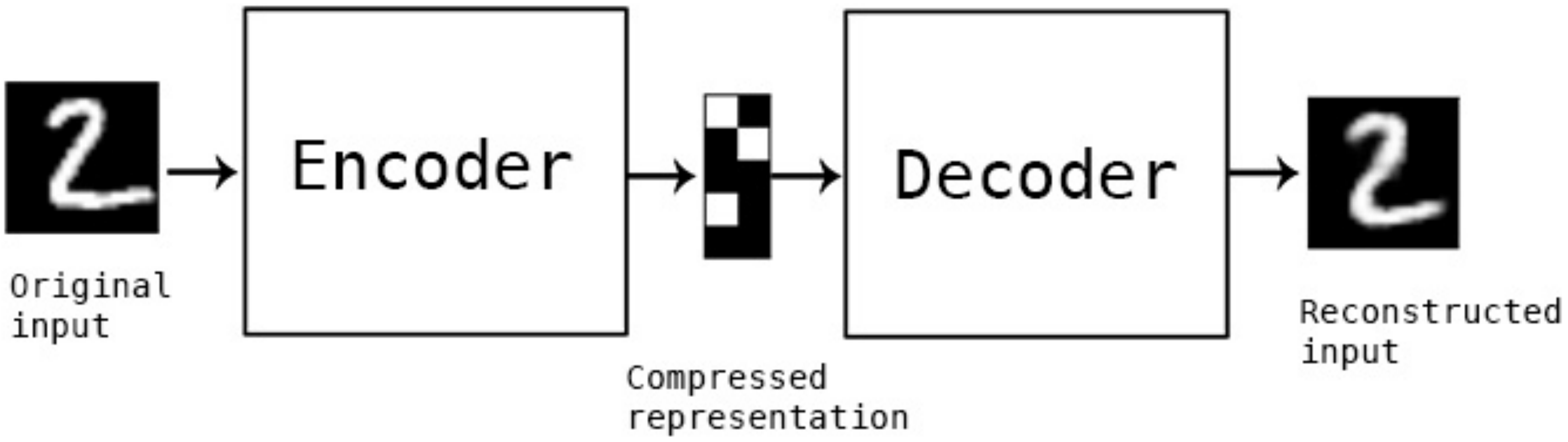
STAT430: Machine Learning for Financial Data

LarryHua.com/teaching
Spring 2019

Variational autoencoders

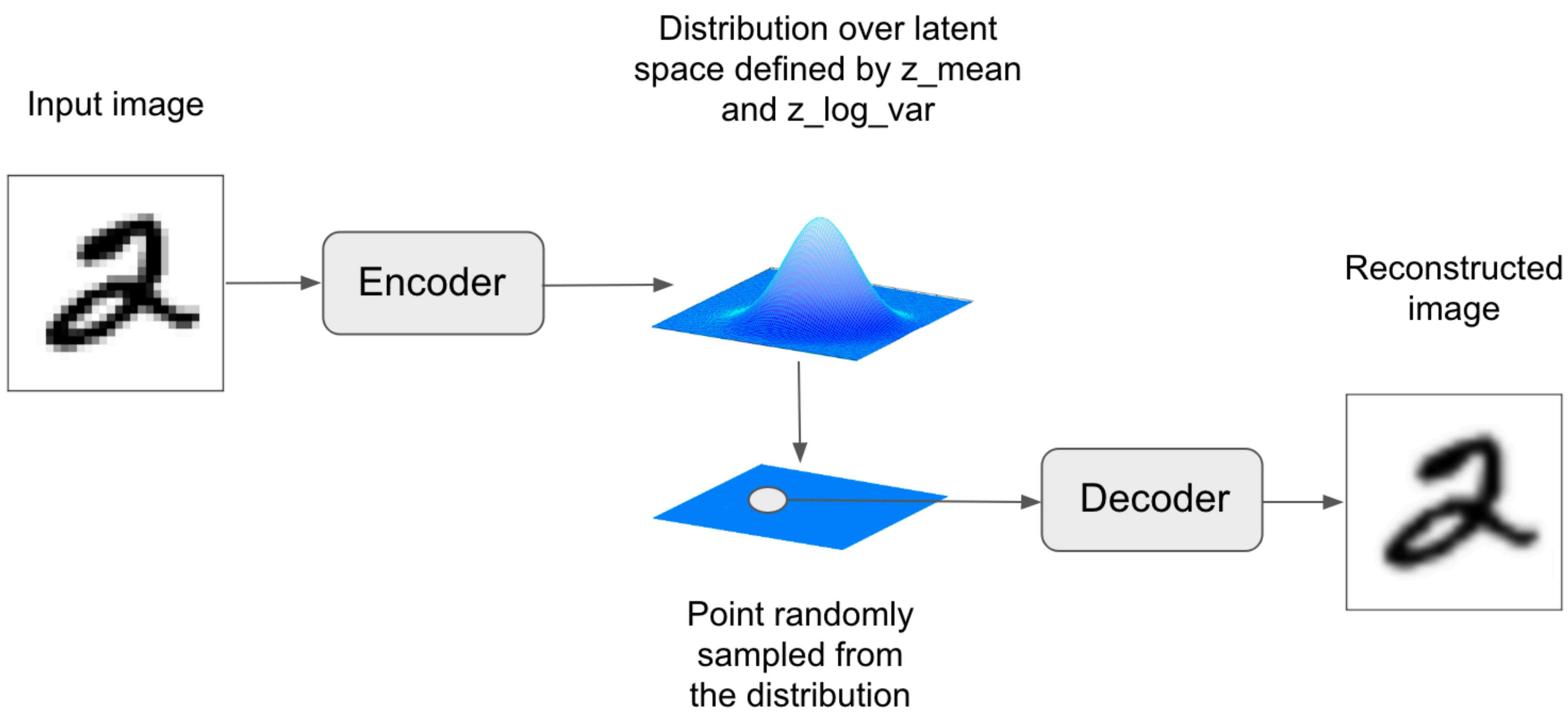
Autoencoders

- encoder: a mapping that transforms an input to a compressed representation
- decoder: a module that takes as input a latent point and outputs an image (a grid of pixels)



Variational autoencoders (VAE)

- instead of compressing its input into a fixed code in the latent space, turns the input into the parameters of a statistical distribution
- VAEs are great for learning latent spaces that are nicely structured, where specific directions encode a meaningful axis of variation in the data



Variational autoencoders (VAE)

- Pseudo codes for VAE

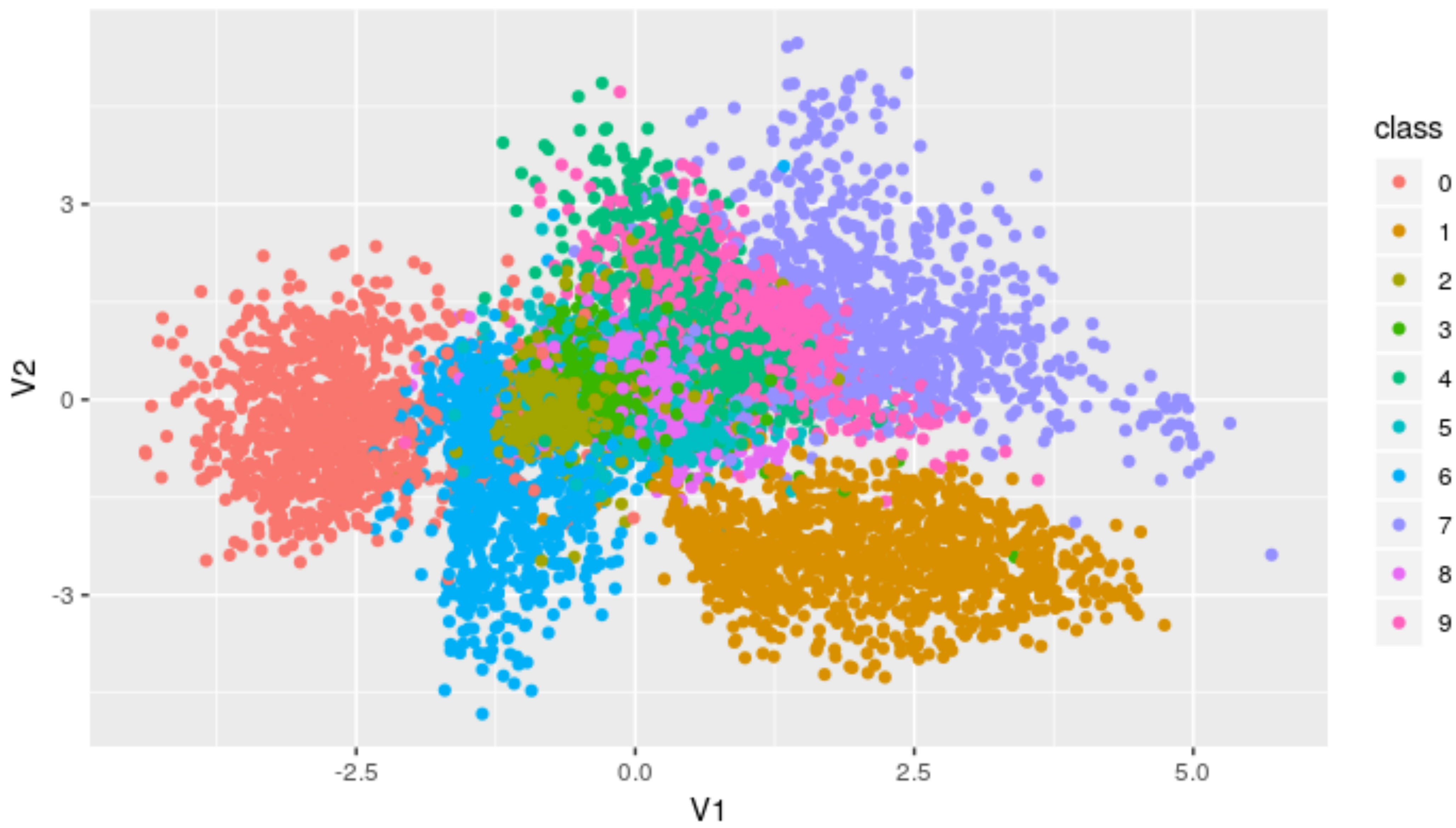
```
# Encode the input into a mean and variance parameter
c(z_mean, z_log_variance) %<% encoder(input_img)

# Draws a latent point using a small random epsilon
z <- z_mean + exp(z_log_variance) * epsilon

# Decodes z back to an image
reconstructed_img <- decoder(z)

# Creates a model
model <- keras_model(input_img, reconstructed_img)

# Then train the model using 2 losses:
# a reconstruction loss and a regularization loss
```

- reconstruction loss: forces the decoded samples to match the initial inputs
- regularization loss: helps learn well-formed latent spaces and reduce overfitting to the training data

Variational autoencoders (VAE)

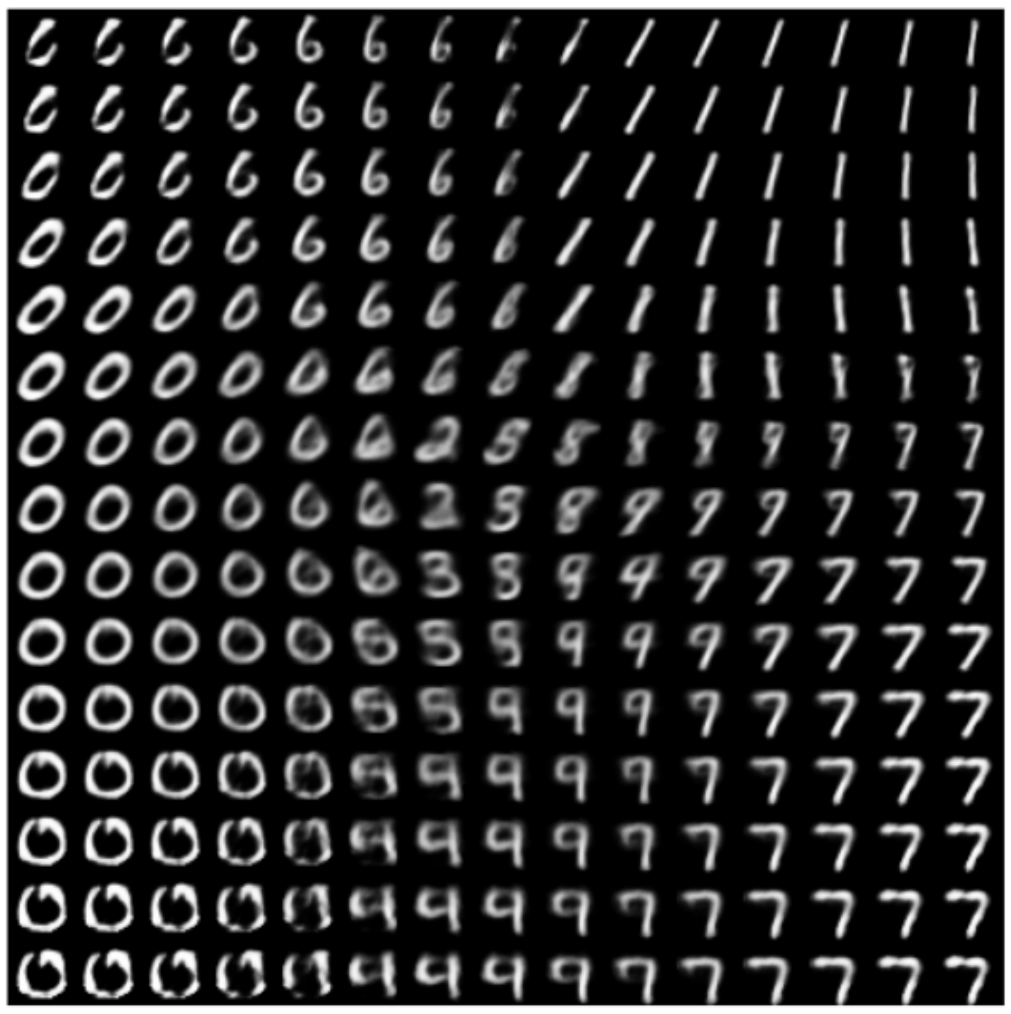
- the latent space is continuously meaningful, and any two close points in the latent space will decode to highly similar images
- continuity, combined with the low dimensionality of the latent space, forces every direction in the latent space to encode a meaningful axis of variation of the data
  - the latent space is forced to be very structured
  - highly suitable to manipulate via concept vectors (eg, smile vector)

VAE - MNIST example



- [Try R]

VAE - MNIST example



- Back to Course Scheduler