Modele generatywne 2: gęstość

Jacek Tabor

13 października 2023

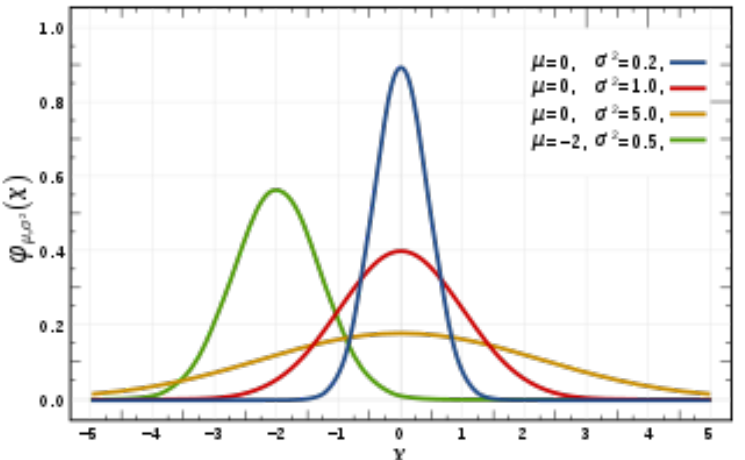
1 Rozkłady normalny

Rozkład jednowymiarowy Rozkład normalny $N(m, \sigma^2)$, gęstość $\frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{1}{2\sigma^2}(x-m)^2)$.

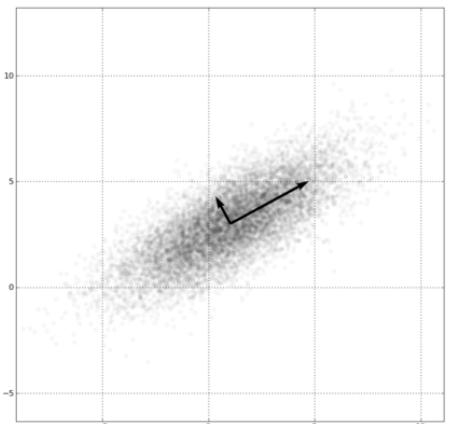
Rozkład normalny wielowymiarowy Standardowy rozkład normalny $N(0, I)$.

Rozkład $N(m, \Sigma)$ o średniej m i macierzy kowariancji Σ, gęstość

$$N(m, \Sigma) = \frac{1}{(2\pi)^{d/2} \det^{1/2} \Sigma} \exp(-\frac{1}{2}(x-m)\Sigma^{-1}(x-m)).$$



Rysunek 1: jednowymiarowy



Rysunek 2: Dwuwymiarowy

Losowanie z rozkładu normalnego Zakładam, że potrafimy losować punkt w z $N(0,1)$.

JEDNOWYMIAROWY:

- aby wylosować z rozkładu $N(m, \sigma^2)$:

$$m + \sigma z.$$

WIELOWYMIAROWY:

- aby wylosować z rozkładu $N(0, I)$, losujemy każdą współrzędną niezależnie z $N(0,1)$
- mając z z $N(0, I)$ aby wylosować z rozkładu $N(m, \mathrm{diag}(\sigma_i^2))$

$$m + \sigma \odot z,$$

gdzie $\odot$ to mnożenie po współrzędnych.

2 Estymacja gęstości

Ostatnim z omawianych tutaj problemów uczenia nienadzorowanego jest estymacja gęstości. Mając daną próbkę ze zbioru danych, chcemy się dowiedzieć, jaki jest rozkład prawdopodobieństwa danych. Estymacja gęstości znajduje zastosowanie między innymi w klastrowaniu danych, gdzie grupy są oddzielone od siebie obszarami o małej gęstości, czy też w generowaniu danych, gdzie chcemy uzyskać nowe przykłady spełniające określone kryteria. W tej sekcji

opiszemy metodę parametrycznej estymacji gęstości bazującą na maksymalizacji funkcji wiarygodności, jak i metodę nieparametryczną używającą estymacji jądrowej.

Dopasowanie rozkładu do danych:

- parametryczne: szukanie optymalnych parametrów
- nieparametryczne: kernelowa (jądrowa estymacja gęstości)

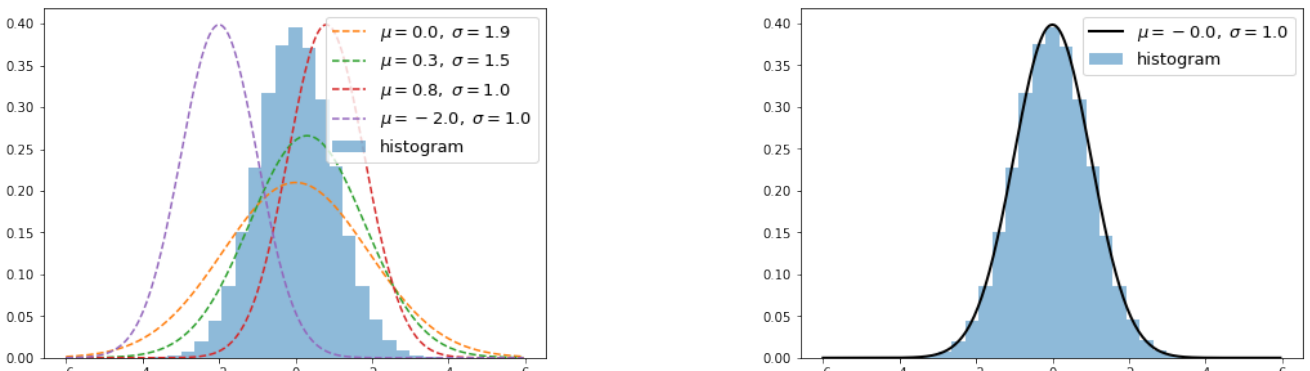
Funkcja wiarygodności (log-likelihood) Zacznijmy od podejścia parametrycznego polegającego na maksymalizacji funkcji wiarygodności. Załóżmy, że mamy daną próbkę $X = (x_i)_{i=1}^N$ i rodzinę gęstości $(f_\theta)_{\theta\in\Theta}$. Chcemy dobrać taką wartość parametru θ, dla którego gęstość f_θ najlepiej odpowiada danej próbce. W przypadku rodziny rozkładów normalnych $N(m, \Sigma)$, poszukujemy parametrów m oraz Σ. Rysunek 3 przedstawia histogram danych pochodzących ze standardowego rozkładu normalnego oraz przykładowe estymacje rozkładami normalnymi o różnych parametrach. Estymacja metodą największej wiarygodności omawiana w tej sekcji znajduje optymalne dopasowanie.

Jako kryterium jakości estymacji przyjmijmy prawdopodobieństwo wylosowania próbki X z tak wybranej gęstości f_θ. Innymi słowy dążymy do znalezienia takich parametrów, które maksymalizują prawdopodobieństwo wyboru próbki (MLE: maximum likelihood estimation). Zakładając, że dane są od siebie niezależne, mamy:

$$f_\theta(X) = f_\theta(x_1) \cdot \ldots \cdot f_\theta(x_N).$$

Aby pozbyć się mnożenia, rozpatrujemy logarytm z powyższej wartości (tak zwana funkcja wiarygodności próbki, ang. log-likelihood):

$$\mathrm{logl}(X; f_\theta) = \log f_\theta(x_1) + \ldots \log f_\theta(x_N).$$



Rysunek 3: Estymacja rozkładu danych pochodzących ze standardowego rozkładu normalnego opisanego histogramem. Metoda największej wiarygodności daje optymalną estymację.

W metodzie największej wiarygodności maksymalizujemy logl albo równoważnie minimalizujemy $-\mathrm{logl}$, co pozwala interpretować ją jako funkcję kosztu. W zależności od wyboru rodziny rozkładów f_θ, mamy jawne wzory na rozwiązanie, bądź też musimy stosować numeryczne metody optymalizacji poznane w poprzednim rozdziale.

Przedstawimy teraz najważniejsze przykłady estymacji.

Rozkład normalny jednowymiarowy Rozważmy sytuację, w której mamy próbkę $X = (x_i)_{i=1}^N \subset \mathbb{R}$, i chcemy do niej dopasować optymalne parametry rozkładu normalnego $\mathrm{N}(m, \sigma^2)$. Funkcja kosztu log-likelihood dla próbki X wynosi:

$$-\mathrm{logl}(X; \mathrm{N}(m, \sigma^2)) = -\sum_{i=1}^{N} \log(\mathrm{N}(m, \sigma^2)(x_i)) = \tfrac{N}{2} \ln(2\pi) + \tfrac{N}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - m)^2.$$

Przy ustalonym σ^2, powyższa funkcja minimalizuje się dla:

$$m = \mathrm{mean} X.$$

Przy tak wybranym m, dokonujemy minimalizacji względem σ^2, obliczając pochodną względem σ^2 i przyrównując ją do zera. Otrzymujemy, że estymatorem σ^2 jest wariancja:

$$\sigma^2 = \tfrac{1}{n} \sum_{i=1}^{N} (x_i - m)^2 = \mathrm{var} X.$$

W konsekwencji optymalny wybór dla m i σ^2 to średnia $\mathrm{mean}X$ i wariancja $\mathrm{var}X$ z próbki.

SAMODZIELNIE: rozkład $\frac{1}{\lambda}\exp(-\lambda x)$ na $[0,\infty)$, metodą największej wiarygodności wyliczyć optymalne λ dla zbioru danych $x_i \subset \mathbb{R}_+$.

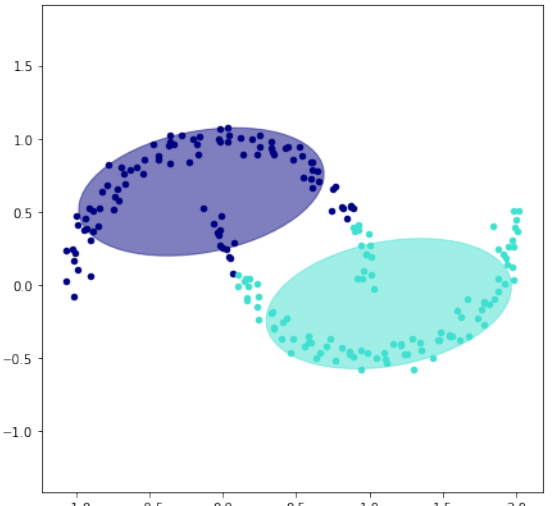
Rozkład normalny wielowymiarowy Powyższą sytuację można rozszerzyć do przypadku wielowymiarowego. Otrzymujemy wtedy, że optymalnymi parametrami wielowymiarowego rozkładu normalnego $\mathrm{N}(\mu,\Sigma)$ dla próbki $X \subset \mathbb{R}^D$ jest średnia i kowariancja X.

Gaussian mixture models Pojedynczy rozkład normalny nie nadaje się zwykle do estymacji gęstości danych. Jednym ze sposobów estymacji rozkładu bardziej złożonych danych jest zastosowanie mieszanki rozkładów normalnych (GMM: Gaussian mixture model) postaci:

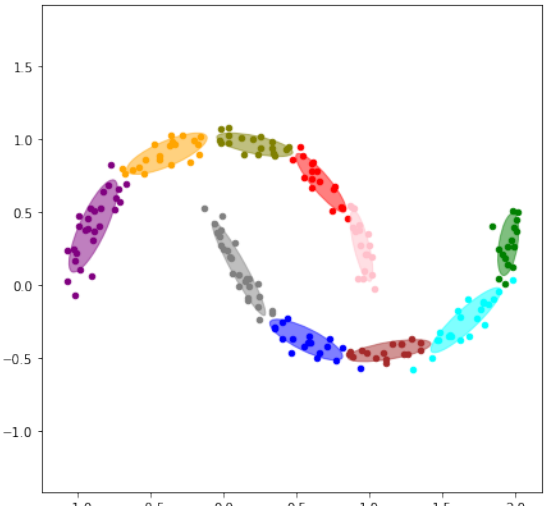
$$\alpha_1 N(m_1,\Sigma_1) + \ldots + \alpha_K N(m_k,\Sigma_K),$$

gdzie $\alpha_i \geq 0$ oraz $\sum_i \alpha_i = 1$. Parametrami tej gęstości są średnie, kowariancje oraz współczynniki mieszanki.

GMM może być z powodzeniem stosowana do grupowania danych, gdzie każda grupa jest opisana pojedynczym rozkładem normalnym. Przeprowadzając algorytm k-means znajdujemy pewien szczególny rodzaj GMM, gdzie $\Sigma_i = I$. Mieszanki dają dość dużą swobodę w estymacji. Zamiast rozkładów normalnych można użyć innych rozkładów, na przykład rozkładów o grubych ogonach. Ogólnie rzecz biorąc GMM jest dość dobrą metodą estymacji, choć dla bardzo dużych wymiarów jego jakość znacząco spada. Rysunek 2 przedstawia estymację za pomocą GMM na 2-wymiarowym przykładzie.



Rysunek 4: GMM k=2



Rysunek 5: GMM k=10

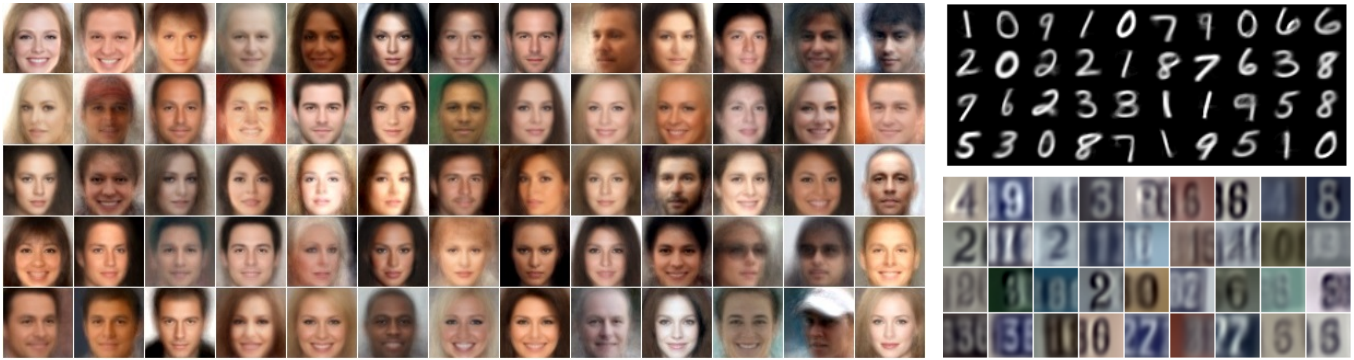


Figure 4: Random samples generated by our MFA model trained on CelebA, MNIST and SVHN

Rysunek 6: Model gmm zastosowany do MNIST, CelebA, SVHN, za pracą „On gans and gmms”.

W przypadku GMM nie istnieją wzory analityczne dające rozwiązanie MLE. Ponadto problem nie jest wypukły, co utrudnia znalezienie globalnego optimum nawet za pomocą metod numerycznych. Standardowo do optymalizacji stosuje się rekurencyjną procedurę zbliżoną do tej używanej w k-means (tak zwana EM: expectation maximization). Rozwiązanie można znaleźć również stosując metodę spadku gradientu, choć jakość rozwiązania będzie gorsza.

Python 1. *Zaimplementuj metodę spadku gradientu dla estymacji GMM. Warto zauważyć, że konieczne jest odpowiednie sparametryzowanie Σ_i oraz α_i, aby zachować wyjściowe ograniczenia. Dokładniej:*

- *dla macierzy kowariancji wybieramy dowolne A i kładziemy $\Sigma = AA^T$. Dzięki temu Σ będzie symetryczne i dodatnio określone (aby zmniejszyć liczbę parametrów możemy od razu rozważać A symetryczne),*
- *aby parametry $\alpha_1, \ldots, \alpha_k$ były nieujemne i sumowały sie do jedynki, bierzemy dowolne $r_i \in \mathbb{R}$ i kładziemy*

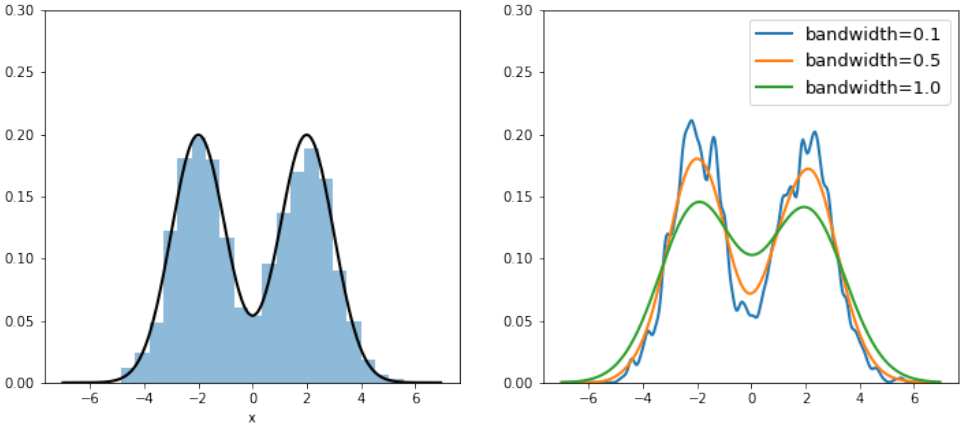
$$\alpha_i = \frac{\exp(r_i)}{\sum_j \exp(r_j)}.$$

Przy takiej parametryzacji zamiast szukać Σ i α_i, szukamy A oraz r_i.

Kernelowa estymacja gęstości Przykładem nieparametrycznej estymacji gęstości jest estymacja kernelowa (jądrowa). Mając daną próbkę $x = (x_i)_{i=1}^N \subset \mathbb{R}$, dokonujemy estymacji gęstości postaci

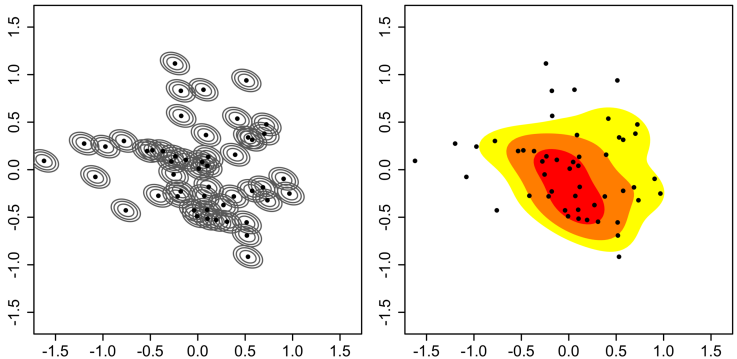
$$f = \tfrac{1}{N} \sum_{i=1}^{N} N(x_i, h^2),$$

gdzie h jest szerokością jądra (w tym przypadku rozpatrujemy jądro gaussowskie). Innymi słowy każdy punkt zbioru X jest reprezentowany przez rozkład normalny o średniej w tym punkcie i ustalonej wariancji. W konsekwencji jeśli w danym obszarze znajduje się wiele punktów, gęstość prawdopodobieństwa wzrasta.



Rysunek 7: Kernelowa estymacja gęstości dla danych pochodzących z mieszanki 2 rozkładów normalnych z różną wartościa szerokości jądra.

Łatwo zauważyć, że jeżeli h jest za duże lub za małe (patrz rysunek 7), to tak utworzona gęstość nie jest bliska tej, z której zostały wygenerowane dane. Jeżeli jest za małe, to nadmiernie dopasowujemy się do istniejących danych (overfitting), jeżeli jest za duże, to można mówić o underfitting (nadmiernie wygładzamy). Powstaje więc w sposób naturalny pytanie, w jaki sposób należy dobrać właściwe h.



Rysunek 8: Kernelowa estymacja gęstości.

Reguła Silvermana jest jednym ze sposobów estymacji szerokości jądra. Wzór dla podzbioru $\mathbb{R}$ o liczności n i odchyleniu standardowym σ wynosi:

$$h = \left(\frac{4}{3N}\right)^{\frac{1}{5}} \sigma \approx 1.06\sigma N^{-1/5}. \tag{1}$$

Daje on dobre wyniki głównie w sytuacji, gdy dane pochodzą z rozkładu normalnego (albo z rozkładu o zbliżonym kształcie), w przypadku ogólnych rozkładów nie daje wartości optymalnych.

Kernelową estymację gęstości stosuje się także w przypadku wyżej wymiarowym, i wtedy przybliża się gęstość za pomocą

$$f = \frac{1}{N} \sum_i N(x_i, \Sigma),$$

gdzie Σ jest typowo brane jako reskalowana macierz kowariancji dla danych, bądź identyczność. Tu również pojawia się problem jak dobrać skalę (szerokość jądra) i można skorzystać z analogicznej reguły Silvermana dla przypadku wyżej wymiarowego. Efektywność estymacji jądrowej jest najlepsza w niskim wymiarze (mniejszym niż 5).

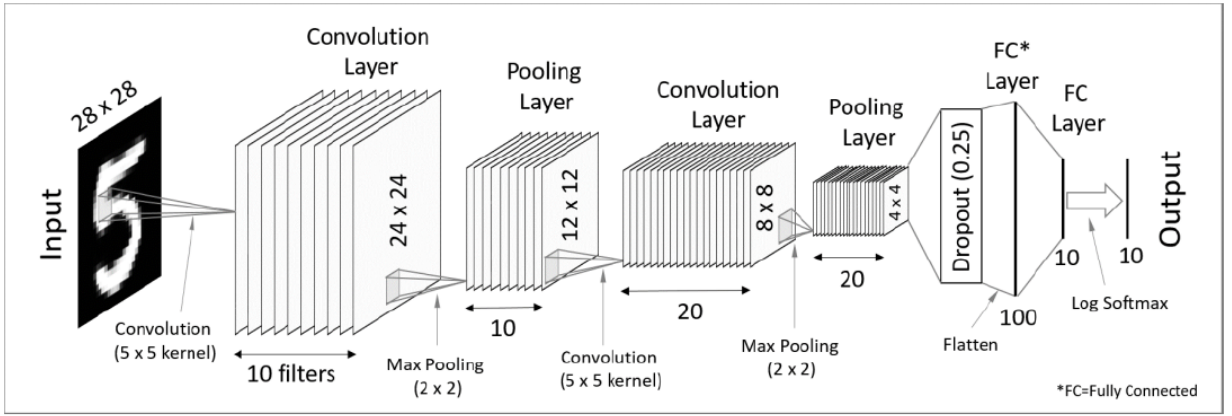
3 Klasyfikacja

Mamy punkty $x_i \in X \subset \mathbb{R}^D$, z których klasa każdego $y_i \in \{1, \ldots, K\}$. Chcemy stworzyć/nauczyć sieć neuronową, która potrafi dobrze przewidzieć klasę y_i dla punktu x_i

Typowym rozwiązaniem jest stworzenie modelu probabilistycznego określającego prawdopodobieństwo przynależności punktu do poszczególnych klas. W tym modelu dla każdego punktu chcemy wyznaczyć tak zwany *rozkład a posteriori* $p(1|x_i), \ldots, p(K|x_i)$ określający przynależność punktu x_i do każdej z K-klas. Jako finalną decyzje o klasyfikacji przyjmujemy tę najbardziej prawdopodobną, czyli:

$$c(x) = \arg \max_{k \in \{1,\ldots,K\}} p(k|x_i).$$

Żeby zbudować taki model, musimy sprarametryzować prawdopodobieństwa a posteriori, a następnie zbudować



funkcję stratu definiującą kryterium optymalizacyjne. W celu sparametryzowania $p(k|\cdot)$, określmy moc przyporządkowania punktu x do klasy k, wykorzystując sieć postaci:

$$f : \mathbb{R}^D \to \mathbb{R}^K$$

gdzie $f = (f_1, \ldots, f_K)$ oraz

$$f_k(x) \in \mathbb{R} \text{ , dla } k = 1, \ldots, K.$$

Softmax W celu transformacji powyższych funkcji do prawdopodobieństw wykorzystamy funkcję *softmax* postaci:

$$p(k|x) = \frac{\exp(z_k)}{\sum_{j=1}^{K} \exp(z_j)} \in (0,1), \tag{2}$$

gdzie $z_k = f_k(x)$. Dzięki zastosowaniu monotonicznej funkcji $\exp$ wyższa wartość $f_k(x)$ przekłada się na wyższe prawdopodobieństwa $p(k|x)$ oraz zapewnia nieujemność. Dodatkowo normalizacja powoduje, że wszystkie wartości sumują się do 1, co stanowi poprawnie zdefiniowany rozkład. Postać (2) nie jest zadana sztucznie, ale można ją również wyprowadzić między innymi z zasady maksymalnej entropii.

Standardowa metoda polega na maksymalizacji funkcji wiarygodności $\prod_{i=1}^{N} p(y_i|x_i)$. Biorąc logarytm (z minusem) otrzymujemy problem minimalizacji:

$$-\mathrm{logl}(X,Y) = -\sum_{i=1}^{N} \log p(y_i|x_i), \tag{3}$$

W terminologii sieci neuronowych powyższą funkcję nazywa się często entropią krzyżową (cross-entropy). Entropia krzyżowa W połączeniu z funkcją softmax stanowi podstawową funkcję kosztu stosowaną w klasyfikacyjnych sieciach neuronowych.

Aby to efektywnie tensorowo zaimplementować w sieci, używamy reprezentacji one-hot:

$$y_i = (0,\ldots,0,\underbrace{1}_{y_i},0,\ldots,0)$$

I wtedy wyrażenie $\log p(y_i|x_i)$ rozpisujemy jako

$$\log p(y_i|x_i) = \sum_{k=1}^{K} y_i^k \log p(k|x_i) = y_i^T \log \mathrm{softmax} f(x_i),$$

gdzie y_i^k oznacza k-tą współrzędną y_i.

Czyli finalnie loss

$$loss = -\sum_i y_i^T \log \mathrm{softmax} f(x_i) = CE(y_i, \mathrm{softmax} f(x_i)),$$

gdzie zaraz wyjaśnię co to jest CE (cross-entropy) [mierzy zgodność dwóch rozkładów, prawdziwego czyli y_i i wytworzonego przez sięć, czyli $\mathrm{softmax} f(x_i)$].

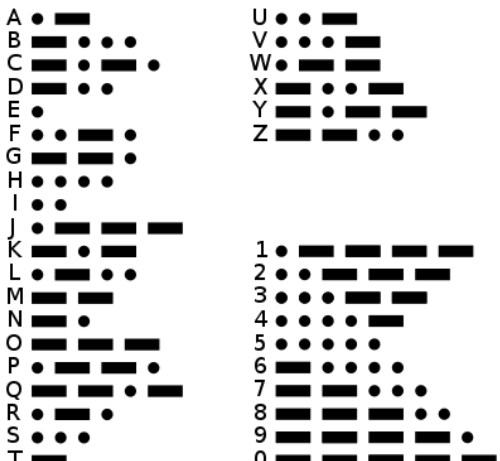
4 Entropia, entropia krzyżowa i DKL

Elementy teorii informacji Załóżmy, że mamy rozkład prawdopodobieństwa na zbiorze $S = \{s_1, \ldots, s_m\}$, czyli litera s_i pojawia się z prawdopodobieństwem p_i Kodujemy te litery (na przykład algorytmem Huffmana) za pomocą kodu binarnego, czyli długość kodu litery s_i wynosi l_i. Oczekiwana długość kodu to

$$\sum_i p_i l_i.$$

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.



Teraz korzystając z nierówności Krafta, można pokazać, że optymalne rozwiązanie dostaje się biorąc $l_i = -\log_2 p_i$. Intuicja jest taka, że im rzadziej się pojawia słowo, tym dłuższy może mieć kod.

Czyli finalnie, nakrótsza długość kodu służąca do zakodowania liter z S wynosi

$$-\sum_i p_i \log_2 p_i.$$

To się nazywa entropią (często zamienia się logarytm z podstawy 2 na naturalny).

Entropia Mamy rozkład p_i. Wtedy entropia rozkładu

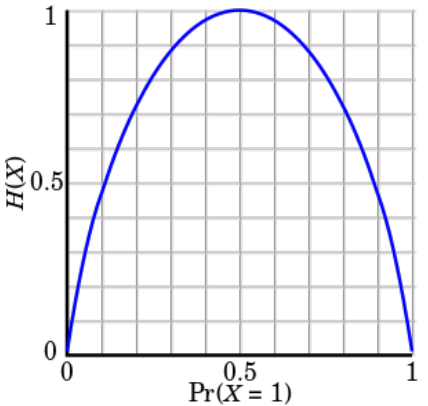
$$-\sum_i p_i \log p_i.$$

Entropia (gdy logarytm o podstawie 2) odpowiada jaka jest optymalna długość kodu. Czyli jeżeli symbol i pojawia się z prawdopodobieństwem p_i, to optymalna długość kodu wynosi $-\log_2 p_i$.

Porównywanie dwóch rozkładów – entropia krzyżowa (Cross-Entropy). Załóżmy, że zbudowaliśmy optymalny kod dla rozkładu q_i, i kodujemy nim rozkład p_i. Uzysukujemy wtedy tak zwaną *entropię krzyżową (cross-entropy)*

Dwa rozkłady dyskretne p, q

$$CE(p, q) = -\sum_i p_i \log q_i.$$



Rysunek 9: Entropia dla rozkładu dwupunktowego o prawdopodobieństwie $p, 1 - p$ßn$[0, 1]$.

Wylicza jaka jest długość kodu rozkładu q, jeżeli kompresujemy za pomocą kodów dopasowanych do p [w uproszczeniu kodowanie Huffmana].

Dywergencja Kullbacka-Leiblera:

$$D_{KL}(p, q) = -\sum_i p_i \log(p_i/q_i)$$

Ile tracimy kompresując p_i za pomocą kodów q_i (w stosunku do optymalnego). Zawsze ≥ 0, równe zero jeżeli mamy równość.

Wersja entropii krzyżowej i dywergencji Kullbacka-Leiblera dla gęstości f, g Entropia krzyżowa – dwa rozkłady ciągłe f, g:

$$CE(f, g) = -\int f(x) \log g(x) dx.$$

Wersja D_{KL} dla gęstości

$$D_{KL}(f\|g) = -\int f(x) \log(g(x)/f(x)) dx.$$

Zawsze ≥ 0, równe zero jeżeli mamy równość.

W szczególności można wyliczyć jawny wzór dla rozkładów normalnych:

$$D_{\mathrm{KL}}\left(N(\mu_0, \Sigma_0)\|N(\mu_1, \Sigma_1)\right) = \frac{1}{2}\left(\mathrm{tr}\left(\Sigma_1^{-1}\Sigma_0\right) - d + (\mu_1 - \mu_0)^{\mathsf{T}} \Sigma_1^{-1} (\mu_1 - \mu_0) + \ln\left(\frac{\det \Sigma_1}{\det \Sigma_0}\right)\right).$$

W ważnym przypadku gdy pierwszy rozkład $N(\mu, \mathrm{diag}(\sigma_i^2))$ ma macierz kowariancji diagonalną $\Sigma = \mathrm{diag}(\sigma_i^2)$, a drugi to $N(0, I)$, dostajemy

$$D_{KL}(N(\mu, \mathrm{diag}(\sigma_i^2))\|N(0, I)) = \frac{1}{2}\big(\sum_i \sigma_i^2 + \sum_i \mu_i^2 - d - \sum_i \log(\sigma_i^2)\big).$$