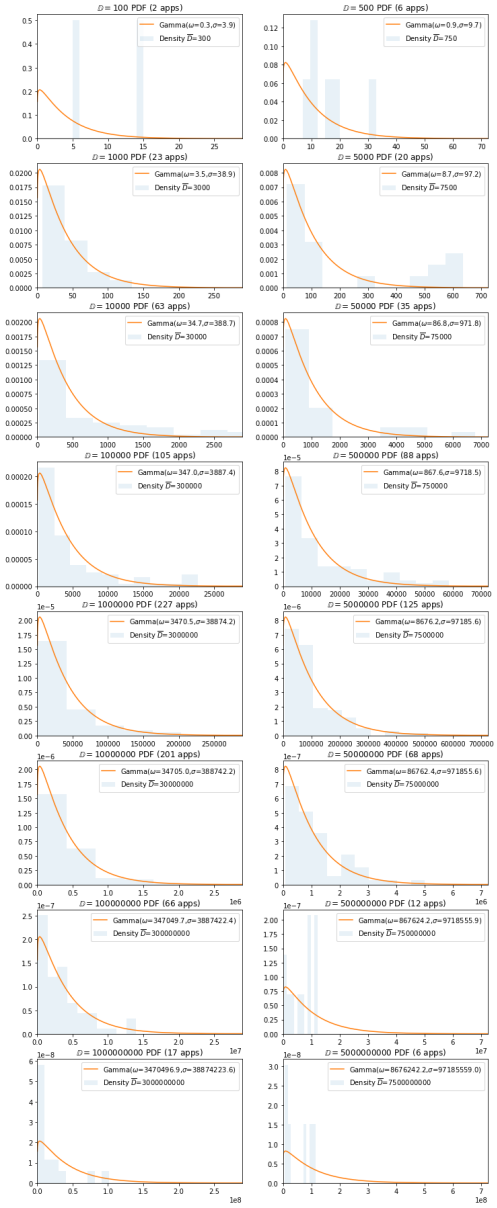

Figure 3: Funzioni di distribuzione delle recensioni (alle ascisse) delle classi di downloads.

Per ottenere la verosimiglianza $P(R^*|D)$, dove R^* è uno specifico valore di recensioni e D sono i possibili downloads, si può creare una matrice $\mathbf{P}(R_c|D_r)$ le cui righe r siano la distribuzione ricavata $P(R|\mathbb{D})$ e le cui colonne c saranno pertanto la verosimiglianza cercata $P(R^*|D)$.

$$\mathbf{P}(R_c|D_r) = \begin{Bmatrix} P(R_1|D_1) & \cdots & P(R_n|D_1) \\ \vdots & \ddots & \vdots \\ P(R_1|D_n) & \cdots & P(R_n|D_n) \end{Bmatrix}$$

Dall'analisi dei dati raccolti delle app disponibili possiamo scegliere $R_{max} = 3 \cdot 10^6$ ovvero 3 milioni di recensioni per le app della classe $\mathbb{D} = 5 \cdot 10^7$. Possiamo suddividere gli intervalli di down-

Table 3: Estratto di tabella del totale delle recensioni ricevute dall'app Immuni.

date	google_reviews	apple_reviews	reviews
2020-06-01	582	647	1229
2020-06-02	4128	2537	6665
2020-06-22	19698	7751	27449
2020-06-23	19870	7806	27676

Table 4: Estratto di matrice delle likelihoods Google $P(R|D)$ per i giorni t.

date	2020-06-01	2020-06-02	2020-06-22	2020-06-23
10000.0	8.919961e-05	4.007836e-17	1.263442e-71	3.156006e-72
20000.0	4.375574e-04	3.213808e-10	1.940974e-37	9.704814e-38
59990000.0	7.225538e-07	8.634189e-07	9.783183e-07	9.788861e-07
60000000.0	7.224223e-07	8.632623e-07	9.781444e-07	9.787121e-07

loads e recensioni, ad esempio, in 6000 steps riducendo così gli elementi delle matrici che verranno processate.

Possiamo così creare una matrice $\mathbf{P}(R_t|D)$ del numero di recensioni R_t ricevute su Play Store dall'app Immuni per ogni giorno t dalla pubblicazione in $t = 1$ a oggi $t = T$ ottenendo le verosimiglianze per i downloads da $D_0 = 0$ a $D_{max} = 6 \cdot 10^7$ (tabelle 4 e 5)

$$\mathbf{P}(R_t|D) = \begin{Bmatrix} P(R_{t=1}|D_0) & \cdots & P(R_{t=T}|D_0) \\ \vdots & \ddots & \vdots \\ P(R_{t=1}|D_{max}) & \cdots & P(R_{t=T}|D_{max}) \end{Bmatrix}$$

Non avendo alcuna informazione sui downloads nominali di App Store, supporremo che la funzione di likelihood derivata sia valida anche per le app di Apple, calcoleremo separatamente i posteriori sulle due piattaforme come indipendenti e sommeremo i risultati finali (tabelle 3 e figura 4).

Table 5: Estratto di matrice delle likelihoods Apple $P(R|D)$ per i giorni t.

date	2020-06-01	2020-06-02	2020-06-22	2020-06-23
10000.0	5.331682e-05	1.441442e-11	8.545378e-30	5.486201e-30
20000.0	3.399631e-04	1.884064e-07	1.528275e-16	1.224941e-16
59990000.0	7.296667e-07	8.268281e-07	9.112699e-07	9.118041e-07
60000000.0	7.295339e-07	8.266779e-07	9.111054e-07	9.116396e-07

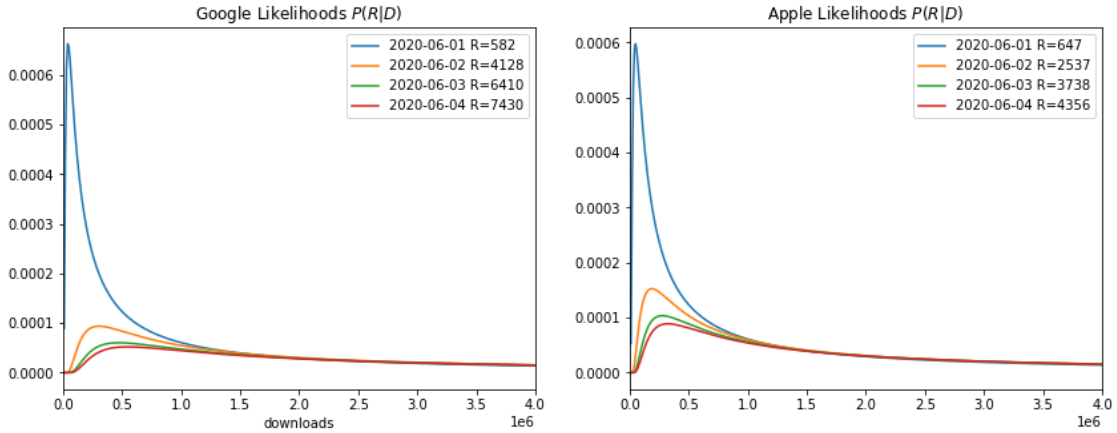

Figure 4: Verosimiglianze $P(R|D)$ per i primi giorni.

1.3 Probabilità marginale

La probabilità marginale $P(R_t)$ è semplicemente ottenibile come sommatoria dei numeratori calcolati per i giorni precedenti ovvero

$$P(R_t) = \sum_{t=1}^{t^*} P(R_t|D)P(D)$$

1.4 Posteriori

Per ogni giorno t, su ciascuna piattaforma, calcoleremo la probabilità a posteriori $P(D_t|R_t)$ assumendo come probabilità a priori $P(D_t)$ il prodotto vettoriale della probabilità a posteriori del giorno precedente per la probabilità a priori $P(D)$

$$P(D_t|R_t) = \frac{P(R_t|D) \cdot P(D)P(D_{t-1}|R_{t-1})}{\sum_{t=1}^{t^*} P(R_t|D)P(D)}$$

Le probabilità a priori di ogni giorno t verranno pertanto ridistribuite in base alla probabilità a posteriori del giorno precedente (figura 5) e otterremo una matrice delle probabilità a posteriori per ogni valore di downloads nell'ambito dell'intervallo scelto da 0 a D_{max} (figura 6).

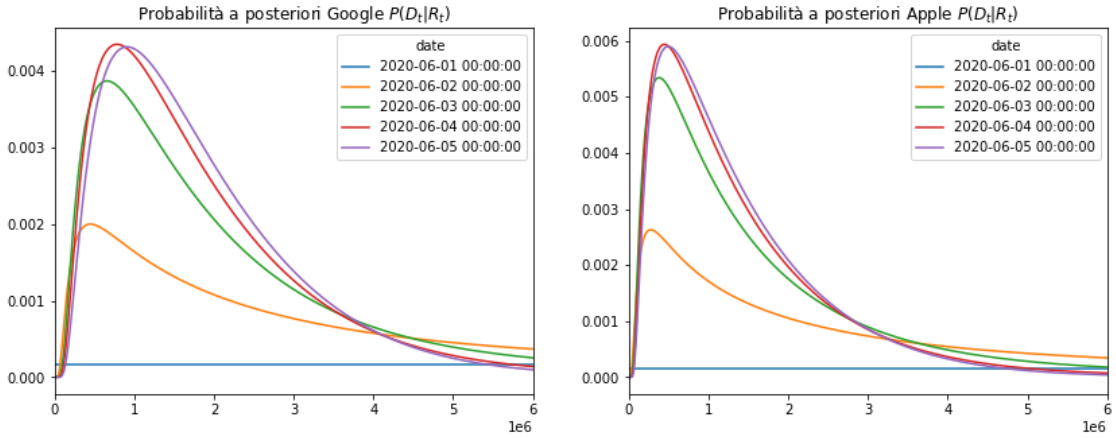

Figure 5: Probabilità a posteriori $P(D_t|R_t)$ per i primi giorni.

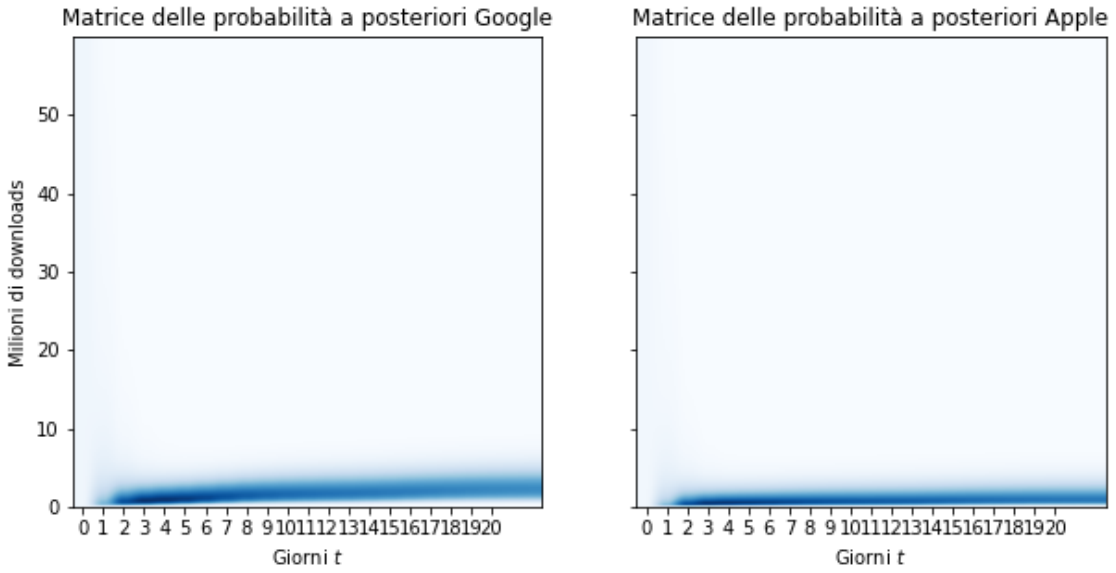

Figure 6: Matrice delle probabilità a posteriori calcolate.

2 Risultati

Dai risultati dei posteriori (totali in tabella 8) otteniamo la conferma di circa 1 milione di downloads per il 3 Giugno 2020 e circa 3 milioni per il 21 Giugno 2020.

Per ogni giorno, calcoliamo gli intervalli di credibilità bayesiana (HDI, Highest Density Intervals) al 95% e al 50% e costruiamo il grafico dei downloads stimati (per piattaforma e totali) e della percentuale di popolazione che ha verosimilmente installato l'app (figure 7, 8 e 9).

Table 6: Estratto di matrice dei risultati a posteriori Google $P(D_t|R_t)$.

date	ML	Low_50	High_50	Low_95	High_95
2020-06-01	10000.0	10000.0	30020000.0	10000.0	57020000.0
2020-06-02	450000.0	80000.0	5020000.0	50000.0	45680000.0
2020-06-03	660000.0	210000.0	1770000.0	70000.0	11190000.0
2020-06-04	790000.0	310000.0	1610000.0	110000.0	5030000.0
2020-06-22	2240000.0	1350000.0	3230000.0	530000.0	5910000.0
2020-06-23	2270000.0	1430000.0	3320000.0	530000.0	5950000.0

Table 7: Estratto di matrice dei risultati a posteriori Apple $P(D_t|R_t)$.

date	ML	Low_50	High_50	Low_95	High_95
2020-06-01	10000.0	10000.0	30020000.0	10000.0	57020000.0
2020-06-02	280000.0	40000.0	4230000.0	20000.0	44620000.0
2020-06-03	380000.0	120000.0	1300000.0	40000.0	8390000.0
2020-06-04	450000.0	140000.0	1130000.0	50000.0	4020000.0
2020-06-22	900000.0	390000.0	1660000.0	150000.0	4130000.0
2020-06-23	900000.0	440000.0	1710000.0	150000.0	4140000.0

Table 8: Estratto di matrice dei risultati a posteriori totali $P(D_t|R_t)$.

date	ML	Low_50	High_50	Low_95	High_95
2020-06-01	20000.0	20000.0	60040000.0	20000.0	114040000.0
2020-06-02	730000.0	120000.0	9250000.0	70000.0	90300000.0
2020-06-03	1040000.0	330000.0	3070000.0	110000.0	19580000.0
2020-06-04	1240000.0	450000.0	2740000.0	160000.0	9050000.0
2020-06-22	3140000.0	1740000.0	4890000.0	680000.0	10040000.0
2020-06-23	3170000.0	1870000.0	5030000.0	680000.0	10090000.0

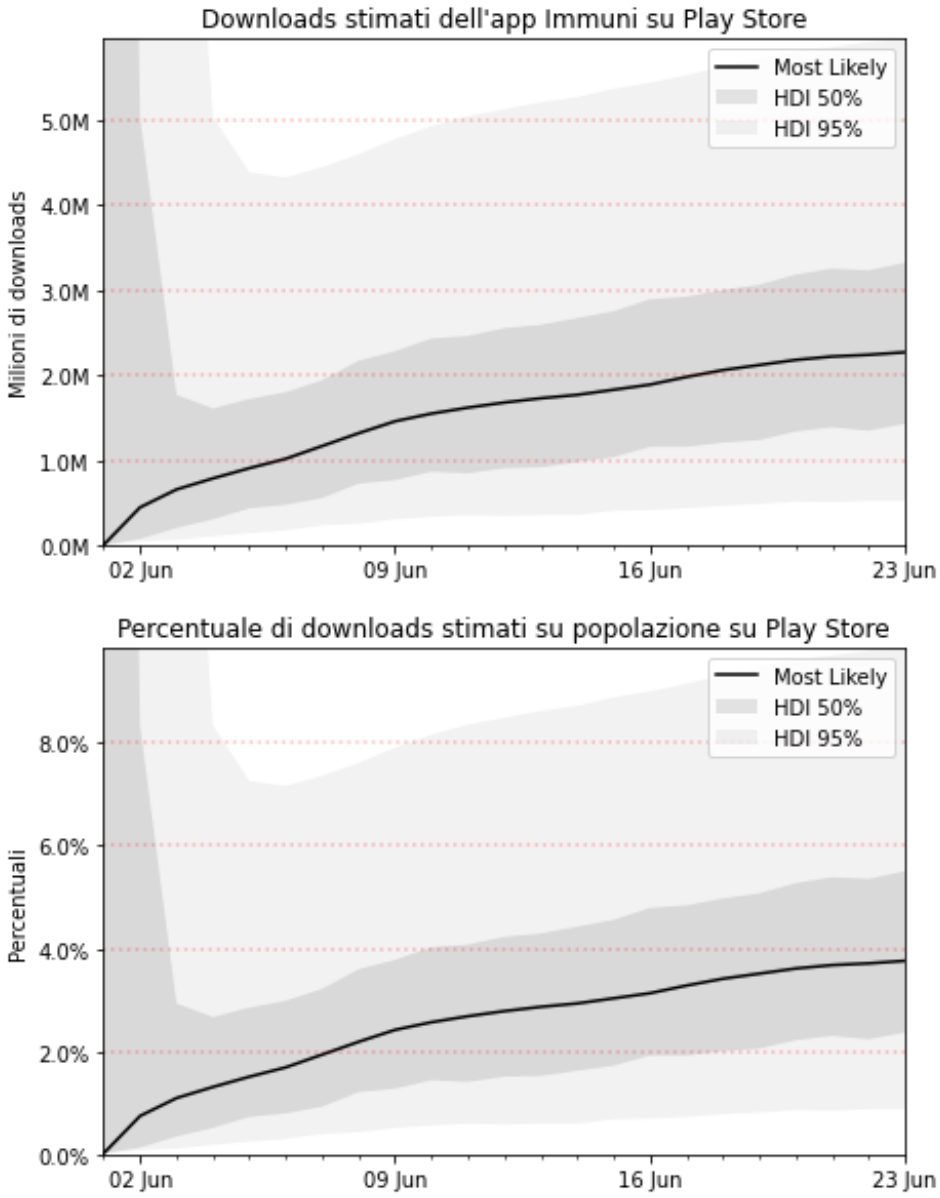

Figure 7: Risultati e intervalli di credibilità.

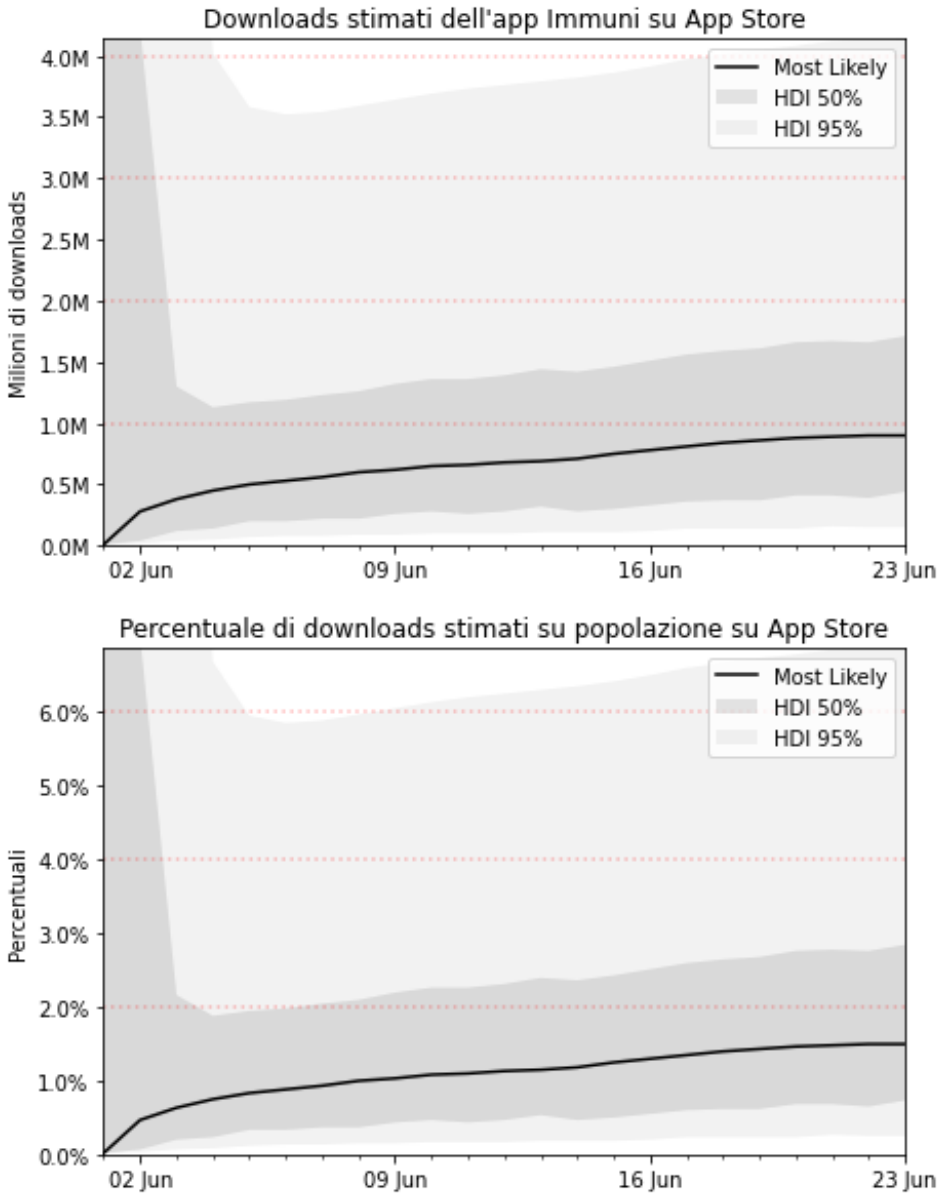

Figure 8: Risultati e intervalli di credibilità.

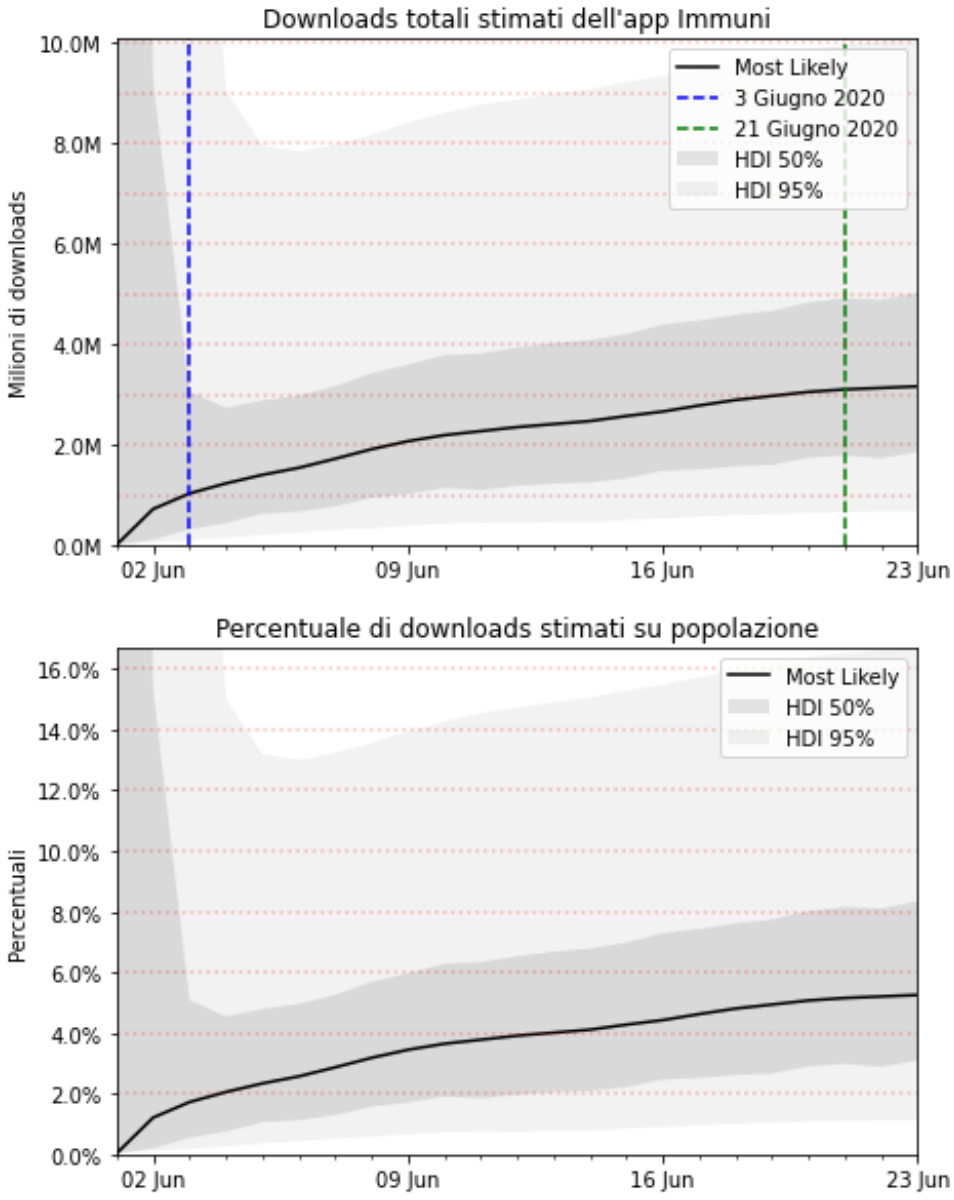

Figure 9: Risultati e intervalli di credibilità.