Advanced Machine Learning:
Deep Generative Models

Introduction

Lecturer: Prof. Dr. Stephan Günnemann

www.daml.in.tum.de

Summer Term 2023

Data Analytics and
Machine Learning

Roadmap

- Introduction
  1. **What will you learn in this lecture?**
  2. Organizational aspects + project tasks

What is this course about?

- In short: A continuation of our intro ML lecture (IN2064) now focusing on advanced learning principles and deep generative models

- Focus on algorithms and general principles, not limited to a single domain

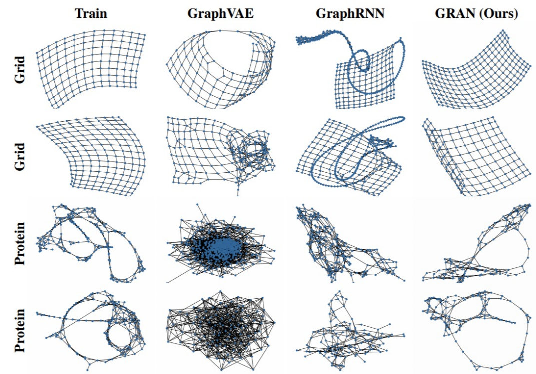
- Project tasks will give you hands-on experience

➢ At the end you should also be able to extend existing techniques and adapt them to different domains and applications

How do you learn complex distributions?

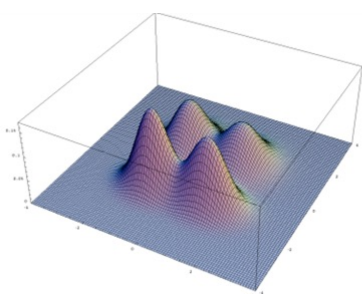
- How can we learn probability distributions over complex real-world data such as images, graphs, and audio signals?

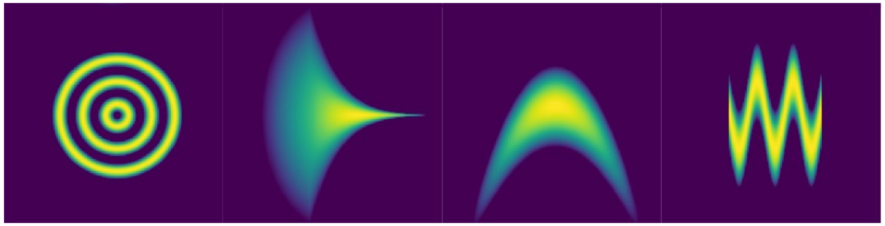
[Khan, 2019]



[Liao+, 2019]

- Distributions in high dimensions defy our intuition



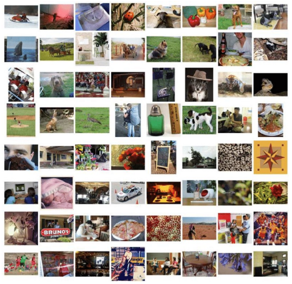


- How do we design flexible and efficient models for these settings?

Generative Models

- Deterministic Generative Models
  - Image = Renderer(object=cube, color=red, size=, position=, …)
  - Image = Renderer(object=cylinder, color=blue, size=, position=, …)

- Statistical Generative Models



+

- Model family
- Loss function
- Optimization algorithm
- ….

learning

=

$$p(\boldsymbol{x})$$

Data

Prior Knowledge

Probability Distribution

Desiderata for Statistical Generative Models

1. Efficient Sampling
   - Should be easy to sample a new instance $\boldsymbol{x}_{new} \sim p(\boldsymbol{x})$
   - Sampled/generated instances $\boldsymbol{x}_{new}$ should be similar to the training data

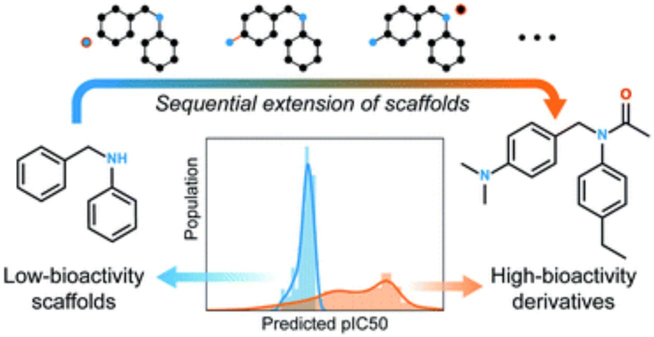
2. Efficient Likelihood Evaluation
   - Should be easy to evaluate $p(\boldsymbol{x})$ for any instance $\boldsymbol{x}$, e.g. $p($  $)$

3. (Optionally) Extract Features
   - For any instance $\boldsymbol{x}$ extract latent features/representations
   - Capture/summarize the important aspects of the instance/image

Some Applications of Generative Models

- Image generation
  - https://www.thispersondoesnotexist.com/
  - https://www.youtube.com/watch?v=p5U4NgVGAwg
- 3D graphics & fluid dynamics
  - https://www.youtube.com/watch?v=i6JwXYypZ3Y
- Speech & music synthesis
  - https://deepmind.com/blog/article/wavenet-generative-model-raw-audio
- Drug discovery

[Achlioptas+, 2017]

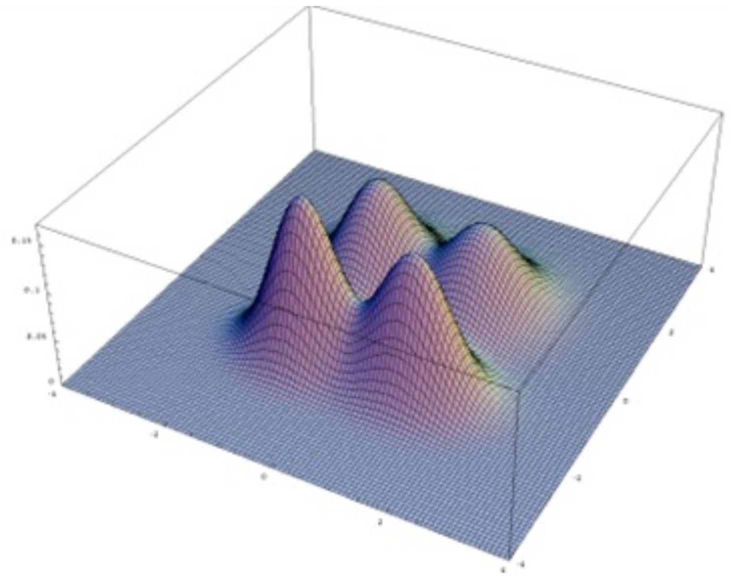


[Lim+, 2019]

[Karras+, 2018]

Continuous Distributions over High-dimensional Data

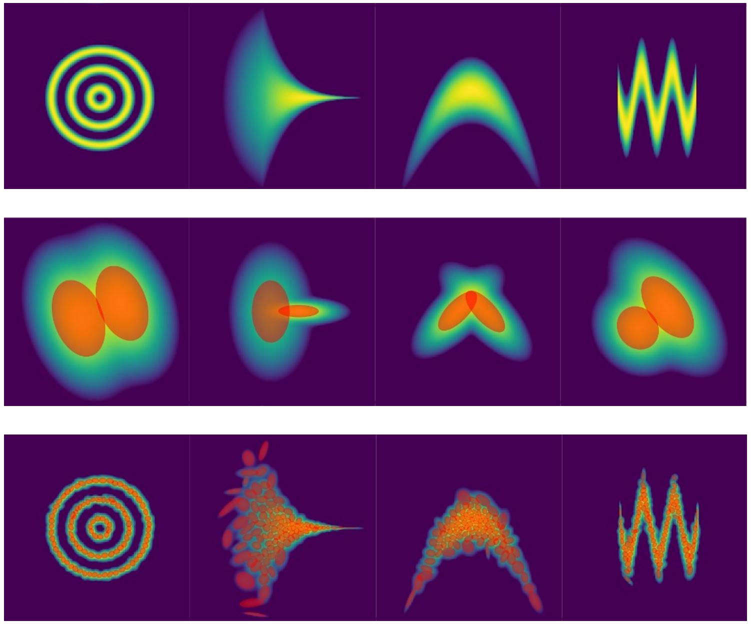
- "Classic" probability distributions (e.g. multivariate normal) do not capture the complexity of real-world datasets
  - Real distributions are multi-modal, asymmetric



- Can we use mixture models to capture this behavior?

Mixture models

- In theory, a mixture with enough components can represent any density
  - How many is “enough”?



[Vergari+, 2019]

- Even for simple 2D densities we need hundreds of mixture components!
  - The situation gets (exponentially) worse as we increase the dimensionality

Discrete Distributions over High-dimensional Data

- What about discrete distributions?

- Suppose x_1, x_2, x_3 are binary variables
  - $p(x_1, x_2, x_3)$ can be specified with $2^3 - 1 = 7$ parameters
  - $p(0,0,0)$, $p(0,0,1)$, … $p(1,1,0)$, $\cancel{p(1,1,1)}$

- For an image with N black or white pixels need to specify $2^N - 1$ values
  - The number of parameters grows exponentially with dimension

Challenges of High-dimensional Data

- "Classic" distributions
  - Do not capture the complexity of the data
- (Finite) mixture models
  - Require ridiculous amounts of parameters to specify even simple densities
  - Do not work in higher dimensions
- For discrete distributions – combinatorial explosion

- In this lecture, you will learn how to design flexible and efficient generative models for high-dimensional data using deep learning techniques

Contents of this Course

- Deep Generative Models
  - Normalizing flows
  - Variational inference / Variational Autoencoder
  - Generative Adversarial Networks
  - Denoising Diffusion

Roadmap

- Chapter: Introduction
  1. What will you learn in this lecture?
  2. **Organizational aspects + project tasks**

Course Organization

- Lecturer
  - Prof. Dr. Stephan Günnemann
- Teaching assistants
  - Dominik Fuchsgruber, Marten Lienen, David Lüdke

- 3 ECTS
- Language: English

- Ungraded exercise sheets
- Graded project tasks

- Final exam + repeat exam
- Project tasks can grant a bonus of up to 0.3

Schedule

- In-person lectures and exercises
  - Lecture Thursday 09:00 – 11:00
  - Exercise roughly every third Thursday 09:00 – 11:00
- Practice material and exercises uploaded to Moodle
  - Ungraded exercise sheets every third week
  - Project tasks every third week

Preliminary Timetable

Week		Topic	Project
1	20.04.2023	**Orga / Normalizing Flow**	
2	27.04.2023	Exercise	Normalizing Flows
3	04.05.2023	**Variational Inference**	
4	11.05.2023	**Variational Inference**	
		No lecture and Exercise	
5	25.05.2023	Exercise	
6	01.06.2023	**VAE**	
		No lecture and Exercise	
7	15.06.2023	**VAE/GANs**	
8	22.06.2023	**GANs**	VAE’s
9	29.06.2023	Exercise	
10	06.07.2023	**Diffusion**	
11	13.07.2023	**Diffusion**	Diffusion
12	20.07.2023	Exercise	

Prerequisites

- The course is designed for Master students of Computer Science (and specializations such as Data Engineering and Analytics, Games Engineering, etc.)

- **This course can not be taken by students that passed MLGS in previous years**

- Prerequisites:
  - Knowledge about the standard Machine Learning concepts (i.e. content of our lecture IN2064)
    - We assume the basic concepts are clear; no repetition!
    - We strongly recommend that you attend IN2064 first before taking this class
  - Knowledge about:
    - Algorithms and Data structures
    - Programming
    - Mathematics: Linear Algebra, Statistics, Optimization

Course Material + Announcements

- All course materials (slides, exercises) will be uploaded to Moodle
  - Video recordings of previous years lectures are accessible via link on Moodle
- Project submission on Artemis
- Use Piazza to ask questions! (please avoid sending e-mails)

https://piazza.com/tum.de/summer2023/cit4230003

Access Code: **dgm2023**

- Please read the guidelines for using Piazza

Exercises and Project Tasks

- Exercise sheets
  - Exam preparation
  - Solutions in the tutorials
  - Due to the high number of registrations, we are unable to provide corrections to your solutions

- Project tasks
  - Get hands-on experience with advanced machine learning methods
  - Improve your final grade! (details later)

Project Format

- Format of programming tasks
  - Tasks will be published via Artemis
  - artemis.in.tum.de
- How to solve programming tasks?
  - Clone template repository from Artemis exercise
  - Solve tasks described in the repository
  - Push repository with filled-in solutions
- Bonus regulations
  - 10 points for each programming sheet
  - A Bonus of 0.1, 0.2 and 0.3 grade points will be granted upon correct completion of 25%, 50% and 75% of all project points, respectively

Project topics

- Three project tasks on the following topics
  - Normalizing flows
  - Variational Autoencoder
  - Denoising Diffusion

- The specific tasks and all details will be described in the corresponding descriptions on Artemis

Exam & Grading Scheme

- Written final exam: 90 minutes
  - Date will be announced via TUMonline
  - We currently plan with an on-site exam
  - One handwritten two-sided A4 sheet with notes

```
def final_grade(exam_grade, project_grade):
    if exam_grade > 4.0:
        return exam_grade
    else:
        return max(1.0, exam_grade - bonus)
```

→ The project is voluntary and can only improve the final grade. The project bonus applies only if you passed, and you cannot improve beyond 1.0

Our Group's Focus

Reliable Machine Learning for **Non-Independent** Data

Robustness/Uncertainty

- Data corruptions, adversaries
- Certificates

Non-independent data

- Temporal/sequence data
- Graph data

- Interested? We offer:
  - Bachelor/Master theses, Guided Research projects, HiWi positions
- More details on specific topics closer to the end of the semester

References

Figures taken from

- Goodfellow et al. 2014, https://arxiv.org/abs/1412.6572
- Akbik et al. 2018, https://research.zalando.com/welcome/mission/research-projects/flair-nlp/
- Khan 2019, https://heartbeat.fritz.ai/stylegans-use-machine-learning-to-generate-and-customize-realistic-images-c943388dc672
- Liao et al. 2019, https://arxiv.org/abs/1910.00760
- Lim et al. 2019, Scaffold-based molecular design with a graph generative model
- Achlioptas et al. 2017, https://arxiv.org/abs/1707.02392
- Karras et al. 2019, https://github.com/NVlabs/stylegan
- Vegari et al. 2019, https://web.cs.ucla.edu/~guyvdb/slides/TPMTutorialUAI19.pdf