P2PMID

Antonio Pace

April 2022

1 Prima parte

1.1 Introduzione

Per l'analisi del dataset ho utilizzato Python con Pandas, e Matplotlib per la rappresentazione grafica della distribuzione dei dati.

1.2 Descrivi come una reale transazione Bitcoin è astratta da una transazione del dataset (quali campi sono eliminati, quali sono astratti e come).

transactions	
id	identifier of the transaction
block_{id}	id of the block containing this transaction
inputs	
id	unique id of this input
tx_{id}	transaction this input is part of
sig_{id}	scriptSig public key id, 0 if coinbase tx, - 1 if nonstandard scripts used
$\text{output}_{\text{id}}$	id of the previous output being referenced, - 1 if coinbase tx
outputs	
id	unique id of this output
tx_{id}	transaction this output is part of
pk_{id}	scriptPubKey public key id, - 1 if nonstandard scripts used

Figura 1: Astrazione di una transazione nel dataset

Come specificato nel testo, tutti gli hash sono sostituiti con numeri interi, come ad esempio l'id delle transazioni. Inoltre, in una reale transazione Bitcoin, sono presenti campi che rappresentano la dimensione della transazione (in bytes), il numero di input e output, la versione (che indica le regole del protocollo da utilizzare per quella specifica transazione) e il locktime (quando una transazione può essere aggiunta alla blockchain). Un'altra astrazione riguarda gli script di input e output. Nel dataset si può vedere come i campi

sig_{id} per l'input e pk_{id} per l'output, contengano semplicemente le chiavi pubbliche e non gli script in sè (ad esempio l'input dovrebbe contenere la firma per permettere lo sblocco). Essendo rappresentate solo le transazioni e non i blocchi, nel dataset è presente il campo $block_{id}$ per associare la transazione al blocco di appartenenza.

1.3 Controlla se tutti i dati contenuti nel dataset sono consistenti, e se dei dati non sono validi, descrivi il problema e rimuovili dal dataset

Analizzando il dataset ho trovato alcuni dati non consistenti.

- La transazione **15698** ha un input che non si riferisce a nessun precedente output, pur non essendo una transazione Coinbase.

Input	tx_input	Sender	prev_output	tx_output	value	Output	Receiver	value_output
16461	15698	109902				15738	39538	9200000000

Figura 2: Transazione invalida

- Sono presenti 4 double spending (transazioni **12152**,**30446**,**61845**,**207365**) evidenziati in arancione in Figura 3, dove sono rappresentati gli input di ogni transazione con i relativi output precedenti. Infatti si può notare come stessi output precedenti siano "consumati" da input diversi.

Input	tx_input	pk	prev_output	tx_output	value
8666	8231	7941	7998	7971	5000000000
12820	12152	7941	7998	7971	5000000000
33113	30446	21807	21928	21878	5000000000
33114	30446	21807	21928	21878	5000000000
76747	61843	138980	65403	61842	5000000000
76750	61845	138980	65403	61842	5000000000
275614	204751	163625	249860	207362	41381000000
279609	207365	163625	249860	207362	41381000000

Figura 3: Double spending

- La transazione **100929** è invalida poichè ha un output totale maggiore del valore degli input.

- La transazione **105281** è invalida poichè ha un output totale negativo (output 123672).

1.4 Calcolare l'importo totale delle UTXO (Unspent Transaction Output) esistenti a partire dall'ultimo blocco del dataset, ovvero la somma di tutti i saldi delle transazioni in uscita sul'insieme di UTXO dell'ultimo blocco. Quale UTXO (TxId, blockId, output, address) ha il valore associato più alto?

La transazione **140479**. Ho cercato il valore massimo fra gli output non associati a nessun input successivo.

out_id	pk_id	value	tx_id	block_id
170430	138895	9000000000000	140479	90532

Figura 4: UXTO con valore maggiore

Tutte le transazioni correlate a quelle invalide sono state rimosse a catena.

1.5 Calcolare la distribuzione del numero di transazioni per ogni blocco e mostrare l'evoluzione della dimensione dei blocchi per ogni mese.

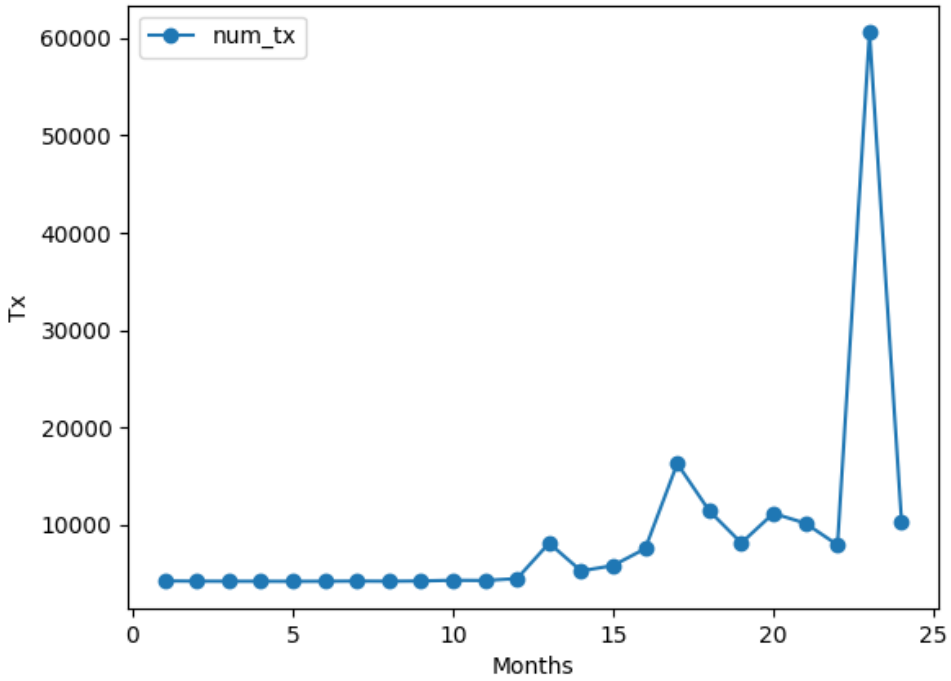

Figura 5

1.6 Calcolare l'ammontare di bitcoin ricevuti da ogni chiave pubblica che ha ricevuto almeno una transazione Coinbase, e mostrare una distribuzione dei valori

Ho effettuato una conversione dei valori ricevuti da Satoshi a Bitcoin.

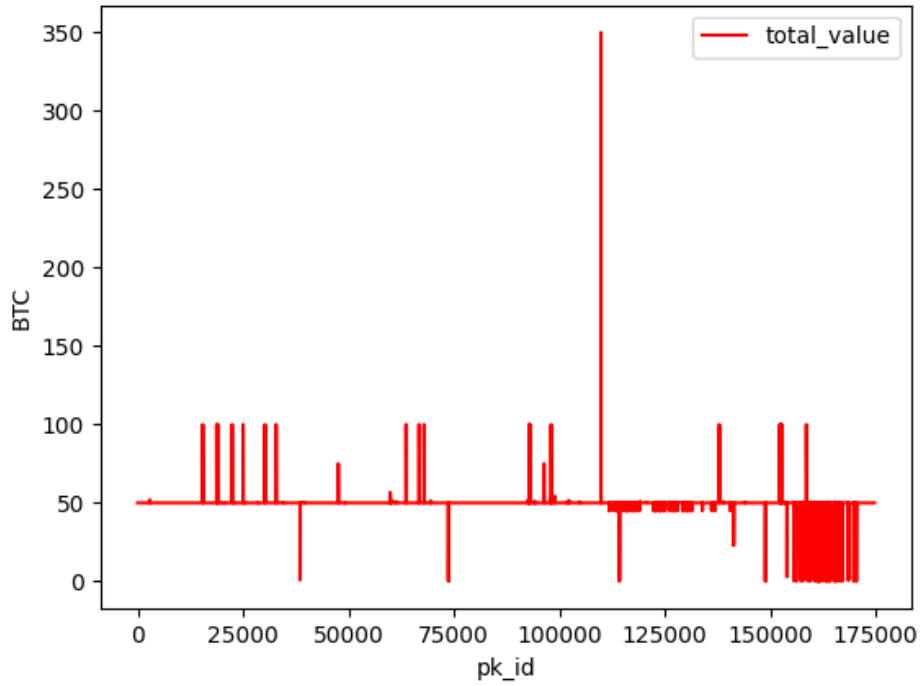

Figura 6

1.7 Calcolare la distribuzione delle fees spese in ogni transazione

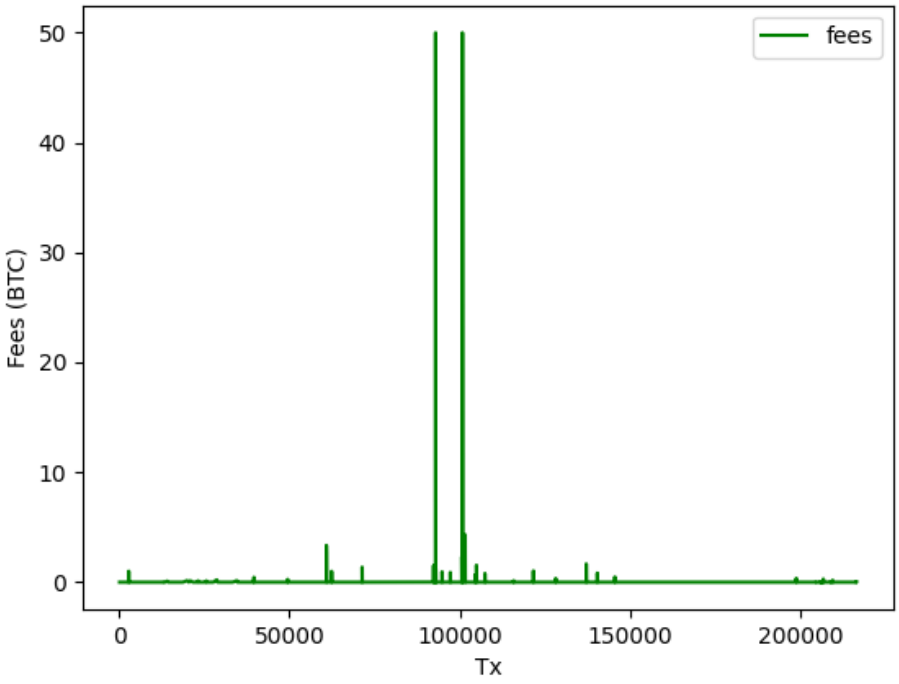

Figura 7

1.8 Proporre un'analisi a scelta

La chiave privata che ha speso più BTC (**273750.23**) è **148105**.

2 Seconda parte

Considera una rete Kademlia con ID a 8 bits e k=4. I k-buckets del peer con ID 11001010 è la seguente:

- *7 : 01001111, 00110011, 01010101, 00000010*
- *6 : 10110011, 10111000, 10001000*
- *5 : 11101010, 11101110, 11100011, 11110000*

- *4 : 11010011, 11010110*
- *3 : 11000111*
- *2 :*
- *1 :*
- *0 :*

2.1 Arrivano messaggi dai seguenti nodi in quest'ordine: 01101001, 10111000, 10101010, 11100011, 11111111. Come cambiano i buckets?

Ho calcolato le distanze di ogni nodo dal peer con l'operatore XOR per assegnarli ai buckets:

- Il nodo 01101001 è diretto al bucket 7 che è pieno, verrà aggiunto in coda solo se il nodo in testa (01001111) non risponderà al ping. Il nodo in testa verrà promosso in coda se risponde al ping.
- Il nodo 10111000 viene aggiunto in coda al bucket 6 in quanto contiene solo 3 nodi.
- Il nodo 10101010 è diretto al bucket 6, che è pieno dopo l'aggiunta del nodo precedente, verrà aggiunto in coda solo se il nodo in testa (10110011) non risponderà al ping. Il nodo in testa verrà promosso in coda se risponde al ping.
- Il nodo 11100011 è diretto al bucket 5 che è pieno, verrà aggiunto in coda solo se il nodo in testa (11101010) non risponderà al ping. Il nodo in testa verrà promosso in coda se risponde al ping.
- Il nodo 11111111 è diretto al bucket 5 che è pieno, verrà aggiunto in coda solo se il nodo in testa (11101110) non risponderà al ping. Il nodo in testa verrà promosso in coda se risponde al ping.

2.2 Viene rilevato che il nodo 11101110 non può essere più raggiunto. Qual è la reazione?

Se un nodo ha abbandonato la rete, il peer aggiorna i buckets tramite informazioni ricevute dalle query o in alternativa tramite aggiornamenti periodici cercando nodi con ID nel range del bucket.

2.3 Che indirizzo restituisce il peer ad un lookup sull'ID 11010010?

Il nodo 11010011 nel bucket 4 perchè ha una distanza di 1 dall'ID richiesto.