

# Projekt: Wypełnianie dziur w obrazkach z wykorzystaniem MFA

Paweł Goliszewski

20 lutego 2020

## 1 Wprowadzenie

Poniższy dokument opisuje projekt dotyczący uzupełniania dziur w obrazach zbioru MNIST. Jest to zbiór ręcznie pisanych cyfr w skali szarości. Opis jak korzystać z kodu jest w pliku readme. Projekt powstał we współpracy z dr. Markiem Śmieją (Uniwersytet Jagielloński).

## 2 Opis modelu

Mamy zbiór danych (obrazków)  $X$ , gdzie punkt  $x \in X$  może mieć nieznane wartości na pewnych współrzędnych. Taki brakujący punkt oznaczmy przez  $(x, J)$ , gdzie  $x \in R^D$  to pewien reprezentant, a  $J \subset \{1, \dots, D\}$  służy do oznaczenia, które współrzędne są brakujące. Każdy z obrazów ma wymiary 28x28 pikseli.

Do wytrenowania modelu wykorzystano 5000 próbek ze zbioru MNIST - w każdym wykonano dziurę o wymiarach 10x10 pikseli w pobliżu środka obrazu. Jako, że konwolucyjna sieć neuronowa nie przyjmie obrazka "z dziurą" lub fragmentu wypełnionego wartościami nan, to dokonujemy ściągnięcia dziury do punktu i wykonanie odpowiedniego przekształcenia o czym więcej w kolejnym rozdziale. Dodajemy także dodatkowe warstwy oznaczające pierwotne położenie punktów  $x$  oraz  $y$ . Jako, że w efekcie dostajemy współrzędne pikseli niecałkowite, to dokonujemy interpolacji liniowej na współrzędne całkowite. Tak przygotowany obrazek przepuszczamy przez 2 warstwy konwolucyjne (po każdej z nich dokonujemy batch normalization [1]) a następnie 2 warstwy liniowe, na których wyjściu dostaniemy parametry Gaussian Mixture Model.

Zformalizujmy nieco zapis. Celem jest estymacja rozkładu brakujących współrzędnych dla danego  $(x, J)$ . Dokładniej, chcemy znaleźć gęstość  $F_{x,J}$  zdefiniowaną na  $R^N$ , gdzie  $N = |J|$ , która mówi jakie jest prawdopodobieństwo poszczególnych wartości dla  $x_J$ .

W celu wypełnienia dziury posłużymy się parametrycznym modelem MFA (mixture of factor analyzers)[2] dla zdefiniowania rodziny gęstości rozkładów  $F_{x,J}$ :

$$F_{x,J} = \sum_{i=1}^k p_i N(m_i, \Sigma_i) \quad (1)$$

Gdzie  $N(m, \Sigma)$  oznacza rozkład normalny o średniej  $m$  i macierzy kowariancji  $\Sigma$ . W tym modelu  $F_{x,J}$  jest mieszkanką rozkładów gaussowskich, w której każdy Gauss ma średnią  $m_i \in R^N$  oraz macierz kowariancji  $\Sigma_i = A_i A_i^T + \text{diag}(d_i)$ , gdzie  $A_i \in R^{N \times L}$ , oraz  $d_i \in R$ , D ( $\text{diag}(d_i)$  jest macierzą diagonalną z diagonalą  $d_i$ ). Zatem do wyestymowania są parametry  $p_i, m_i, A_i, d_i$ . Te parametry bierzemy z wyjścia sieci neuronowej. Brakujące współrzędne wypełniamy średnią ważoną:  $\frac{1}{k} \sum_{i=1}^k p_i m_i$ . Więcej o tym w sekcji 4.

### 3 Ściąganie dziur do punktu

Jako, że sieć nie przyjmie brakujących współrzędnych, przed umieszczeniem obrazu w sieci dokonano odpowiedniego przekształcenia, które jest zobrazowane na rysunku 1. Przykładowo dla dolnego białego paska dokonujemy przekształcenia:

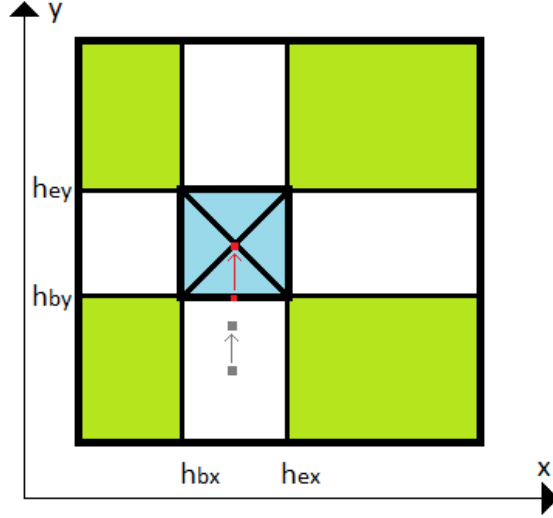
$$y_{new} = y \cdot \left( c \left| x - \frac{h_{bx} + h_{ex}}{2} \right| + d \right) \quad (2)$$

Gdzie:

$$d = \frac{h_{by} + h_{ey}}{2h_{by}} \quad (3)$$

$$c = \frac{2 - 2d}{h_{ex} - h_{bx}} \quad (4)$$

Założenie jest takie, że przekształcamy współrzędną  $y$  przez przemnożenie jej o wartość zależną liniowo od współrzędnej  $x$  danego punktu.



Rysunek 1: Przekształcenie przed umieszczeniem w sieci. Zielone obszary nie zostają przekształcone. Na niebiesko zaznaczono dziurę. Białe obszary zostaną "rozciągnięte", tak aby zakryć dziurę. Przykładowo dolna pozycja czerwonego punktu zostanie przesunięta w górną, podobnie z szarym punktem.

## 4 Funkcja kosztu

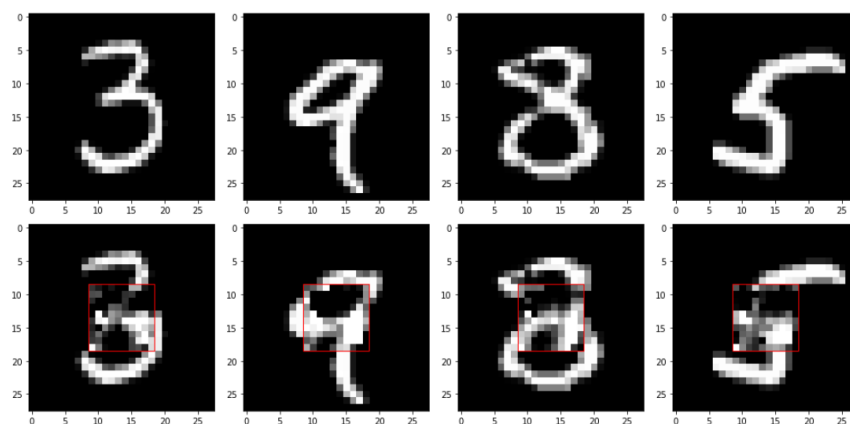
Żeby sprawdzić, jak dobry rozkład znaleźliśmy definiujemy funkcję kosztu (ujemny log likelihood):

$$-\sum_{i=1}^k \log [N(m_i, A_i A_i^T + \text{diag}(d_i))(x_J)] \quad (5)$$

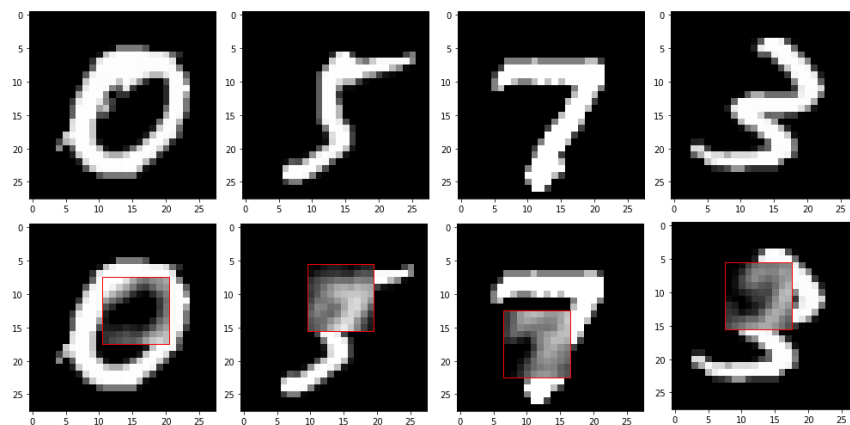
gdzie  $x_J$  oznacza wzięcie współrzędnych  $J$  z  $x$  (prawdopodobieństwo danych współrzędnych mając dany rozkład). Tę funkcję minimalizujemy.

## 5 Przykładowe wyniki

Na rysunku 2 przedstawiono przykładowe wyniki dla dziury w stałym miejscu. Na rysunku 3 przedstawiono przykładowe wyniki dla dziury w losowym miejscu (w odległości nie większej niż 4 piksele od środka).



Rysunek 2: W górnym rzędzie przedstawiono oryginalne obrazy, a poniżej uzupełnienia.



Rysunek 3: W górnym rzędzie przedstawiono oryginalne obrazy, a poniżej uzupełnienia.

## 6 Wnioski

Model lepiej sobie radzi z dziurą w stałym miejscu. Zauważono większą stabilność uczenia po do dodaniu warstw batch normalization. Dla niestałego miejsca uzyskano uzupełnienia takie samo niezależnie od pozycji dziury - prawdopodobnie model pomija dane dotyczące pikseli. Podejrzewa się, że dobranie odpowiednich hiperparametrów polepszyłoby wyniki i dało ostrzejszy obraz, lecz wymaga to większej ilości prób oraz mocy obliczeniowej, co wykracza poza zakres projektu.

## Literatura

- [1] Sergey Ioffe, Christian Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, arXiv:1502.03167v3 [cs.LG]  
<https://arxiv.org/pdf/1502.03167v3> [dostęp 20 lutego 2020]
- [2] Eitan Richardson, Yair Weiss, *On GANs and GMMs*  
<https://papers.nips.cc/paper/7826-on-gans-and-gmms.pdf> [dostęp 20 lutego 2020]