Tien Ly

(408) 329-8732 | tien.n.ly@sjsu.edu | linkedin.com/in/tien-ly-468bb41a2/

EDUCATION

San Jose State University — San Jose, CA
Master of Science in Bioinformatics — *Aug 2023 – Expected May 2025*
- GPA: 4.0
- Relevant Coursework: Bioinformatics II, Molecular Ecology, Data Visualization, Processing Big Data, Machine Learning, Applied Probability and Statistics

University of California, Davis — Davis, CA
Bachelor of Science in Computer Science — *Sep 2019 – Jun 2022*
- GPA: 3.98
- Relevant Coursework: Data Structures, Algorithm Design and Analysis, Web Programming, Essentials of Life on Earth, Principles of Ecology and Evolution, Biodiversity and the Tree of Life

SKILLS

Programming Languages: Python, R, Bash Command Line, C/C++, HTML, CSS, JavaScript, SQL
Developer Tools: Docker, Google Cloud Platform, Git/GitHub, RStudio, Jupyter Notebook, Google Colab, NCBI databases, Visual Studio Code, Microsoft 365
Operating Systems: Linux, macOS, Windows
Frameworks: Node.js, React

EXPERIENCE

Graduate Student Researcher — Aug 2023 – Present
SJSU Castillo Vardaro Molecular Ecology Lab — *San Jose, CA*
- Perform whole-genome sequencing analysis of historic and modern American pika populations to identify signals of local adaptation and responses to climate change on a high-performance computing (HPC) cluster
- Develop a bioinformatics pipeline using various command line tools to analyze high-throughput sequencing data, covering quality control through variant calling stages

Satellite Data Scientist Intern — Jun 2024 – Aug 2024
NOAA National Marine Fisheries Service, CoastWatch, PolarWatch Node — *Remote*
- Led the analysis of over 40 years of satellite data, utilizing Python to identify significant sea ice concentration anomalies and trends, resulting in actionable insights for climate research
- Developed and optimized a Python-based workflow for processing and visualizing large datasets, improving data accessibility for PolarWatch users and facilitating research in polar region changes
- Built a predictive model using deep learning techniques to forecast sea ice patterns

Bioinformatics Programmer — Sep 2021 – Aug 2023
UC Davis Genomic Variation Laboratory — *Davis, CA*
- Developed R scripts to analyze, visualize, and interpret large amounts of next-generation sequencing (NGS) data, improving species detection and identification
- Optimized bioinformatics workflows for DNA metabarcoding datasets, resulting in a 50% increase in processing speed
- Authored detailed protocols for data analysis, ensuring consistency and reproducibility in research methods
- Presented progress updates and research findings to an interdisciplinary research team during weekly meetings

PROJECTS

Support Vector Machine (SVM) Prediction of BRAF V600E Inhibitors | *Python (scikit-learn, pandas, seaborn)*
- Applied machine learning techniques to predict novel small molecules targeting BRAF V600E mutation, contributing to cancer-related drug discovery
- Applied preprocessing, including addressing constant features and utilizing SelectKBest method for feature selection, to a PubChem dataset of 243 compounds with 357 descriptors

- Developed an SVM model achieving an impressive 98% accuracy in identifying small molecule inhibitors of BRAF V600E target
- Evaluated SVM model through cross-validation, confusion matrix analysis, and ROC curve, demonstrating its robust performance

Benchmarking of Two De Novo Genome Assemblers for Short Reads | *Bash, R*
- Conducted a comparative analysis of two popular de novo genome assemblers, SPAdes and MEGAHIT, using microbial datasets to assess assembly contiguity and inform researchers on optimal assembly strategies for microbial genomes
- Utilized high-throughput, short sequence, and trimmed reads from 12 paired-end datasets, obtained from MicrobesNG, and sequenced with an Illumina HiSeq instrument
- Employed SPAdes 3.7 with recommended settings for 250-bp paired-end reads and MEGAHIT 1.2.9 for genome assembly, demonstrating proficiency in both assembler selection and usage
- Leveraged ggplot2 package in R to create informative plots visualizing assembly contiguity metrics, including total contigs, N50, and L50, generated through QUAST analysis

LIBUTHREAD | *C, Makefile, Git, GitHub*, Linux
- Developed a lightweight user-level thread library for Linux
- Programmed a FIFO queue API to manage thread scheduling
- Created a uthread library API for concurrent execution of independent threads
- Employed alarms and signal actions to enable thread preemption

Fitness Tracker | *HTML, CSS, JavaScript, Node.js, D3, SQLite*
- Crafted a full-stack web application to monitor users' exercise routines
- Implemented AJAX requests for seamless data exchange with web server on Replit
- Engineered a streamlined SQL database to securely store user activity sessions
- Leveraged Passport.js for third-party login integration with Google

Completely Fair Scheduler | *C++, Makefile, Google Test, Linux*
- Built a simplified version of Completely Fair Scheduler, default process scheduler of Linux kernel, leveraging a multimap data structure created using a Left-Leaning Red-Black tree
- Wrote an extensive suite of relevant unit tests to rigorously evaluate implementation and maximize test coverage

REFERENCES AVAILABLE UPON REQUEST