

# Breaking Cloud AutoAI Models

Evaluating the Robustness of AutoAI Models to Adversarial Attacks

Term Project for CSCI-GA 3033-085 Cloud and Machine Learning  
Prepared by Aashka Trivedi (aht323) and Anya Trivedi (aht324)

# Image Classification on Cloud Auto AI Models

# AutoAI

## Automating the Artificial Intelligence Lifecycle

- Ubiquity and performance of Machine Learning Models
- Automation on the Cloud
- Spending on Cloud AI will grow to \$75Billion in 2022<sup>1</sup>
- IBM Cloud's AutoAI Service
- Google Cloud Platform's AutoML Tool

[1] <https://techjury.net/blog/how-many-companies-use-cloud-computing/>

# How Easily Fooled are Cloud AutoAI Models?

Spoiler: Very Easily

# IBM Cloud

## Training AutoAI Models for MNIST

- CSV input
- 3 Models with 4 optimization pipelines
- Evaluation Criteria: Validation Accuracy (10-fold cross validation)

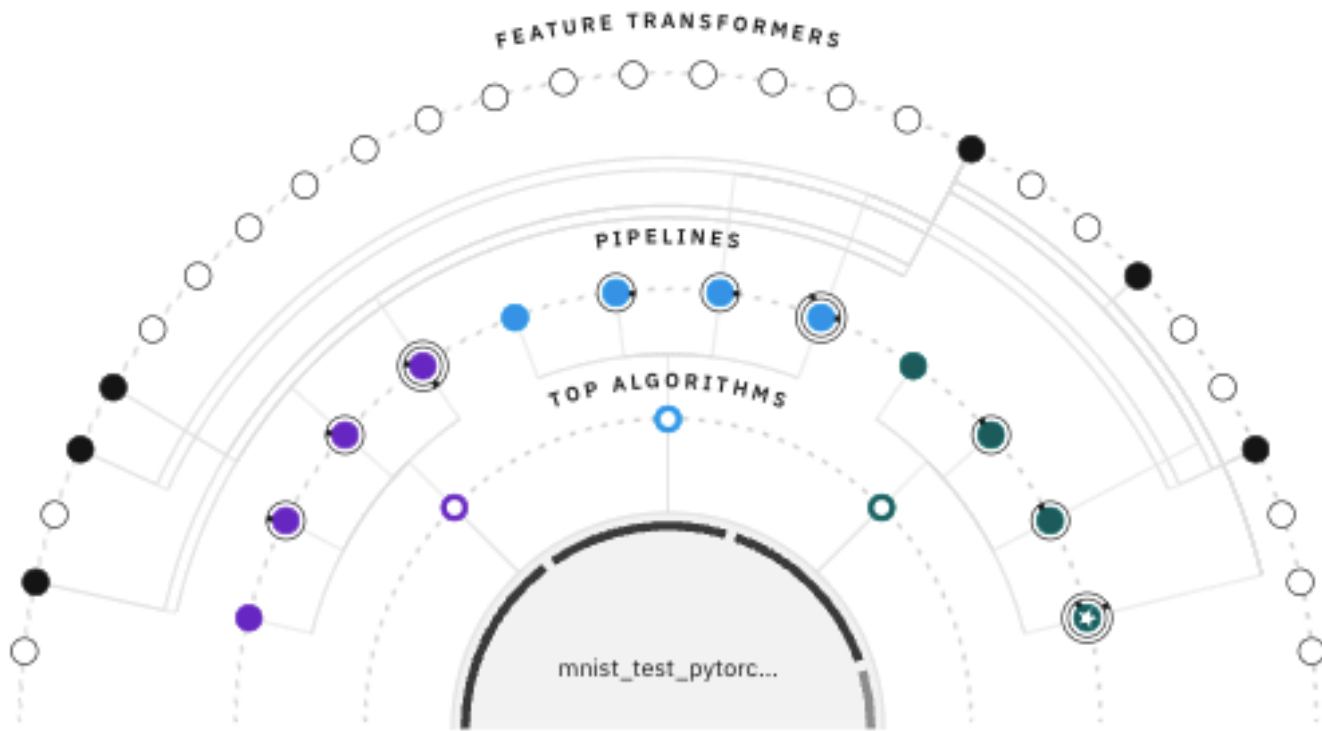
Experiment summary

Pipeline comparison

★ Rank by: Accuracy (Optimized) | Cross validation score 🔗

Relationship map ⓘ

Prediction column: label



Progress map

Swap view 🔗



Experiment completed 🟢

12 PIPELINES GENERATED

12 pipelines generated from algorithms. See pipeline leaderboard below for more detail.

Time elapsed: 2 hours

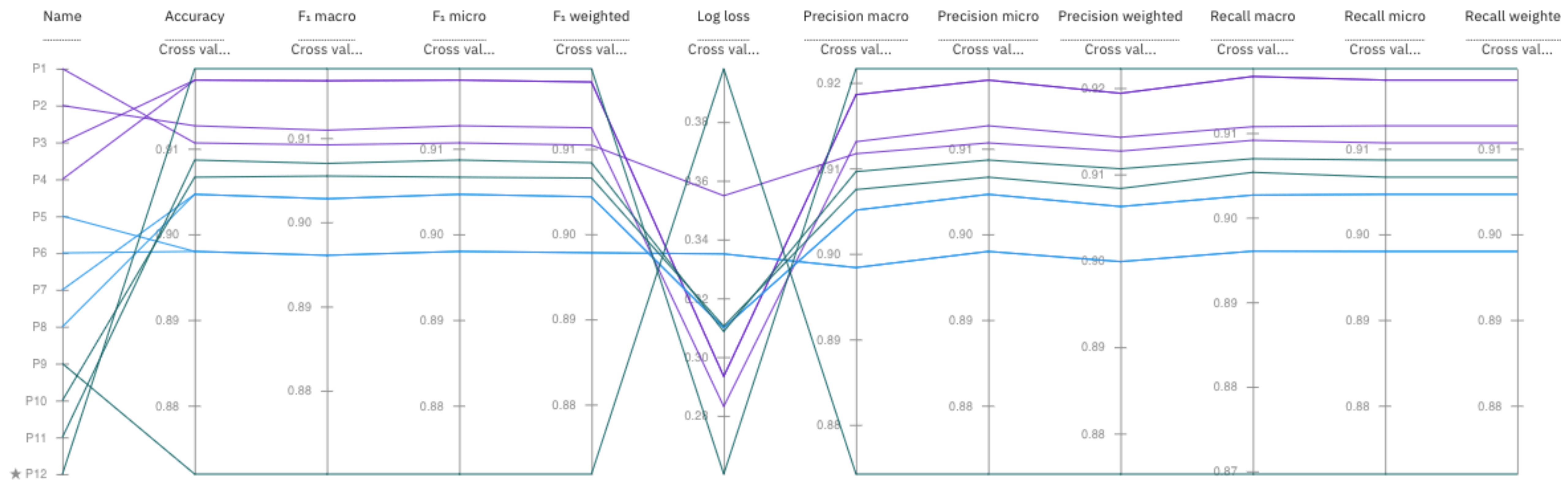
View log

Save code

Pipeline leaderboard ⌵

	Rank	↑	Name	Algorithm	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1		Pipeline 12	🟢 Gradient Boosting Classifier	0.919	HPO-1 FE HPO-2	00:11:08
	2		Pipeline 3	🟡 LGBM Classifier	0.918	HPO-1 FE	00:16:09
	3		Pipeline 4	🟡 LGBM Classifier	0.918	HPO-1 FE HPO-2	00:09:26
	4		Pipeline 2	🟡 LGBM Classifier	0.913	HPO-1	00:05:49
	5		Pipeline 1	🟡 LGBM Classifier	0.911	None	00:01:12
	6		Pipeline 11	🟢 Gradient Boosting Classifier	0.909	HPO-1 FE	00:29:23
	7		Pipeline 10	🟢 Gradient Boosting Classifier	0.907	HPO-1	00:12:12
	8		Pipeline 7	🟡 XGB Classifier	0.905	HPO-1 FE	00:15:51
	9		Pipeline 8	🟡 XGB Classifier	0.905	HPO-1 FE HPO-2	00:22:31
	10		Pipeline 5	🟡 XGB Classifier	0.898	None	00:01:13





Comparison of Experiment Pipelines on IBM AutoAI Platform

# Google Cloud Platform

## Training AutoML Models for MNIST

- CSV input- [Tables](#), Image Input- Vision
- Black Box
- Evaluation Criteria: Log Loss (Only Criteria for MultiClass Classification)



Model

mnist\_image\_classification\_model

Multi-class classification model  
Nov 29, 2021, 10:14:12 AM  
Training cost: 1 node hour

Target	Feature columns	Optimized for	AUC PR	AUC ROC	Precision	Recall	Log loss
label	<a href="#">784 included</a> 5,910 test rows	Log loss	0.997	0.999	98.9%	98.1%	0.057

Micro-averaged precision and recall are generated using a score threshold of 0.5

→ EXPORT PREDICTIONS ON TEST DATASET TO BIGQUERY

You have up to 30 days to export your test dataset to BigQuery

Filter

Filter labels

3

6

7

0

2

4

8

9

5

All

Score threshold

0.50

F1 score	0.985
Precision	98.9% (5,800/5,862)
True positive rate (Recall)	98.1% (5,800/5,910)
False positive rate	0.001 (62/53,190)

The score threshold determines the minimum level of confidence needed to make a prediction positive. [Learn more about model evaluation](#)

Precision

0%100%

AUC: 0.997

PRC

True positive rate

0%100%

AUC: 0.999

ROC

100%0%0.01.0

Score threshold

RecallPrecision

Precision & Recall

GCP AutoML Model Performance

# Comparison

## IBM Cloud AutoAI vs GCP AutoML

	IBM Cloud AutoAI	GCP AutoML
MNIST Validation Accuracy	96.5	97.3
MNIST Test Accuracy	99.6	98.6
Input Data	CSV Only	CSV, Image, Textual Data
Access to Models	Easily convertible to Jupyter Notebooks	User needs to link model to Notebook
Hyperparameter Optimization	More Transparent	Less Transparent
Training Time	Shorter (per pipeline)	Larger
Deployment and Testing Time	Very Long	Shorter
Online Resources	Very Widely Available	Not up to date

# Adversarial Robustness

# Adversarial Examples

## Generating Adversarial Examples for MNIST

- Small Perturbation to input (Hyperparameter: Epsilon)
- Targeted Attacks vs Untargeted Attacks
- Whitebox vs Blackbox attacks<sup>1</sup>
- Transferability Property and Substitute Models<sup>2</sup>

[1] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, Cho-Jui Hsieh, 2017. ZOO: Zeroth Order Optimization based Black-box Attacks to Deep Neural Networks without Training Substitute Models

[2] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy, 2015. Explaining and Harnessing Adversarial Examples

# Fast Gradient Sign Method

## Whitebox Adversarial Example Generation

- Proposed by Goodfellow et. al<sup>1</sup>
- Add noise in direction of gradient to maximise loss
- Hyperparameter Epsilon: how much noise
- Fast, easy, not the most powerful
- Whitebox
- Substitute Model- LeNet 50 trained on MNIST
- Epsilon: 0 (no distortion), 0.10, 0.15, 0.20, 0.25, 0.30

[1] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy, 2015. Explaining and Harnessing Adversarial Examples

Original Label : 6  
Lenet Prediction : 6  
IBMAutoAI Prediction:6  
GCPAutoML Prediction:6



Original Label : 9  
Lenet Prediction : 9  
IBMAutoAI Prediction:9  
GCPAutoML Prediction:9



Original Label : 3  
Lenet Prediction : 3  
IBMAutoAI Prediction:3  
GCPAutoML Prediction:3



Original Label : 4  
Lenet Prediction : 4  
IBMAutoAI Prediction:4  
GCPAutoML Prediction:4



Original Label : 1  
Lenet Prediction : 1  
IBMAutoAI Prediction:1  
GCPAutoML Prediction:1



Original Label : 6  
Lenet Prediction : 1  
IBMAutoAI Prediction:6  
GCPAutoML Prediction:6



Original Label : 9  
Lenet Prediction : 4  
IBMAutoAI Prediction:7  
GCPAutoML Prediction:7



Original Label : 3  
Lenet Prediction : 5  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 4  
Lenet Prediction : 6  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 1  
Lenet Prediction : 8  
IBMAutoAI Prediction:1  
GCPAutoML Prediction:1



Original Label : 6  
Lenet Prediction : 0  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:6



Original Label : 9  
Lenet Prediction : 4  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:7



Original Label : 3  
Lenet Prediction : 5  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:7



Original Label : 4  
Lenet Prediction : 9  
IBMAutoAI Prediction:9  
GCPAutoML Prediction:8



Original Label : 1  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 6  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 9  
Lenet Prediction : 4  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:7



Original Label : 3  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 4  
Lenet Prediction : 9  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:0



Original Label : 1  
Lenet Prediction : 2  
IBMAutoAI Prediction:7  
GCPAutoML Prediction:7



Original Label : 6  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 9  
Lenet Prediction : 8  
IBMAutoAI Prediction:5  
GCPAutoML Prediction:7



Original Label : 3  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 4  
Lenet Prediction : 8  
IBMAutoAI Prediction:9  
GCPAutoML Prediction:9



Original Label : 1  
Lenet Prediction : 8  
IBMAutoAI Prediction:4  
GCPAutoML Prediction:8



Original Label : 6  
Lenet Prediction : 4  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:7



Original Label : 9  
Lenet Prediction : 2  
IBMAutoAI Prediction:5  
GCPAutoML Prediction:8



Original Label : 3  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:0



Original Label : 4  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8



Original Label : 1  
Lenet Prediction : 8  
IBMAutoAI Prediction:8  
GCPAutoML Prediction:8

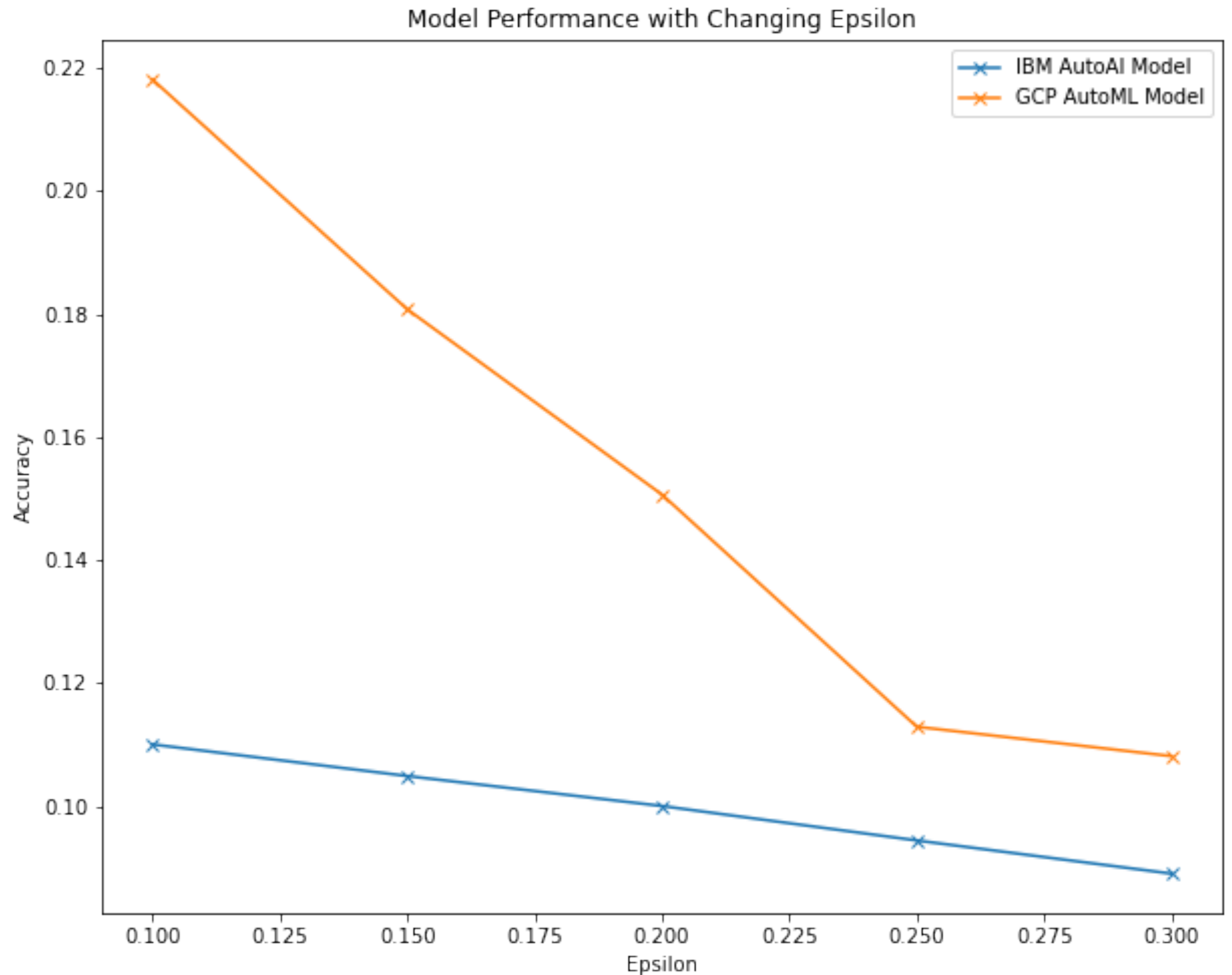


Generated Adversarial Examples with Different Values of Epsilon



# Robustness of Auto AI Model

## Accuracy of models on Adversarial Examples



Model Performance with Changing Epsilon.  
The IBM Model and the GCP Model gave 0.99 and 0.98 accuracy on the original test case (eps=0)

# Defensive Distillation



# Distilling Defensively

## A Defense against Adversarial Attacks

- Proposed by Papernot et. al<sup>1</sup>
- Use distillation<sup>2</sup> to improve robustness
  - Teach a student how a teacher “learns”
  - Prediction Probabilities (“logits”) as soft target labels
- Not effective as a measure of defense<sup>3,4</sup>

[1] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha and Ananthram Swami, 2016. Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks

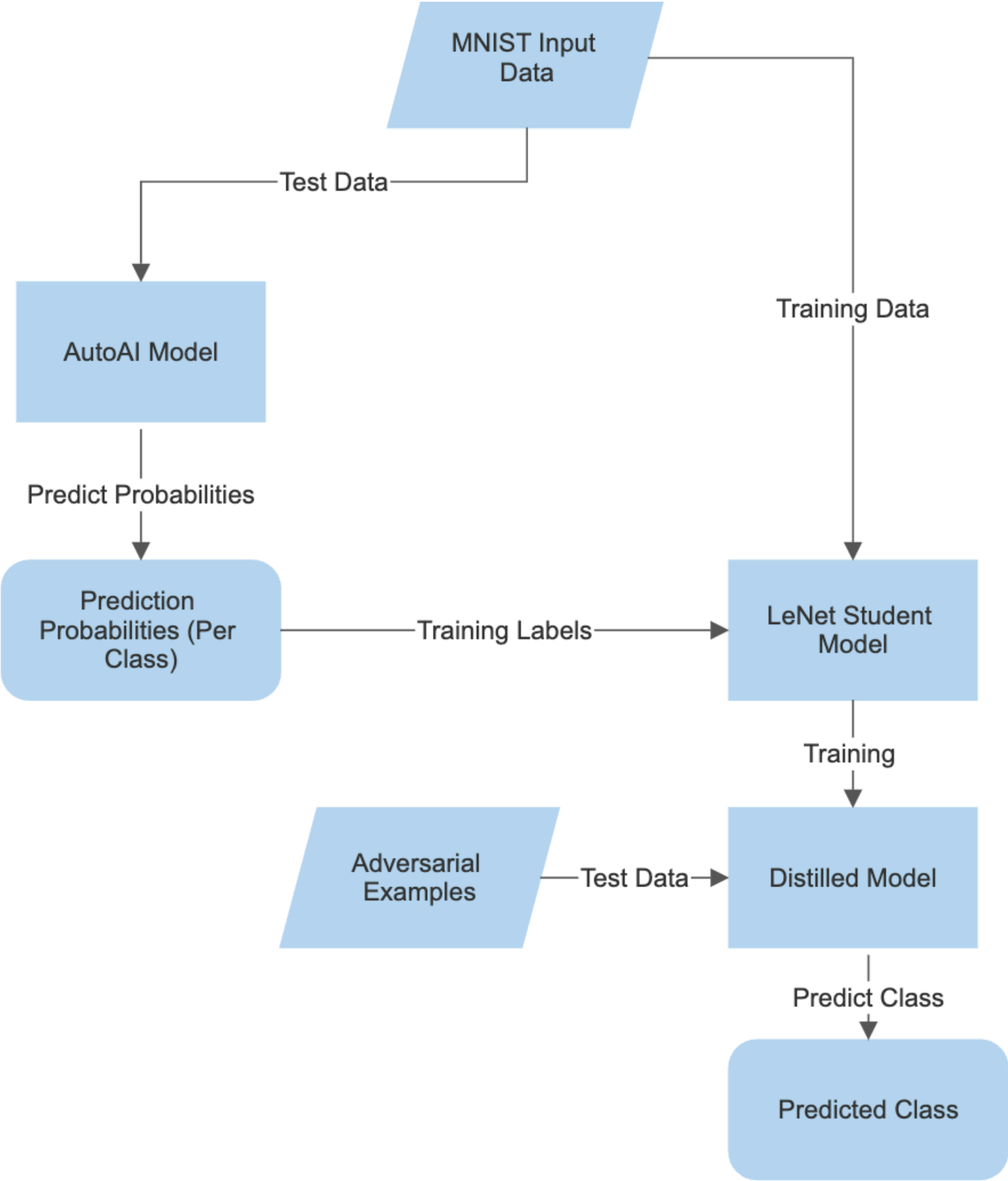
[2] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, 2015. Distilling the Knowledge in a Neural Network

[3] Nicholas Carlini and David Wagner, 2016. Defensive Distillation is Not Robust to Adversarial Example

[4] Nicholas Carlini and David Wagner, 2017. Towards Evaluating the Robustness of Neural Networks

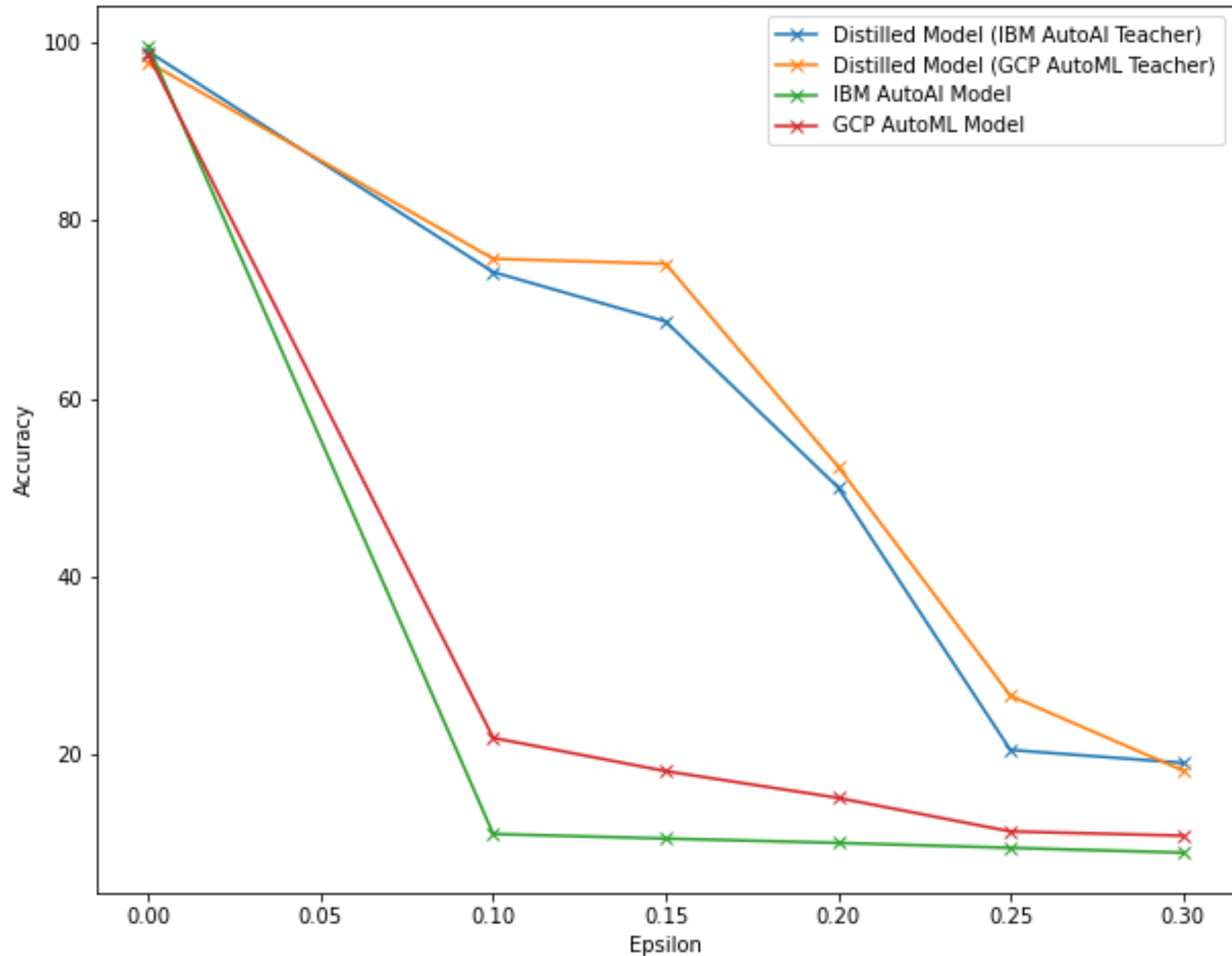
# Distilling AutoAI Models

## Defensive Distillation for IBM Cloud's and GCP's AutoAI Model



Flowchart for Model Distillation

Performance of Models on Adversarial Examples (With and Without Distillation)



Model Robustness with and Without Distillation

# Conclusion

## Major Findings

- Using AutoAI to build Image Classification Models
- AutoAI models are not robust against adversarial attacks
  - GCP vs IBM AutoAI
  - Most Guessed Label
- Defensive distillation is not completely robust to adversarial attacks, but may improve performance to an extent

Github Repository: <https://github.com/aashka-trivedi/cloud-autoai-adversarial-robustness>

# Questions