CPS 491 - Capstone II

Source: https://github.com/cps491sp21-team13/cps491sp21-main-repo.git

University of Dayton

Department of Computer Science

CPS 491 - Capstone II, Semester Year

Instructor: Dr. Phu Phung

Capstone II Project

Compressible Learning Agents for Autonomous Cyber-Physical Systems

Team members

1. Zachary Rowland, rowlandz1@udayton.edu

Company Mentor

Matthew Clark, Principal Scientist

Galois

444 E 2nd Street

Dayton, OH 45402

Project Management Information

Management board (private access): https://trello.com/b/kVF2rdUV/cps-491-team-13

Source code repository (private access): https://github.com/cps491sp21-team13/cps491sp21-main-repo

Project homepage (public): https://cps491sp21-team13.github.io/

Revision History

Date	Version	Description
16/02/2021	0.1	Details of Taylor and convolutional approximations
16/03/2021	0.2	Details data-driven SINDy approach/change-of-basis strategy for building neural networks
05/04/2021	0.3	Preliminary results of applying regression-based model fitting to neural networks

Overview

The goal of this project is to explore and document the correlation between the configuration of deep neural networks and the analytical mathematical functions that they model. A long term goal, possibly not explored in this project specifically, is to develop methods for *decomposing* an arbitrary deep neural network into smaller networks with intuitive behavior, and *simplifying* the network into the minimum network needed to model the desired mathematical function.

Project Context and Scope

This project is a contribution to research into the development of machine learning models that learn intuitively.

High-level Requirements

By the end of the semester, we hope to have reasonably comprehensive documentation of the different ways simple neural networks can approximate simple functions. The process of compiling this documentation should involve careful inspection of trained neural networks guided by mathematics.

Implementation

Most work on this project is being done in a Jupyter notebook using markdown for mathematics and Tensorflow for implementing neural networks.

For now, I am focused on fitting functions to pretrained neural networks using SINDy, a regression based technique. Here is a code snippit that illustrates how we can fit discover a function of the form $r\sin(x + \phi) + b$. For more details, see the notebook itself.

```
# Construct some sample data
f = lambda x: 0.5*tf.sin(x + 0.79) + 0.2
xdata = tf.random.uniform([100], minval=0.0, maxval=math.pi * 2.0)
ydata = f(xdata) + tf.random.normal([100], stddev=0.05)

# A function for converting from cartesian to polar coordinates
# (x,y) -> (r,theta)
def cart2polar(x, y):
  r = math.sqrt(x**2.0 + y**2.0)
  theta = math.acos(x/r) if y >= 0.0 else 2*math.pi - math.acos(x/r)
  return [r,theta]

# Fit regression against a basis of functions {sin(x), cos(x)}
# to get w0, w1, b
X = np.column_stack([tf.sin(xdata), tf.cos(xdata)])
reg = sklearn.linear_model.LinearRegression()
reg.fit(X, ydata)
w0, w1 = reg.coef_
b = reg.intercept_
```

```
# Output model equation in two different forms
print(f"Model: {w0:.2f}sin(x) + {w1:.2f}cos(x) + {b:.2f}")
r, theta = cart2polar(w0, w1)
print(f"Model: {r:.2f}sin(x + {theta:.2f}) + {b:.2f}")
print(f"Model R^2 value: {reg.score(X, ydata):.2f}")
```

Software Process Management

Scrum process

Sprint 0

Duration: 20/01/2021-27/01/2021

Completed Tasks:

1. Studied fuzzy inference systems from material provided from Galois
2. Studied automatic differentiation.

Sprint Retrospection:

Good	Could have been better	How to improve?
learned a lot of preliminary information	documentation of progress	take more notes on what I read about/discover

Sprint 1

Duration: 27/01/2021-03/02/2021

Completed Tasks:

1. Learned tensor manipulation and learning models in Tensorflow
2. Experimented with small neural networks in a Jupyter notebook

Sprint Retrospection:

Good	Could have been better	How to improve?
good pace aquiring new knowlege and documenting what I have learned	could have started experimenting with small networks sooner	reach out to ask questions sooner

Sprint 2

Duration: 03/02/2021-09/02/2021

Completed Tasks:

1. Learned about convolution from Linear Systems.
2. Detailed a technique for constructing neural networks that approximate arbitrary functions inspired by convolution.
3. Detailed a technique for approximating arbitrary polynomials with neural networks based on Taylor-series expansion.

Sprint Retrospection:

Good	Could have been better	How to improve?
documentation was thorough and clear	direction of work could have been more focused toward project goal	more formal communication about the precise goal of the project

Sprint 3

Duration: 28/02/2021-09/02/2021

Completed Tasks:

1. Explored Taylor series approximations and detailed a method of building a neural network using a Taylor series approximation
2. Explored Fourier series approximations

Sprint Retrospection:

Good	Could have been better	How to improve?
Documentation was clear	Better communication with company sponsor regarding the direction of the project	smaller, more frequent communication

Sprint 4

Duration: 01/03/2021-16/03/2021

Completed Tasks:

1. Learned about spline-theory as it relates to deep learning.
2. Learned about SINDy approach to fitting functions to data.

Sprint Retrospection:

Good	Could have been better	How to improve?
Learning was done quickly and efficiently	Could have implemented ideas more instead of just detailing them in math	organize notebook and code

Sprint 5

Duration: 16/03/2021-24/03/2021

Completed Tasks:

1. Applied SINDy to a network trained on the MNIST dataset.
2. Built Python infrastructure for manipulating equations and data as required by SINDy.
3. Compared linear and quadratic models learned from SINDy for classifying the a digit.

Sprint Retrospection:

Good	Could have	How to improve?

	been better	
Very productive in terms of documentation and implementation. Concrete results were found.	NA	Continue with this line of work.

Sprint 6

Duration: 24/03/2021-05/04/2021

Completed Tasks:

1. Wrote project introduction and motivation.
2. Literature review on approximation theory of neural networks and model compression.

Sprint Retrospection:

Good	Could have been better	How to improve?
Productivity	NA	Continue literature review

Acknowledgments

I would like to thank Mr. Matthew Clark, Principal Scientist at Galois, for mentoring this project.