**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**МОСКОВСКИЙ ЭНЕРГЕТИЧЕСКИЙ ИНСТИТУТ**

**Отчет к лабораторной работе №12**

**Дисциплины «cGAN с MNIST.»**

**Выполнили студенты группы А-13-22**

***Кокляева Мария и Сафонова София***

**Преподаватель: *Бартеньев О.В.***

**Москва 2024 г.**

# ***1. model.summary();***

**Дискриминатор:**

**def discriminator():**

model = Sequential()

model.add(Dense(512, input\_dim=np.prod(img\_shape)))

model.add(LeakyReLU(alpha=0.2))

model.add(Dense(512))

model.add(LeakyReLU(alpha=0.2))

model.add(Dropout(0.4))

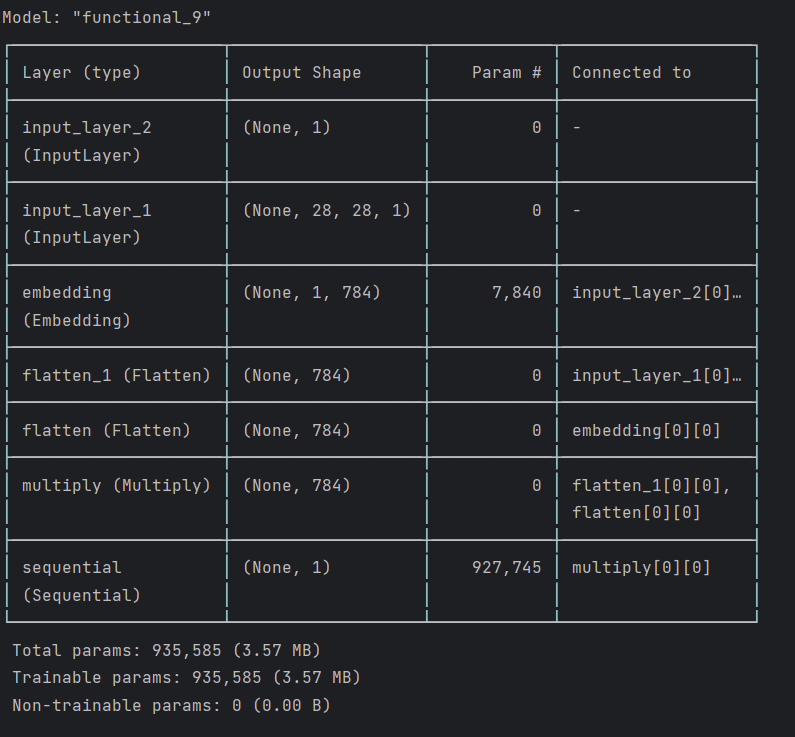
model.add(Dense(512))

model.add(LeakyReLU(alpha=0.2))

model.add(Dropout(0.4))

model.add(Dense(1, activation='sigmoid'))

model.summary()



**Генератор:**

**def generator():**

model = Sequential()

model.add(Dense(256, input\_dim=latent\_dim))

model.add(LeakyReLU(alpha=0.2))

model.add(BatchNormalization(momentum=0.8))

model.add(Dense(512))

model.add(LeakyReLU(alpha=0.2))

model.add(BatchNormalization(momentum=0.8))

model.add(Dense(1024))

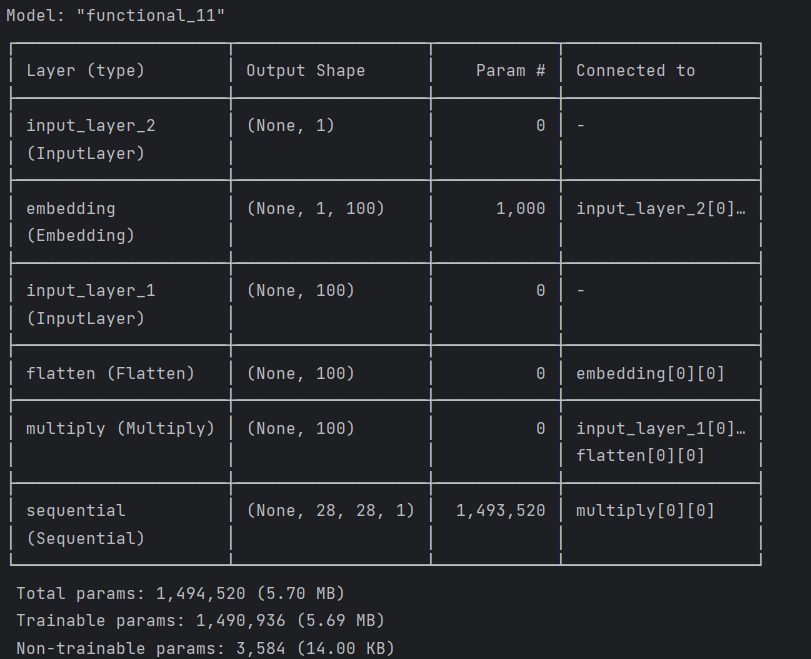
model.add(LeakyReLU(alpha=0.2))

model.add(BatchNormalization(momentum=0.8))

model.add(Dense(np.prod(img\_shape), activation='tanh'))

model.add(Reshape(img\_shape))

model.summary()

****

# ***2. Функции активации и потерь;***

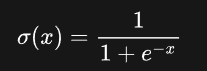
## ***Дискриминатор:***

***Функция активации:***

*model.add(Dense(1, activation='****sigmoid****'))*

*Сигмоидная функция активации - это нелинейная функция, которая преобразует входное значение в диапазоне от отрицательной бесконечности до положительной бесконечности в значение от 0 до 1. Эта функция активации часто используется в нейронных сетях для задач бинарной классификации.*

*Математически сигмоидная функция активации определяется следующим образом:*

**

*Графически сигмоидная функция активации выглядит как S-образная кривая, которая монотонно возрастает и имеет асимптоты на 0 и 1. В частности, если x > 0, то f(x) > 0.5, а если x < 0, то f(x) < 0.5. Значение 0.5 достигается при x = 0.*

*Сигмоидная функция активации используется для преобразования выходного значения нейрона в вероятность, т.е. вероятность того, что входное значение относится к классу 1, если мы работаем с задачей бинарной классификации. Если значение сигмоидной функции близко к 1, то вероятность того, что входное значение относится к классу 1, высока. Если значение близко к 0, то вероятность того, что входное значение относится к классу 1, низкая.*

*Однако сигмоидная функция активации имеет недостаток, который называется проблемой затухания градиента (vanishing gradient problem). Это означает, что при использовании сигмоидной функции активации в глубоких нейронных сетях градиенты могут становиться очень маленькими, что затрудняет обучение. В таких случаях часто используется другая функция активации, например, ReLU (Rectified Linear Unit).*

***Функция потерь:***

*discriminator.compile(loss=[****'binary\_crossentropy'****], optimizer=optimizer, metrics=['Точность'])*

***Перекрёстная энтропия****измеряет разницу между предсказанной вероятностью и истинной вероятностью.*[*3*](https://tr-page.yandex.ru/translate?lang=en-ru&url=https%3A%2F%2Fwww.geeksforgeeks.org%2Fwhat-is-cross-entropy-loss-function%2F)*Она стремится к минимуму, когда предсказанные вероятности приближаются к реальным вероятностям. Штрафует модель за неправильные прогнозы, особенно в случаях, когда предсказания сильно отклоняются от реальных вероятностей.*[*1*](https://yourtodo.ru/ru/posts/analiz-funktsij-poter-i-metrik-v-mashinnom-obuchenii/)

## ***Генератор:***

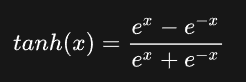
***Функция активации:***

*model.add(Dense(np.prod(img\_shape), activation='****tanh****'))*

*Гиперболический тангенс*

*Гиперболический тангенс (tanh) является одной из наиболее распространенных функций активации в нейронных сетях. Он используется как для классификации, так и для регрессии, а также для обработки изображений и других типов данных.*

*Это функция активации, которая преобразует входные значения в диапазоне от -1 до 1. Формула для вычисления гиперболического тангенса выглядит следующим образом*

**

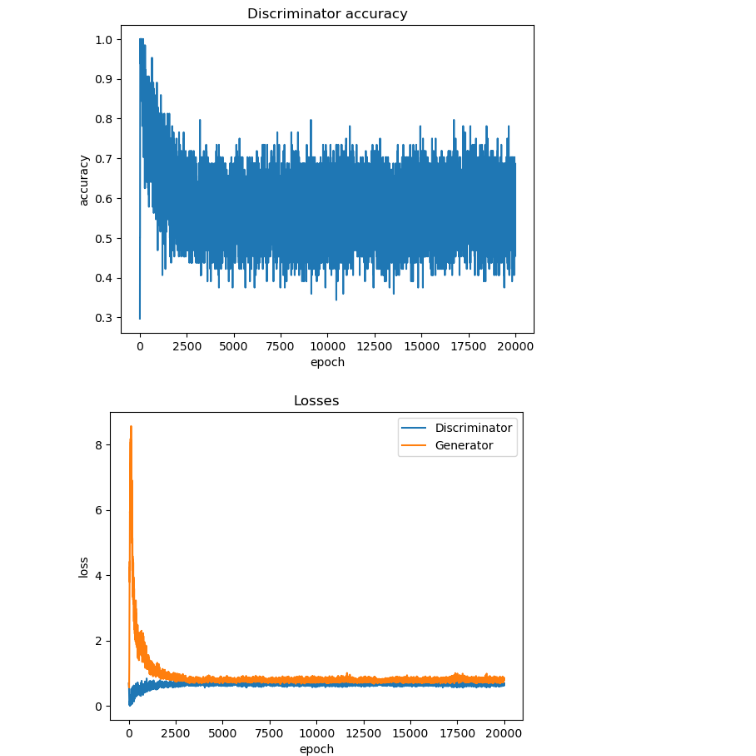
*Гиперболический тангенс очень похож на сигмоидальную функцию, которая также используется в нейронных сетях. Он принимает входные значения и преобразует их в диапазон от -1 до 1, что может использоваться для задач регрессии. Значения, близкие к -1, интерпретируются как отрицательные значения, а значения, близкие к 1, как положительные значения. Значения, близкие к нулю, обрабатываются как нейтральные.*

*По сравнению со сигмоидальной функцией, гиперболический тангенс имеет более пологую кривую, что позволяет сети лучше распознавать сложные зависимости в данных. Также гиперболический тангенс имеет гладкую производную, что позволяет использовать алгоритмы оптимизации, которые требуют вычисления градиента.*

***Модель(дискриминатор + генератор):***

*combined.compile(loss=['binary\_crossentropy'], optimizer=optimizer)*

# ***3.Графики обучения;***



# ***4. Примеры работы обученной модели:***

- изображения из ОМ с метками, поступившими на вход генератора;



- соответствующие им сгенерированные изображения.

