# Zane Alumbaugh

San Francisco Bay Area | zanedma@gmail.com | (510)599-6558 | Portfolio Site | LinkedIn | Github

## Education

**BS in Computer Science,** *University of California - Santa Cruz* Sep 2017 – Jun 2021

- Achievements: UCSC Dean's Honors List - Winter 2021, Spring 2020, Winter 2020, Winter 2018

## Skills

**Coding Languages**
Python, C/C++, Golang, HTML/CSS, Java, JavaScript, Scala, Prolog, Haskell

**Libraries/Frameworks**
PyTorch, NumPy, Pandas, SciPy, matplotlib, BeautifulSoup, Django, Bootstrap, Flask

**Software**
Docker, Git, Bash, UNIX, Windows, MacOS

**General**
Machine Learning, Reinforcement Learning, Algorithms, Computer Networking

## Professional Experience

**Student Research Assistant,** *University of Washington* Jun 2020 – present | Remote

- Extended existing policy gradient and soft-actor critic reinforcement learning algorithms to multiplayer environments building on OpenAI's Spinningup Deep Reinforcement Learning package and PyTorch.
- Developed and implemented multiplayer reinforcement learning algorithms using PyTorch based on game-theoretic models of player interaction including linear quadratic dynamic games.
- Co-Author in: Liyuan Zheng, Tanner Fiez, Zane Alumbaugh, Benjamin Chasnov, Lillian J. Ratliff. Stackelberg Actor-Critic: A Game-Theoretic Perspective, Association for the Advancement of Artificial Intelligence Reinforcement Learning and Games (AAAI RLG) Workshop, 2021.
- Supervisor: Lillian Ratliff, Assistant Professor, UW. Mentor: Ben Chasnov, PhD Student, UW

**Freelance Software Engineer,** *BuddhiBox* Jul 2020 – Aug 2020 | Remote

- Created a bot that scrubs daily update emails from "Help a Reporter" and notifies the user of any links containing keywords/phrases. Implemented using Python, Gmail API, and BeautifulSoup.

**Student Research Assistant,** *University of Washington* Jul 2019 – Sep 2019 | Seattle, WA

- Implemented gradient-based learning algorithms for training Generative Adversarial Networks (GANs) using the PyTroch Framework.
- Implemented an optimizer class for PyTorch that used second-order derivatives to predict future optimizations and allowed for the learning agent to act on these predictions .

## Projects

**Causally Consistent Distributed Key/Value Store,** *Final project for my Distributed Systems capstone course*

- Designed and implemented a distributed Key/Value Store (KVS) using Golang that:
  - Was fault-tolerant and partition-tolerant using data replication.
  - Key/Value pairs were causally consistent on all nodes.

**Movie Review Classifier,** *Final project for my Machine Learning course* ↗

- Implemented a movie review sentiment classifier that is 93% accurate using a Bidirectional Encoder Representations from Transformers (BERT) neural network.