

Project 2: Presentation

STAT GU4243

Applied Data Science

Cynthia Rush
Columbia University

February 28, 2018

PROJECT DESCRIPTION

- ▶ Carry out model assessment and selection for image classification.
- ▶ Evaluate different modeling/analysis strategies and decide what's best.
- ▶ Present sound evidence in the form of model assessment, validation, and comparison.
- ▶ Communicate your decision and supporting evidence clearly and convincingly in an accessible fashion.

MODEL COMPARISON

Baseline Model

- ▶ Use provided SIFT descriptors as features.
- ▶ Implement a gradient boosting machine on decision stumps.

Task 1 will be implementing the above strategy and tuning it correctly

Proposed Strategy

- ▶ Consider better features and better models
- ▶ Implement structured comparison to establish the value added by new features and new methods

SUBMISSION OF PROJECT 2

- ▶ A well-documented GitHub repo (following starter codes instruction).
 - ▶ Your README files should explain how the Github is organized.
 - ▶ Subfolder READMEs should documents the files in the subfolder and how they are used in the project.
- ▶ A `main.rmd` that carries out the project.

`main.rmd` uses `feature.R`

(1) A file of feature processing codes (`feature.R`):

- ▶ Takes as input folder of images
- ▶ Outputs a folder of “feature” objects with features for the images
 - ▶ Format is `RData`, or other R readable file
 - ▶ Make sure you keep track of the file names of the images.

SUBMISSION OF PROJECT 2

`main.rmd` uses `train.R`

(2) A file of training codes (`train.R`):

- ▶ Inputs a path for training image features
- ▶ Inputs a file containing training image names and labels
- ▶ Outputs trained classifiers (in the form of RData, or other R readable file); One for the baseline model and one for the new model.

Note that model training should include any necessary parameter tuning.

`main.rmd` uses `test.R`

(3) A file of testing codes (`test.R`):

- ▶ Inputs a path for test image features
- ▶ Inputs a trained classifier from the output of `train.R`
- ▶ Output predicted labels

PROJECT 2 SUBMISSION

On Monday

- ▶ We'll fork all project repos to save a time-stamped version of your code
- ▶ You'll be given 1850 test images (no label) and SIFT features
- ▶ Run your `feature.R` (or `feature.py`, whatever)
- ▶ Run `test.R` to give test image prediction
- ▶ Time limit 30 mins
- ▶ We'll ask you to submit test image classifications using your advanced and baseline model

`train.R` is prepared before Monday

- ▶ Can use any methods to generate features
- ▶ Want base and advanced model **saved in your repo so we can access them**
- ▶ Training model can take > 30 mins (you won't do this again in class)
- ▶ Include any tuning step here

SUBMISSION OF PROJECT 2

You should also prepare a presentation (12 min/group) for this project

- ▶ Methodology details
 - ▶ How did you perform model selection?
 - ▶ How did you perform model assessment?
- ▶ Model details
 - ▶ What features are you using?
 - ▶ What classifier are you using?
 - ▶ What do you estimate to be its accuracy?
- ▶ Performance comparison between baseline and new model
 - ▶ Time/cost analysis.

Some more info

- ▶ Not everyone has to be 'on stage' during the presentation.
- ▶ Can use Powerpoint or other tools.
- ▶ I will let you know the order of the presentations on Sunday.

EVALUATION OF PROJECT 2

Ease of reproducibility by the client (5 points)

- ▶ Are codes for the proposed methods well-annotated and documented?
- ▶ Can the analysis be re-run nearly automatically using the ‘main.rmd’?

Level of reproducibility (5 points)

- ▶ Can the client derive the same evaluation conclusion as presented in the team’s final presentation?
- ▶ How close are the reported performances (presentation and online) to the reproduced performances?

EVALUATION OF PROJECT 2

Portability of proposed strategies (5 points)

- ▶ Computational speed for feature extraction and model training.
- ▶ Computational speed for prediction.
- ▶ Memory use for model training and prediction.

Presentation and organization (5 points)

- ▶ Is the the intuition behind the proposed strategies convincing?
- ▶ Is it supported by adequate and appropriate evidence?
- ▶ Is the GitHub organized and prepared so that it's easy to understand the proposed strategies and their advantages and limitations?