# Anvith Thumma

(650)-520-4546    GitHub | LinkedIn    arthumma@asu.edu / anvith.thumma@gmail.com

## EDUCATION

**Arizona State University**, *Tempe*
*Bachelors of Science in* Data Science; *Minor in* Business                                                                 **Expected May 2026**

## RELEVANT COURSEWORK

*Deep Learning Structures | Exploring Data in R and Python | Machine Learning | Data Structures and Algorithms | Computer Organization | Applied Linear Algebra | Discrete Mathematical Structures | Stat Model Inference Data Sci | Industry Tools Data Science*

## SKILLS

**Programming Languages:** *Python, SQL, Java, C/C++, C#, JavaScript, R*
**Data Science & Machine Learning:** *TensorFlow, Scikit-Learn, Pandas, NumPy*
**Database Management:** *MongoDB, MySQL*
**Data Visualization:** *Matplotlib, Seaborn, Folium, Dash*
**Tools:** *GitHub, Jupyter Notebook, MongoDB Compass, Studio3t*

## CERTIFICATIONS

- *[PCEP-30-02] PCEP™ – Certified Entry-Level Python Programmer*
- *AWS Certified Cloud Practitioner: Expires on September 01, 2026*
- *IBM Data Science Professional Certificate - Coursera*
- Stanford & DeepLearning.AI Machine Learning Specialization - Coursera

## PERSONAL PROJECTS

**Breast Cancer Prediction Model**                                                                                                         **November 2024**
- *Built and evaluated Decision Tree, K-Nearest Neighbors (KNN), and Logistic Regression models to classify breast tumors (benign vs. malignant) using the Wisconsin Breast Cancer Diagnostic dataset (569 records, 30 features).*
- *Achieved 96% accuracy with Logistic Regression on both full and reduced datasets (top 10 important features) and demonstrated a balance in precision and recall, minimizing false negatives.*
- *Conducted feature importance analysis highlighting the role of size and shape metrics in diagnosis.*
- *Compared model performances between full and reduced datasets, illustrating the robustness of Logistic Regression and the sensitivity of KNN and Decision Tree to feature dimensionality.*

**SpaceX Falcon 9 Launch Success Prediction (IBM Data Science Capstone Project)**                                           **July 2024**
- *Developed a predictive model to determine the successful landing of the SpaceX Falcon 9 first stage, helping decrease launch costs from $165 million to $62 million.*
- *Gathered data from the SpaceX API and performed web scraping from Wikipedia, resulting in a dataset with over 1000 launch records.*
- *Conducted data wrangling and cleaning using Pandas and NumPy, reducing missing data by 95%; utilized SQL for exploratory data analysis to extract key insights.*
- *Created 20+ visualizations with Matplotlib, Seaborn, and Folium to identify patterns and trends; developed interactive dashboards with Dash to display launch outcomes and correlations.*
- *Built and evaluated models using Logistic Regression, SVM, Decision Tree, and KNN with Scikit-learn; achieved highest performance with Decision Tree (GridSearchCV best score: 0.875), improving prediction accuracy by 15%.*

**Collaborative Filtering Recommender System (Stanford & DeepLearning.AI Machine Learning Lab)**                          **June 2024**
- *Implemented a collaborative filtering algorithm to build a movie recommender system, processing 20,000+ user ratings.*
- *Preprocessed and explored the MovieLens dataset (9000 movies, 600 users) using Pandas to extract relevant features and understand user preferences.*
- *Trained the model on 25,521 user-movie rating data points to learn latent features and generate personalized movie recommendations using TensorFlow.*
- *Evaluated and tuned the model using Scikit-Learn, achieving a 15% improvement in recommendation accuracy, resulting in a functional system that effectively predicted user preferences.*

**Neural Networks for Handwritten Digit Recognition Multiclass (Stanford & DeepLearning.AI Machine Learning Lab)**        **June 2024**
- *Designed and implemented a neural network to classify handwritten digits (0-9) using the MNIST dataset.*
- *Utilized Python and NumPy for reshaping MNIST images from 28x28 pixels to flat vectors of 784 features, and normalized pixel values to a range of 0 to 1.*
- *Applied data augmentation techniques like rotation, scaling, and shifting to enhance training data diversity; used Python to split the dataset into training, validation, and test sets.*
- *Designed a neural network architecture using TensorFlow with ReLU activation and softmax for multiclass classification; trained the model with optimized hyperparameters and applied dropout to improve generalization.*