Distribuzione degli amino acidi nelle catene proteiche RNA-binding

Benedetta Barbetti
Michaela Servi

October 22, 2014

1 Introduzione

Le interazioni RNA-proteina sono essenziali per molti processi biologici, tra cui la regolarizzazione dell'espressione genica, la sintesi di proteine, la replicazione e l'assemblaggio di molti virus.
Per capire i meccanismi di questa interazione l'idea è utilizzare tecniche di machine learning.
In questo lavoro si cerca di trovare una distribuzione generativa della catena proteica, che in seguito possa essere usata come punto di partenza per un apprendimento di tipo supervisionato.
Per questo scopo è stata utilizzata una Restricted Boltzmann Machine (RBM), un modello generativo parametrico che rappresenta una distribuzione di probabilità. Dato un insieme di osservazioni, il training set, apprendere una RBM significa aggiustare i parametri in modo tale che la distribuzone di probabilità rappresentata fitti i dati del training set nel miglior modo possibile. Una RBM può essere vista come un Markov Random Field(MRF) associato con un grafo non orientato bipartito. Nella Sezione 2 verranno introdotti i modelli grafici e quindi i MRF; nella Sezione 3 saranno formalmente descritte le RBM; i metodi di approssimazione per la funzione di training della RBM verranno presentati nella Sezone 4. La Sezione 5 mostrerà l'implementazione del modello e i risultati dei test.

2 Modelli Grafici

I modelli grafici sono modelli probabilistici che rappresentano con una struttura a grafo dipendenze e indipendenze condizionate tra un insieme di variabili casuali; due variabili X_1 e X_2 si dicono *condizionalmente indipendenti* dato X_3 se $p(X_1, X_2|X_3) = p(X_1|X_3)p(X_2|X_3)$.

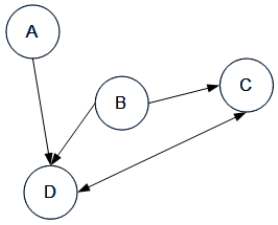


Figure 1: Un esempio di modello grafico orientato; ogni freccia indica una dipendenza: D dipende da A, B e C, C dipende da B e D

Esistono diversi modelli grafici associati con differenti tipi di strutture a grafo, per esempio i *factor graph* e le *Bayesian networks*, associate con grafi orientati, e i *Markov random fields* o *Markov networks*, associati con grafi non orientati. Nel prossimo paragrafo verranno presentati i modelli grafici non orientati, che sono i modelli a cui sono associate le RBM.

2.1 Grafi non orientati e Markov Random Fields

Un grafo non orientato è una tupla $G = (V, E)$ dove V è un insieme finito di nodi ed E è un insieme di archi non orientati, dove un arco è una coppia di nodi $(u, v) \in V$. Un *cammino* da v_1 a v_m è una sequenza di nodi $v_1, v_2...v_m \in V$, con $(v_i, v_{i+1}) \in E, i = 1, .., m-1$. Un set $V' \subset V$ *separa* due nodi $u \notin V'$ e $v \notin V'$ se ogni cammino da u a v contiene un nodo in V'. Si definisce $N_v := \{u \in V : (v, u) \in E\}$ l'insieme di nodi connessi a v. Una *cricca* è un sottoinsieme di V in cui per ogni coppia di nodi si ha un arco che li collega. Una cricca si dice massimale se non è possibile aggiungere un altro nodo in modo tale che il nuovo insieme rappresenti una cricca.
Se si associa una variabile casuale X_v, che prende valori in uno spazio degli stati Ω, con ogni nodo v del grafo G, queste variabili X_v vengono chiamate Markov Random Fields se la distribuzione di probabilità condizionata p soddisfa la proprietà globale di Markov rispetto al grafo: per ogni $A, B, S \subset V$ disgiunti, dove tutti i nodi in A e B sono separati da S, $p(x_{a\in A})|x_{t\in S\cup B}) = p(x_{a\in A})|x_{t\in S})$ con $x \in \Omega^{|V|}$.
Un set di nodi $MB(v)$ viene definito *Markov Blanket* del nodo V se, per ogni $B \subset V$ con $v \notin B$, $p(v|MB(V), B) = p(v|MB(v))$, che significa che v è condizionalmente indipendente da ogni altra variabile dato $MB(v)$.
In un MRF la Markov Blanket di un nodo v coincide con N_v, questà proprietà viene chiamata *proprietà locale di Markov*.

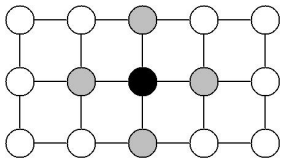


Figure 2: Dati i nodi grigi, il nodo nero è condizionalmente indipendente da tutti gli altri nodi

Per il *teorema di Hammersley-Clifford*, la distribuzione di probabilità di ogni MRF può essere generalizzata a una forma fattorizzata di distribuzioni ed espressa dalla *distribuzione di Gibbs*:

$$p(x) = \frac{1}{Z}\prod_{c\in C}\phi_c(x_c) = \frac{1}{Z}e^{\sum_{c\in C} ln\phi_c(x_c)} = \frac{1}{Z}e^{-E(x)}$$

dove $Z = \sum_x \prod_{c\in C}\phi_c(x_c)$ è chiamata *funzione di partizione* e $E := \sum_{c\in C} ln\phi_c(x_c)$ è chiamata *funzione energia*.

3 Restricted Boltzmann Machine

Una RBM è un MRF associato con un grafo non orientato biartito, come mostrato in figura 3. Consiste di m unità visibili $V = (V_1, ..., V_m)$ per rappresentare i dati osservati e n unità nascoste $H = (H_1, ..., H_n)$ che modellano le dipendenze tra le variabili osservate.
Nelle RBM binarie le variabili casuali (V, H) prendono valori $(v, h) \in \{0, 1\}_{m+n}$

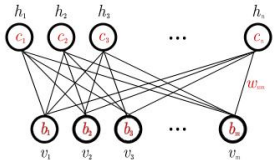


Figure 3: Grafo non orientato di una RBM con n unità nascoste e m unità visibili

e la distribuzione di probabilità congiunta del modello è data dalla distribuzione di Gibbs $p(v, h) = \frac{1}{Z}e^{-E(v,h)}$ con funzione di energia

$$E(v, h) := -\sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i,$$

con w, b, c rispettivamente vettore dei pesi, vettore di bias associato con le unità visibili e vettore di bias associato con le unità nascoste.
Una RBM non ha connessioni tra variabili dello stesso livello, che in termini di probabilità significa che le variabili nascoste sono indipendenti dato lo stato delle variabili visibili e viceversa:

$$p(h|v) = \prod_{i=1}^{n} p(h_i|v)$$

$$p(v|h) = \prod_{i=1}^{m} p(v_i|h).$$

La distribuzione marginale delle variabili visibili viene così calcolata come:

$$p(v) = \sum_h p(v|h) = \sum_h \frac{1}{Z} e^{-E(v,h)} = \frac{1}{Z} e^{-F(v)},$$

dove $F(v) = -log \sum_h e^{-E(v,h)}$, chiamata *Energia libera*, e $Z = \sum_v e^{-F(v)}$.

3.1 Problema di training

Un metodo standard per stimare i parametri di un modello statistico è la *Maximum Likelihood*.
Applicata a MRFs, questo metodo consiste nel trovare i parametri che massimizzano la probabilità del dataset secondo la distribuzione di MRF; il training corrisponde quindi a trovare i parametri w,b e c che massimizzano la likelihood dato il training set:

$$\max_{w,b,c} \frac{1}{m} log(\mathcal{L}(w,b,c|V)), \text{ con}$$

$$\mathcal{L}(w,b,c|V) = \prod_{j=1}^{m} p(v; w, b, c)$$

La derivata della log-likelihood può essere scritta come la somma di due valori attesi: il valore atteso sul dataset e il valore atteso su tutte le possibili configurazioni di input v:

$$\begin{aligned} \frac{1}{m} \sum_{v \in V} \frac{\partial log(\mathcal{L}(w,b,c|V))}{w_{ij}} &= < v_i h_j >_{p(h|v)q(v)} - < v_i h_j >_{p(h,v)} \\ &\propto < v_i h_j >_{dati} - < v_i h_j >_{modello} \end{aligned} \tag{1}$$

dove q è la distribuzione empirica. Il primo può essere computato efficientemente perchè facilmente fattorizzabile data la stuttura del grafo; il secondo è intrattabile perchè cresce esponenzialmente con la grandezza del vettore di input.
Per evitare la complessità esponenziale, il secondo termine viene approssimato campionando dalla distribuzione del modello, basandosi su tecniche di tipo MCMC.

4 Catene di Markov e metodi Markov Chain Monte Carlo

Le catene di Markov giocano un ruolo importante nel processo di training di una RBM in quanto forniscono un metodo per estrarre campioni da distribuzioni di probabilità "complesse" come la distribuzione di Gibbs di un MRF.
Una catena di Markov è un processo stocastico a tempo discreto per cui vale la proprietà di Markov, i.e. una famiglia di variabili random $X = \{X^{(k)} | k \in N_0\}$ che prende valori in un set Ω e per cui $\forall k \geq 0$ e $\forall j, i, i_0, ..., i_{k-1} \in \Omega$ si ha

$$p_{ij}^{(k)} := P(X^{(k+1)} = j | X^k = i, X^{(k-1)} = i_{k-1}, ..., X^{(0)} = i_0) = P(X^{(k+1)} = j | X^{(k)} = i).$$

Questo vuol dire che il prossimo stato del sistema dipende soltanto da quello corrente e non dalla sequenza di eventi che lo hanno preceduto. Se per ogni $k \geq 0$ i $p_{ij}^{(k)}$ hanno lo stesso valore p_{ij}, la catena è detta *omogenea* e la matrice $P = (p_{ij})_{ij \in \Omega}$ è detta *matrice di transizione* della catena di Markov omogenea. Se la distribuzione di probabilità iniziale $\mu^{(0)}$ (i.e. la distribuzione di probabilità di $X^{(0)}$) è data dal vettore $\mu^{(0)} = (\mu^{(0)}(i))_{i \in \Omega}$ con $\mu^{(0)}(i) = P(X^{(0)} = i)$ la distribuzione $\mu^{(k)}$ di $X^{(k)}$ è data da $\mu^{(k)T} = \mu^{(0)T} P^k$

Una distribuzione π per cui vale che $\pi^T = \pi^T P$ è detta *distribuzione stazionaria.* Se la catena di Markov per un tempo k raggiunge la distribuzione stazionaria $\mu^{(k)} = \pi$ tutti gli stati conseguenti saranno distribuiti in base ad essa, cioè $\mu^{(k+n)} = \pi$ per ogni $n \in N$. Una condizione sufficiente (ma non necessaria) perchè una distribuzione π sia stazionaria in riferimento alla catena di Markov descritta dalle probabilità di transizione $p_{ij}, i, j \in \Omega$ è che $\forall i, j \in \Omega$ sia:

$$\pi(i) p_{ij} = \pi(j) p_{ji}$$

Questa è chiamata *detailed balance condition.*
Rilevanti sono le catene di Markov per cui è nota l'esistenza di una distribuzione stazionaria unica. Per un Ω finito questo avviene se la catena di Markov è *irriducibile.*
Una catena di Markov è irrudicibile se è possibile, da uno stato in Ω raggiungerne ogni altro in un numero finito di transizioni, o più formalmente

$$\forall i, j \in \Omega \; \exists k > 0 \text{ con } P(X^{(k)} = j | X^{(0)} = i) > 0.$$

Una catena è chiamata *aperiodica* se per ogni $i \in \omega$ il massimo comun divisore di $\{k | P(X^{(k)} = i | X^{(0)} = i) > 0 \bigwedge k \in N_0\}$ è 1. Si può mostrare che per una catena di Markov irriducibile e aperiodica su uno spazio degli stati finito è garantita la convergenza alla sua distribuzione stazionaria. Infatti, per un'arbitraria distribuzione di partenza μ si ha

$$\lim_{k \to \inf} d_V(\mu^T P^k, \pi^T) = 0$$

dove d_V è la *distanza di variazione*. Per due distribuzioni α e β in uno spazio degli stati finito Ω, la distanza di variazione è definita come

$$d_V(\alpha, \beta) = \frac{1}{2}|\alpha - \beta| = \frac{1}{2}\sum_{x \in \Omega} |\alpha(x) - \beta(x)|$$

I metodi Markov chain Monte Carlo utilizzano questo teorema di convergenza per produrre campioni da determinate distribuzioni di probabilità, creando una catena di Markov che converge alla distribuzione desiderata.
Supponiamo di voler campionare da una distribuzione q con uno spazio degli stati finito. Si construisce una catena di Markov irriducibile e aperiodica con distribuzione stazionaria $\pi = q$. Questo è un problema non banale. Se t è grande abbastanza, un campione $X^{(t)}$ preso dalla catena costruita è approssimativamente un campione da π e quindi da q. Il Gibbs Sampling è un metodo MCMC.

4.1 Gibbs Sampling

Il Gibbs Sampling appartiene alla classe degli algoritmi Metropolis-Hastings. È un semplice algoritmo MCMC per produrre campioni dalla distribuzione congiunta di variabili casuali multiple. L'idea di base è aggiurnare ogni variabile sequenzialmente basandosi sulla distribuzione condizionata dati gli stati delle altre variabili.
Vediamo come il Gibbs Sampling può essere usato per simulare la distribuzione

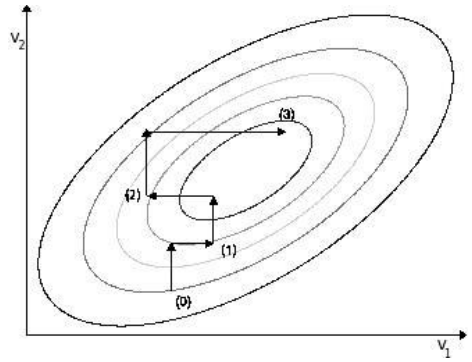


Figure 4

di Gibbs di un MRF.
Consideriamo un MRF $X = (X_1, ..., X_N)$ rispetto a un grafo $G = (V, E)$ dove $V = 1, .., N$.
Le variabili casuali $X_i, i \in V$ prendono valori in un insieme finito Λ e $\pi(x) = \frac{1}{Z}e^{\epsilon(x)}$ è la distribuzione di probabilità congiunta di X. Se assumiamo che MRF cambi il suo stato nel tempo, consideriamo $X = \{X^{(k)} | k \in N_0\}$ come una catena

di Markov che prende valori in $\Omega = \Lambda^N$ dove $X^{(k)} = (X_1^{(k)}, ..., X_N^{(k)})$ descrive lo stato del MRF al tempo $k \geq 0$.
Ad ogni passo di transizione si sceglie una variabile casuale X_i, $i \in V$ con probabilità q_i data dalla distribuzione di probabilità strettamente positiva q su V e si campiona un nuovo valore X_i basato sulla distribuzione condizionata dato lo stato $x_{v \in V_1}$di ogni altra variabile $X_{v \in V_1}$, cioè basata su $\pi(X_i | x_{v \in V_1}) = \pi(X_i | x_{w \in N_1})$.
La probabilità di transizione p_{xy} per due stati x, y di un MRF X con $x \neq y$ è quindi data da:

$$p_{xy} = \begin{cases} q(i)\pi(y_i | x_{v \in V_1}), & \text{se } \exists i \in V \text{tale che } \forall v \in V \text{con } v \neq i : x_v = y_v \\ 0, & \text{altrimenti.} \end{cases}$$

La probabilità che lo stato del MRF x non cambi è data da:

$$p_{xx} = \sum_{i \in V} q(i)\pi(y_i | x_{v \in V_1}).$$

È facile vedere che la distribuzione congiunta π del MRF è la distribuzione stazionaria della catena di Markov definita da queste probabilità di transizione mostrando che la detailed balance condition regge: per $x = y$ questo segue direttamente; se x e y differiscono nel valore di più di una variabile casuale segue dal fatto che $p_{xy} = p_{yx} = 0$.
Assumiamo che x y differiscano soltanto nello stato di esattamente una variabile X_i, cioè $y_j = x_j \forall j \neq i$ e $y_i \neq x_i$ quindi si ha:

$$\pi(x)p_{xy} = \pi(x)q(i)\pi(y_i | x_{v \in V_1}) = ... = \pi(y)p_{yx}$$

Dal momento che π è strettamente positiva lo sono anche le distribuzioni di probabilit condizionata delle singole variabili. Quindi segue che ogni singola variabile X_i possa assumere ogni stato $x_i \in \Lambda$ in un solo passo di transizione e perciò ogni stato dell'intero MRF possa raggiungerne ogni altro in Λ^N in un numero finito di passi e la catena di Markov sia irriducibile.
Inoltre, dalla positività della distribuzione condizionata, segue che $p_{xx} > 0$ per ogni $x \in \Lambda^N$ e quindi che la catena di Markov è aperiodica.
Aperiodicità e irriducibilità garantiscono la convergenza della catena alla distribuzione stazionaria π.
Nella pratica la scelta della singola variabile casuale aggiornata non è su base random ma è un ordine sequenziale predefinito.
Il corrispondente algoritmo è spesso definito come *Gibbs Sampler periodico.*

4.2 Contrastive Divergence

Ottenere stime unbiased per il gradiente della likelihood usando metodi MCMC solitamente richiede molti passi di sampling; è stato però mostrato che stime ottenute dopo solo pochi passi (k passi) possono essere sufficienti per allenare il modello. *Contrastive Divergence* (CD) è diventato il metodo standard per

allenare le RBMs.
Invece di approssimare il secondo termine del gradiente della log-likelihood campionando dalla distribuzione del modello(che quindi richiederebbe di far girare una catena di Markov fino a raggiungere una distribuzione stazionaria) l'idea del k-step contrastive divergence (CD-k) è di calcolare una catena di Gibbs per soli k passi(solitamente $k = 1$).
La catena è inizializzata con un esempio di training $v(0)$ e produce il campione $v(k)$ dopo k passi.
Ogni passo t consiste nel campionare $h(t)$ da $p(h|v(t))$ e campionare $v(t+1)$ da $p(v|h(t))$. Il gradiente della log-likelihood nell'equazione (1) viene quindi

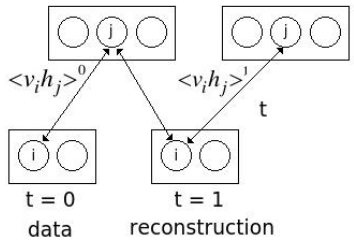


Figure 5: Contrastive Divergence di k passi su una RBM

approssimato, per un pattern di training $v^{(0)}$, come:

$$CD_k(w_{ij}, v^{(0)}) = -\sum_h p(h|v^{(0)}) \frac{\partial E(v^{(0)},h)}{\partial w_{ij}} + \sum_h p(h|v^{(k)}) \frac{\partial E(v^{(k)},h)}{\partial w_{ij}}.$$

Dal momento che $v^{(k)}$ non è un campione della distribuzione stazionaria, l'approssimazione è biased, e diventa unbiased quando $k \to \infty$. Il fatto che CD sia un'approsimazione biased si realizza anche dal fatto che non massimizza la likelihood dei dati, ma la differenza di due divergenze di Kullback-Leibler:

$$KL(q|p) - KL(p_k|p),$$

dove q è la distribuzione empirica e p_k è la distribuzione delle variabil visibili dopo k passi di Makov chain. Se la catena ha già raggiunto la stazionarietà , $p_k = p$ perciò $KL(p_k|p) = 0$ e l'errore di approssimazione di CD sparisce.
Come conseguenza dell'errore di approssimazione, kCD non porta necessariamente alla stima di maximum likelihood dei parametri del modello, ma come possiamo vedere dalla Figura 6 vi si avvicina molto.

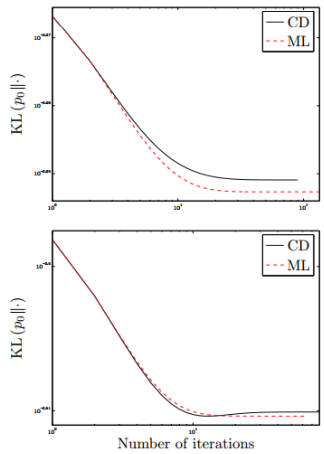


Figure 6: Curve di learning per Maximum Likelihood e Contrastive Divergence per due distribuzioni di dati scelte in maniera casuale

5 Implementazione e risultati

Il modello descritto nella Sezione 3 è stato implementato utilizzando la classe `rbm` della libreria `pylearn2`, usando il modello binario-binario, che assume distribuzione Bernoulli per ogni unità in ingresso e in uscita.
È stato anche testato il modello che assume una distribuzione gaussiana dei dati in ingresso, ma, come ci si aspettava, l'errore di ricostruzione non arriva a convergere.
Per allenare il modello è stato utilizzato un training set non ridondante di 439 catene proteiche RNA-binding, rappresentate, come sequenze di amino-acidi, in formato *fasta*.
Le feature di ingresso alla RBM sono state ottenute aggiungendo un padding

```
>2BBV:C
TQTAPVPQQNVPALTRLSQPGLAFLKCAFAPPDFNTDPGKGIPDRFEGKVVTRKDVLNQSINFTANRD...
```

Figure 7: Esempio di file fasta

di *'dummy'* amino-acidi X e facendo scorrere una finestra di lunghezza fissata N sulle sequenze di amino-acidi; successivamente ogni amino acido è stato codificato con un vettore *one-hot*.

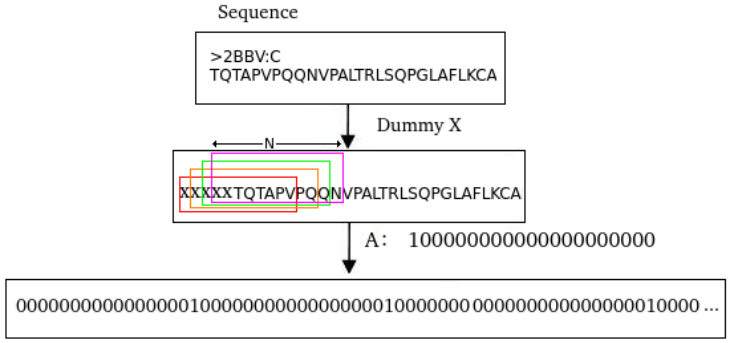
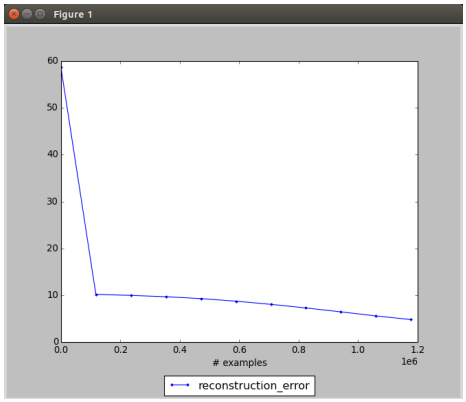
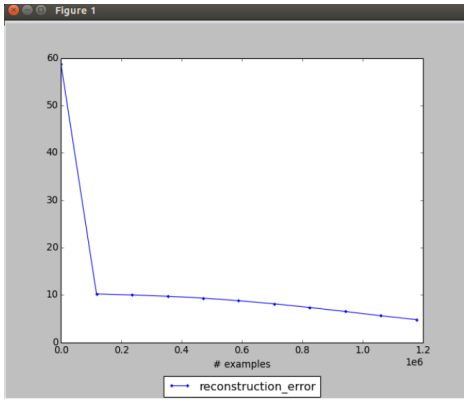


Figure 8: Generazione dei pattern binari

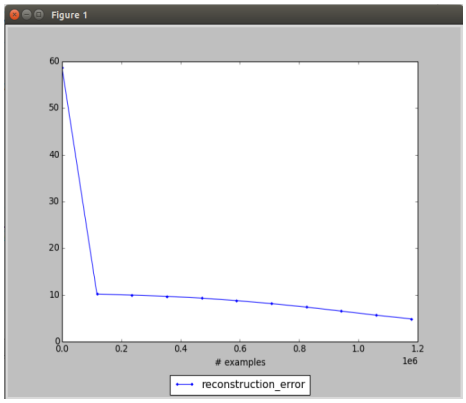
Vediamo nelle seguenti figure il grafico dell'errore di ricostruzione, calcolato come errore quadratico medio tra i valori effettivi e quelli campionati, al crescere del numero di esempi visti dalla rete.
Sono stati fatti variare due parametri: il numero di iterazioni k per ogni passo di Contrastive Divergence e il numero di epoche in cui la rete apprende la distribuzione.
Possiamo notare che l'errore di ricostruzione decade molto velocemente e già con 25 epoche si ottiene un valore prossimo a zero e che al variare del numero di passi di campionamento k si ottengono approssimativamente gli stessi risultati.



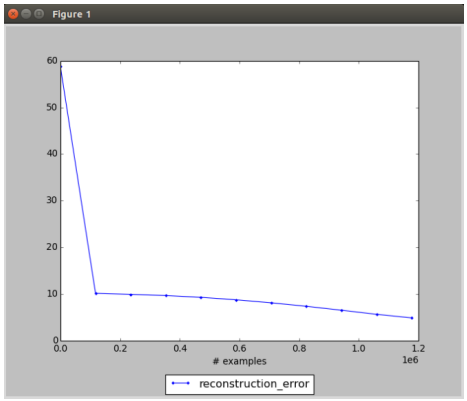
(a) k = 1



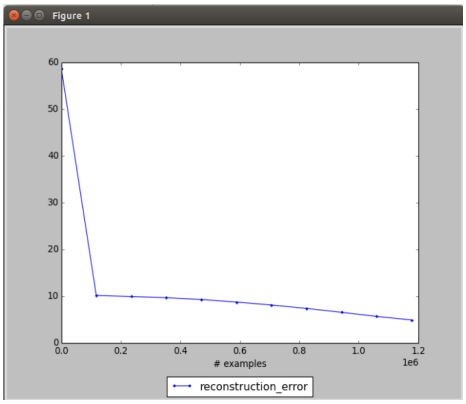
(b) k = 2



(c) k = 5

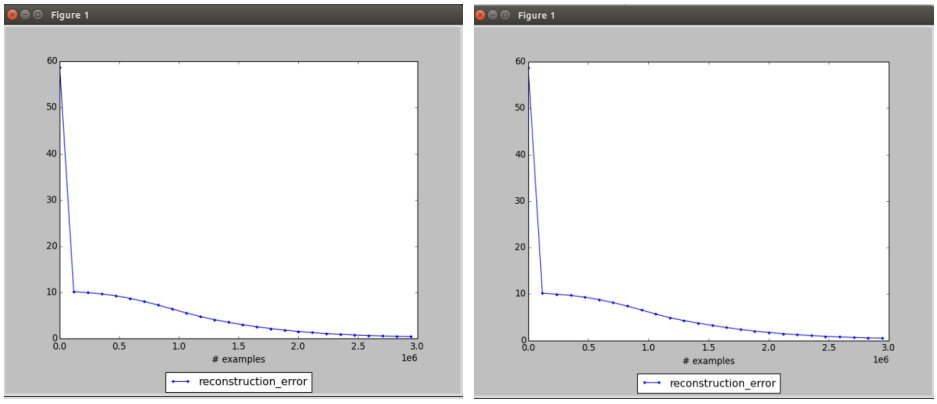


(d) k = 10



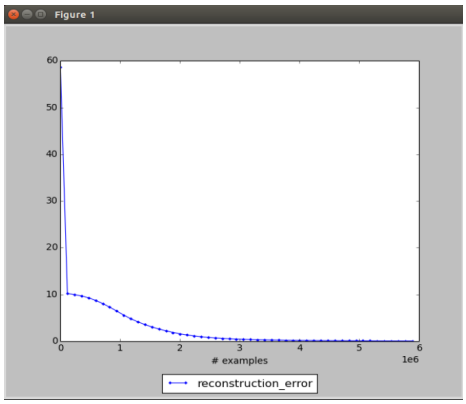
(e) k = 20

Figure 9: Errore di ricostruzione dopo 10 epoche, per differenti valori di k dell'algoritmo k-CD e distribuzione dei dati bernoulliana

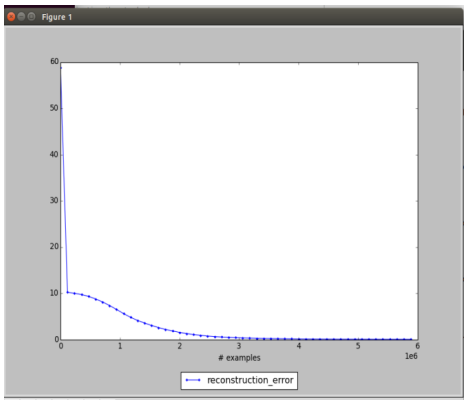


(a) k = 1 (b) k = 10

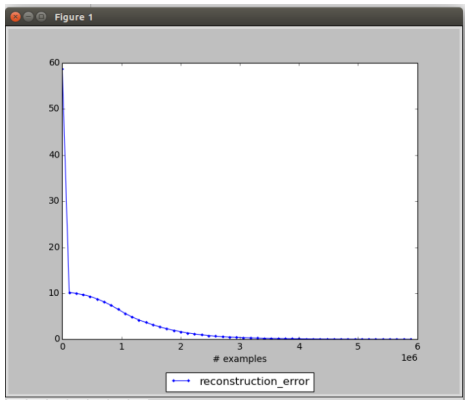
Figure 10: Errore di ricostruzione dopo 25 epoche, per differenti valori di k dell'algoritmo k-CD e distribuzione dei dati bernoulliana



(a) k = 1

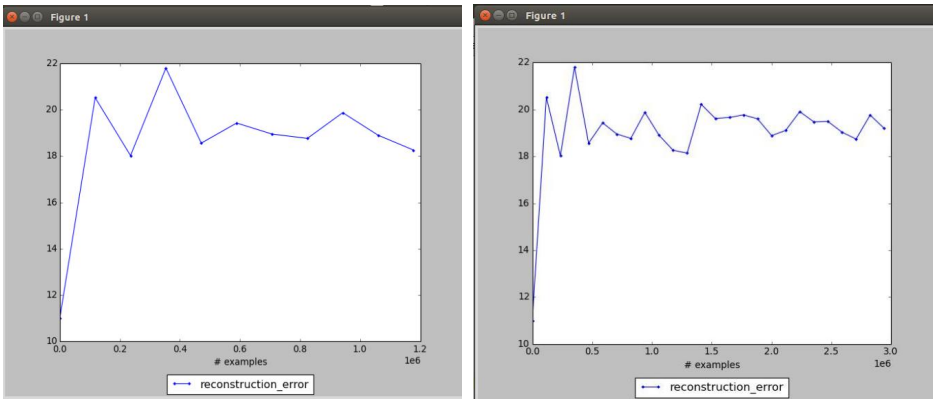


(b) k = 2



(c) k = 5

Figure 11: Errore di ricostruzione dopo 50 epoche, per differenti valori di k dell'algoritmo k-CD e distribuzione dei dati bernoulliana



(a) numero di epoche = 10 (b) numero di epoche = 25

Figure 12: Errore di ricostruzione con k = 1, differenti valori del numero di epoche e distribuzione dei dati gaussiana