1. Na czym polega segmentacja obrazu.
2. W autoenkoderze Wassersetina do klasycznej funkcji kosztu MSE dodany jest dodatkowo człon MMD. Jaką pełni rolę w procesie uczenia?

W autoenkoderze Wasserstein z dodatkowym członem MMD (Maximum Mean Discrepancy) funkcja kosztu składa się z dwóch części: błędu średniokwadratowego (MSE) oraz składowej MMD. Rola składowej MMD w procesie uczenia to zachęcanie modelu do generowania danych z przestrzeni latentnej, które są rozłożone równomiernie i podobne do rozkładu danych treningowych.

MMD jest miarą odległości między dwoma rozkładami prawdopodobieństwa. W przypadku autoenkodera Wasserstein z MMD, chodzi o minimalizację odległości między rozkładem danych w przestrzeni latentnej, a rozkładem danych treningowych. Działa to jako regularyzacja, która pomaga zapobiegać przetrenowaniu i zmusza autoenkoder do generowania bardziej zrównoważonych, bardziej reprezentatywnych próbek.

Wprowadzenie składowej MMD pomaga w uniknięciu problemów związanych z mode collapse (zjawisko, w którym generator generuje jedynie ograniczoną różnorodność przykładów) i sprawia, że autoenkoder generuje bardziej różnorodne i realistyczne dane. MMD pomaga w tym, że penalizuje odchylenia między rozkładem danych w przestrzeni latentnej a rzeczywistym rozkładem danych treningowych, co skutkuje bardziej stabilnym uczeniem.

1. W sieciach GAN, co jest wejściem dla generatora?

W sieciach GAN (Generative Adversarial Networks), generator przyjmuje jako wejście losowy szum, który jest zazwyczaj generowany przy użyciu rozkładu normalnego lub innych rozkładów prawdopodobieństwa. Ten losowy szum jest nazywany wektorem szumu lub wektorem latentnym. Generator używa tego wektora szumu jako punktu startowego do generowania danych, które są podobne do danych treningowych.

Generator w sieci GAN działa jak funkcja przekształcająca wektor szumu w obraz (w przypadku GAN-ów stosowanych w dziedzinie przetwarzania obrazów). Innymi słowy, generator próbuje nauczyć się odwzorowywać rozkład danych treningowych na przestrzeni wektorów szumu.

Proces trenowania GANów polega na dostosowywaniu wag generatora i dyskryminatora w sposób, który prowadzi do generowania przez generator danych, które są trudne do odróżnienia od rzeczywistych danych przez dyskryminatora. To rywalizacyjne podejście pomiędzy generatorem a dyskryminatorem powoduje, że generator staje się coraz lepszy w generowaniu realistycznych danych.