

▼ PROJETO 3 - VAE com Fashion MNIST

Este notebook apresenta uma das duas partes do Projeto 3, focando no uso do dataset Fashion MNIST. O objetivo é explorar e entender o funcionamento dos Modelos Autoencoders Variacionais (VAEs) e sua aplicação em datasets rotulados. As etapas realizadas são descritas a seguir:

Seleção do Dataset

- **Dataset Escolhido:** Fashion MNIST, um dataset amplamente utilizado para tarefas de aprendizado de máquina e visão computacional. Consiste em imagens de 10 categorias diferentes de roupas e acessórios, como camisetas, calças, pulôveres, entre outros.

Treinamento do Modelo VAE

- **Construção e Ajuste do Modelo:** O modelo VAE foi construído e ajustado para este dataset específico.
 - **Estrutura do Modelo:** O modelo consiste em duas partes principais, o encoder e o decoder. O encoder é responsável por mapear as imagens de entrada para um espaço latente, enquanto o decoder reconstrói as imagens a partir deste espaço.
 - **Dimensionalidade:** As imagens foram achatadas para vetores de 784 elementos (28x28 pixels) e a dimensão latente foi definida como 20.
 - **Camadas e Ativações:** Usamos camadas densas com ativação ReLU e a última camada do decoder com ativação sigmóide.

Ajuste do Melhor Modelo

- **Função de Custo e Validação:** A função de custo utilizada foi a soma do erro de reconstrução (binary crossentropy) e a divergência KL. Esta combinação ajuda a balancear a fidelidade da reconstrução com a regularização do espaço latente.
- **Treinamento e Validação:** O modelo foi treinado com o conjunto de treinamento e validado usando um conjunto de validação derivado do dataset Fashion MNIST.

Exploração do Espaço Latente

- **Projeção em 2D com PCA:** Após o treinamento, utilizamos o PCA (Principal Component Analysis) para projetar as representações latentes em um espaço 2D, facilitando a visualização e interpretação dos resultados.
- **Uso dos Rótulos na Projeção:** Cada ponto na projeção 2D foi etiquetado com o rótulo correspondente da categoria de vestuário, permitindo uma análise visual da distribuição das categorias no espaço latente.

Entendo, nesse caso, podemos revisar a análise e o bloco markdown para refletir essa informação. Aqui está a versão atualizada:

Análise Comparativa do Espaço Latente dos Modelos VAE para Fashion MNIST

Este notebook documenta uma análise comparativa detalhada do espaço latente resultante de dois modelos VAE diferentes treinados no dataset Fashion MNIST. Os gráficos a seguir representam as projeções 2D do espaço latente para o modelo inicial e o modelo otimizado.

Projeção 2D do Espaço Latente do Modelo Inicial

Verificar no notebook um pouco abaixo. É a primeira projeção.

Projeção 2D do Espaço Latente do Modelo Otimizado

Verificar no notebook um pouco abaixo. É a segunda projeção.

Formação de Clusters

- **Modelo Inicial:** Apresenta uma formação de clusters que, embora sobrepostos, indicam uma tendência de agrupamento por categorias de roupas.
- **Modelo Otimizado:** Os clusters são mais definidos e apresentam uma separação mais clara, sugerindo uma melhor codificação das características distintas das categorias no espaço latente.

Separação dos Rótulos

- **Modelo Inicial:** A separação dos rótulos é relativamente difusa, com muitas sobreposições entre as categorias.
- **Modelo Otimizado:** Observa-se uma melhoria na separação dos rótulos, com grupos mais distintos correspondendo às categorias de vestuário. Isso sugere uma representação latente mais discriminativa após a otimização.

Variância Capturada pela Projeção

- A projeção PCA em ambos os modelos ilustra uma parte significativa da variância. Contudo, o modelo otimizado parece capturar uma maior variância intraclasse, o que pode ser um indicativo de uma melhor representação latente dos dados.

Considerações Finais

As melhorias observadas no modelo otimizado, como resultado do processo de seleção e ajuste realizado com o Optuna, realçam a importância de uma sintonia fina na arquitetura do VAE. A otimização levou a um espaço latente que não só proporciona melhores visualizações, mas também pode oferecer um desempenho aprimorado em tarefas downstream, como classificação e geração de imagens.

Estes resultados sublinham o potencial de modelos VAE bem ajustados para a criação de representações latentes robustas e úteis para análises complexas de dados.

```
!pip install tensorflow optuna matplotlib scikit-learn
```

```
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (16.0.6)
Requirement already satisfied: ml-dtypes~=0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: numpy<2.0.0,>=1.23.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.23.5)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (23.2)
Requirement already satisfied: protobuf!=4.21.0,!4.21.1,!4.21.2,!4.21.3,!4.21.4,!4.21.5,<5.0.0dev,>=3.20.3 in /usr/local/lib/
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (67.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.4.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (4.5.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.60.0)
Requirement already satisfied: tensorboard<2.16,>=2.15 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.1)
Requirement already satisfied: tensorflow-estimator<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2
Requirement already satisfied: keras<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Collecting alembic>=1.5.0 (from optuna)
  Downloading alembic-1.13.1-py3-none-any.whl (233 kB)
    233.4/233.4 kB 5.5 MB/s eta 0:00:00
Collecting colorlog (from optuna)
  Downloading colorlog-6.8.0-py3-none-any.whl (11 kB)
Requirement already satisfied: sqlalchemy>=1.3.0 in /usr/local/lib/python3.10/dist-packages (from optuna) (2.0.23)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from optuna) (4.66.1)
Requirement already satisfied: PyYAML in /usr/local/lib/python3.10/dist-packages (from optuna) (6.0.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.46.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.5)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.1.1)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.2.0)
Collecting Mako (from alembic>=1.5.0->optuna)
  Downloading Mako-1.3.0-py3-none-any.whl (78 kB)
    78.6/78.6 kB 5.6 MB/s eta 0:00:00
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunparse>=1.6.0->tensorflow)
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from sqlalchemy>=1.3.0->optuna) (3.0.2)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: google-auth-oauthlib<2,>=0.5 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorflow)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorflow)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorflow)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from google-auth-oauthlib<2,>=0.5->tensorflow)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug>=1.0.1->tensorflow)
Requirement already satisfied: pyasn1<0.6.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn1-modules>=0.2.1->google-auth)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib)
Installing collected packages: Mako, colorlog, alembic, optuna
Successfully installed Mako-1.3.0 alembic-1.13.1 colorlog-6.8.0 optuna-3.5.0
```

```
import tensorflow as tf
from tensorflow.keras.datasets import fashion_mnist
from tensorflow.keras.layers import Input, Dense, Lambda, Flatten, Reshape
from tensorflow.keras.models import Model
from tensorflow.keras import backend as K
from tensorflow.keras.losses import binary_crossentropy

# Carregando o dataset Fashion-MNIST
(x_train, y_train), (x_test, y_test) = fashion_mnist.load_data()

# Normalizando os dados para o intervalo [0, 1]
x_train = x_train.astype('float32') / 255.
x_test = x_test.astype('float32') / 255.

# Achatar as imagens para vetores de 28*28 = 784 elementos
x_train = x_train.reshape((len(x_train), -1))
x_test = x_test.reshape((len(x_test), -1))

# imprimindo as dimensões para verificar se está correto
print(x_train.shape, x_test.shape)
```

```

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
29515/29515 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26421880/26421880 [=====] - 1s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
5148/5148 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4422102/4422102 [=====] - 1s 0us/step
(60000, 784) (10000, 784)

def build_vae(input_shape, latent_dim, intermediate_dims):
    # Encoder
    inputs = Input(shape=input_shape, name='encoder_input')
    x = inputs
    for intermediate_dim in intermediate_dims:
        x = Dense(intermediate_dim, activation='relu')(x)
    z_mean = Dense(latent_dim, name='z_mean')(x)
    z_log_var = Dense(latent_dim, name='z_log_var')(x)

    # Função para amostrar do espaço latente
    def sampling(args):
        z_mean, z_log_var = args
        batch = K.shape(z_mean)[0]
        dim = K.int_shape(z_mean)[1]
        epsilon = K.random_normal(shape=(batch, dim))
        return z_mean + K.exp(0.5 * z_log_var) * epsilon

    z = Lambda(sampling, output_shape=(latent_dim,), name='z')([z_mean, z_log_var])

    # Encoder model
    encoder = Model(inputs, [z_mean, z_log_var, z], name='encoder')

    # Decoder
    latent_inputs = Input(shape=(latent_dim,), name='z_sampling')
    x = latent_inputs
    for intermediate_dim in reversed(intermediate_dims):
        x = Dense(intermediate_dim, activation='relu')(x)
    outputs = Dense(input_shape[0], activation='sigmoid')(x)

    # Decoder model
    decoder = Model(latent_inputs, outputs, name='decoder')

    # VAE model
    outputs = decoder(encoder(inputs)[2])
    vae = Model(inputs, outputs, name='vae_mlp')

    # VAE loss
    reconstruction_loss = binary_crossentropy(inputs, outputs) * input_shape[0]
    kl_loss = 1 + z_log_var - K.square(z_mean) - K.exp(z_log_var)
    kl_loss = K.sum(kl_loss, axis=-1)
    kl_loss *= -0.5
    vae_loss = K.mean(reconstruction_loss + kl_loss)
    vae.add_loss(vae_loss)
    vae.compile(optimizer='adam')

    return encoder, decoder, vae

# Criando o VAE
encoder, decoder, vae = build_vae((784,), latent_dim=20, intermediate_dims=[256])

# Resumo do modelo
encoder.summary()
decoder.summary()
vae.summary()

```

tf.cast (TFOpLambda)	(None, 784)	0	['encoder_input[0][0]']
tf.convert_to_tensor (TFOpLambda)	(None, 784)	0	['decoder[0][0]']
tf.math.subtract (TFOpLambda)	(None, 20)	0	['tf.__operators__.add[0][0]', 'tf.math.square[0][0]']
tf.math.exp (TFOpLambda)	(None, 20)	0	['z_log_var[0][0]']
tf.keras.backend.binary_crossentropy (TFOpLambda)	(None, 784)	0	['tf.cast[0][0]', 'tf.convert_to_tensor[0][0]']
tf.math.subtract_1 (TFOpLambda)	(None, 20)	0	['tf.math.subtract[0][0]', 'tf.math.exp[0][0]']
tf.math.reduce_mean (TFOpLambda)	(None,)	0	['tf.keras.backend.binary_crossentropy[0][0]']
tf.math.reduce_sum (TFOpLambda)	(None,)	0	['tf.math.subtract_1[0][0]']
tf.math.multiply (TFOpLambda)	(None,)	0	['tf.math.reduce_mean[0][0]']
tf.math.multiply_1 (TFOpLambda)	(None,)	0	['tf.math.reduce_sum[0][0]']
tf.__operators__.add_1 (TFOpLambda)	(None,)	0	['tf.math.multiply[0][0]', 'tf.math.multiply_1[0][0]']
tf.math.reduce_mean_1 (TFOpLambda)	(,)	0	['tf.__operators__.add_1[0][0]']
add_loss (AddLoss)	(,)	0	['tf.math.reduce_mean_1[0][0]']

=====
Total params: 418104 (1.59 MB)
Trainable params: 418104 (1.59 MB)
Non-trainable params: 0 (0.00 Byte)

#Treinamento VAE Simples para testar

Parâmetros de treinamento

epochs = 15
batch_size = 256

Treinamento do modelo
vae.fit(x_train, x_train,
 epochs=epochs,
 batch_size=batch_size,
 validation_data=(x_test, x_test))

Epoch 1/15
235/235 [=====] - 7s 8ms/step - loss: 312.8518 - val_loss: 274.0730
Epoch 2/15
235/235 [=====] - 1s 6ms/step - loss: 265.6434 - val_loss: 262.3207
Epoch 3/15
235/235 [=====] - 1s 6ms/step - loss: 257.7445 - val_loss: 257.8029
Epoch 4/15
235/235 [=====] - 1s 6ms/step - loss: 253.8148 - val_loss: 253.6622
Epoch 5/15
235/235 [=====] - 1s 6ms/step - loss: 251.4539 - val_loss: 252.1136
Epoch 6/15
235/235 [=====] - 1s 6ms/step - loss: 249.8911 - val_loss: 250.3888
Epoch 7/15
235/235 [=====] - 2s 7ms/step - loss: 248.5707 - val_loss: 249.9203
Epoch 8/15
235/235 [=====] - 2s 8ms/step - loss: 247.5230 - val_loss: 248.9756
Epoch 9/15
235/235 [=====] - 2s 7ms/step - loss: 246.7254 - val_loss: 247.6066
Epoch 10/15
235/235 [=====] - 1s 6ms/step - loss: 245.9844 - val_loss: 247.2307
Epoch 11/15
235/235 [=====] - 1s 6ms/step - loss: 245.5556 - val_loss: 246.8256
Epoch 12/15
235/235 [=====] - 1s 6ms/step - loss: 245.0308 - val_loss: 246.4045
Epoch 13/15
235/235 [=====] - 1s 6ms/step - loss: 244.6665 - val_loss: 246.0214
Epoch 14/15
235/235 [=====] - 1s 6ms/step - loss: 244.3191 - val_loss: 245.6206
Epoch 15/15
235/235 [=====] - 1s 6ms/step - loss: 243.9880 - val_loss: 245.7212
<keras.src.callbacks.History at 0x795f78d7a4d0>

```
import numpy as np
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Definir o tamanho do batch para previsão
batch_size = 128

# Obtendo as representações latentes
z_mean, _, _ = encoder.predict(x_test, batch_size=batch_size)

# Usando PCA para reduzir a dimensionalidade para 2D
pca = PCA(n_components=2)
z_pca = pca.fit_transform(z_mean)

# Plotando a projeção do espaço latente
plt.figure(figsize=(12, 10))

# Mapeamento de rótulos numéricos para os nomes das classes
labels_map = {
    0: 'T-shirt/top',
    1: 'Trouser',
    2: 'Pullover',
    3: 'Dress',
    4: 'Coat',
    5: 'Sandal',
    6: 'Shirt',
    7: 'Sneaker',
    8: 'Bag',
    9: 'Ankle boot'
}

# Convertendo os rótulos numéricos em nomes de classes
label_names = np.vectorize(labels_map.get)(y_test)

scatter = plt.scatter(z_pca[:, 0], z_pca[:, 1], c=y_test, alpha=0.5, cmap='viridis')
plt.colorbar(scatter)
plt.xlabel('Componente Principal 1')
plt.ylabel('Componente Principal 2')
plt.title('Projeção 2D do Espaço Latente do Fashion-MNIST')

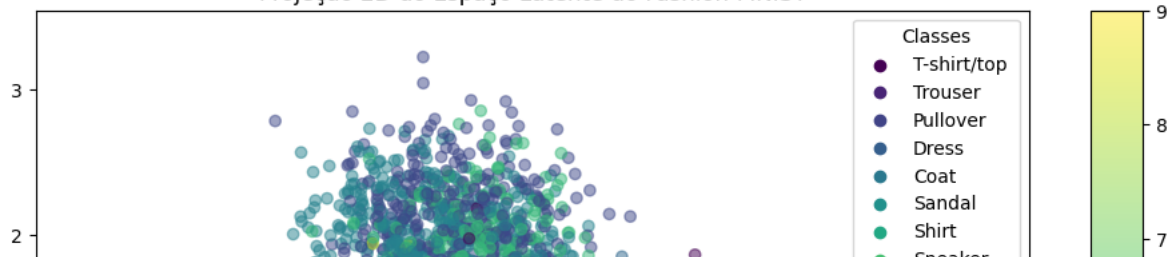
# Adicionando legendas com os nomes das classes
for i, label in labels_map.items():
    plt.scatter([], [], label=label, color=plt.cm.viridis(i / 10))

plt.legend(title='Classes')
plt.show()
```



79/79 [=====] - 0s 3ms/step

Projeção 2D do Espaço Latente do Fashion-MNIST



```
import optuna
from optuna.visualization import plot_optimization_history, plot_parallel_coordinate
from tqdm.auto import tqdm

# Definindo a função de objetivo do VAE para o Optuna
def vae_objective(trial):
    # Hiperparâmetros do Optuna
    num_layers = trial.suggest_int('num_layers', 1, 4)
    latent_dim = trial.suggest_int('latent_dim', 1, 20)
    intermediate_dims = [trial.suggest_int(f'intermediate_dim_{i}', 128, 1024, log=True) for i in range(num_layers)]
    batch_size = trial.suggest_categorical('batch_size', [64, 128, 256, 512])
    epochs = 30 # Pode ser fixo ou também otimizado pelo Optuna

    # Construção do modelo VAE com os hiperparâmetros sugeridos
    encoder, decoder, vae = build_vae((784,), latent_dim=latent_dim, intermediate_dims=intermediate_dims)

    # Treinamento do modelo
    history = vae.fit(x_train, x_train,
                     epochs=epochs,
                     batch_size=batch_size,
                     validation_data=(x_test, x_test),
                     verbose=0) # verbose=0 para não imprimir o progresso

    # Retorna a última perda de validação
    val_loss = history.history['val_loss'][-1]
    return val_loss

# Definindo um nome para o estudo e habilitando o progress bar
study_name = 'mnist_vae_optimization' # Nome do estudo
study = optuna.create_study(direction='minimize', study_name=study_name)

# Callback para acompanhar o progresso
def print_progress(study, trial):
    last_value = trial.value if trial.value is not None else float('nan')
    print(f"Trial {trial.number}: val_loss = {last_value:.3f}, params = {trial.params}")

# Função de progresso personalizada
def optuna_progress_bar(study, trial):
    bar.update(1)

# Inicializando o progress bar
bar = tqdm(total=30, desc=study_name)

# Otimizando com progress bar e print callback
study.optimize(vae_objective, n_trials=30, callbacks=[print_progress, optuna_progress_bar])

# Fechando o progress bar
bar.close()

# Mostrar os melhores hiperparâmetros
print(f"Melhores hiperparâmetros: {study.best_params}")
```

```
[I 2023-12-21 01:16:17,606] A new study created in memory with name: mnist_vae_optimization
mnist_vae_optimization: 100% 30/30 [46:34<00:00, 87.06s/it]

[I 2023-12-21 01:17:41,790] Trial 0 finished with value: 239.48912048339844 and parameters: {'num_layers': 2, 'latent_dim': 7, 'intermediate_dim_0': 324, 'intermediate_dim_1': 310, 'batch_size': 256}
[I 2023-12-21 01:20:48,842] Trial 1 finished with value: 256.6214904785156 and parameters: {'num_layers': 3, 'latent_dim': 2, 'intermediate_dim_0': 170, 'intermediate_dim_1': 299, 'batch_size': 256}
[I 2023-12-21 01:22:09,431] Trial 2 finished with value: 240.9254150390625 and parameters: {'num_layers': 2, 'latent_dim': 6, 'intermediate_dim_0': 235, 'intermediate_dim_1': 175, 'batch_size': 256}
[I 2023-12-21 01:22:52,417] Trial 3 finished with value: 240.51364135742188 and parameters: {'num_layers': 1, 'latent_dim': 17, 'intermediate_dim_0': 988, 'batch_size': 256}
[I 2023-12-21 01:26:16,578] Trial 4 finished with value: 243.84519958496094 and parameters: {'num_layers': 2, 'latent_dim': 4, 'intermediate_dim_0': 281, 'intermediate_dim_1': 182, 'batch_size': 256}
[I 2023-12-21 01:27:41,280] Trial 5 finished with value: 243.2742156982422 and parameters: {'num_layers': 4, 'latent_dim': 7, 'intermediate_dim_0': 976, 'intermediate_dim_1': 138, 'batch_size': 256}
[I 2023-12-21 01:29:06,282] Trial 6 finished with value: 240.45726013183594 and parameters: {'num_layers': 2, 'latent_dim': 8, 'intermediate_dim_0': 261, 'intermediate_dim_1': 131, 'batch_size': 256}
[I 2023-12-21 01:30:30,502] Trial 7 finished with value: 241.04029846191406 and parameters: {'num_layers': 1, 'latent_dim': 8, 'intermediate_dim_0': 230, 'batch_size': 128}
[I 2023-12-21 01:32:54,950] Trial 8 finished with value: 239.59132385253906 and parameters: {'num_layers': 3, 'latent_dim': 9, 'intermediate_dim_0': 563, 'intermediate_dim_1': 143, 'batch_size': 256}
[I 2023-12-21 01:35:04,406] Trial 9 finished with value: 263.1184997558594 and parameters: {'num_layers': 1, 'latent_dim': 2, 'intermediate_dim_0': 230, 'batch_size': 64}
[I 2023-12-21 01:35:48,214] Trial 10 finished with value: 242.18435668945312 and parameters: {'num_layers': 4, 'latent_dim': 14, 'intermediate_dim_0': 444, 'intermediate_dim_1': 908, 'batch_size': 256}
[I 2023-12-21 01:37:21,410] Trial 11 finished with value: 239.02593994140625 and parameters: {'num_layers': 3, 'latent_dim': 12, 'intermediate_dim_0': 455, 'intermediate_dim_1': 389, 'batch_size': 256}
[I 2023-12-21 01:38:55,508] Trial 12 finished with value: 238.87364196777344 and parameters: {'num_layers': 3, 'latent_dim': 13, 'intermediate_dim_0': 409, 'intermediate_dim_1': 439, 'batch_size': 256}
[I 2023-12-21 01:39:24,700] Trial 13 finished with value: 241.0811309814453 and parameters: {'num_layers': 3, 'latent_dim': 13, 'intermediate_dim_0': 431, 'intermediate_dim_1': 580, 'batch_size': 256}
[I 2023-12-21 01:41:49,384] Trial 14 finished with value: 242.76988220214844 and parameters: {'num_layers': 4, 'latent_dim': 19, 'intermediate_dim_0': 581, 'intermediate_dim_1': 507, 'batch_size': 256}
[I 2023-12-21 01:43:24,141] Trial 15 finished with value: 240.1028289794922 and parameters: {'num_layers': 3, 'latent_dim': 12, 'intermediate_dim_0': 419, 'intermediate_dim_1': 454, 'batch_size': 256}
[I 2023-12-21 01:44:54,729] Trial 16 finished with value: 241.74441528320312 and parameters: {'num_layers': 3, 'latent_dim': 16, 'intermediate_dim_0': 128, 'intermediate_dim_1': 706, 'batch_size': 256}
[I 2023-12-21 01:45:27,186] Trial 17 finished with value: 242.21322631835938 and parameters: {'num_layers': 4, 'latent_dim': 11, 'intermediate_dim_0': 577, 'intermediate_dim_1': 372, 'batch_size': 256}
[I 2023-12-21 01:46:18,643] Trial 18 finished with value: 240.15188598632812 and parameters: {'num_layers': 3, 'latent_dim': 15, 'intermediate_dim_0': 364, 'intermediate_dim_1': 245, 'batch_size': 256}
[I 2023-12-21 01:48:43,465] Trial 19 finished with value: 238.20431518554688 and parameters: {'num_layers': 2, 'latent_dim': 19, 'intermediate_dim_0': 500, 'intermediate_dim_1': 379, 'batch_size': 256}
[I 2023-12-21 01:50:05,667] Trial 20 finished with value: 238.21412658691406 and parameters: {'num_layers': 2, 'latent_dim': 20, 'intermediate_dim_0': 710, 'intermediate_dim_1': 695, 'batch_size': 256}
[I 2023-12-21 01:51:30,480] Trial 21 finished with value: 238.54417419433594 and parameters: {'num_layers': 2, 'latent_dim': 20, 'intermediate_dim_0': 708, 'intermediate_dim_1': 691, 'batch_size': 256}
[I 2023-12-21 01:52:55,605] Trial 22 finished with value: 238.43634033203125 and parameters: {'num_layers': 2, 'latent_dim': 20, 'intermediate_dim_0': 719, 'intermediate_dim_1': 920, 'batch_size': 256}
[I 2023-12-21 01:54:21,389] Trial 23 finished with value: 238.1331024169922 and parameters: {'num_layers': 2, 'latent_dim': 18, 'intermediate_dim_0': 719, 'intermediate_dim_1': 920, 'batch_size': 256}

!pip install plotly

import plotly
print(plotly.__version__)

Requirement already satisfied: plotly in /usr/local/lib/python3.10/dist-packages (5.15.0)
Requirement already satisfied: tenacity>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from plotly) (8.2.3)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from plotly) (23.2)
5.15.0

#Usando o modo offline do plotly, ja que o outro nao funcionou.

from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)

!pip install "notebook>=5.3" "ipywidgets>=7.2"
!jupyter nbextension enable --py widgetsnbextension

Requirement already satisfied: notebook>=5.3 in /usr/local/lib/python3.10/dist-packages (6.5.5)
Requirement already satisfied: ipywidgets>=7.2 in /usr/local/lib/python3.10/dist-packages (7.7.1)
Requirement already satisfied: Jinja2 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (3.1.2)
Requirement already satisfied: tornado>=6.1 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (6.3.2)
Requirement already satisfied: pyzmq<25,>=17 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (23.2.1)
Requirement already satisfied: argon2-cffi in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (23.1.0)
Requirement already satisfied: traitlets>=4.2.1 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (5.7.1)
Requirement already satisfied: jupyter-core>=4.6.1 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (5.5.1)
Requirement already satisfied: jupyter-client<8,>=5.3.4 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (6.1.12)
Requirement already satisfied: ipython-genutils in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (0.2.0)
Requirement already satisfied: nbformat in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (5.9.2)
Requirement already satisfied: nbconvert>=5 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (6.5.4)
Requirement already satisfied: nest-asyncio>=1.5 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (1.5.8)
Requirement already satisfied: ipykernel in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (5.5.6)
```

```
Requirement already satisfied: Send2Trash>=1.8.0 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (1.8.2)
Requirement already satisfied: terminado>=0.8.3 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (0.18.0)
Requirement already satisfied: prometheus-client in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (0.19.0)
Requirement already satisfied: nbclassic>=0.4.7 in /usr/local/lib/python3.10/dist-packages (from notebook>=5.3) (1.0.0)
Requirement already satisfied: widgetsnbextension~=3.6.0 in /usr/local/lib/python3.10/dist-packages (from ipywidgets>=7.2) (3.6.6)
Requirement already satisfied: ipython>=4.0.0 in /usr/local/lib/python3.10/dist-packages (from ipywidgets>=7.2) (7.34.0)
Requirement already satisfied: jupyterlab-widgets>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from ipywidgets>=7.2) (3.0.9)
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2)
Collecting jedi>=0.16 (from ipython>=4.0.0->ipywidgets>=7.2)
  Downloading jedi-0.19.1-py2.py3-none-any.whl (1.6 MB)
```

1.6/1.6 MB 11.8 MB/s eta 0:00:00

```
Requirement already satisfied: decorator in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2) (4.4.2)
Requirement already satisfied: pickleshare in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2) (0.7.5)
Requirement already satisfied: prompt-toolkit!=3.0.0,!<3.0.1,<3.1.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from ipyth
Requirement already satisfied: pygments in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2) (2.16.1)
Requirement already satisfied: backcall in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2) (0.2.0)
Requirement already satisfied: matplotlib-inline in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2)
Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets>=7.2) (4.9
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.10/dist-packages (from jupyter-client<8,>=5.3.4->not
Requirement already satisfied: platformdirs>=2.5 in /usr/local/lib/python3.10/dist-packages (from jupyter-core>=4.6.1->notebook>=5
Requirement already satisfied: jupyter-server>=1.8 in /usr/local/lib/python3.10/dist-packages (from nbclassic>=0.4.7->notebook>=5
Requirement already satisfied: notebook-shim>=0.2.3 in /usr/local/lib/python3.10/dist-packages (from nbclassic>=0.4.7->notebook>=5
Requirement already satisfied: lxml in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (4.9.3)
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (4.11
Requirement already satisfied: bleach in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (6.1.0)
Requirement already satisfied: defusedxml in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (0.7.1)
Requirement already satisfied: entrypoints>=0.2.2 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (0
Requirement already satisfied: jupyterlab-pygments in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (2.1
Requirement already satisfied: mistune<2,>=0.8.1 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (0
Requirement already satisfied: nbclient>=0.5.0 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (0.9
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (23.2)
Requirement already satisfied: pandocfilters>=1.4.1 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3)
Requirement already satisfied: tinycss2 in /usr/local/lib/python3.10/dist-packages (from nbconvert>=5->notebook>=5.3) (1.2.1)
Requirement already satisfied: fastjsonschema in /usr/local/lib/python3.10/dist-packages (from nbformat->notebook>=5.3) (2.19.0)
Requirement already satisfied: jsonschema>=2.6 in /usr/local/lib/python3.10/dist-packages (from nbformat->notebook>=5.3) (4.19.2)
Requirement already satisfied: ptyprocess in /usr/local/lib/python3.10/dist-packages (from terminado>=0.8.3->notebook>=5.3) (0.7.0
Requirement already satisfied: argon2-cffi-bindings in /usr/local/lib/python3.10/dist-packages (from argon2-cffi->notebook>=5.3) (
Requirement already satisfied: parso<0.9.0,>=0.8.3 in /usr/local/lib/python3.10/dist-packages (from jedi>=0.16->ipython>=4.0.0->ip
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=2.6->nbformat->notebook
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=2
Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=2.6->nbformat->not
Requirement already satisfied: rfc3339 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=2.6->nbformat->notebook
```

```
from optuna.visualization import plot_optimization_history, plot_parallel_coordinate
import plotly
```

```
# Plotando o histórico da otimização
fig1 = plot_optimization_history(study)
fig1.show()
```

```
# Plotando coordenadas paralelas dos hiperparâmetros
fig2 = plot_parallel_coordinate(study)
fig2.show()
```



```
#Salvar estudo Optune
import joblib

joblib.dump(study, 'study.pkl')

['study.pkl']

import numpy as np
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score

# 1. Construir o modelo VAE com os melhores hiperparâmetros
best_params = study.best_params

# Extraindo o número de camadas e as dimensões intermediárias
num_layers = best_params['num_layers']
intermediate_dims = [best_params[f'intermediate_dim_{i}'] for i in range(num_layers)]

encoder, decoder, vae = build_vae(
    input_shape=(784,),
    latent_dim=best_params['latent_dim'],
    intermediate_dims=intermediate_dims
)

# 2. Treinar modelo com o conjunto de dados completo
vae.fit(x_train, x_train, epochs=15, batch_size=best_params['batch_size'])

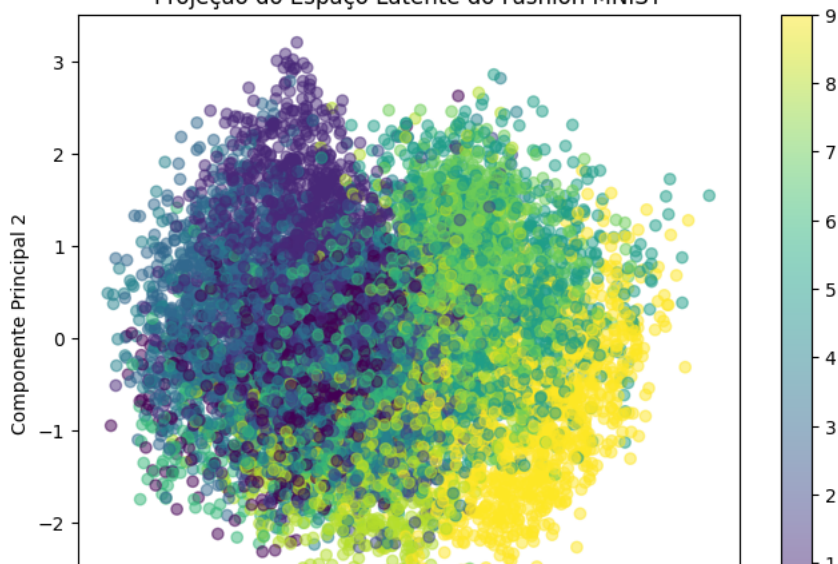
# 3. Projeção do espaço latente em 2D
z_mean, _, _ = encoder.predict(x_test, batch_size=best_params['batch_size'])
pca = PCA(n_components=2)
z_pca = pca.fit_transform(z_mean)
plt.figure(figsize=(8, 6))
plt.scatter(z_pca[:, 0], z_pca[:, 1], c=y_test, alpha=0.5, cmap='viridis')
plt.colorbar()
plt.xlabel('Componente Principal 1')
plt.ylabel('Componente Principal 2')
plt.title('Projeção do Espaço Latente do Fashion MNIST')
plt.show()
```

```

Epoch 2/15
469/469 [=====] - 7s 14ms/step - loss: 250.1254
Epoch 3/15
469/469 [=====] - 3s 6ms/step - loss: 246.3312
Epoch 4/15
469/469 [=====] - 3s 6ms/step - loss: 244.3682
Epoch 5/15
469/469 [=====] - 3s 6ms/step - loss: 242.9420
Epoch 6/15
469/469 [=====] - 4s 8ms/step - loss: 241.8700
Epoch 7/15
469/469 [=====] - 3s 6ms/step - loss: 241.0913
Epoch 8/15
469/469 [=====] - 3s 6ms/step - loss: 240.3932
Epoch 9/15
469/469 [=====] - 3s 6ms/step - loss: 239.7835
Epoch 10/15
469/469 [=====] - 4s 8ms/step - loss: 239.3119
Epoch 11/15
469/469 [=====] - 3s 7ms/step - loss: 239.0422
Epoch 12/15
469/469 [=====] - 3s 6ms/step - loss: 238.6888
Epoch 13/15
469/469 [=====] - 3s 6ms/step - loss: 238.3015
Epoch 14/15
469/469 [=====] - 3s 7ms/step - loss: 238.0875
Epoch 15/15
469/469 [=====] - 4s 8ms/step - loss: 237.8685
79/79 [=====] - 0s 2ms/step

```

Projeção do Espaço Latente do Fashion MNIST



```

n = 10 # número de imagens para gerar
digit_size = 28
figure = np.zeros((digit_size * n, digit_size * n))

#latent_dim = best_params['latent_dim']
latent_dim = 18

# Amostra aleatória de pontos no espaço latente
# Assegurando que cada ponto z_sample tenha a dimensão correta
grid_x = np.linspace(-15, 15, n)
grid_y = np.linspace(-15, 15, n)

for i, yi in enumerate(grid_x):
    for j, xi in enumerate(grid_y):
        # Aqui nós criamos um vetor latente com a dimensão correta
        z_sample = np.random.normal(size=(1, latent_dim))
        x_decoded = decoder.predict(z_sample)
        digit = x_decoded[0].reshape(digit_size, digit_size)
        figure[i * digit_size: (i + 1) * digit_size,
              j * digit_size: (j + 1) * digit_size] = digit

plt.figure(figsize=(10, 10))
plt.imshow(figure, cmap='Greys_r')
plt.grid(False)
plt.show()

```

```
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 45ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
```

```
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 16ms/step
```

