



# Variational Autoencoders GUI

Christos Kormaris

# How to Run the GUI



First, browse to the directory "**vaes\_gui**" of the thesis project and install the Python dependency libraries, by typing:

**pip install -r dependencies.txt**

Open a console (terminal) in Unix/Linux or a command prompt in Windows with Python 3 installed and run:

**python gui.py**

## How to Run the GUI (part 2)

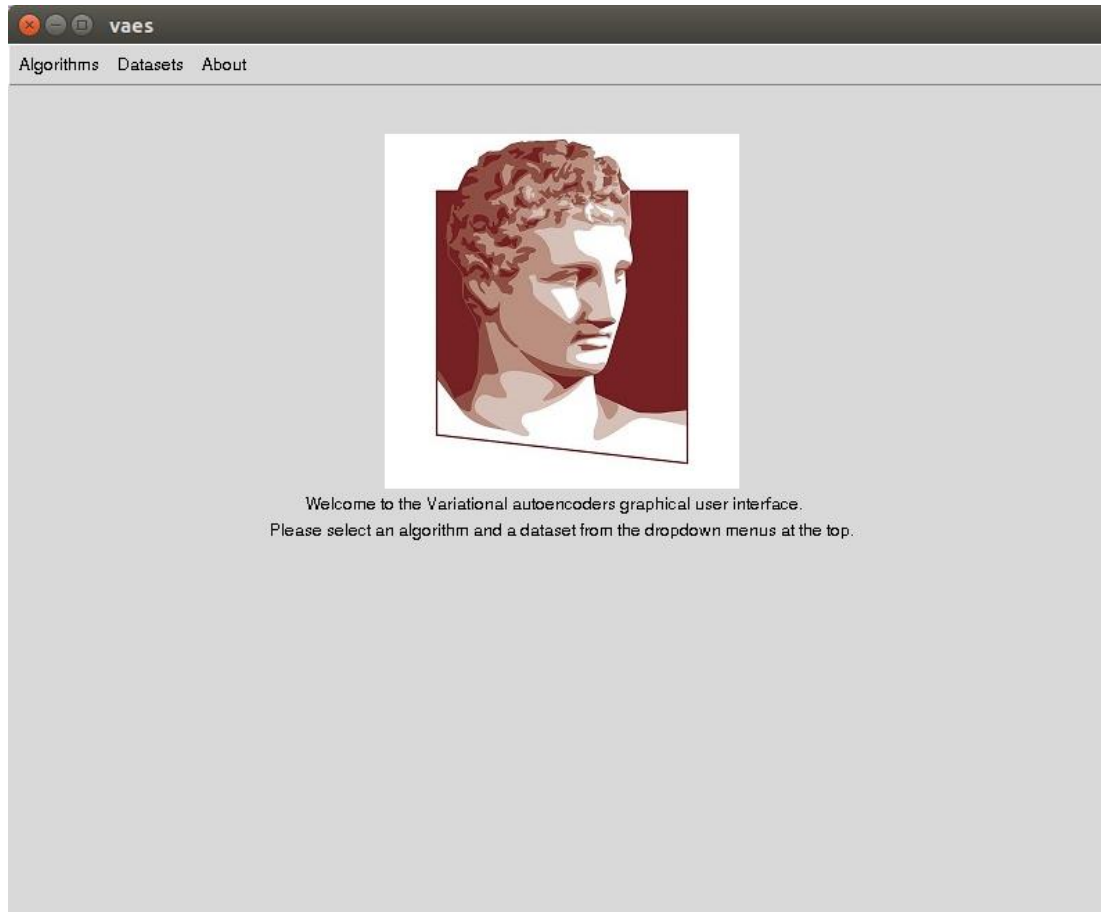


To create an executable file for the GUI (".exe"), which you can run anytime from a Windows environment, type:

```
pip install pyinstaller  
pyinstaller gui.spec
```

Then, download all the datasets from the URLs in the file "**datasets\_urls.txt**" and move them to the newly created "**dist**" folder. Inside, there should be a folder with the name "**vaes\_gui**", which contains the executable file "**vaes\_gui.exe**".

# GUI Welcome page.



# GUI Algorithms and Datasets dropdown menus.



VAE in TensorFlow

VAE in PyTorch

VAE in Keras

VAE Missing Values in TensorFlow

VAE Missing Values in PyTorch

K-NN Missing Values

**GUI Algorithms dropdown menu.**

MNIST

Binarized MNIST

CIFAR-10

OMNIGLOT

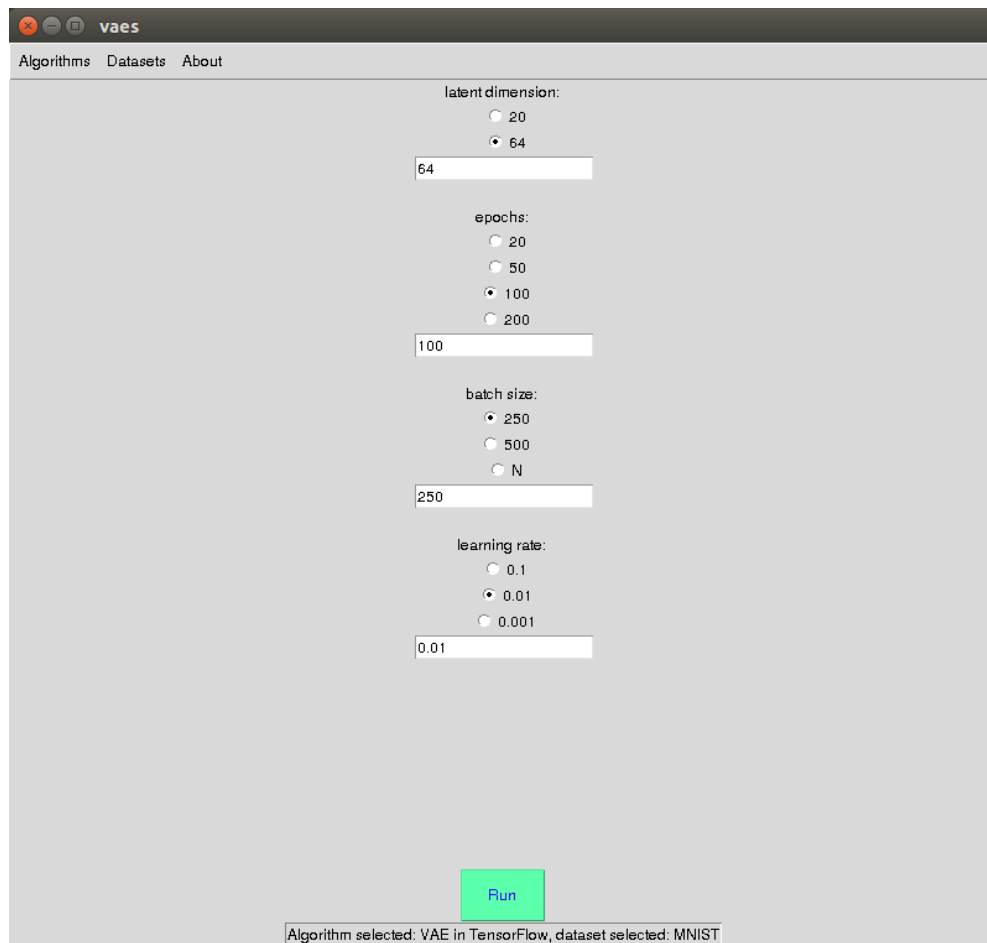
YALE Faces

The Database of Faces

MovieLens

**GUI Datasets dropdown menu.**

# GUI VAE in TensorFlow, MNIST dataset.




The image shows a graphical user interface (GUI) for training a Variational Autoencoder (VAE) in TensorFlow on the MNIST dataset. The window is titled "vae" and has a menu bar with "Algorithms", "Datasets", and "About".

The main configuration area includes the following settings:

- latent dimension:** Radio buttons for 20, 64 (selected), and a text input field containing 64.
- epochs:** Radio buttons for 20, 50, 100 (selected), and 200, with a text input field containing 100.
- batch size:** Radio buttons for 250 (selected), 500, and N, with a text input field containing 250.
- learning rate:** Radio buttons for 0.1, 0.01 (selected), and 0.001, with a text input field containing 0.01.

A green "Run" button is located at the bottom center. At the bottom of the window, a status bar displays: "Algorithm selected: VAE in TensorFlow, dataset selected: MNIST".

# GUI VAE in TensorFlow, CIFAR-10 dataset.



vaes

Algorithms Datasets About

latent dimension:

☐ 20

☒ 64

64

epochs:

☐ 20

☐ 50

☒ 100

☐ 200

100

batch size:

☒ 250

☐ 500

☐ N

250

learning rate:

☐ 0.1

☒ 0.01

☐ 0.001

0.01

grayscale or RGB:

☒ grayscale

☐ RGB

Run

Algorithm selected: VAE in TensorFlow, dataset selected: CIFAR-10

# GUI VAE in TensorFlow, OMNIGLOT dataset.



vaes

Algorithms Datasets About

latent dimension:

☐ 20

☒ 64

64

epochs:

☐ 20

☐ 50

☒ 100

☐ 200

100

batch size:

☒ 250

☐ 500

☐ N

250

learning rate:

☐ 0.1

☒ 0.01

☐ 0.001

0.01

language:

☒ English

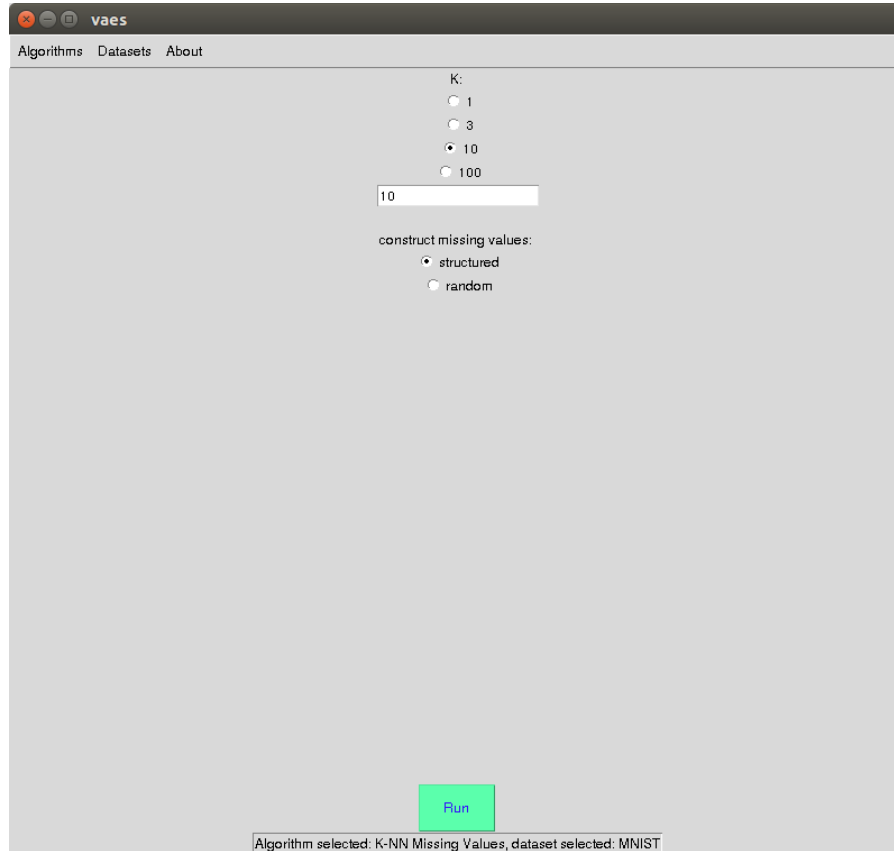
☐ Greek

Run

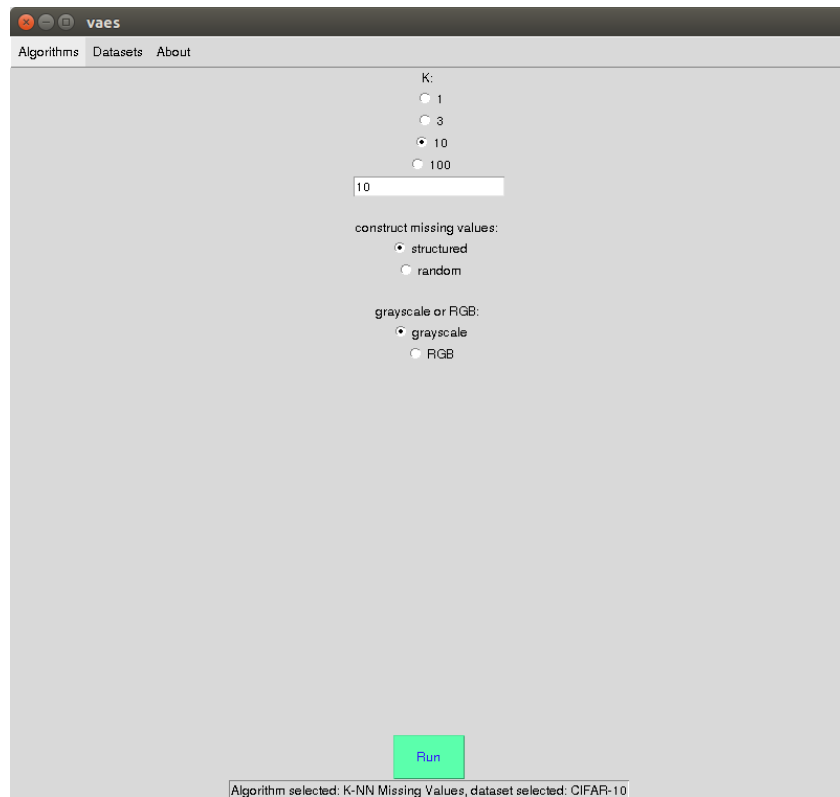
Algorithm selected: VAE in TensorFlow, dataset selected: OMNIGLOT



# GUI K-NN Recommendation System, MNIST dataset.



# GUI K-NN Recommendation System, CIFAR-10 dataset.



# GUI K-NN Recommendation System, OMNIGLOT dataset.

The image shows a screenshot of a graphical user interface (GUI) titled 'vaes'. The interface has a dark grey header bar with the title 'vaes' and three window control buttons (minimize, maximize, close). Below the header is a light grey navigation bar with three tabs: 'Algorithms', 'Datasets', and 'About'. The main content area is a light grey rectangle. In the center, there are several settings: 'K:' with four radio buttons labeled '1', '3', '10', and '100', where '10' is selected; a text input field containing the number '10'; 'construct missing values:' with two radio buttons labeled 'structured' and 'random', where 'structured' is selected; and 'language:' with two radio buttons labeled 'English' and 'Greek', where 'English' is selected. At the bottom center, there is a green rectangular button labeled 'Run'. At the very bottom, a status bar displays the text 'Algorithm selected: K-NN Missing Values, dataset selected: OMNIGLOT'.

# GUI datasets details.

Datasets Details	
MNIST dataset	
# TRAIN data: 55000, # TEST data: 10000, # VALIDATION data: 5000	
# Classes: 10	
Dimension: 28x28 pixels	
<a href="#">MNIST dataset link</a>	
Binarized MNIST dataset	
# TRAIN data: 50000, # TEST data: 10000, # VALIDATION data: 10000	
# Classes: 10	
Dimension: 28x28 pixels	
<a href="#">Binarized MNIST dataset link</a>	
CIFAR-10 dataset	
# TRAIN data: 50000, # TEST data: 10000	
# Classes: 10	
RGB Dimension: 32x32x3 pixels,	
Grayscale Dimension: 32x32x1 pixels	
<a href="#">CIFAR-10 and CIFAR-100 datasets link</a>	
OMNIGLOT dataset	
English Alphabet	# TRAIN data: 390, # TEST data: 130, # Classes: 26
Greek Alphabet	# TRAIN data: 360, # TEST data: 120, # Classes: 24
Dimension: 28x28 pixels	
<a href="#">OMNIGLOT dataset link</a>	
YALE Faces dataset	
# of data: 2442	
# Classes: 38	
Dimension: 168x192 pixels	
<a href="#">YALE Faces dataset link</a>	
ORL Face Database	
# of data: 400	
# Classes: 40	
Dimension: 92x112 pixels	
<a href="#">ORL Face Database link</a>	
MovieLens 100k dataset	
# TRAIN ratings: 90570	# TEST ratings: 9430
# of users: 943	# of movies: 1682
# of total ratings: 1586126 non-missing values percentage: 5.7 %	
<a href="#">MovieLens dataset link</a>	

[Download all datasets here](#)

# GUI About.

