Autoencodeurs variationnels (VAE)

=> Autoencodeur dont la distribution des encodages est régularisée pendant l'entrainement afin de s'assurer que son espace latent a de bonnes propriétés permettant de générer de nouvelles données.

Combinaison de trois idées :

- autoencodeurs
- approximation et borne inférieure variationnelle
- trick de reparamétrisation.

=> les autoencodeurs présentent certaines limites pour la génération de nouveau contenu

- Difficile de s'assurer que l'encodeur organisera l'espace latent de manière intelligente et compatible avec le processus de génération de données (normal rien ne l'y oblige).
- Il aura donc tendance pendant son entrainement à profiter de toutes les possibilités de surapprentissage pour réaliser sa tâche au mieux.

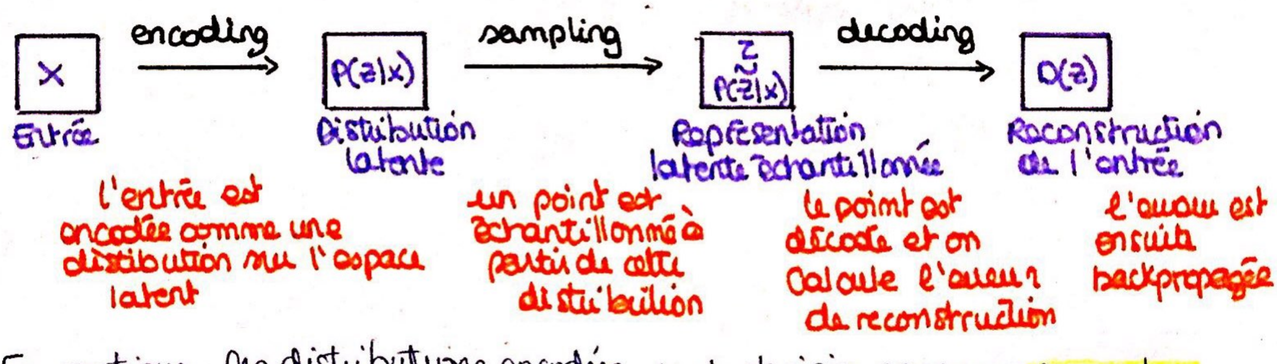
Solution : le régulariser.

* Définition des autoencodeurs variationnels.

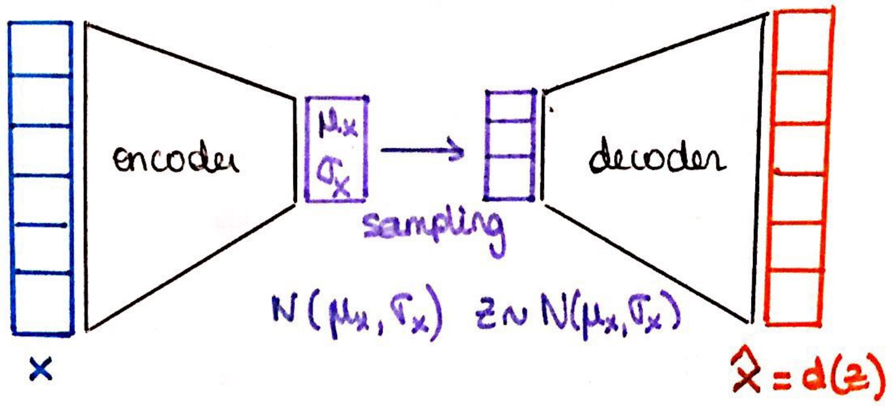
=> on veut utiliser le décodeur pour générer des données.

= Architecture composée d'un encodeur et d'un décodeur qui est entrainée pour minimiser l'erreur de reconstruction.

Pour introduire une régularisation de l'espace latent
=> encodage de l'entrée comme une distribution et non un point.



En pratique, les distributions encodées sont choisies comme normales afin que l'encodeur puisse être entrainé à retourner la moyenne et la matrice de covariance qui décrivent ces distributions gaussiennes.



fonction de perte

$$loss = \|x - \hat{x}\|^2 + KL[N(\mu_x, \sigma_x), N(0, I)]$$

terme de reconstruction
=> cherche la perf de l'encodeur-decodeur

terme de régularisation
=> organise l'espace latent en rendant les distributions proche d'une dist normale.
= Kulback-Leibler divergence

La régularité qui est attendue de l'espace latent afin de rendre possible le processus de génération des données peut-être exprimée par deux propriétés principales:

- La continuité : deux points proches dans l'espace latent ne devraient pas donner deux contenus ≠ une fois décodés.
- La complétude : pour une distribution donnée, un point échantillonné de l'espace latent devrait donner un contenu "significatif" une fois décodé.

=> Pour satisfaire ces deux propriétés, on doit régulariser la matrice de covariance et la moyenne des distributions renvoyées par l'encodeur. (on force les distributions à être proches d'une loi normale centrée-réduite).

=> La régularisation crée un "gradient" sur les informations encodées dans l'espace latent.

* Détails mathématiques des VAE

Cadre probabilistique et hypothèses

Supposons que nos données x sont générées à partir d'une variable latente z (la représentation encodée). Pour chaque point on a donc :

(1) une représentation latente z est échantillonné à partir de la distribution prior $p(z)$.

(2) des données x sont échantillonnées à partir de la distribution conditionnelle $p(x|z)$

$$\textcircled{z} \longrightarrow \textcircled{x}_N$$

Ainsi :

- Encodeur probabiliste : distribution de la variable encodée par rapport à la variable décodée. $p(z|x)$
- Decodeur probabiliste : distribution de la variable décodée par rapport à la variable encodée $p(x|z)$

=> des représentations encodées z dans l'espace latent sont supposées suivre la distribution prior $p(z)$

Bayes : $$p(z|x) = \frac{p(x|z)\,p(z)}{p(x)} = \frac{p(x|z)\,p(z)}{\int p(x|u)\,p(u)\,du}$$

En supposant que :

- $p(z) \sim \mathcal{N}(0, I)$
- $p(x|z) \sim \mathcal{N}(f(z), cI)$ $\quad f \in F \quad c > 0$

non spécifiée ici

fonction déterministe

En théorie, pour f définie et fixée, comme on connait $p(z)$ et $p(x|z)$, on peut calculer $p(z|x)$ = inférence bayésienne classique

=> en fait, ce type de calcul est souvent insoluble (en raison de l'intégrale au dénominateur)

=> utilisation de techniques d'approximation (inférence variationnelle)

Inférence variationnelle

=> permet d'approcher des distributions complexes en définissant une famille de distribution paramétrée et de rechercher la meilleure approximation de notre distribution cible parmi cette famille. Le meilleur est celui qui minimise une mesure d'erreur d'approx° donnée (divergence de Kullback-Leibler entre l'approx° et la cible) et est trouvé par descente de gradient sur les paramètres de la famille.

VAE = approximation de $p(z|x)$ par une distribution gaussienne $q(z|x)$ de moyenne et covariance définies par g et h du paramètre x.

$$q(z|x) \sim N(g(x), h(x)) \quad g \in G \quad h \in H$$

(famille de candidats)

deux familles paramétrées

=> trouver la meilleure approximation en optimisant g et h pour minimiser la divergence de Kullback-Leibler entre l'approx° et la cible $p(z|x)$.

$$
\begin{aligned}
(g^*, h^*) &= \underset{(g,h)\in G\times H}{\arg\min} \; KL(q(z|x), p(z|x)) \\
&= \arg\min \left(E_{z\sim q_x}(\log(q(z|x)) - E\left(\log \frac{p(x|z)p(z)}{p(x)}\right)\right) \\
&= \arg\min \left(E(\log(q(z|x))) - E(\log(p(z))) - E(\log(p(x|z))) + E(\log(p(x)))\right) \\
&= \arg\max \left(E(\log(p(x|z))) - KL(q(z|x), p(z))\right) \\
&= \arg\max \left(\underbrace{E\left(\frac{-\|x - f(z)\|^2}{2c}\right)}_{\text{erreur de reconstruction}} - \underbrace{KL(q(z|x), p(z))}_{\text{régularisation}}\right)
\end{aligned}
$$

compromis entre la maximisation de la probabilité des observations et le fait de rester proche de la distribution antérieure (prior)

=> en pratique, on ne connait pas f (qui définit la moyenne du décodeur). Or la régularité et donc la performance dépend fortement du choix de f (pour faire des optimisations, les deux seuls leviers sont le paramètre c (variance de la probabilité) et la fonction f (moyenne de la proba))

Donc, on recherche :

$$
\begin{aligned}
f^* &= \underset{f\in F}{\arg\max} \; E_{z\sim \overbrace{q_x^*}^{\text{dépends de } f}}(\log(p(x|z))) \\
&= \underset{f\in F}{\arg\max} \; E_z \sim q_x^*\left(\frac{-\|x - f(x)\|^2}{2c}\right)
\end{aligned}
$$

Pour x donné, on veut maximiser $P(\hat{x}=x)$ quand $z \sim q(z|x)^*$ puis $\hat{x} \sim p(x|z)$.

Plus c est élevé, plus la variance autour de $f(z)$ est élevée (décodeur) plus on privilégie le terme de régularisation par rapport au terme de reconstruction.

Reparamétrisation Trick

Modèle probabiliste :

$$(f^*, g^*, h^*) = \underset{(f,g,h)\in F\times G\times H}{\operatorname{argmax}} \left[E_{z\sim q_x}\left(\frac{-\|x-f(z)\|^2}{2c}\right) - KL(q(z|x), p(z)) \right]$$

=> on décide d'exprimer f, g et h sous forme de réseau de neurones

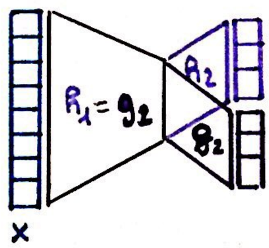
En pratique, g et h partagent une partie de leur architecture et de leurs poids

$$g(x) = g_2(g_1(x)) \qquad h(x) = h_2(h_1(x)) \qquad g_1(x) = h_1(x)$$

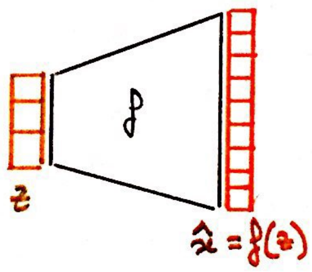
censée être une matrice carrée

Pour simplifier le calcul et réduire le nombre de paramètres, on suppose que les variables sont indépendantes (matrice de covariance diagonale). => $h(x)$ est le vecteur des éléments diagonaux de la mat de covariance. => approximation de $p(z|x)$ sera moins précise

Partie encodeur



Partie décodeur.



=> le processus d'échantillonnage doit être exprimé de manière à permettre la rétropropagation de l'erreur à travers le réseau.

Solution : astuce de reparamétrage

=> rend possible la descente de gradient malgré l'échantillonnage aléatoire qui se produit à mi chemin de l'architecture et consiste à utiliser le fait que si z est une variable aléatoire suivant une distribution gaussienne avec une moyenne $g(x)$ et une cov $h(x)$, alors :

$$z = g(x) + z_0 h(x) \qquad z_0 \sim N(0, I)$$