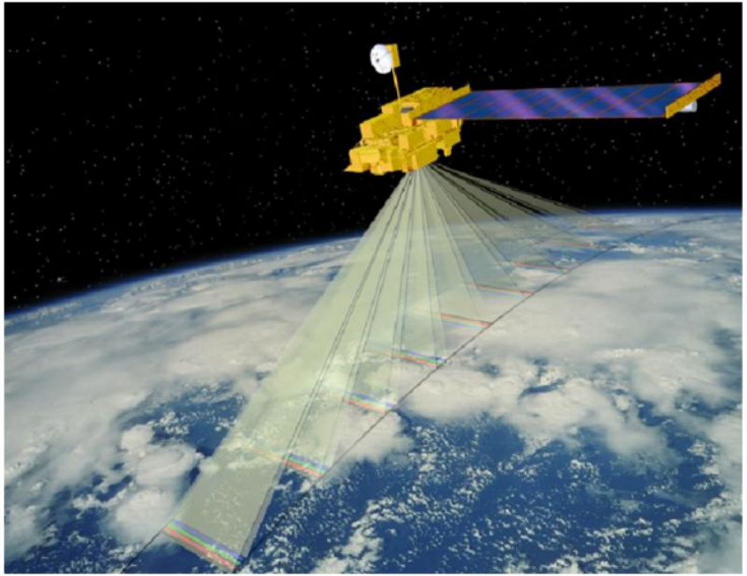
STAT 214 Spring 2025
Week 5

Austin Zane

Lab 1 Reminders

- Make sure Lab 1 submission is formatted correctly (see announcement)
- **Do not push** /data/ folder (a lot of students pushed the data for lab 0 to their repo)
- Lab reports are anonymous (no name, student id, email, etc.)
- Download, update, and push **`info.txt`**

Lab 2: Remote Sensing & Cloud Detection



Overview of Lab 2

Goal: Develop a cloud detection model using remote sensing images from NASA's Terra satellite.

Key Concepts: Transfer Learning, Autoencoders, Feature Engineering, Classification Models.

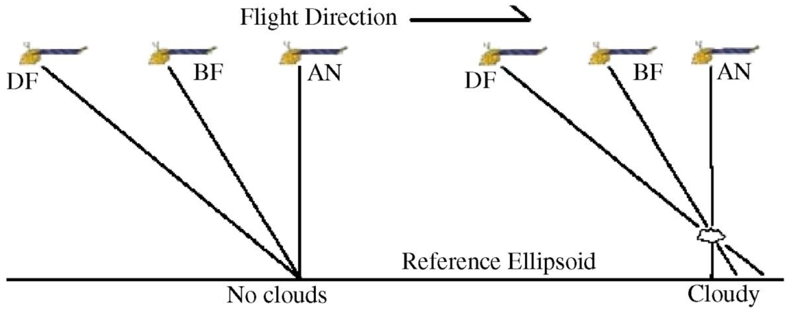
Submission Deadline: March 21st, 23:59 via GitHub.

Why is Cloud Detection Important?

Climate Models: Clouds affect global temperature by both trapping heat and reflecting sunlight.

Challenge: Clouds and ice in polar regions appear visually similar (both white and cold).

Solution: Use Multi-angle Imaging SpectroRadiometer (MISR) data to distinguish clouds based on altitude differences.



Dataset Overview

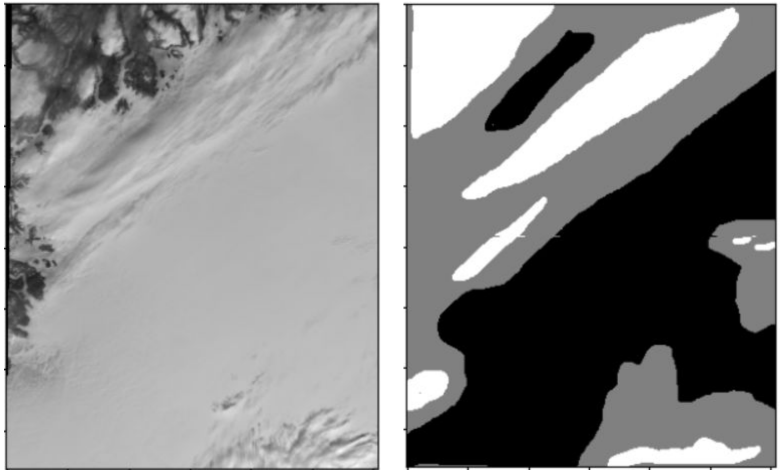
164 images from the MISR sensor (stored in .npz format).

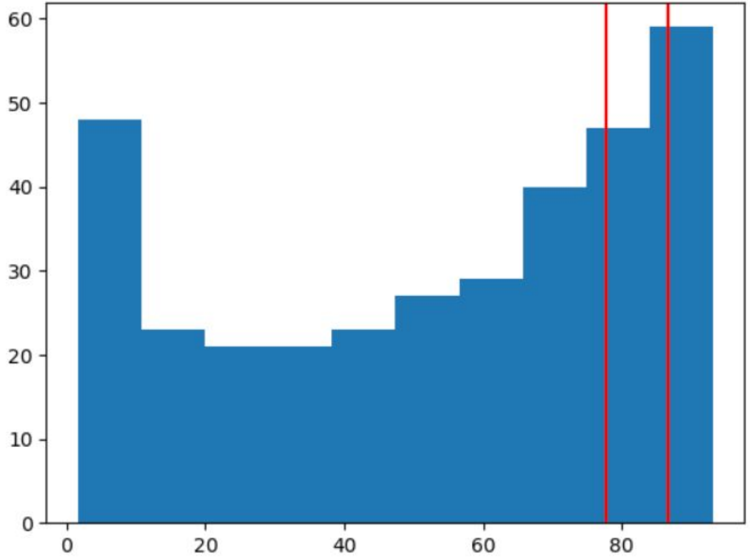
Features include:

- X, Y coordinates
- Radiance at multiple angles
- Precomputed features: NDAI, SD, CORR (derived from domain knowledge)
- **Only 3 images have expert-labeled cloud masks** (Images: 13257, 13490, 18616)
- Labels: +1 (Cloud), -1 (No Cloud), 0 (Unlabeled)

Lab Structure - Part 1: Exploratory Data Analysis (EDA)

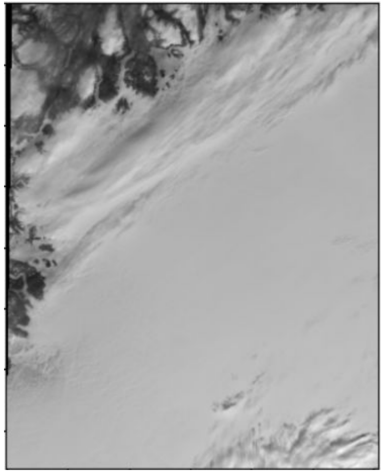
- Visualize cloud labels in the 3 labeled images.
- Examine feature relationships (e.g., radiance vs. cloud presence).
- Split data into training, validation, and test sets.

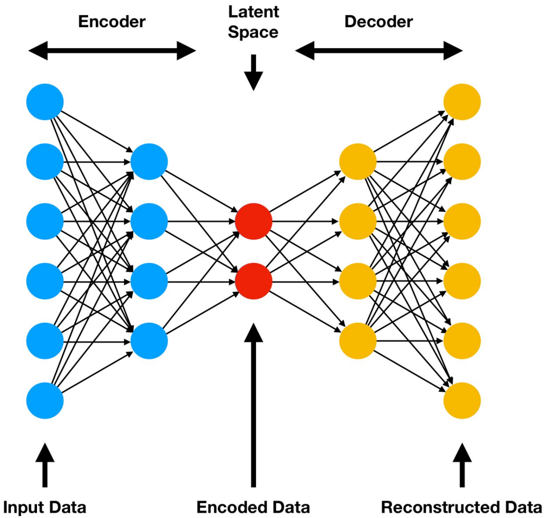


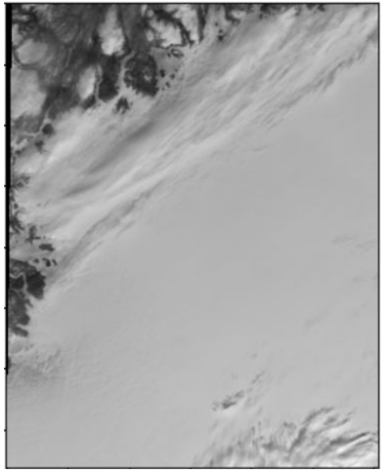


Lab Structure - Part 2: Feature Engineering

- Identify **most informative features** using statistics & visualizations.
- **Create new features** (e.g., using neighborhood pixels).
- Introduce **Transfer Learning** via autoencoders.

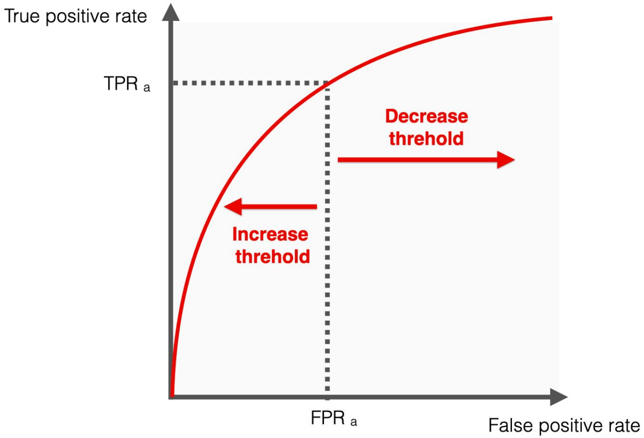






Lab Structure - Part 3: Predictive Modeling

- Train and compare **at least three** classifiers.
- Justify with visualizations and performance metrics (ROC, AIC, etc.).
- Conduct **error analysis** & evaluate real-world applicability.



What is Transfer Learning?

Definition: Training a model on one task and adapting it to another.

Our approach:

- Pre-train an autoencoder on unlabeled data.
- Fine-tune it on labeled data to extract meaningful representations.

Implementation: Modify **`autoencoder.py`**, tune hyperparameters, and test latent feature representations.

Classification Models

Train multiple models (e.g., Logistic Regression, Random Forest, Neural Networks).

Evaluate using:

- Cross-validation or data splits
- Feature importance analysis
- Error distribution & model stability tests

Peer Grading Process (*subject to change*)

Each student will peer-review two reports.

Feedback Criteria:

- Clarity & communication
- Data cleaning & preprocessing
- Visualization quality
- Reproducibility of results

Submission: Complete the Google questionnaire after grading.

Key Takeaways

Objective: Build a cloud classification model from remote sensing data.

Steps: EDA → Feature Engineering → Transfer Learning → Classification.

Emphasis: Careful data analysis, reproducibility, and scientific reasoning.

Final Submission Deadline: March 21st, 23:59 (*No Late Submissions!*)

Working with PSC Bridges-2

`stat-214-gsi/computing/psc-instructions.md`

Connecting to Bridges-2

- SSH Connection:
  - `ssh psc_username@bridges2.psc.edu`
- Web Interface:
  - Use Open OnDemand: **ondemand.bridges2.psc.edu**
- Reminder:
  - Login nodes are for file management, environment setup, and job submission only—do not run research code on them.

Setting Up Your Bridges-2 Environment

1. SSH to a login node:
   - `ssh psc_username@bridges2.psc.edu`
2. Load Conda
   - `module load anaconda3`
3. Create your environment:
   - `conda env create -f /ocean/projects/mth240012p/shared/214/environment.yaml`
4. Activate the environment:
   - `conda activate env_214`
5. Install the IPython kernel:
   - `python -m ipykernel install --user --name env_214 --display-name env_214`
6. Clean up Conda files:
   - `conda clean --all`

Testing Your Environment – Command Line

1. Start an interactive GPU job:
   - **`interact -gpu -t 00:10:00`**
2. Check GPU details:
   - **`nvidia-smi`**
3. Test PyTorch
   - Load Conda and activate your environment:
     - **`module load anaconda3`**
     - **`conda activate env_214`**
   - Start Python:
     - **`python`**
   - In the Python shell, run:
     - **`import torch`**
     - **`print(torch.version.cuda)`**
     - **`print(torch.cuda.is_available())`**

Testing Your Environment – JupyterLab

1. Launch Jupyter Lab via Bridges-2 Web Interface:
    a. Go to **https://ondemand.bridges2.psc.edu**
    b. Select "**Jupyter Lab: Bridges2**"
2. Set Parameters:
    a. Number of hours: 1
    b. Number of nodes: 1
    c. Account: mth240012p
    d. Partition: GPU-shared
    e. Extra Slurm Args: **`--gpus=1`**

Testing Your Environment – JupyterLab

- Verify:
  - In JupyterLab, create a notebook using the “env_214” kernel.
  - In a notebook cell, run:
    - **`import sys`**
    - **`print(sys.executable)`**
    - **`import torch`**
    - **`print(torch.version.cuda)`**
    - **`print(torch.cuda.is_available())`**
- Open a terminal in JupyterLab and run:
  - **`nvidia-smi`**

Avoiding Wasted GPU Resources

Development Workflow:

- Develop and test code locally.
- Use small datasets and shorter runs for debugging.
- Transfer code to Bridges-2 for production runs.

Resource Management:

- Use short time limits on jobs.
- Ensure jobs terminate when complete.
- Shut down idle interactive sessions.

Always shut down your Jupyter Lab server when finished (via File → Shut Down or through the web dashboard).

Only use for interactive experimentation, not for production runs.

Final Reminders

- Use the Right Partition:
  - **GPU-shared** for 1 GPU; reserve **GPU** for jobs that truly need 8 GPUs.
- Monitor Usage:
  - Check job status with **`squeue -u your_username`** and GPU usage with **`nvidia-smi`**.
- Class-Related Work Only:
  - Bridges-2 is for work directly related to this class.
- Need Help?
  - Contact a GSI if issues arise.