Demo: Convolutional Neural Network (CNN)
SAS Viya

Dr. Goutam Chakraborty

SAS® Professor of Marketing Analytics

Director of MS in Business Analytics and Data Science* (http://analytics.okstate.edu/mban/)
Director of Graduate Certificate in Business Data Mining (http://analytics.okstate.edu/certificate/grad-data-mining/)
Director of Graduate Certificate in Marketing Analytics (http://analytics.okstate.edu/certificate/grad-marketing-analytics/)

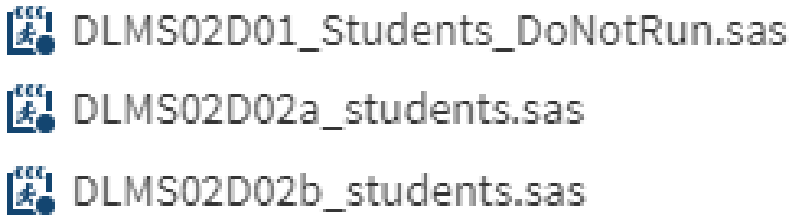
- *Name change pending internal approval.
- Note some of these slides are copyrighted by SAS® and used with permission. Reuse or redistribution is prohibited.
- Some of the slides were developed by Mr. Sanjoy Dey, used with permission.

Outline

- Setting up data folders for CNN in SAS Viya
- Running CNN in SAS Viya
- Scoring CNN in SAS Viya

Setting Up Data Folder in SAS Viya for CNN

- For this class, we are loading all images to CAS public library for you.
  - So, please do not copy images in your SAS drive
- Images need to be in separate sub-folders for each class.
  - For example, for 2-class images (such as airplane and bird), we need 2 sub-folders containing images of each class
- The demo program is set for a 10-class problem.
  - But, I am using a subset of 2-class problem because OSU's SAS server does not have GPU.
- Codefiles (students do not run the first one):



Results: DLMS02D01_Students_DoNotRun

- This code sets the data up for CAS public library, partitions it in 80 (train/validation) 20 (holdout for scoring) and then shuffles the images before being passed on to CNN modeling

The PARTITION Procedure

Stratified Sampling Frequency				
Index	_label_	Number of Obs	Sample Size 1	Sample Size 2
0	Airplane	1000	800	200
1	Automobile	1000	800	200

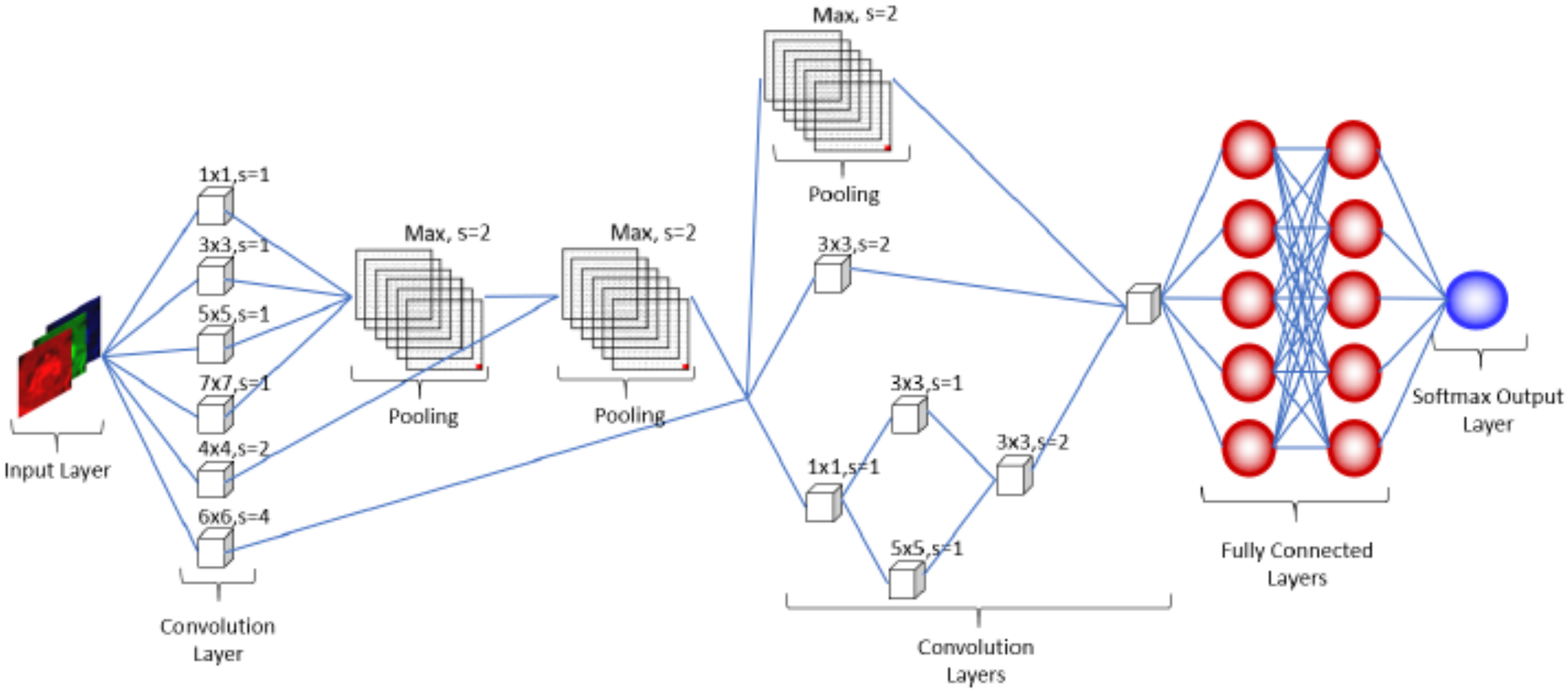
Output CAS Tables			
CAS Library	Name	Number of Rows	Number of Columns
Public	SMALLIMAGEDATA	2000	11

DLMS02D02a

- Students can run it
- Setting up CNN layers such as:
  - Multiple convolutional layers
  - Concatenation layers
  - Max pooling layer
  - A fully-connected later with batch normalization
- Estimating weights using ADAM optimizer and options

CNN Model Architecture

1. First, an image of the model architecture is printed.



DLMS02D02b

- Students can run it
- Scoring holdout data using weights trained in previous program

Results from deepLearn.dlScore

Score Information for SMALLIMAGEDATASHUFFLED	
Number of Observations Read	1600
Number of Observations Used	1600
Misclassification Error (%)	4.5625
Loss Error	0.119615

Output CAS Tables

CAS Library	Name	Number of Rows	Number of Columns
imagelib	Layer_data	1600	1
CASUSER(gchakra)	ScoredData	1600	6