Michael Newman

Los Angeles, CA | (213) 868-5115 | michaelnewman.cs@gmail.com | github.com/mn-cs | michael-newman.dev
Authorized to work in the U.S. for any employer

Summary

Data Scientist with hands-on experience in large-scale data processing, machine learning, and distributed computing on HPC infrastructure. Focused on data pipelines, statistical analysis, and predictive modeling.

Education

- **University of California San Diego** San Diego, CA
  Master of Science in Data Science — GPA: 4.0 *September 2024 – June 2027*
- **Armenian State University** Yerevan, Armenia
  Bachelor of Science in Engineering *August 2013 – July 2017*

Skills

- **Languages**: Python, SQL | **Technologies**: PySpark, Pandas, AWS, Git | **Tools**: VS Code

Projects

- **FineWeb-Edu Large-Scale Analysis — GitHub** *(Group Project)*
  - Processed **9.67 million** educational web documents stored in 14 Parquet files using Apache Spark on SDSC Expanse HPC (128 GB RAM, 32 cores).
  - Built a distributed Spark ML pipeline with RegexTokenizer, StopWordsRemover, Word2Vec embeddings, and feature normalization for end-to-end preprocessing.
  - Addressed severe class imbalance (86.7% majority class) using stratified sampling, enabling fair model training across quality score buckets.
  - Trained and compared two distributed Random Forest classifiers, improving test accuracy from 61.9% to **68.7%** by tuning tree depth and ensemble size.
- **Success Factors — GitHub**
  - Built an end-to-end data pipeline integrating **Forbes API** data with scraped Wikipedia biographies, with modular cleaning functions to normalize schemas and transform JSON into structured Pandas DataFrames.
  - Conducted SQL-based exploratory analysis on **2,919 records**, applying statistical methods and demographic segmentation to identify key patterns and insights.
  - Created multi-panel visualizations using Matplotlib and Seaborn to analyze wealth distributions, geographic patterns, and demographic trends.
  - Ensured reproducibility through virtual environments, Docker containerization, and version-controlled analysis workflows.
- **Feature Representation Analysis — GitHub**
  - Designed controlled experiments comparing **four feature extraction methods** (raw pixels, HOG, pretrained and random CNN embeddings) with k-NN classification.
  - Applied PCA and t-SNE to reduce dimensionality, visualize high-dimensional feature spaces, and improve computational efficiency.
  - Evaluated CNN architecture performance using statistical tests and accuracy metrics across feature representations.