Project 1: Digit Classification and Model Analysis

Machine Learning

Quanshi Zhang

Objective

- The objective is to help you have an deep insight on current models, including the intuitive understanding of the feature and the model, and the discrimination power of different models.
  - Implement digit classification using different methods
  - Visualize features
  - Analyze models
  - Discuss advantages and disadvantages of different methods
    - Design your own experiments to prove them.
  - Write a report

Objective

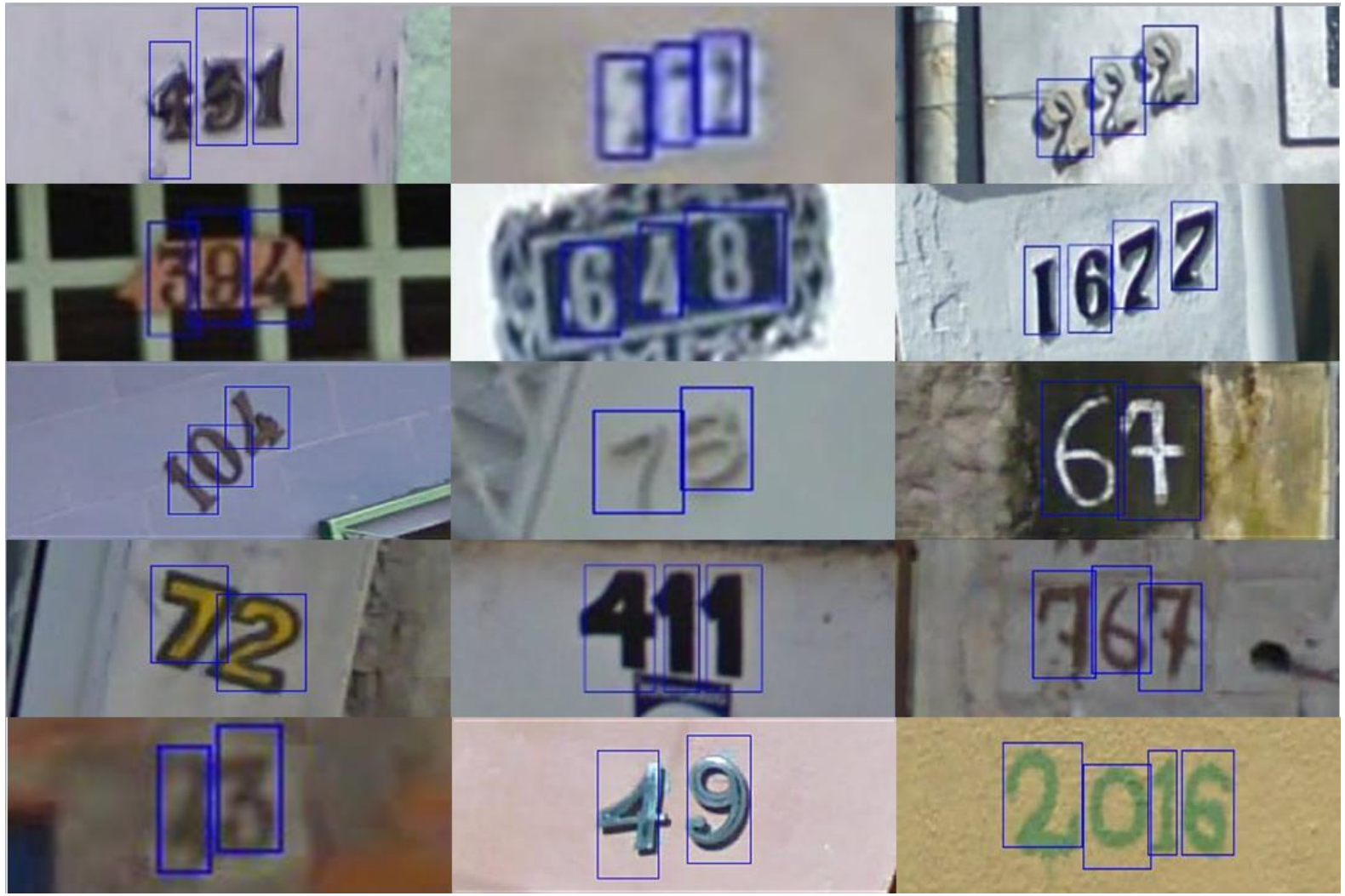
- As long as your code does not have a bug (e.g., the classification accuracy is significantly lower than usually), I do not care about the accuracy. It is not a competition of the accuracy.
- Instead, I hope you can
  - Have deep understanding of basic machine-learning techniques.
  - Learn how to design an experiment to prove or verify your ideas.
    - I.e. learn how to ensure the scientific rigor and how to obtain convincing results, when you design your experiments.
  - Propose new hypotheses or your own understanding of some machine-learning methods, and try to verify your opinions through experiments.
    - NOTE THAT “whether your hypotheses themselves are correct or not” is not the most important. Instead, I am interested in the process how you prove them correct or incorrect.

Dataset: The SVHN Database of handwritten digits

- You need to download the database.
  - http://ufldl.stanford.edu/housenumbers/
  - SVHN is a real-world image dataset for developing machine learning and object recognition algorithms with minimal requirement on data preprocessing and formatting. It can be seen as similar in flavor to MNIST (e.g., the images are of small cropped digits), but incorporates an order of magnitude more labeled data (over 600,000 digit images) and comes from a significantly harder, unsolved, real world problem (recognizing digits and numbers in natural scene images). SVHN is obtained from house numbers in Google Street View images.

  - It is OK to use a subset of training images, if you do not have a powerful computer.
    - For example, you can "RANDOMLY" sample 10000 examples from the all training samples for training.
    - Please crop the digit number using its bounding box and resize the image to 32x32 pixels.
    - You must clarify the size of your own training set in the report.

Digits in the SVHN database



Your task: classification

- You can learn the following classifiers
  - Logistic regression
  - LDA
  - SVM (Linear and non-linear with various kernels)
  - Logistic regression + Ridge loss
  - Logistic regression + LASSO loss
  - Kernel-based logistic regression + LASSO loss
  - Neural networks (with various layer numbers, various architectures, various losses)
  - Gaussian mixture models (EM methods)
    - You may use deep features from neural networks to conduct GMM
  - … …
- For the binary classifier, the 10-category classification can be implemented using 10 models of single-category classification.
- For neural networks, the 10-category classification needs to be implemented using the softmax operation, which simultaneously classifies 10 categories.
- You must write your own codes to implement these models.
  - Except for the SVM. You may use existing tool packages to implement SVMs.

Your task (optional): generative models

- (Optional) You may learn GANs, VAEs, or other generative models to synthesize digit images.

If you do not have a powerful computer to learn a DNN

- You may use learn a small DNN.
- You may reduce the number of layers.
- You may reduce the number of channels in each layer.
- … …
- Anyway, I do not care about the accuracy, as long as there is not a strange accuracy that indicates a bug.
- However, you must clearly introduce the network architecture.

If you have a powerful computer to learn a DNN

- All standard network architectures for CIFAR-10 images (also 32x32 pixels) can be applied to this project.
  - You may read papers and do a survey on the network architecture, but you do NOT need to try all architectures. Just learn >=2 DNNs.

What you need to write in the report

- You need to write how you implement these models
- You need to visualize features
  - I will introduce methods for feature visualization and network diagnosis in future classes.
- You need to diagnose the logic of the model based on the learned parameters.
- You need to design your own experiments to verify some conclusions introduced in the class.
  - Introduce how to design your experiments
  - Whether the experiment is fair or not
  - Is the conclusion verified or not? Why? Or why not?

What you need to write in the report

- You need to propose your own hypotheses on some techniques and design experiments to verify them
  - How to design your experiments
  - Whether the experiment is fair or not
  - Is the conclusion verified or not? Why? Or why not?

- **I also encourage you to conduct some extended experiments. You may implement techniques not taught in the class or conduct new tasks based on the SVHN database, in order to get better scores.**

- The deep understanding and experimental verification are the most important.

Accuracy/success is NOT what I want

- Honesty
  - Report the true number in your experiments, even if they may not look good.
  - Do not try to cheat in the accuracy number.
  - Do not try to cheat in order to prove more conclusions than you can prove.
  - Facts are always more important than the book. If the book says “A,” but you get “B” in experiments. Honestly report your experimental results. The book just introduces a general conclusion, which is not necessarily valid for all special cases.

Accuracy/success is NOT what I want

- **If I find anyone cheating in the report, he/she will get 0 in Project 1.**
  - Nobody will be exceptional.
  - Please do not try to make your results “too good to be true.” Take care!

- Although you do not need to submit your codes, please do not delete your codes before July. If your results are suspicious, I will require you to submit your code to check whether results in your report are real or not.
  - If the suspicious result is caused by a bug, it will not be a big problem. Thus, **do not worry** if your strange results are caused by bugs.

Workload

- I do not require you to implement all tasks, which I mentioned in previous slides.
- You may focus on a few tasks. Anyway, I encourage you to do more work to get better scores in this project.

- Compared to learning more models, I suggest you design experiments to explore/verify more hypotheses. The deep understanding of current techniques is more important than the large number of techniques.

Submit a report

- Code & report
  - **Do not copy others' codes or others' reports**
    - **The teaching assistant will check the code and the report.**
  - **Make sure your code can run**
    - **I will randomly run codes of 30% students.**
  - **Do not create numbers without writing a code**
- Deadline
  - Friday in the 17th week
- Please ask questions if you have.

- **Take it easy. The purpose of this project is not to distinguish bad and good students. Have fun!**