

Austen Nelson

ajn6@pdx.edu — Software Engineer / Data Scientist

Work Experience

Self Employed | 2014 – 2017 | Math and Science Tutor

Operated a small personal business where I mostly tutored calculus to engineering students

Portland State University | 2019 – 2021 | Technical Course Support Specialist

Karla Fant teaches a majority of the lower division courses (CS162, CS163, CS202) at PSU

Responsibilities included:

- Facilitate homework recitation and labs
- Proctor programming proficiency demos and written exams
- Create course materials
- Grading and administrative tasks

Education

Portland State University — BCS

Bachelor's of Computer Science completed in May of 2021 with a 3.8 GPA

Programming Languages

- **Rust** — My strongest, most modern, and favorite tool to use. Rust is a innovative and exciting project started by Mozilla in 2010. The goals of the project are to create a C++ replacement with a focus on performance, safety, and concurrency. These values are mostly achieved by the novel ownership system that allows for powerful static analysis of the language. The language has seen tremendous growth in the recent years and I would love to keep expanding my Rust knowledge.
- **C++** — Two years of experience in C++ 98 through school projects and tutoring. I have very limited exposure to modern C++ concepts.
- **Python** — Almost all of my data science and AI/ML projects have been done using python and the popular libraries of Numpy, Pandas, Matplotlib, Scikit Learn, and Requests. I have also used Python to set up automatic deployment of cloud resources using the GCP via their Deployment Manager API.
- **Latex** — Quite familiar using Latex to create beautiful documents that can be tracked and collaborated easily with a VCS like git.
- Limited exposure to Java, R, Javascript, HTML/CSS, Bootstrap, Haskell, C, and Bash.

Tools

- **Google Cloud Platform (GCP)** — My capstone project for graduation was to design a codelab to teach various cloud security and vulnerability topics using the GCP. Concepts I am familiar with include IAM, service accounts, compute and storage resources, deployment management and automation, and least privilege. See projects section for more information.

- **Git** — Confident using version control in a team environment. Proficient in branching, pull requests, and merge conflict resolution.
- **SQL** — Basic understand of SQL concepts and techniques. Most experience working with the Postgres software.
- **Agile** — Familiar with the general agile concepts and how they can be used in software developed. Used Agile techniques in group projects such as capstone.
- **Linux** — Use Linux as my primary development environment. Familiar with Bash as well as basic system administration concepts such as init systems (systemd), users and groups, permissions, and file systems. Vim »> Emacs
- **Docker** — Use docker compose for projects and personal projects to simplify build processes.

Concepts

- Artificial Intelligence and Machine Learning — Strong background in applied statistics and linear algebra. Familiar with statistical techniques such as naive Bayes, regression analysis, and descriptive analysis. Decent understanding of different types of multi-layer perceptrons (CNN, RNN, LSTM, GAN, etc.) and gradient decent for supervised learning. See projects section for an example of a 2 layer fully connected perceptron classifier I implemented from scratch to do MNIST classification in my college machine learning course. Familiar with unsupervised learning techniques such as k-means clustering and modularity based graph algorithms for community detection in large networks. See projects section for a graph focused research project where I identified communities of cooking ingredients using modularity to generate novel recipes.
- Security, Malware, and Hacking — Took two classes at university focused on computer security. One was an introduction to malware where the primary exercises were using disassembly tools (IDA) to solve CTFs. The second was a web and cloud course where we completed exercises to demonstrate improper authentication techniques, brute force and fuzzing, improper access controls, server-side request forgeries, cross site scripting, injection attacks (SQL, XML, Bash, etc.), improper CORS, cross site request forgeries, clickjacking techniques, improper deserialization, and improper cloud configuration. I have also competed in a couple different CTF competitions and would love to do more!
- Systems Programming — As a Rust evangelist I have an interest in systems level programming such as data encoding/processing, operating systems, embedded systems, and networking. I am currently working on a fullstack web application project where the API is implemented using the warp async/await Rust library implemented on top of the Tokio runtime with a Postgres database (see projects). I enjoy working with and designing backend systems and APIs to create scalable applications.

Personal Interests and Projects

Undergraduate Research Mentorship Program: Culinary Computation

Monte Carlo Simulation of Set[®]

I am interested in the probabilities of this game.

Motivations and an analysis I did can be found here: Set Simulator

Linear Algebra

I am co-developed an open source linear algebra library with the help of Waylon Cude when I was in an advanced linear algebra course at PSU. The library was to demonstrate my familiarity with sparse matrix operations and algorithms. As a final project for the class I reimplemented an iterative solver known as

GMRES.

Here is a link to the project: [matrixlab](#)

Thunder CTF: Cloud Audit and Incident Response

Primitive Guitar Pedal

Multi-Layer Perceptron

Gardens App

References

- Karla Fant — Senior Instructor, PSU MCECS — (503) 725-5394
- Panayot S Vassilevski — Computational Research Mathematician, Lawrence Livermore National Labs