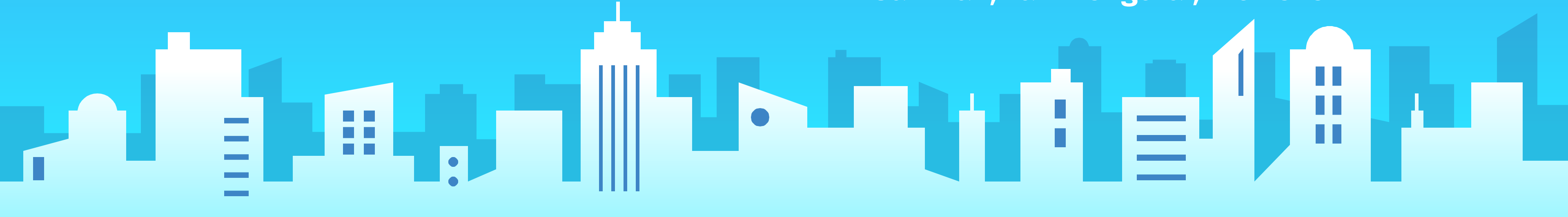
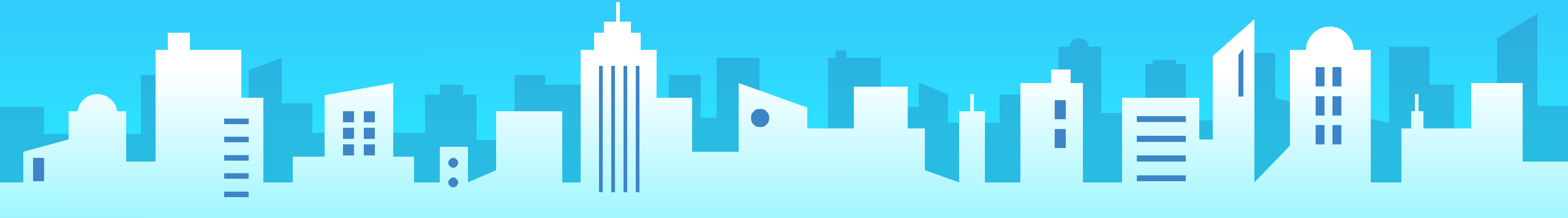
Creating Data w/ GANs

Sam Yan, Tai Thongthai, Rich Chen



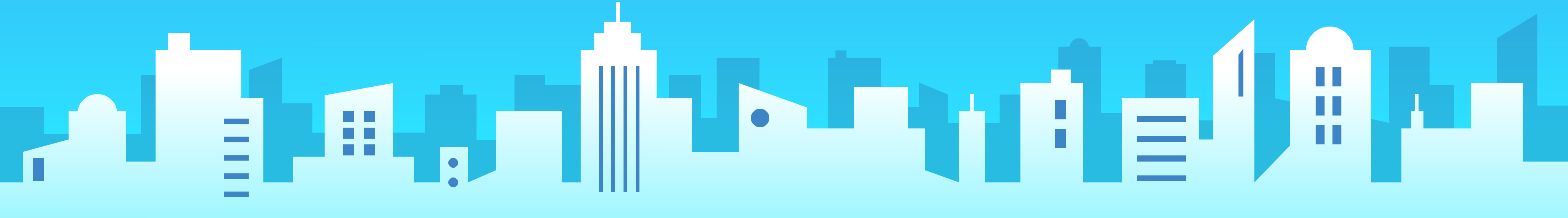
Sign Post

1. Motivation & Research Questions
2. Dataset Choice and Preliminary GANs Showcasing
3. Training Classifiers
4. Results and Interpretations
5. Conclusions and Future Work

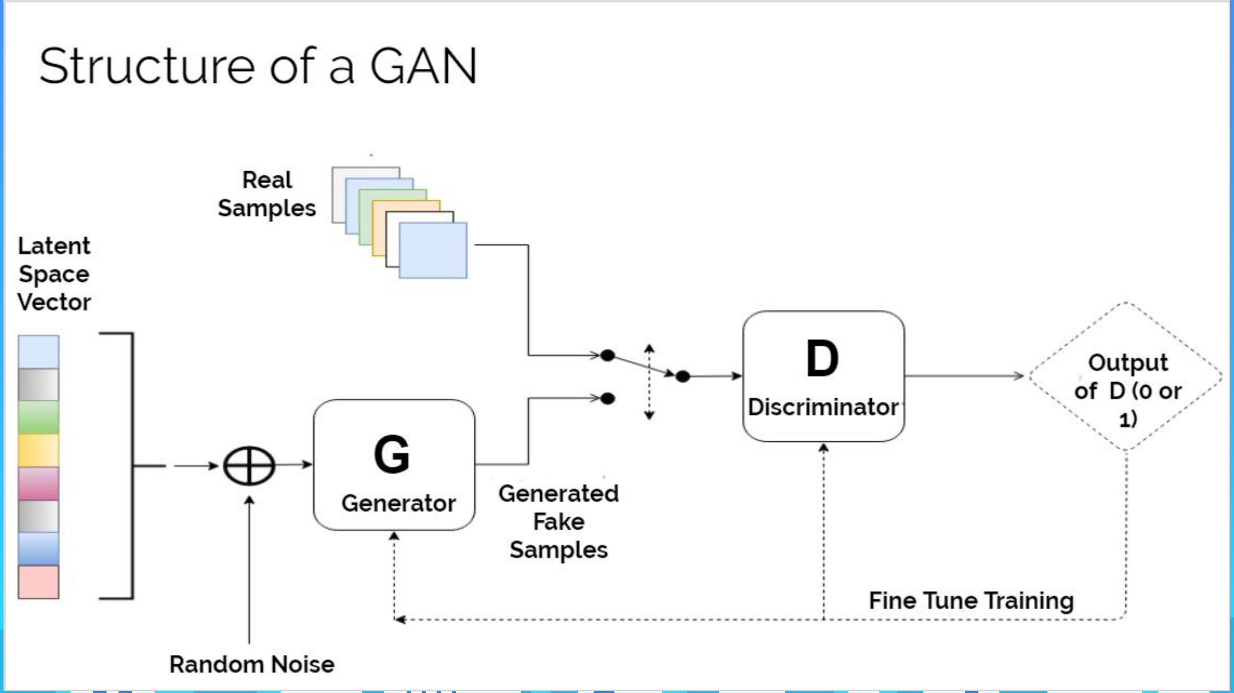


Motivation:

“Problems with Not Enough or Missing Data”



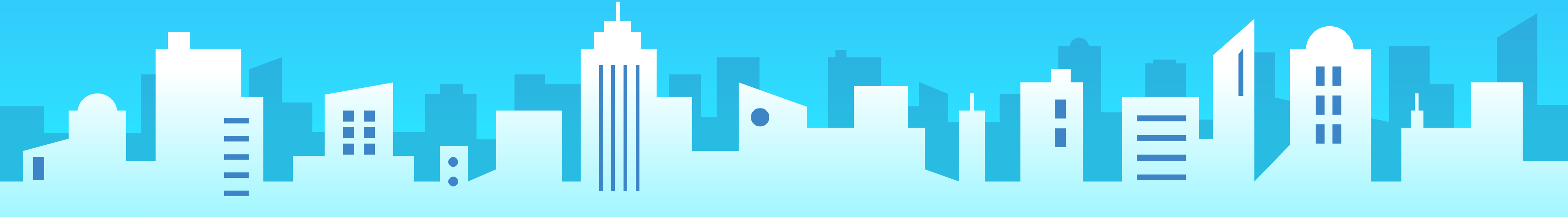
So What Exactly is GANs?



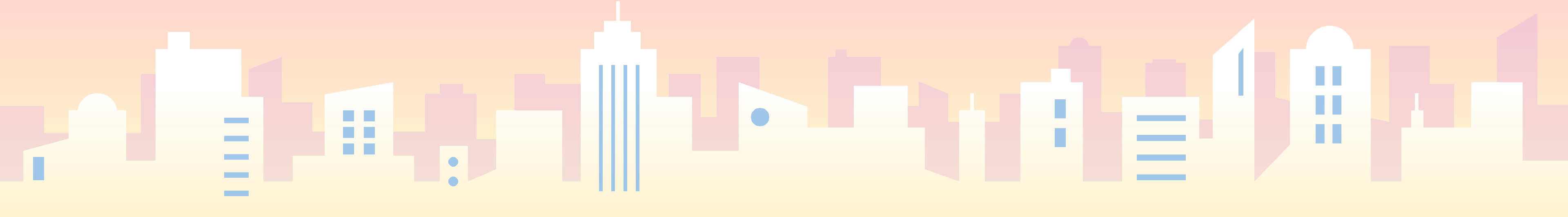
Research Question

"Can we successfully create dataset that will 'trick' a classifier?"

- Analyze how effective GANs are at mimicking real data



Data and Methods



Just a Bit About Library Used: KERAS

- A user friendly library for FNNs and CNNs
- Works like an assembly line!
  - Declaration of model
    - model = Sequential()
  - Adding Layers
    - model.add(Dense(…))
  - Adding Activation Functions
    - model.add(LeakyReLU(…))

Datasets

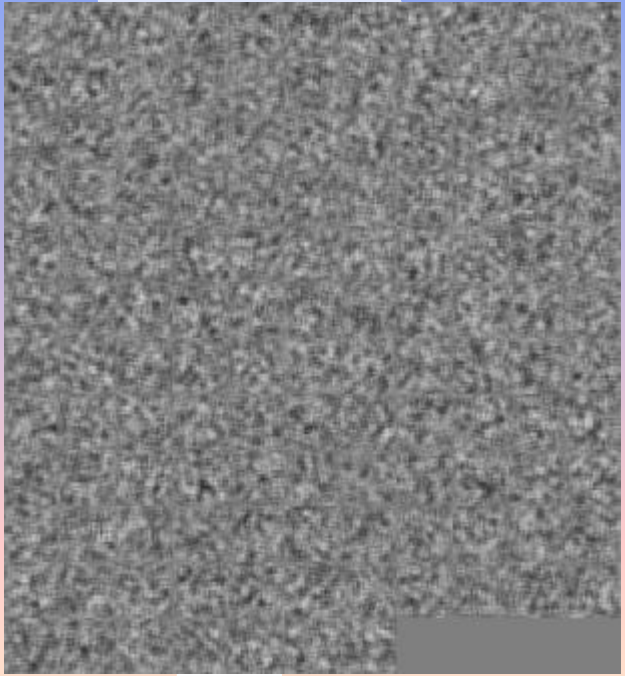
MNIST

- 28 x 28 pixels
- Black and White
- 60,000 training examples

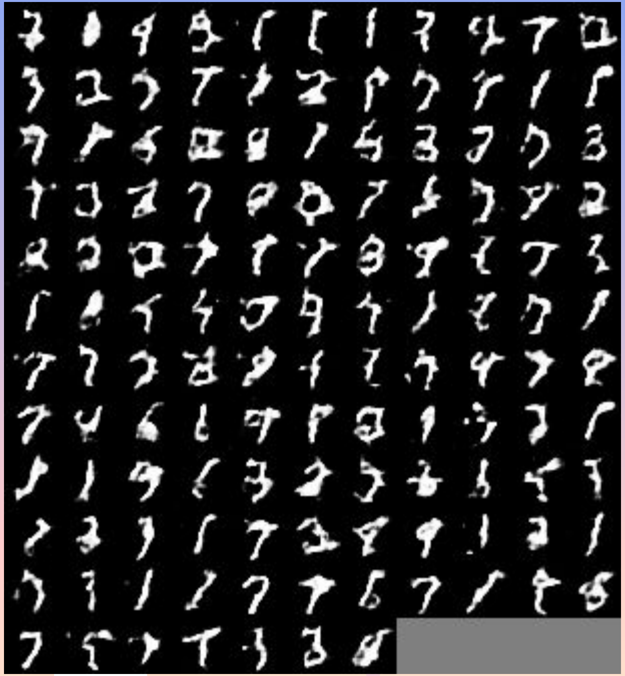
Cifar-10

- 32 x 32 pixels
- Colored (RGB)
- 60,000 training examples

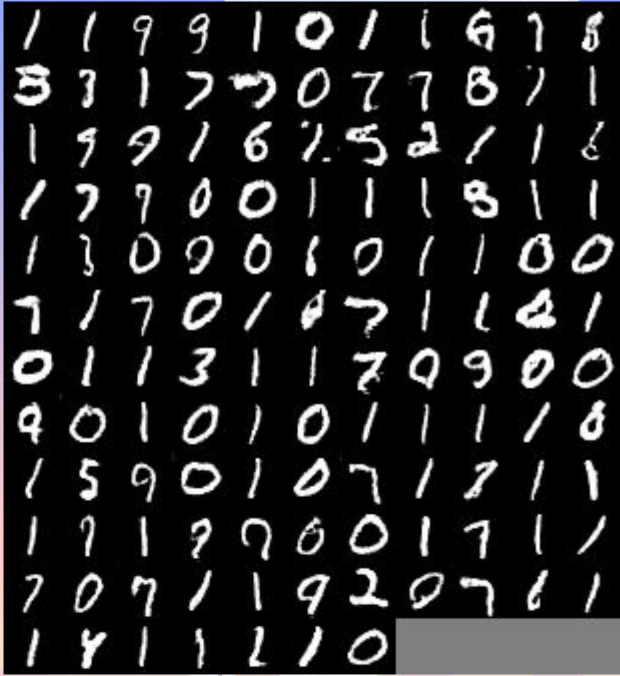
MNIST Results:



1st Epoch



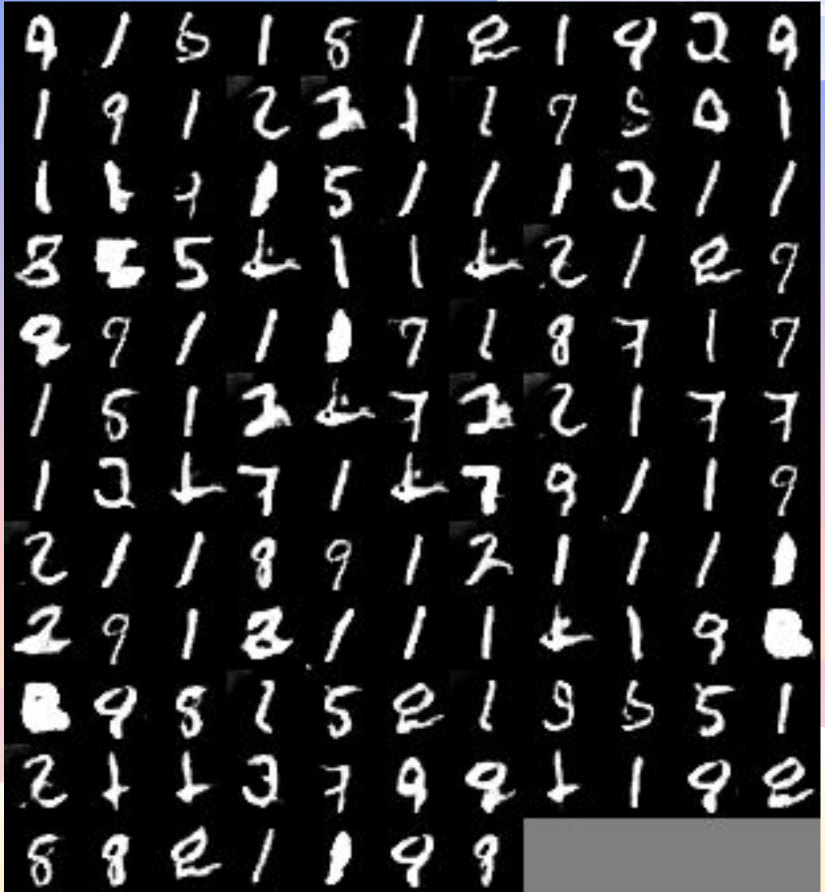
10th Epoch



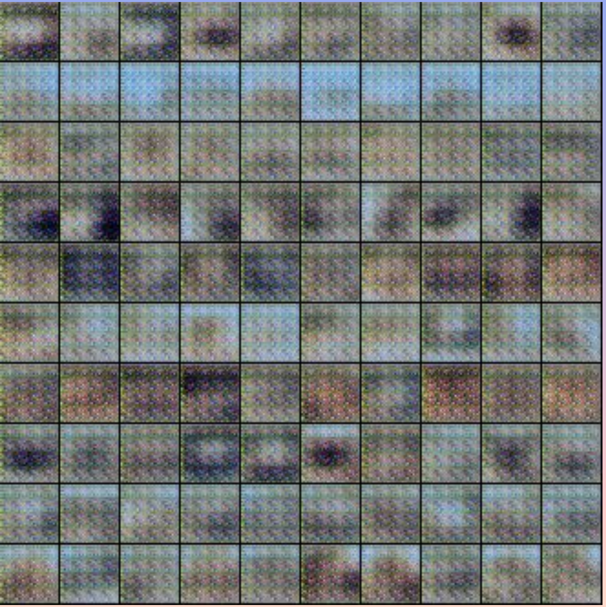
100th Epoch

MNIST Results: (cont.)

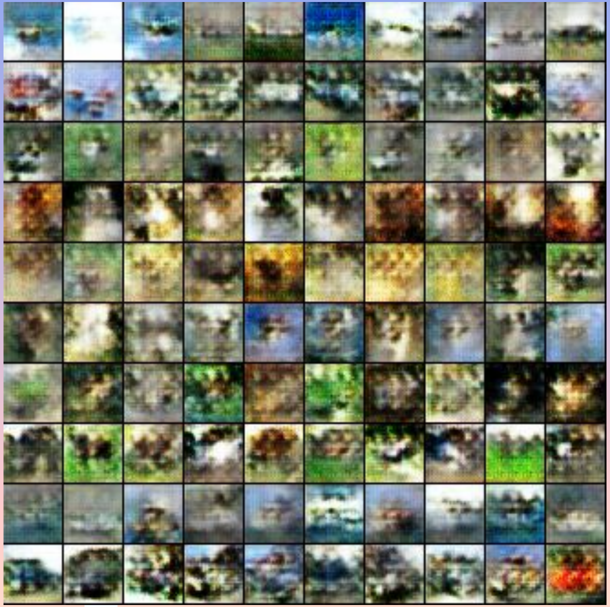
1000th (and Final) Epoch:



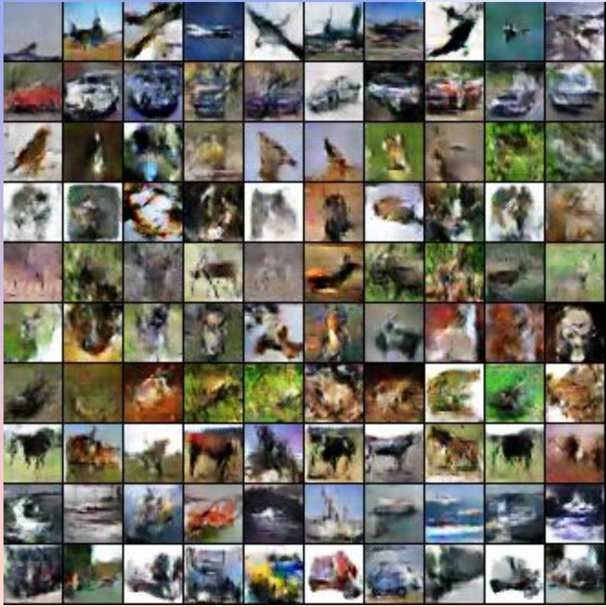
Cifar-10 Results:



1st Epoch

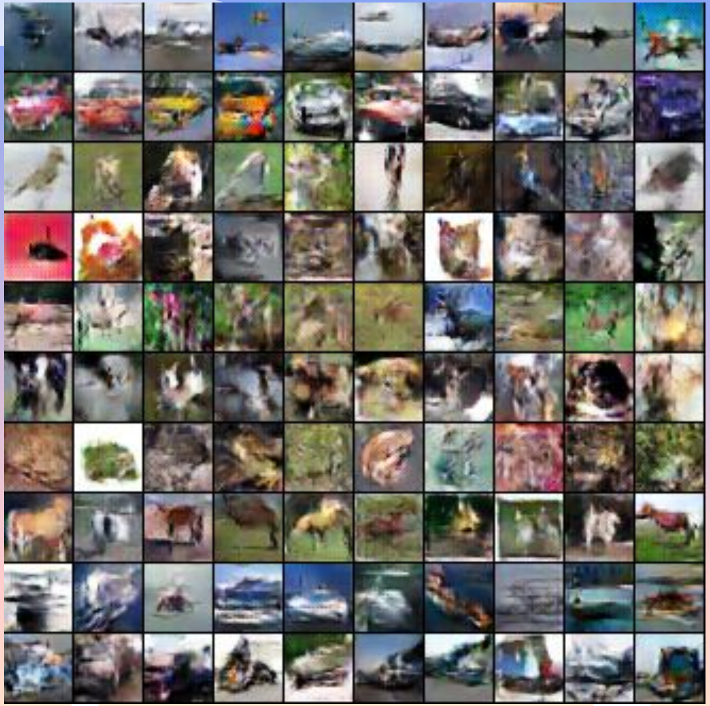


10th Epoch

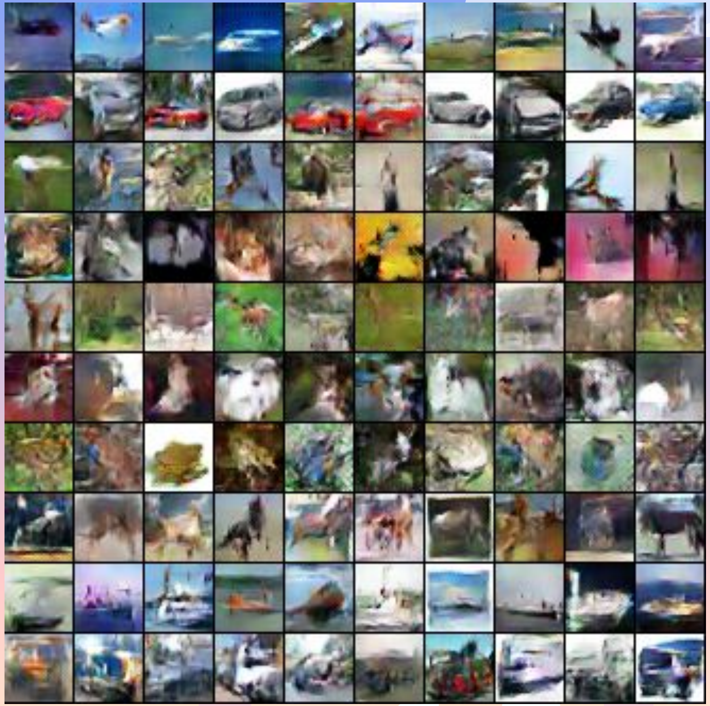


100th Epoch

Cifar-10 Results: (cont.)



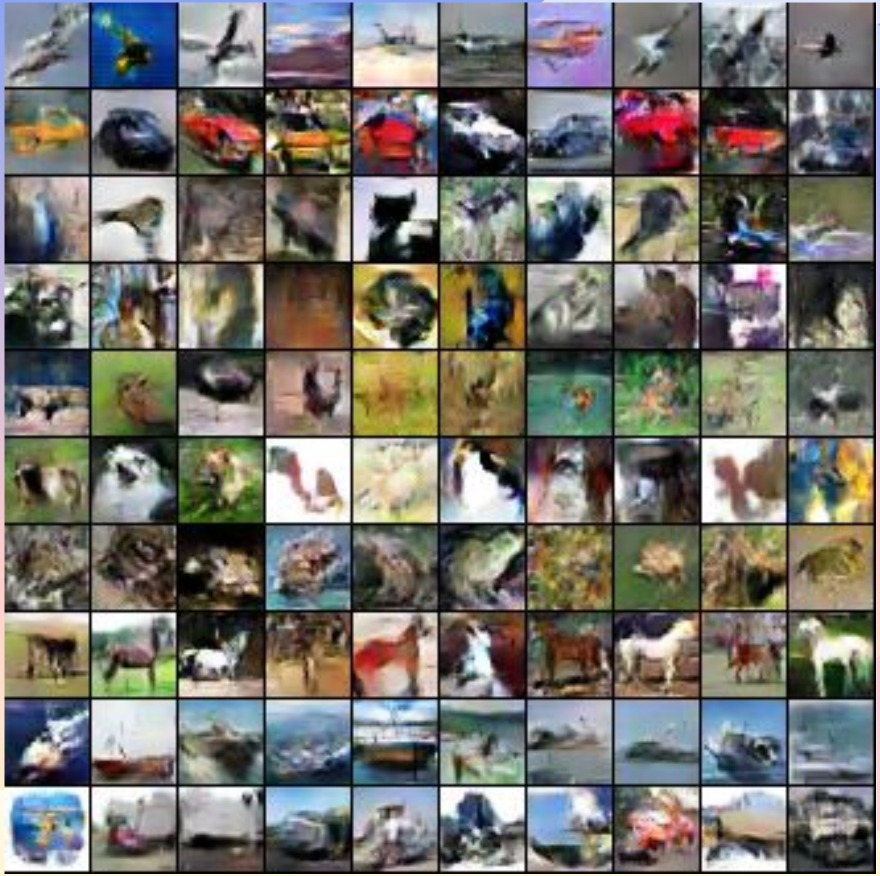
250th Epoch



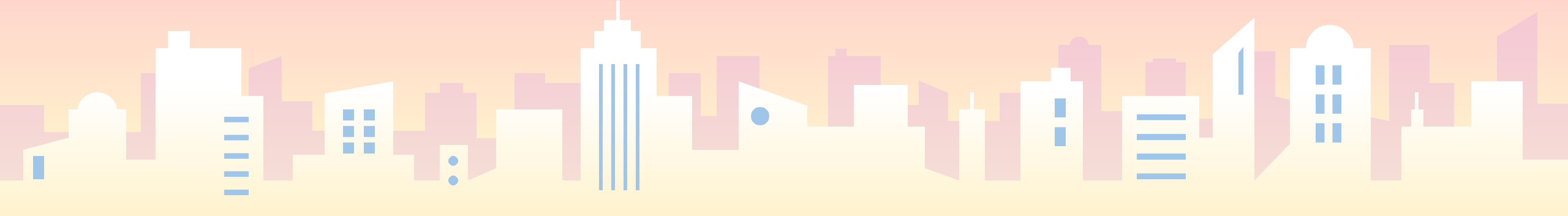
500th Epoch

Cifar-10 Results: (cont.)

1000th (and Final) Epoch:



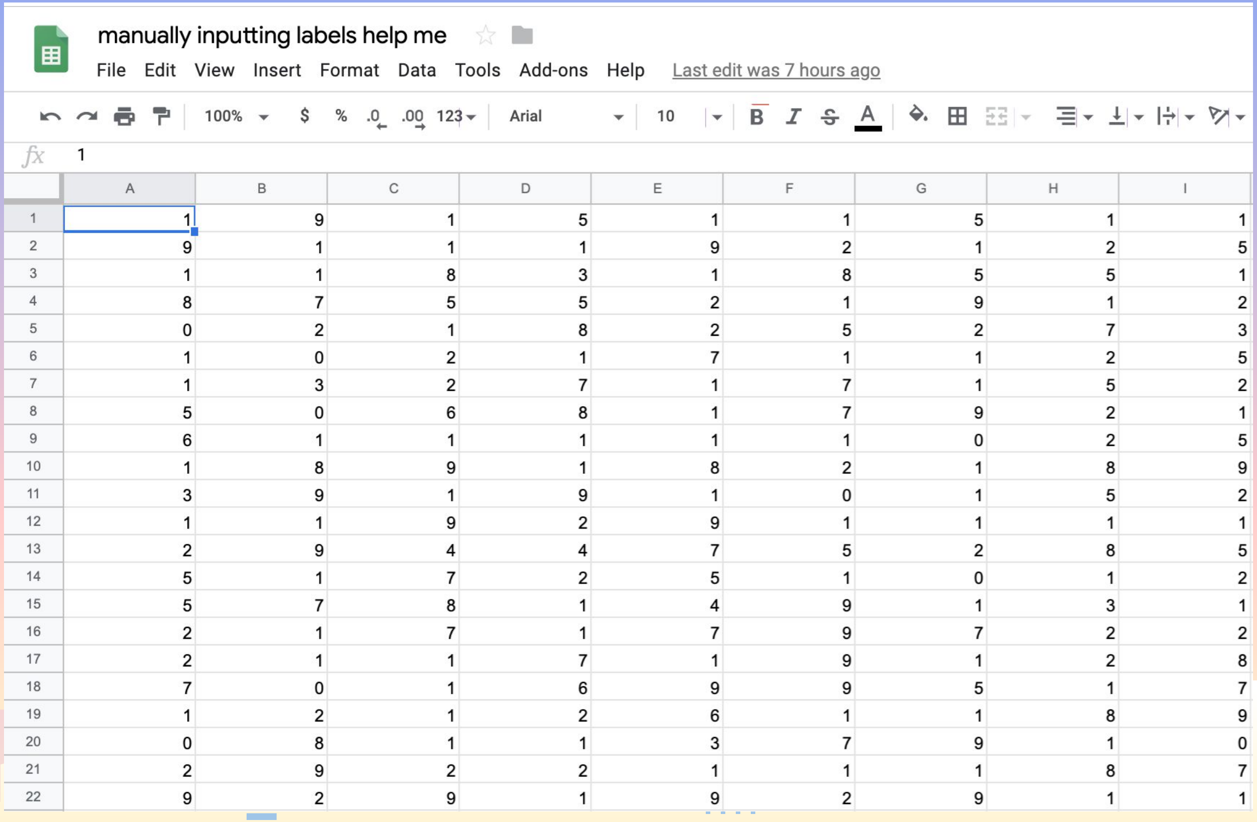
Training Classifiers



MNIST

- Used and modified CNN code from a public git repo
- N = 1408
- Num_Labels = 10

MNIST Labels

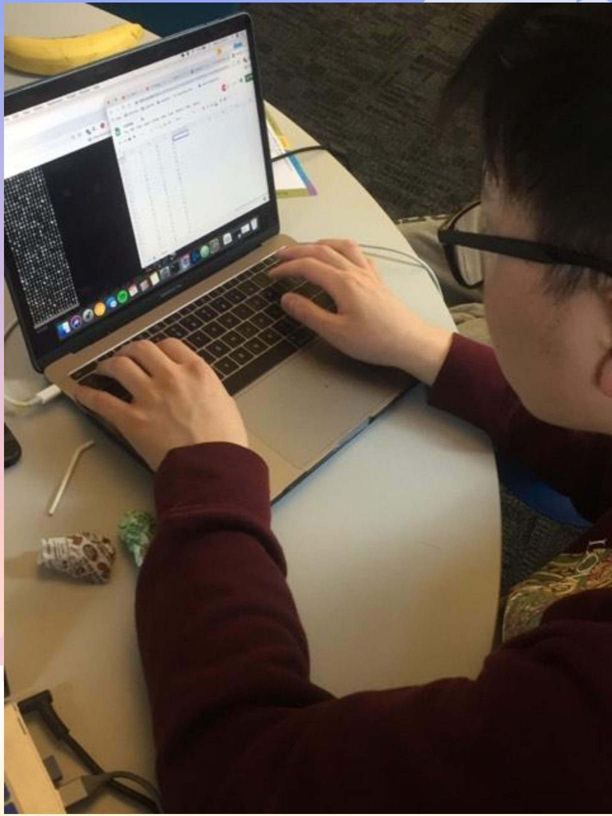


manually inputting labels help me

Last edit was 7 hours ago

fx 1

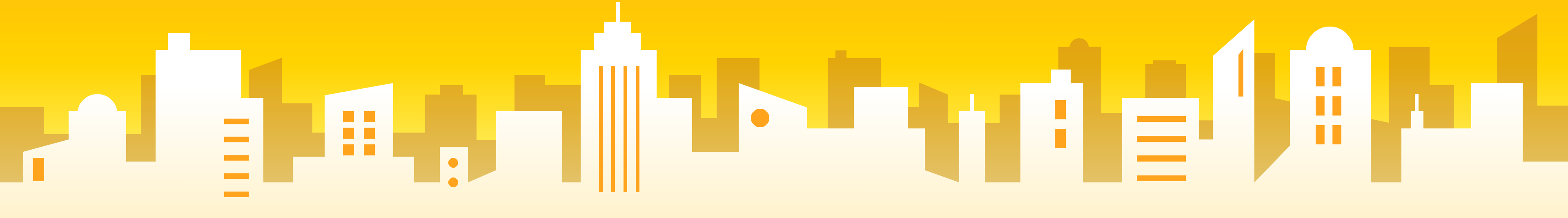
	A	B	C	D	E	F	G	H	I
1	1	9	1	5	1	1	5	1	1
2	9	1	1	1	9	2	1	2	5
3	1	1	8	3	1	8	5	5	1
4	8	7	5	5	2	1	9	1	2
5	0	2	1	8	2	5	2	7	3
6	1	0	2	1	7	1	1	2	5
7	1	3	2	7	1	7	1	5	2
8	5	0	6	8	1	7	9	2	1
9	6	1	1	1	1	1	0	2	5
10	1	8	9	1	8	2	1	8	9
11	3	9	1	9	1	0	1	5	2
12	1	1	9	2	9	1	1	1	1
13	2	9	4	4	7	5	2	8	5
14	5	1	7	2	5	1	0	1	2
15	5	7	8	1	4	9	1	3	1
16	2	1	7	1	7	9	7	2	2
17	2	1	1	7	1	9	1	2	8
18	7	0	1	6	9	9	5	1	7
19	1	2	1	2	6	1	1	8	9
20	0	8	1	1	3	7	9	1	0
21	2	9	2	2	1	1	1	8	7
22	9	2	9	1	9	2	9	1	1



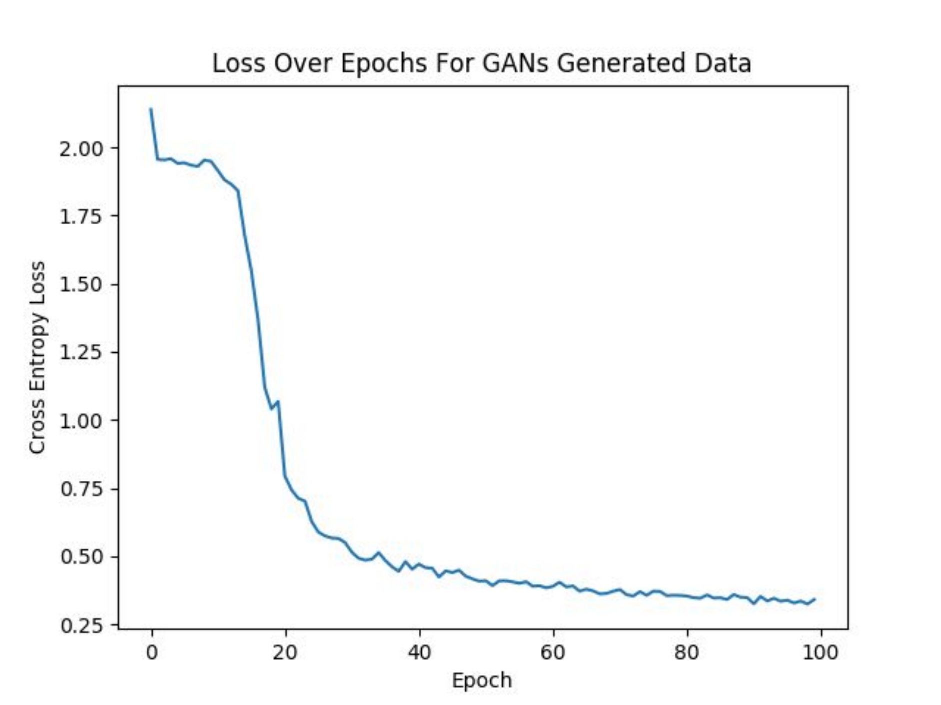
Cifar-10

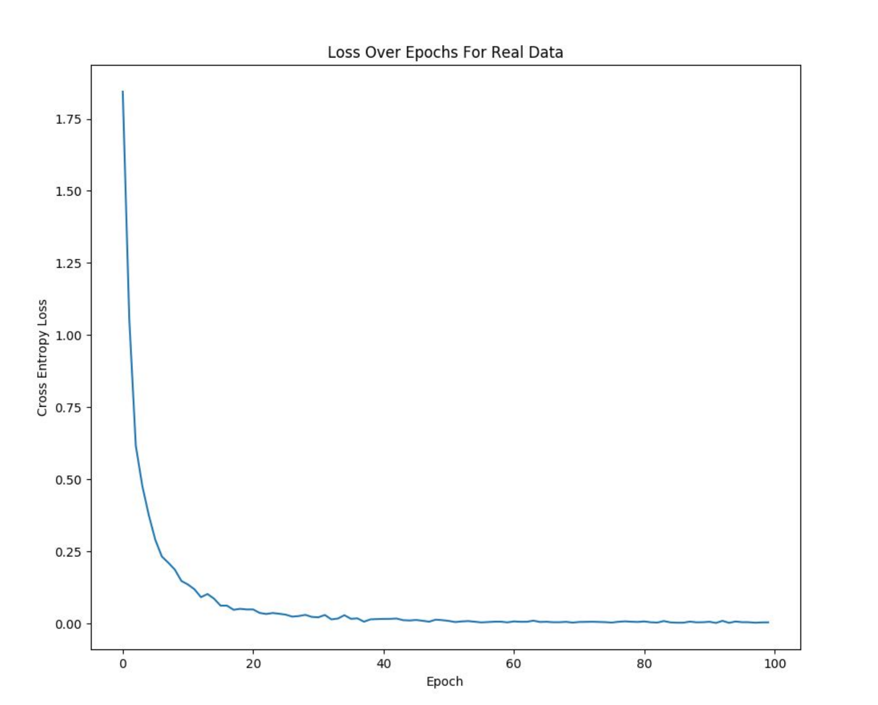
- Expanded upon code from Lab07
- Used generated Data to Train
- Tested on Real Cifar-10 Data
    - Want to see how it would compare to training on real Cifar-10 data

Results and Interpretations

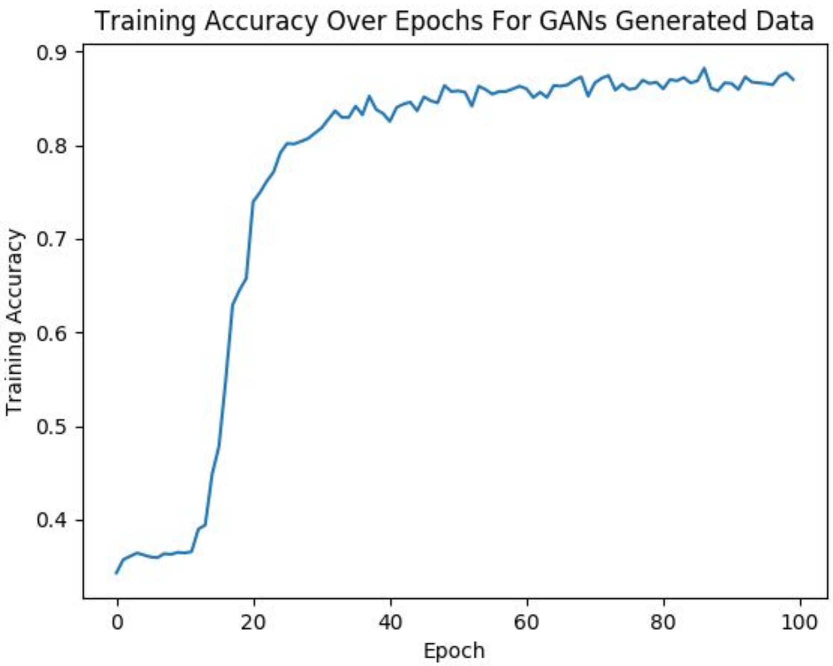


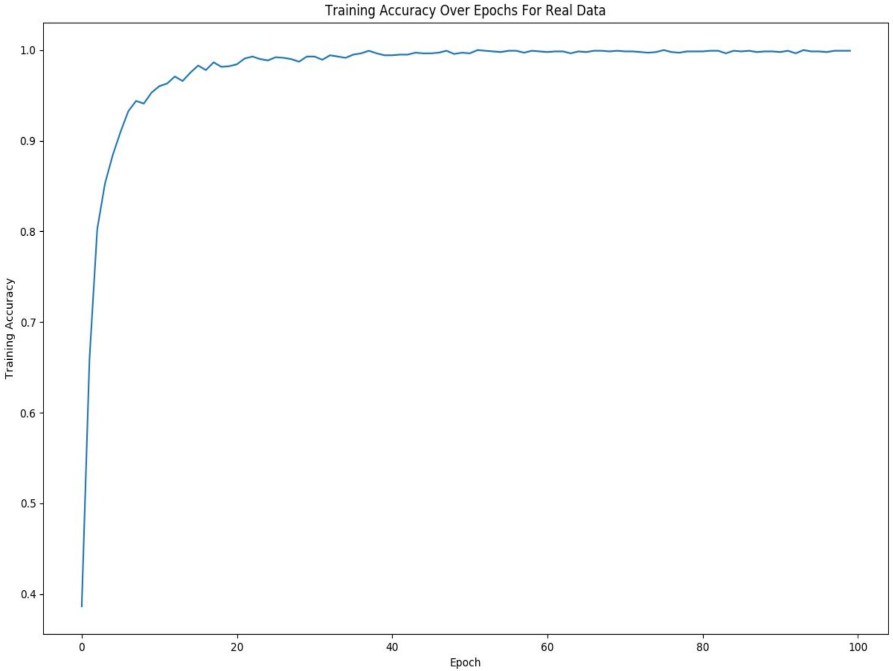
MNIST Training Losses: Fake Data vs Real Data



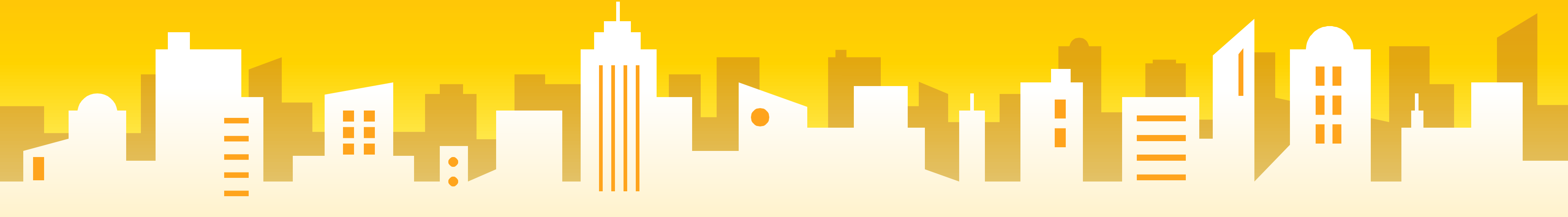


MNIST Training Accuracies: Fake Data vs Real Data





So how good is GANs generated data on training classifiers?



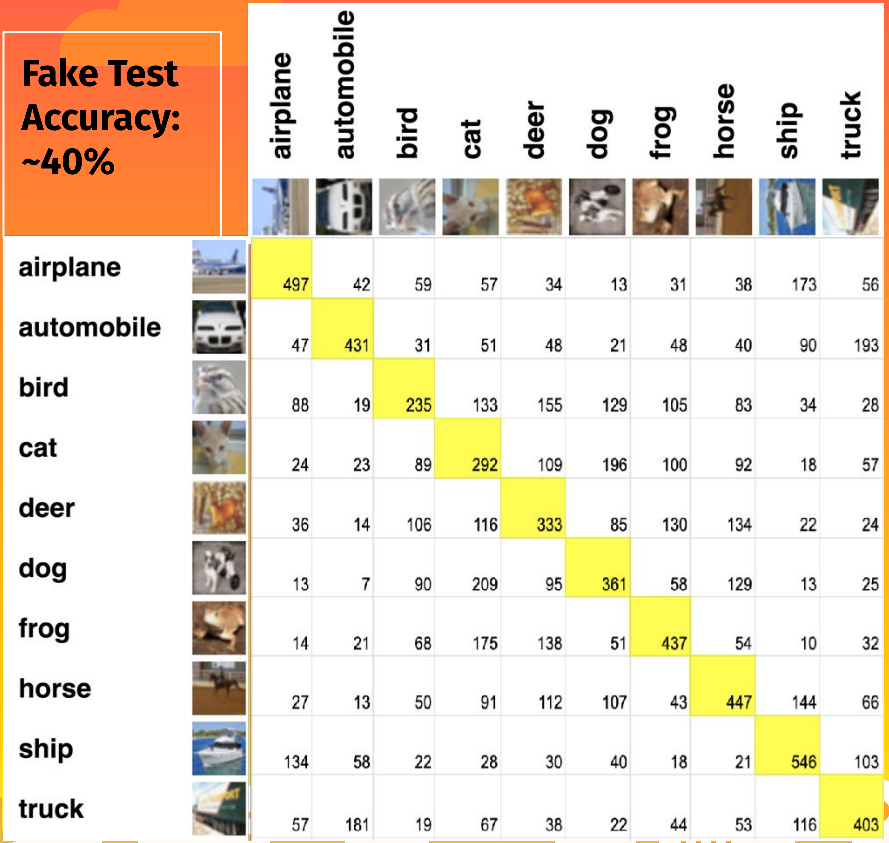
Its straight up bad dude..

Classifying Generated v Real

	Test Accuracy	
Epochs	GANs Generated	Real Data
12	~15%	>93%
100	~49%	>93%
500	~51%	>93%

CIFAR: Fake Data vs Real Data

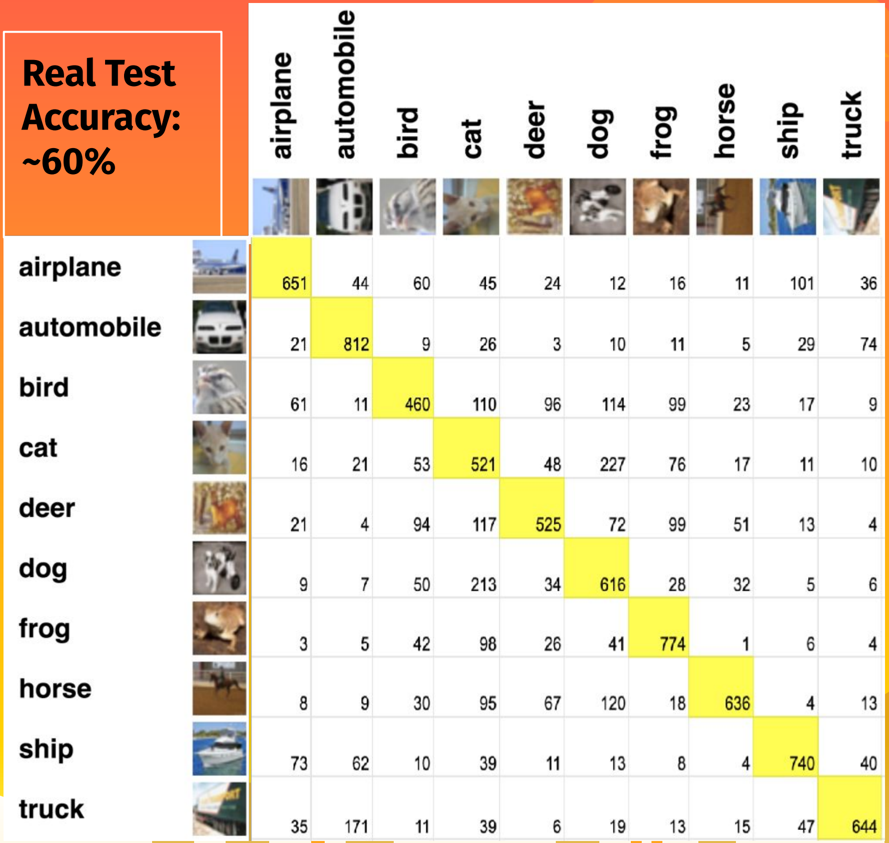
Fake Test Accuracy: ~40%



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	497	42	59	57	34	13	31	38	173	56
automobile	47	431	31	51	48	21	48	40	90	193
bird	88	19	235	133	155	129	105	83	34	28
cat	24	23	89	292	109	196	100	92	18	57
deer	36	14	106	116	333	85	130	134	22	24
dog	13	7	90	209	95	361	58	129	13	25
frog	14	21	68	175	138	51	437	54	10	32
horse	27	13	50	91	112	107	43	447	144	66
ship	134	58	22	28	30	40	18	21	546	103
truck	57	181	19	67	38	22	44	53	116	403

“Fake” Cifar-10 Confusion Matrix

Real Test Accuracy: ~60%



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	651	44	60	45	24	12	16	11	101	36
automobile	21	812	9	26	3	10	11	5	29	74
bird	61	11	460	110	96	114	99	23	17	9
cat	16	21	53	521	48	227	76	17	11	10
deer	21	4	94	117	525	72	99	51	13	4
dog	9	7	50	213	34	616	28	32	5	6
frog	3	5	42	98	26	41	774	1	6	4
horse	8	9	30	95	67	120	18	636	4	13
ship	73	62	10	39	11	13	8	4	740	40
truck	35	171	11	39	6	19	13	15	47	644

Real Cifar-10 Confusion Matrix

Conclusions and Future Work

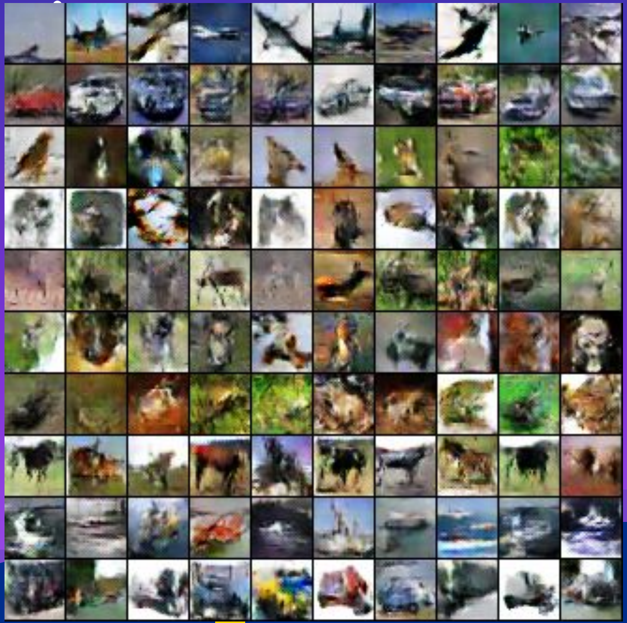
Conclusion

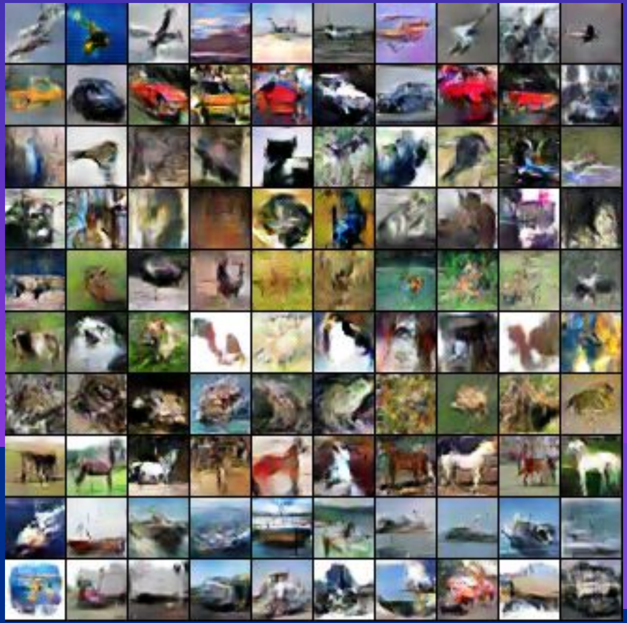
- In both MNIST and Cifar-10, Generated Data worse than Real Data

- Generated Data received lower accuracy score than Real Data:
  - MNIST: ~50% vs >93% Test Accuracy
  - CIFAR: ~40% vs ~60% Test Accuracy

Cifar-10 Thoughts

- Might have a discriminator and generator cap
  - After a certain amount of epochs, quality plateaued





(Left)
100th Epoch

(Right)
1000th Epoch

Moving Forward

- Test Various different Generator and Discriminator Functions
- Fine-tune the Density Functions more
- Test Various Cost Functions
- Try Other GANs Implementation
  - eg. DC GANS (Deep Convolutional GANs), Cycle GANs

Implications

- If GANs can be improved, it can result in:
  - Provide more realistic Images
  - Reinforced Learning (simulate models)
  - Creation of multi-media works
  - Simulate Missing Data or More Data
  - AND MORE!

“

Computer science inverts the normal. In normal science, you're given a world, and your job is to find out the rules. In computer science, you give the computer the rules, and it creates the world.

-Alan Kay

Thank You!