How to Start Your 24786 Project?

Shang Zhu, Alexander Bills

April 27th, 2023

Logistics

Homework 6: Open-Ended Projects (40 pts) (Pick one)

1. This is your last homework, where you only need to **pick one** of the below two projects to work on.

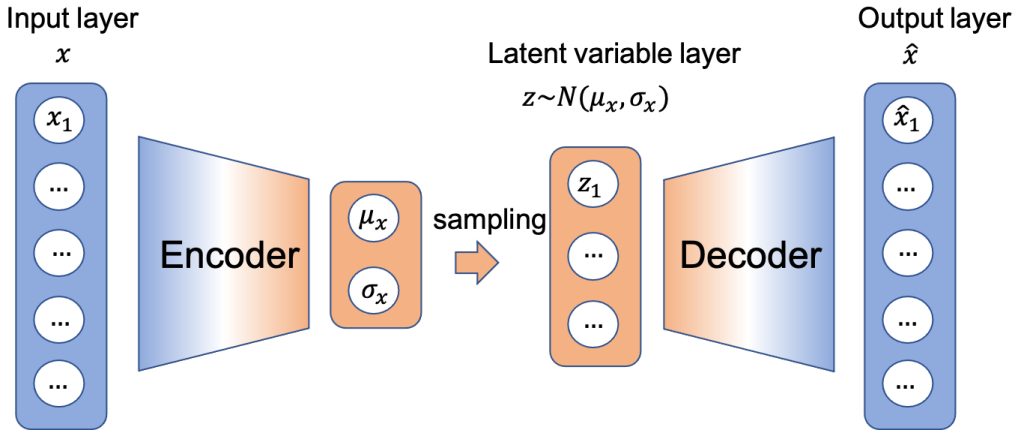
2. This homework is out on **April 21st**, and will be due on **May 12th**. We do not allow extensions this time, so please start early. If you are **graduating** this semester (so you may need grades uploaded before the due date), reach out to Professor and TA as soon as possible.

3. We will host a tutorial and Q&A session on **April 27th**, Thursday. Please be prepared with your questions, especially if you choose the second one.

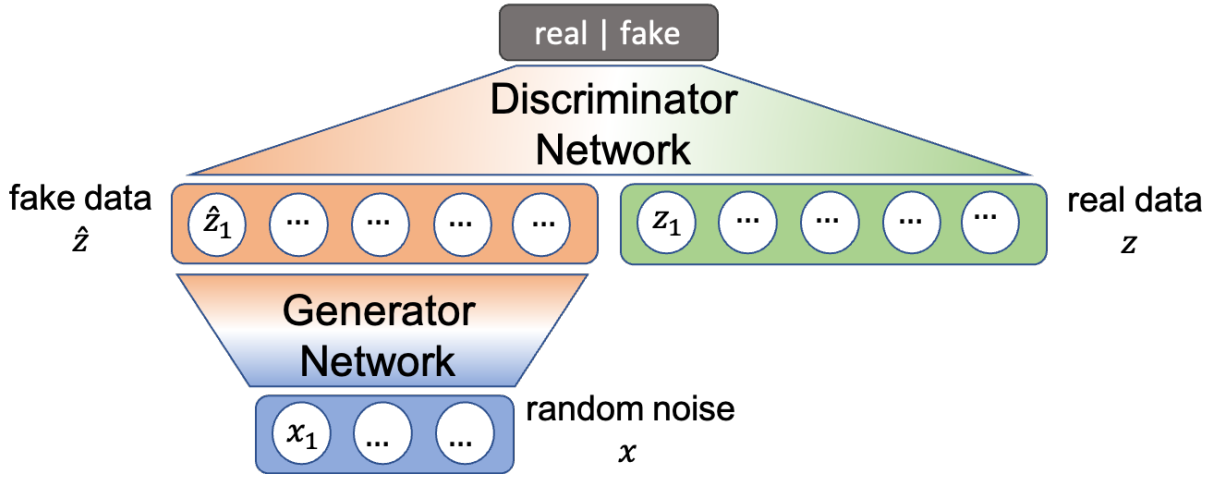
Project 1: Inverse Design for Airfoils

1. CNN for supervised learning, as you did in your last homework.

2. Generative Models for inverse design: we provided tutorials for GANs, VAEs, Diffusion Models in both MATLAB and Python (PyTorch), but feel free to use other open-sourced packages. You should mention which part of the code is borrowed from other resources.



Variational Autoencoder

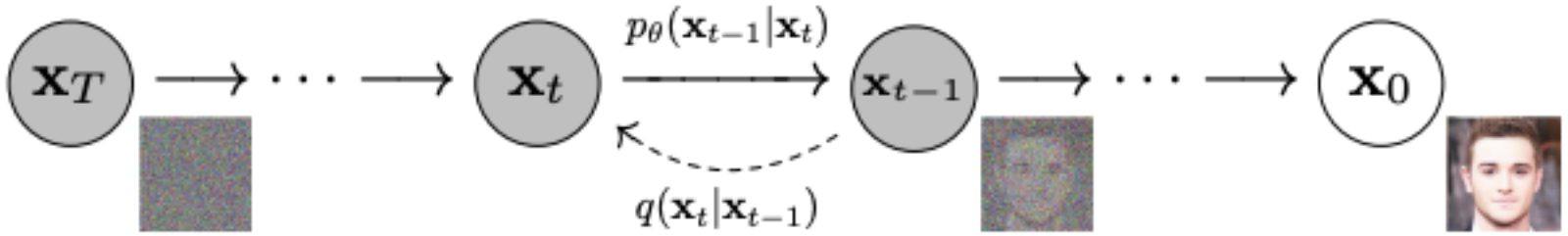


Generative Adversarial Network

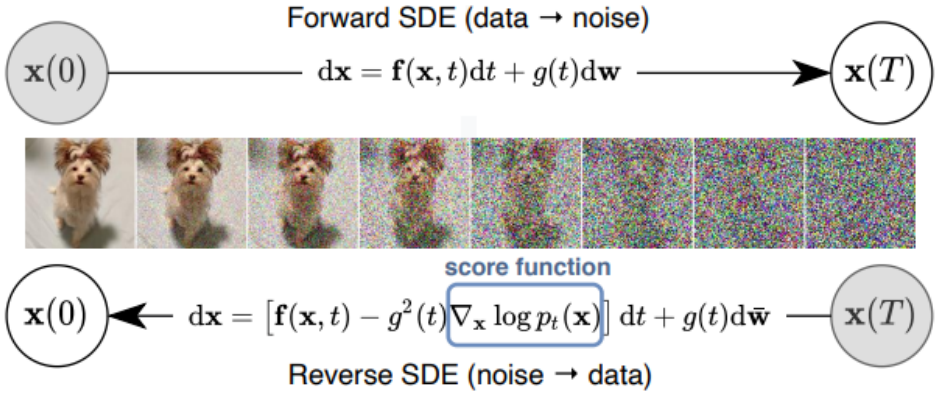
https://www.deeplearningbook.org/

Project 1: Inverse Design for Airfoils

3. Diffusion Model on Image Synthesis



Denoising Diffusion Probabilistic Models [Code Demo on google colab]



Solving a reverse-time SDE yields a score-based generative model.

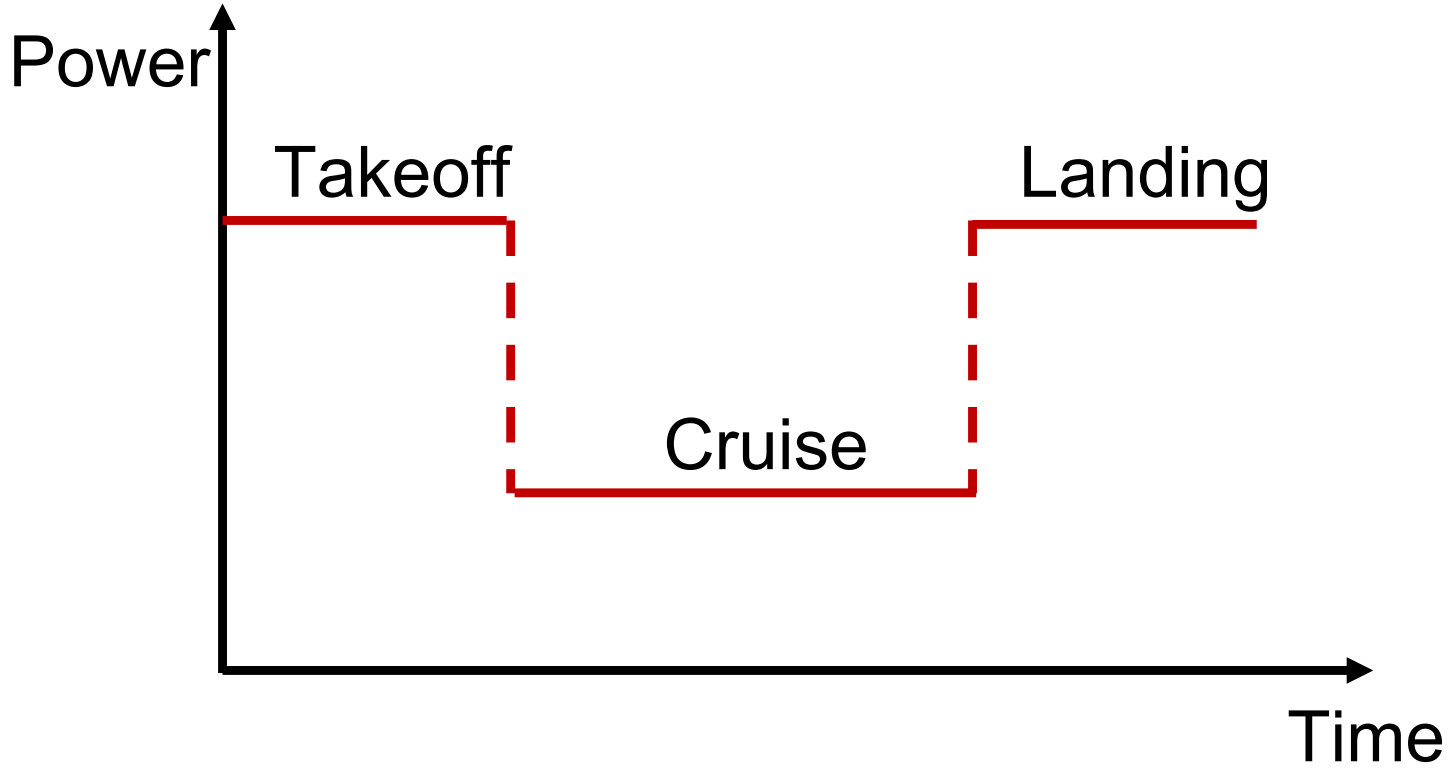
https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf

https://arxiv.org/pdf/2011.13456.pdf

Resources

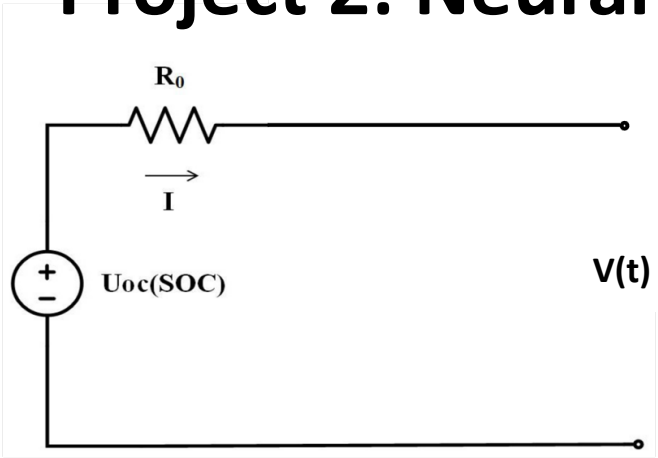
1.Google Colab with GPU support (Tesla T4)
2.Open-sourced Tutorials/Codes (Please tell us which package you were using)

Project 2: Neural ECM: Differentiable Equivalent Circuit Model

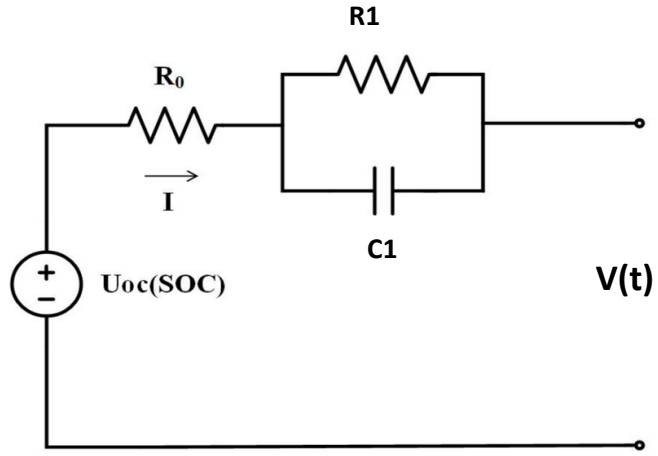


eVOTL Mission Profile

Project 2: Neural ECM: Differentiable Equivalent Circuit Model



(a) 0-RC pair ECM



(b) 1-RC pair ECM

ECM

You have predefined Uoc(z) in the provided Julia demo

(a) U_{oc}: open circuit voltage, which is typically a function of SOC (z), in the unit of V; a good initial guess is 4 V;

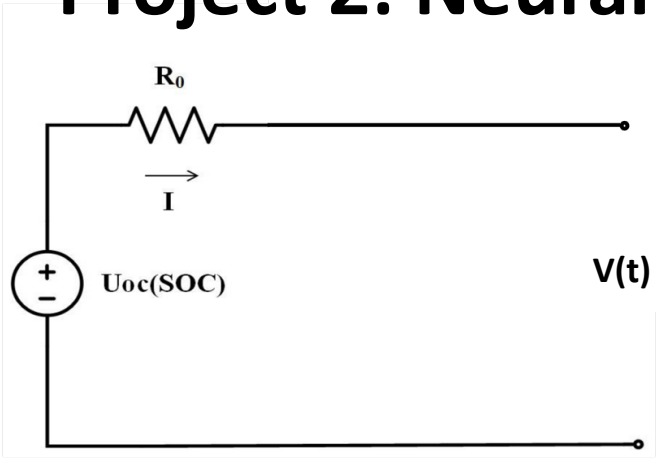
(b) R_0: a separate resistor, in the unit of milli ohm; a good initial guess is 20 $m\Omega$;

(c) R_1, C_1: R-C (resistor-capacitor) pair to capture more complicated dynamics; a good initial guess for R_1, C_1 is 200 $m\Omega$ and 10000 farads (F);

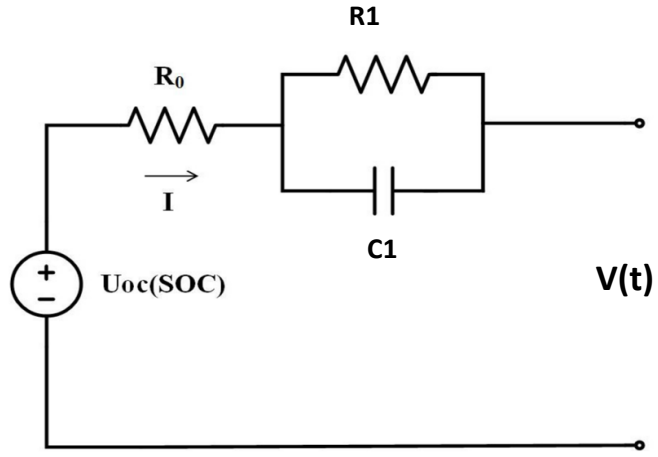
(d) $V(t)$, $I(t)$: measurable voltage and current, in the unit of V and A, respectively. This is what we collected in the data;

Parameters

Project 2: Neural ECM: Differentiable Equivalent Circuit Model



(a) 0-RC pair ECM



(b) 1-RC pair ECM

ECM

Be careful with the units, and sign!

For a simple 0 pair ECM model (also known as an OCV-R model), the system can be described by:

$$V(t) = U_{oc}(z(t)) - I_t(t) \cdot R_0$$

For any RC-pair, the current across the resistor branch can be described by

$$\frac{dI_1}{dt} = \frac{I(t) - I_1(t)}{R_1 C_1},$$

where I_1 is the current across the resistor R_1. Therefore, the voltage drop across the RC pair is

$$V_{RC} = I_1 R_1,$$

and the "terminal" (measured) voltage of the cell is

$$V(t) = U_{oc}(z(t)) - I(t)R_0 - I_1(t)R_1.$$

Equations

Project 2: Neural ECM: Differentiable Equivalent Circuit Model

Julia Code Demo

https://github.com/abillscmu/NeuralECM.jl/tree/main/scripts

Project 2: Neural ECM: Differentiable Equivalent Circuit Model

PyTorch Code Demo

Q&A