**GitHub and Git Documentation Guide by Somnath Das**

**Table of Contents**

1. Project Overview
2. GitHub Repository Structure
3. Git Commands Reference
4. Daily Workflow
5. Troubleshooting Guide
6. Best Practices

**Project Overview**

**VCU Fall 2025 - Road Accident Risk Prediction**

**Repository**: <https://github.com/somnathdas75/VCU_Fall2025>

**Project Description**: Advanced machine learning pipeline for predicting road accident risk using ensemble methods and comprehensive feature engineering.

**Key Features**:

* 6-Model Ensemble (Random Forest, Gradient Boosting, Ridge, Lasso)
* 22 Engineered Features
* Cross-Validation with Performance Weighting
* Comprehensive EDA and Visualization
* Kaggle Competition Ready

**GitHub Repository Structure**

text

VCU\_Fall2025/

├── README.md

└── RoadAccidentPrediction/

├── accdrisk\_v4\_somnathdas\_adv.py # Advanced Ensemble Model

├── accdrisk\_v3\_somnathdas\_py.py # Basic Pipeline

├── train.csv # Training Dataset

├── test.csv # Testing Dataset

├── submission.csv # Kaggle Submission

├── advanced\_ensemble\_submission.csv # Enhanced Submission

├── accident\_risk\_analysis.png # EDA Visualization

└── feature\_importance.png # Model Insights

**Git Commands Reference**

**Basic Configuration**

bash

*# Set user identity*

git config --global user.name "Somnath Das"

git config --global user.email "your.email@example.com"

*# Check configuration*

git config --list

**Repository Management**

bash

*# Initialize new repository*

git init

*# Clone existing repository*

git clone https://github.com/somnathdas75/VCU\_Fall2025.git

*# Check remote connections*

git remote -v

*# Add remote repository*

git remote add origin https://github.com/somnathdas75/VCU\_Fall2025.git

*# Remove remote*

git remote remove origin

**Daily Workflow Commands**

bash

*# Check status*

git status

*# View changes*

git diff

*# Add files to staging*

git add filename.py *# Single file*

git add \*.py *# All Python files*

git add . *# All files*

*# Commit changes*

git commit -m "Descriptive commit message"

*# View history*

git log

git log --oneline *# Compact view*

git log --graph --oneline *# Visual history*

**Branch Management**

bash

*# Create and switch to new branch*

git checkout -b feature-branch

*# List branches*

git branch

git branch -a *# Show all branches*

*# Switch branches*

git checkout main

git checkout feature-branch

*# Merge branches*

git checkout main

git merge feature-branch

*# Delete branch*

git branch -d feature-branch

**Remote Operations**

bash

*# Push to remote*

git push origin master

git push -u origin master *# Set upstream*

*# Pull changes*

git pull origin master

*# Fetch without merging*

git fetch origin

**Daily Workflow**

**Standard Development Process**

1. **Check Status**: git status
2. **Review Changes**: git diff
3. **Stage Files**: git add .
4. **Commit Changes**: git commit -m "meaningful message"
5. **Push to GitHub**: git push origin master

**Commit Message Convention**

text

feat: Add new feature

fix: Resolve bug

docs: Update documentation

style: Code formatting changes

refactor: Code restructuring

test: Add tests

chore: Maintenance tasks

**Example Workflow Session**

bash

*# Start working session*

git status

*# Make changes to files...*

*# Then stage and commit*

git add .

git commit -m "feat: Enhance ensemble model with additional features"

git push origin master

A screen shot of a computer program

AI-generated content may be incorrect.

**GitHub page**

**A screenshot of a computer

AI-generated content may be incorrect.**

* **TTroubleshooting Guide**

**Common Issues and Solutions**

**1. Repository Not Found Error**

bash

*# Problem: remote: Repository not found*

*# Solution: Check remote URL and repository existence*

git remote -v

git remote remove origin

git remote add origin https://github.com/somnathdas75/VCU\_Fall2025.git

**2. Authentication Issues**

bash

*# Use Personal Access Token instead of password*

*# Create token at: https://github.com/settings/tokens*

**3. Merge Conflicts**

bash

*# Pull latest changes*

git pull origin master

*# Resolve conflicts in files, then:*

git add resolved-file.py

git commit -m "Resolve merge conflicts"

git push origin master

**4. Undoing Changes**

bash

*# Discard uncommitted changes*

git checkout -- filename.py

*# Unstage files*

git reset HEAD filename.py

*# Undo last commit (keep changes)*

git reset --soft HEAD~1

*# Undo last commit (discard changes)*

git reset --hard HEAD~1

**Success Verification**

After push, verify at:  
[**https://github.com/somnathdas75/VCU\_Fall2025**](https://github.com/somnathdas75/VCU_Fall2025)

Check for:

* ✅ Files visible in repository
* ✅ README.md displays correctly
* ✅ Folder structure intact
* ✅ Commit history updated

**Best Practices**

**1. Commit Frequently**

* Make small, focused commits
* Commit related changes together
* Test before committing

**2. Write Good Commit Messages**

* Use imperative mood ("Add" not "Added")
* First line max 50 characters
* Include context in body if needed

**3. Branch Strategy**

* Use feature branches for new development
* Keep main branch stable
* Merge with pull requests

**4. .gitignore Setup**

Create .gitignore file to exclude:

text

# Python

\_\_pycache\_\_/

\*.pyc

\*.pyo

\*.pkl

\*.model

# Environment

.env

.venv

# System

.DS\_Store

Thumbs.db

**5. Regular Maintenance**

bash

*# Pull before push*

git pull origin master

*# Keep branches clean*

git branch --merged

git branch -d old-feature-branch

*# Update remote references*

git fetch --prune

**Advanced Features**

**Stashing Changes**

bash

*# Save temporary changes*

git stash

*# List stashes*

git stash list

*# Apply last stash*

git stash apply

*# Clear stashes*

git stash clear

**Tagging Releases**

bash

*# Create version tag*

git tag v1.0.0

*# Push tags*

git push origin --tags

**GitHub Features to Explore**

1. **Issues**: Bug tracking and feature requests
2. **Projects**: Kanban-style project management
3. **Actions**: CI/CD automation
4. **Wiki**: Project documentation
5. **Pages**: Project website hosting

**Quick Reference Card**

**Essential Commands**

bash

git status *# Check status*

git add . *# Stage all changes*

git commit -m "message" *# Commit changes*

git push origin master *# Push to GitHub*

git pull origin master *# Pull latest changes*

**Useful Aliases (Add to ~/.gitconfig)**

ini

[alias]

co = checkout

br = branch

ci = commit

st = status

last = log -1 HEAD

lg = log --oneline --graph --all

**Support Resources**

* **Git Documentation**: <https://git-scm.com/doc>
* **GitHub Help**: [https://help.github.com](https://help.github.com/)
* **Visual Git Guide**: <https://marklodato.github.io/visual-git-guide>

**Repository URL**: <https://github.com/somnathdas75/VCU_Fall2025>

*Documentation created for VCU Fall 2025 Projects*