
P-HACKING ML
By Jankh

About Me

- Github.com/JankhJankh (Where the slides and exercises will be)
- Pentester
- Done a few hobbyist AI projects
- Probably other things

Artificial Intelligence

- Something that seems smart
- Decision trees
- Machine Learning

Artificial Intelligence

- Something that seems smart
- Decision trees
- Machine Learning
- If statements ☺

Machine Learning

- Something that "learns"
- Usually split up into supervised learning and unsupervised learning
- Most of the "AI" on the news is ML based
- Used in security in things like antivirus and network alerts
- Used in fraud detection
- Some research has been done into ML based encryption

Goals for today

- Get people thinking about ML
- Go through some ML algorithms and their pitfalls
- Do 2 hands-on challenges to trick ML algorithms
- Talk about the future of pentesting regarding ML
- Get people thinking about GANs ☺

Real World AI Problems

- User controlled data was used to train a twitter bot

Real World AI Problems

A bunch of AI are being used to predict stocks, you can go buy one of these AIs and use it to make "basically free money".

Stock Market Predictions Based on Artificial Intelligence: Returns up to 66.14% in 1 Month

July 7, 2019

Stock Market Predictions

The Fundamental Package includes our algorithmic stock market predictions for stocks screened by fundamental criteria. Our algorithms help you find best opportunities for both long and short positions for the stocks within each fundamental screen. The stocks are selected according to five basic valuation categories:

- P/E (price to earnings ratio)
- PEG (price/earnings to growth ratio)
- price-to-book ratio
- price-to-sales ratio
- short ratio

Package Name: Fundamental – High price-to-sales ratio Stocks
Recommended Positions: Long
Forecast Length: 1 Month (06/04/2019 – 07/04/2019)
I Know First Average: 14.73%

Fundamentals

Algorithmic Stock Forecast

1 Month		Updated on 04_Jun_2019		
ARRY	MDCO	HEI	EXAS	MELI
6.22	4.48	4.28	4.11	3.02
0.48	0.17	0.19	0.41	0.37
MKTX	EPZM	EBSB	TSS	NKTR
2.69	2.67	2.67	2.60	2.58
0.32	0.35	0.42	0.32	0.28
2.57	2.55	2.35	2.09	2.09
0.18	0.31	0.15	0.29	0.42

Forecast Performance (long)

Symbol	Forecast June 4th	% Change July 4th	Accuracy
ARRY	↑	66.14%	✓
MDCO	↑	3.54%	✓
HEI	↑	11.29%	✓
EXAS	↑	16.84%	✓
MELI	↑	11.38%	✓
MKTX	↑	15.05%	✓

Real World AI Problems

If you like the sound of that, buy my new ICO ☺

"Its basically free money" –Someone Legitimate

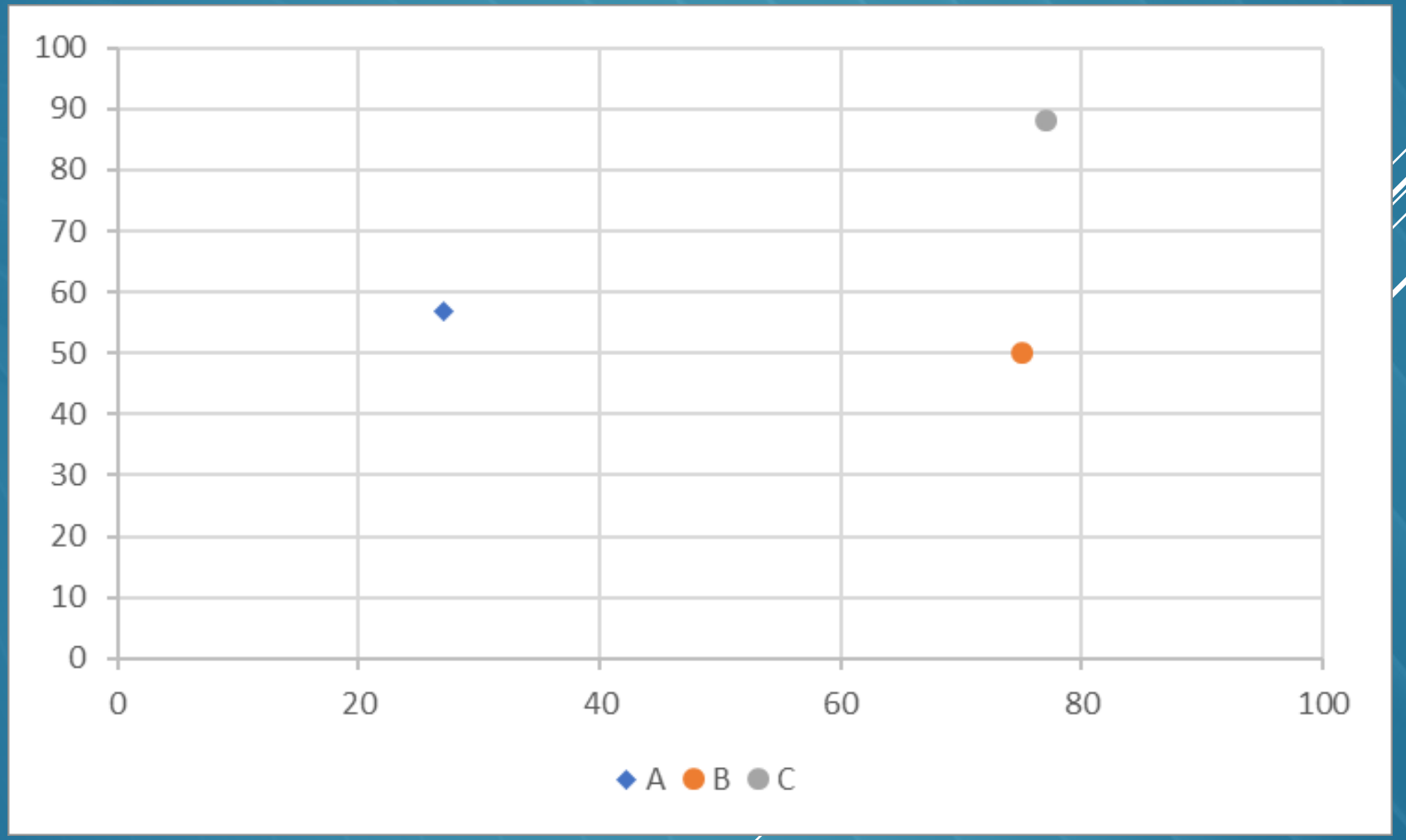
Real World AI Problems

In reality, stocks are insanely volatile and include layers of depth to predict.
While they can be predicted, I wouldn't recommend drinking the AI stock prediction Kool-Aid.
Plus as we will see in a minute, if you can figure out what an AI is doing, you can exploit it (In this case; buy the stocks before the AI, sell before it sells, etc.)

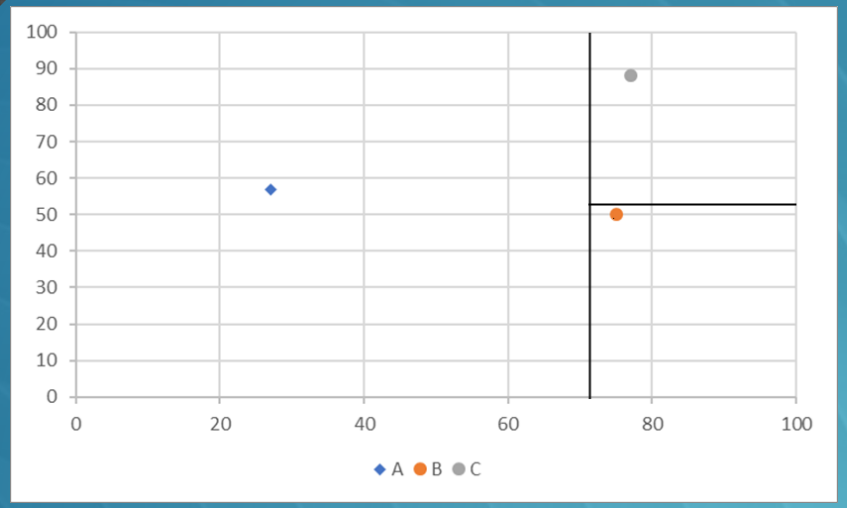
Once again, not recommending anyone try to "beat the system" but use due care and research when looking into this sort of stuff.

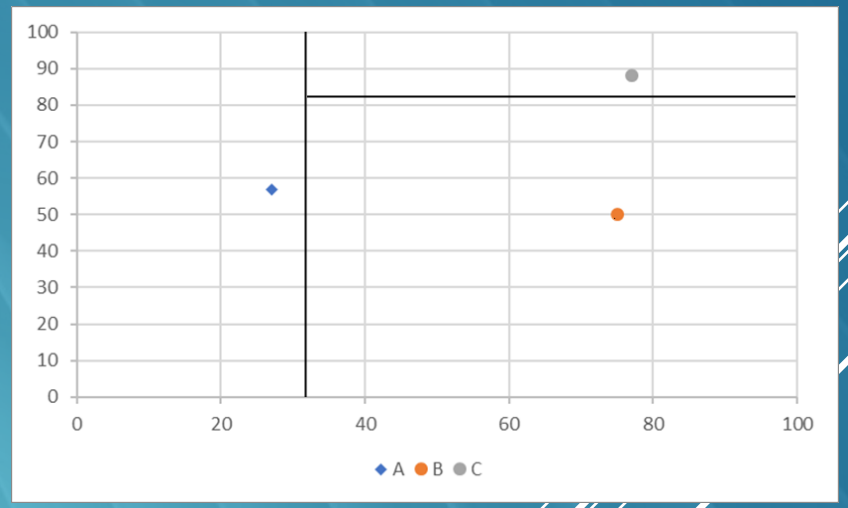
2-Dimensional Data

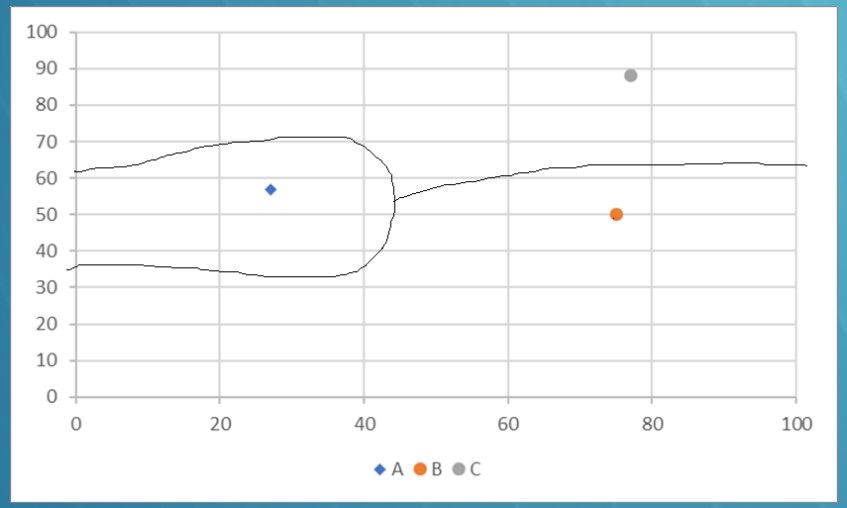
X and Y are the locations of the 1-meter square of land the animal was found on

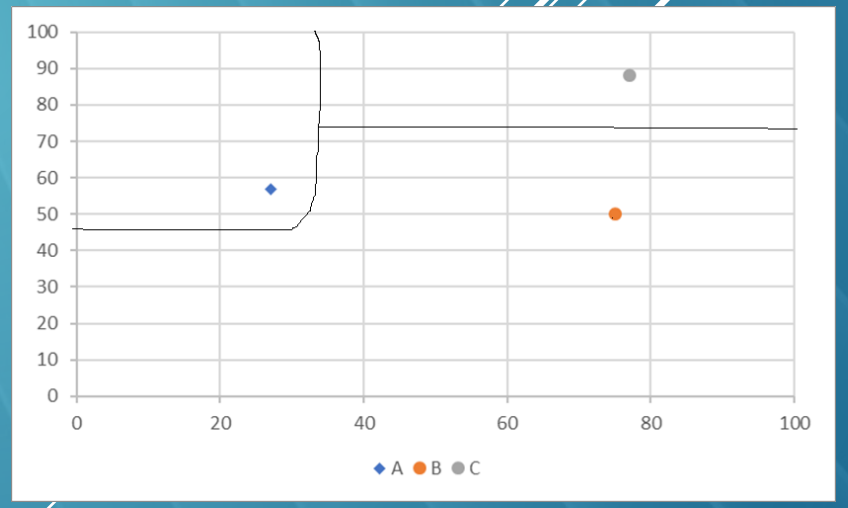


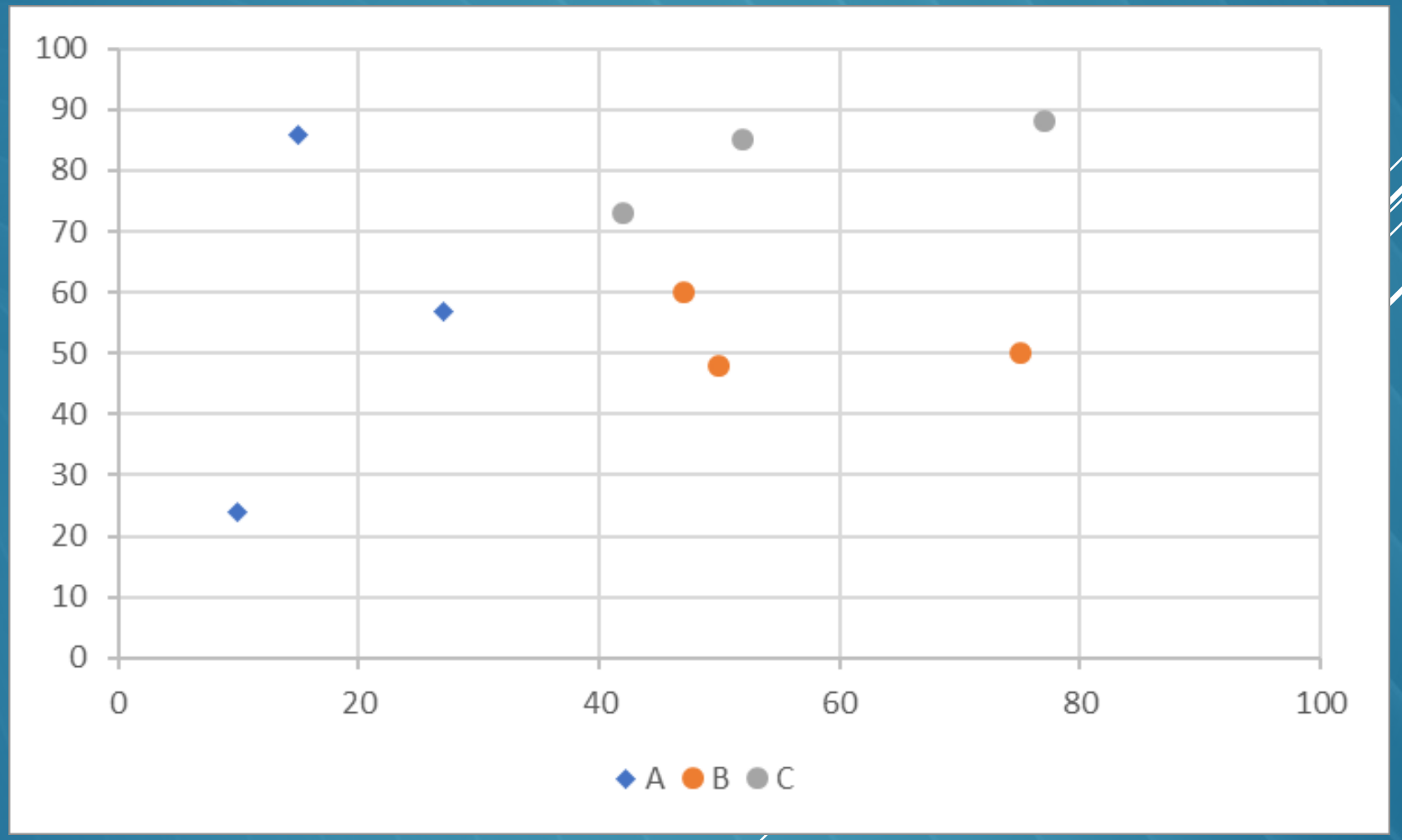
A: Kangaroos
B: Crocodiles
C: Sheep

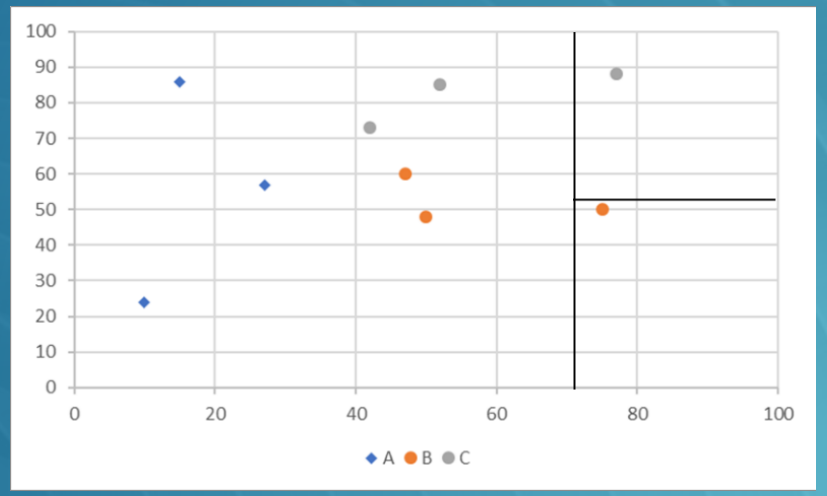

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

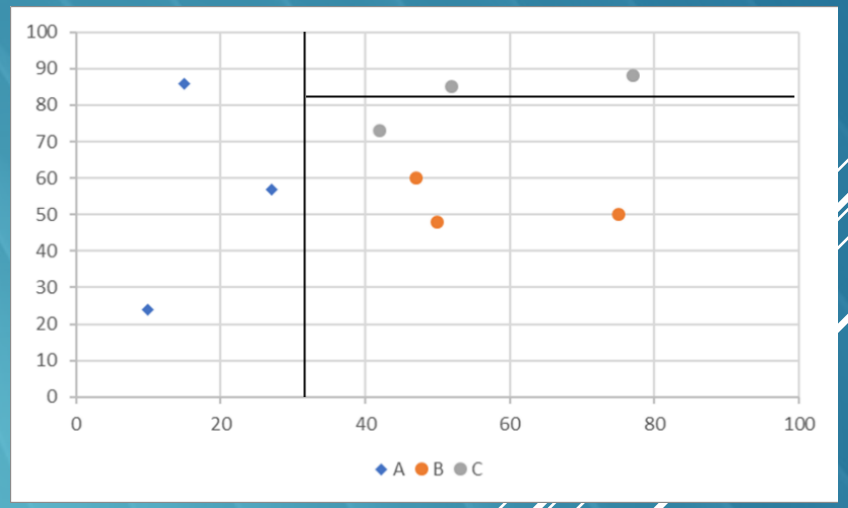

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

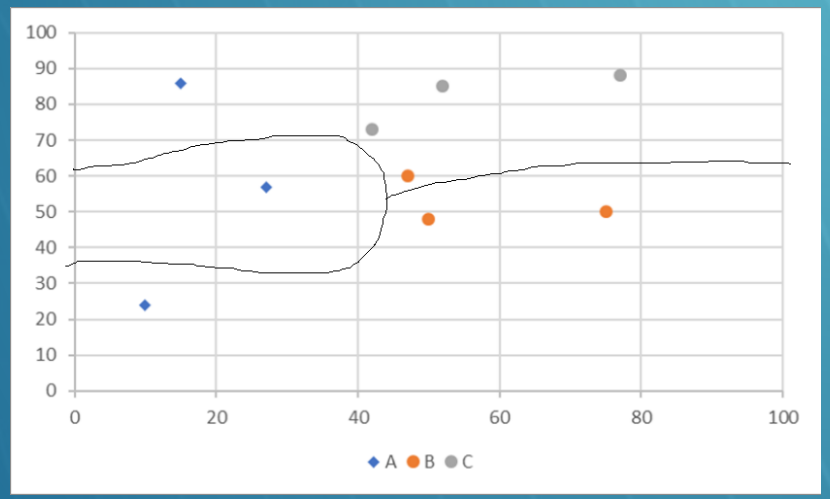

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

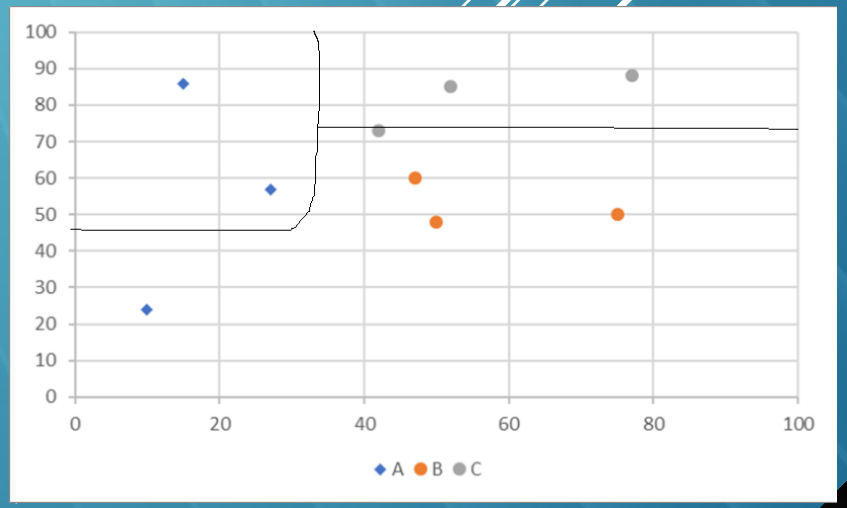

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

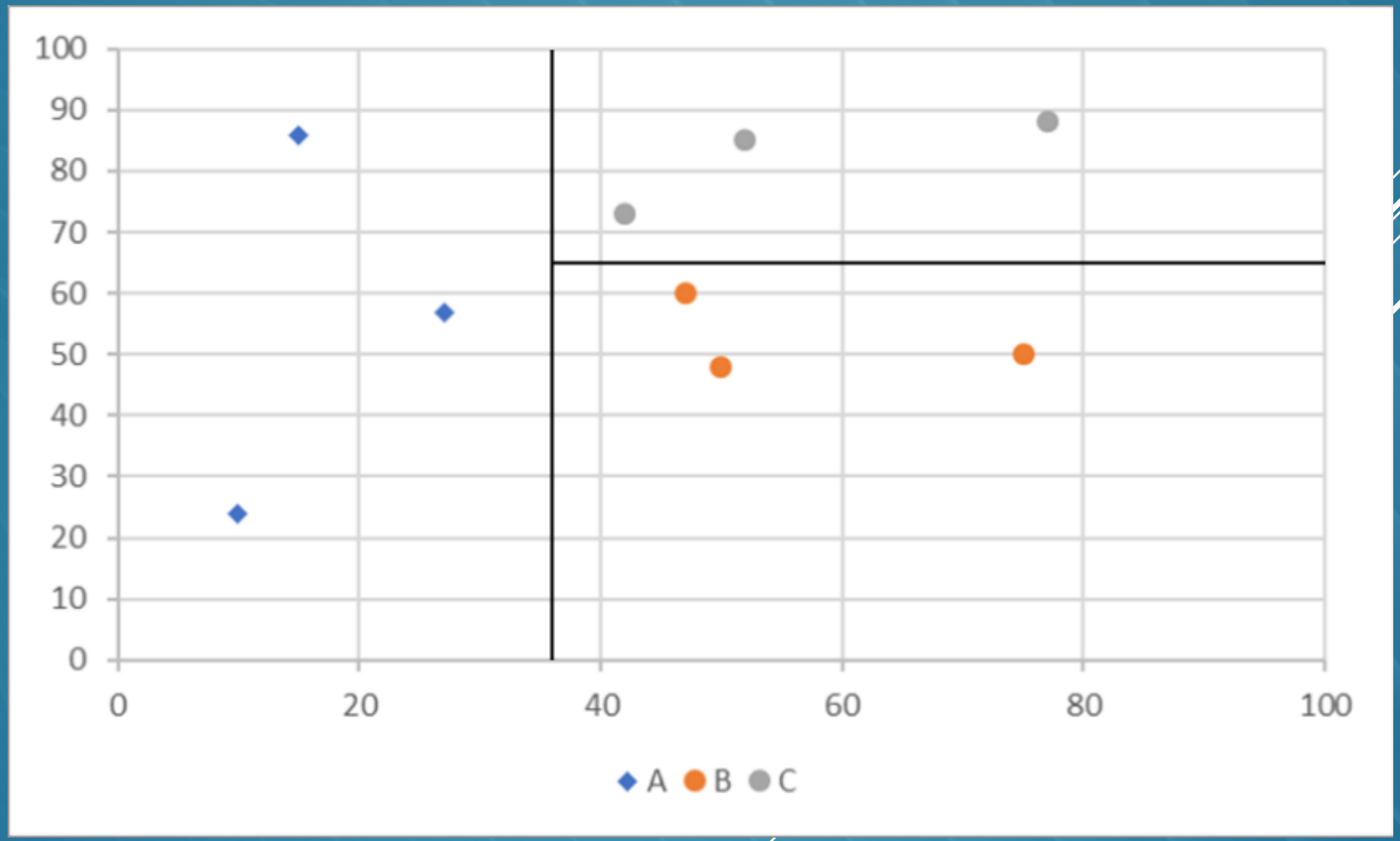

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A
B
C

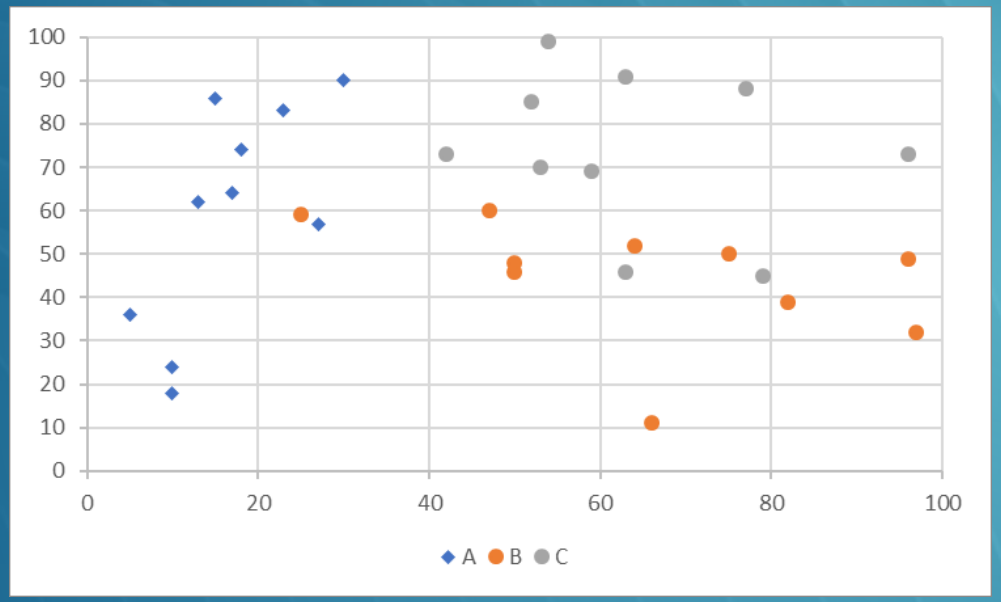

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

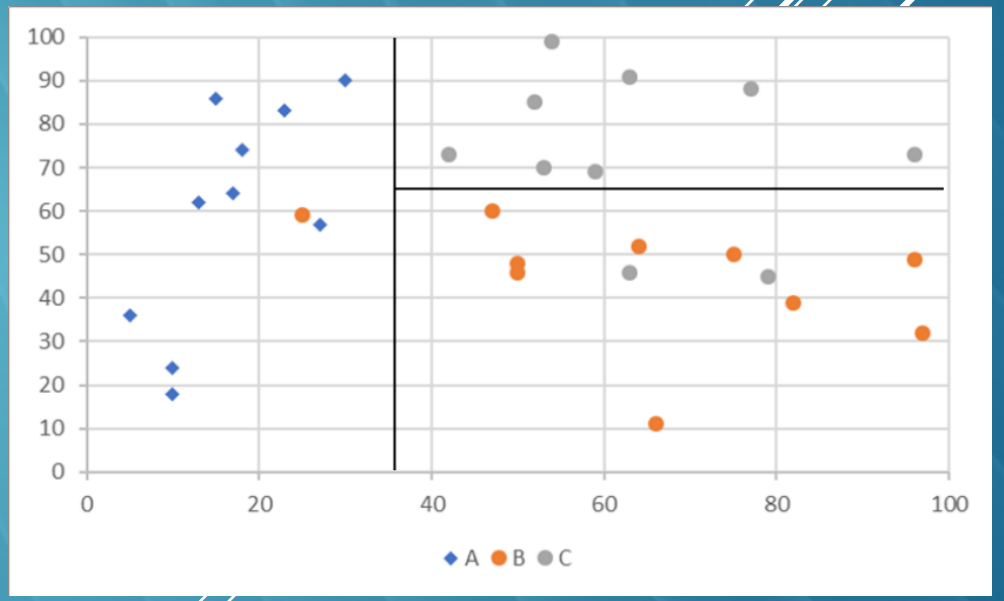

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

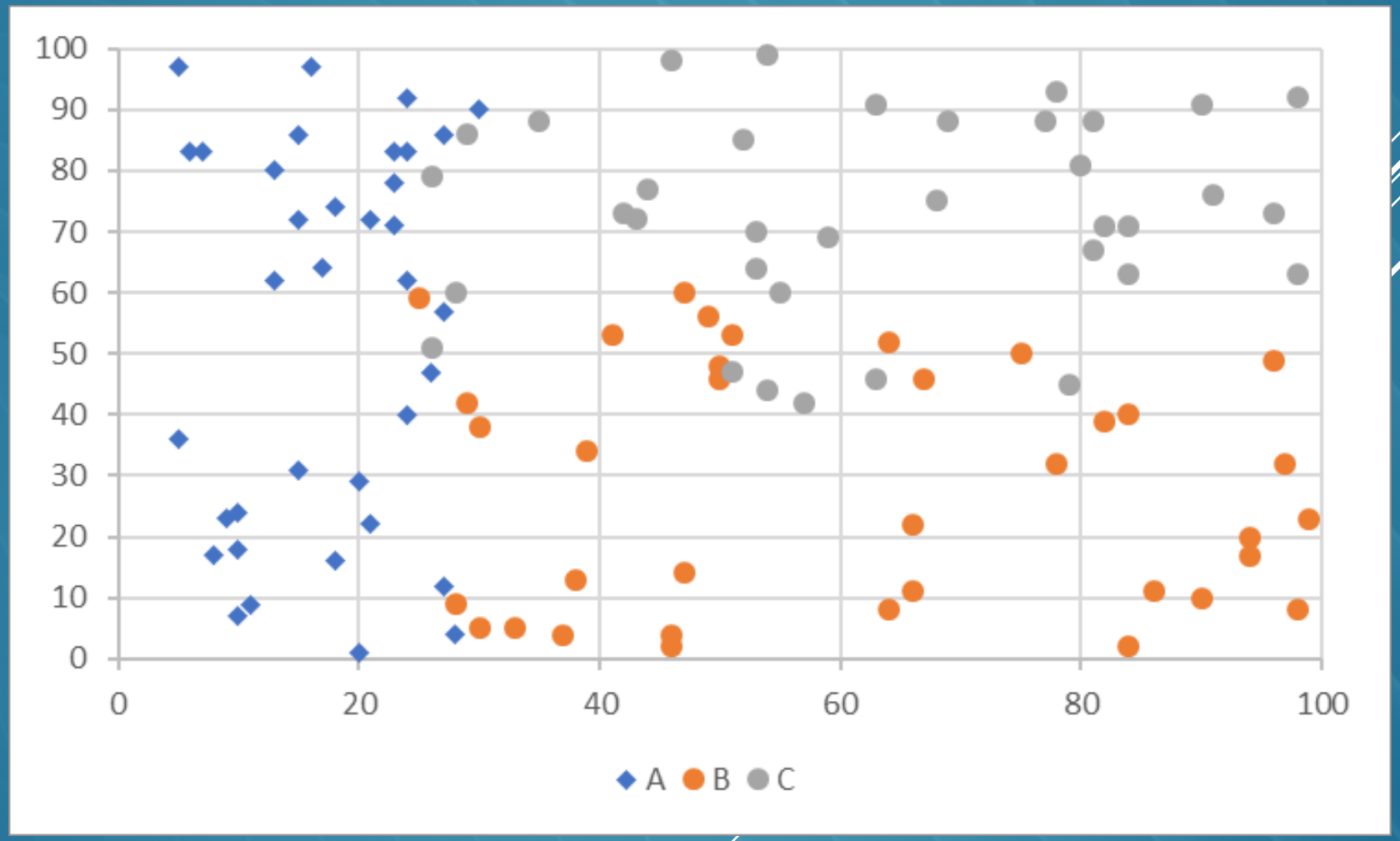

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C

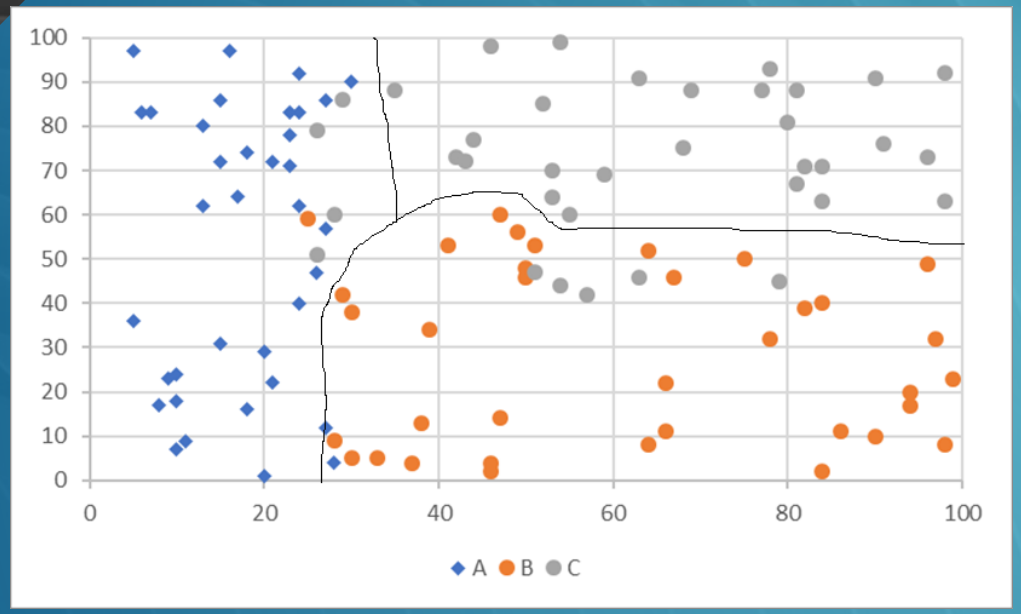

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A B C


100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A
B
C

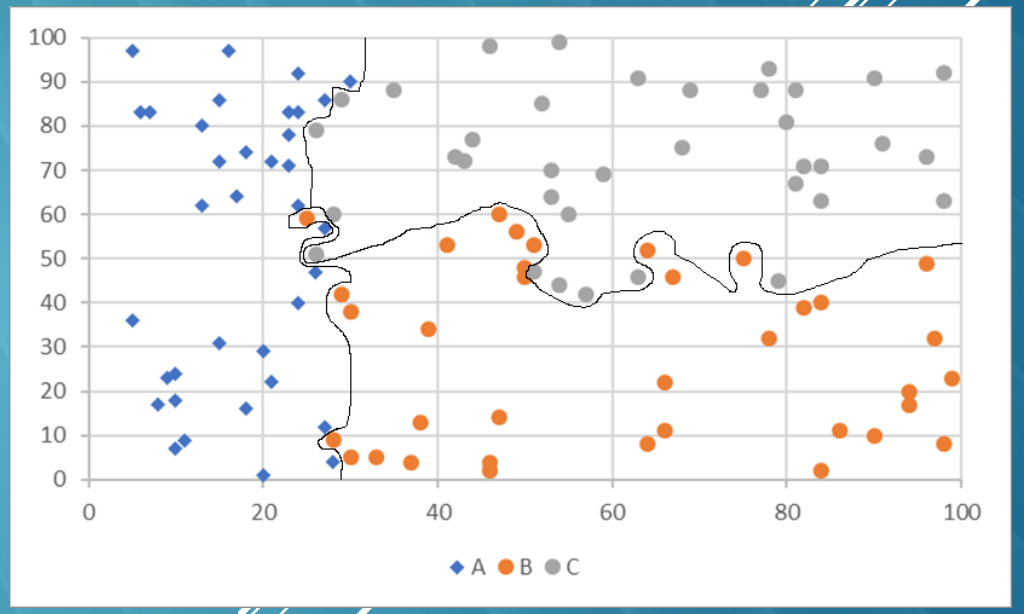

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A
B
C


100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A
B
C

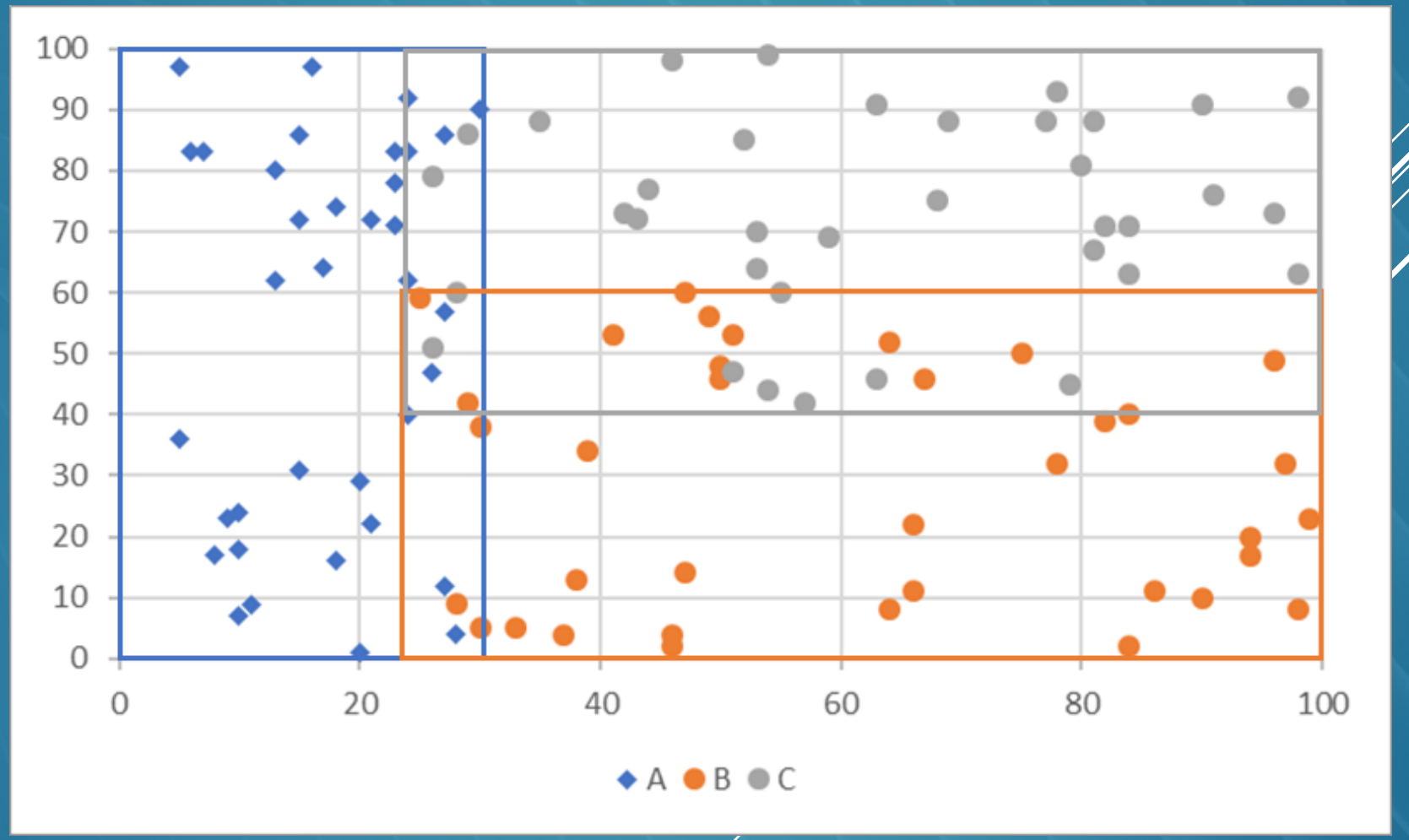

100
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
A
B
C



80% Accuracy



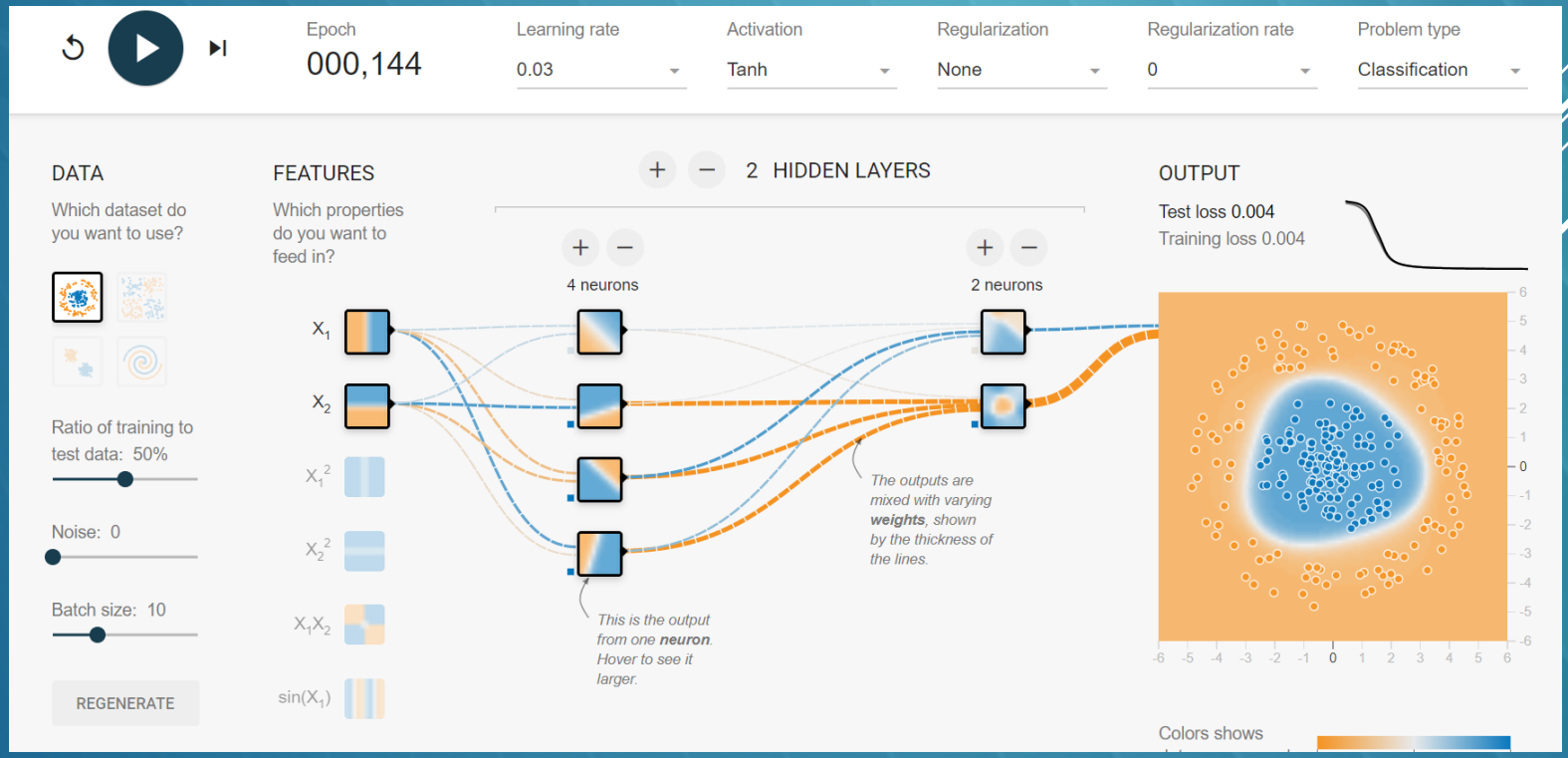
100% Accuracy



This data set is a little bit fuzzy, but if you over-train the network it will generally get you further from the real answer.

TensorFlow Playground

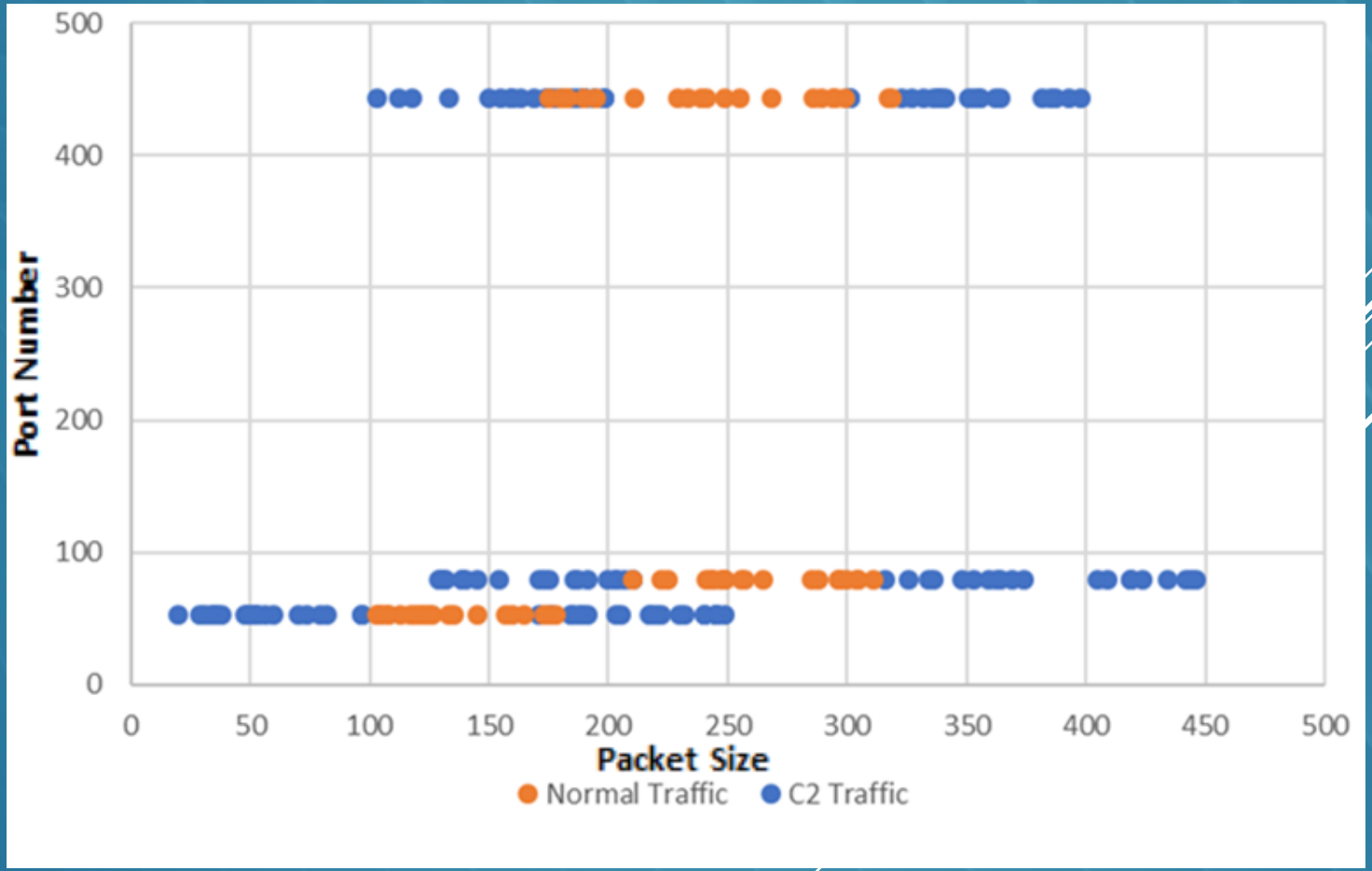
- To get a better idea of how this works with neural networks, checkout
- https://playground.tensorflow.org



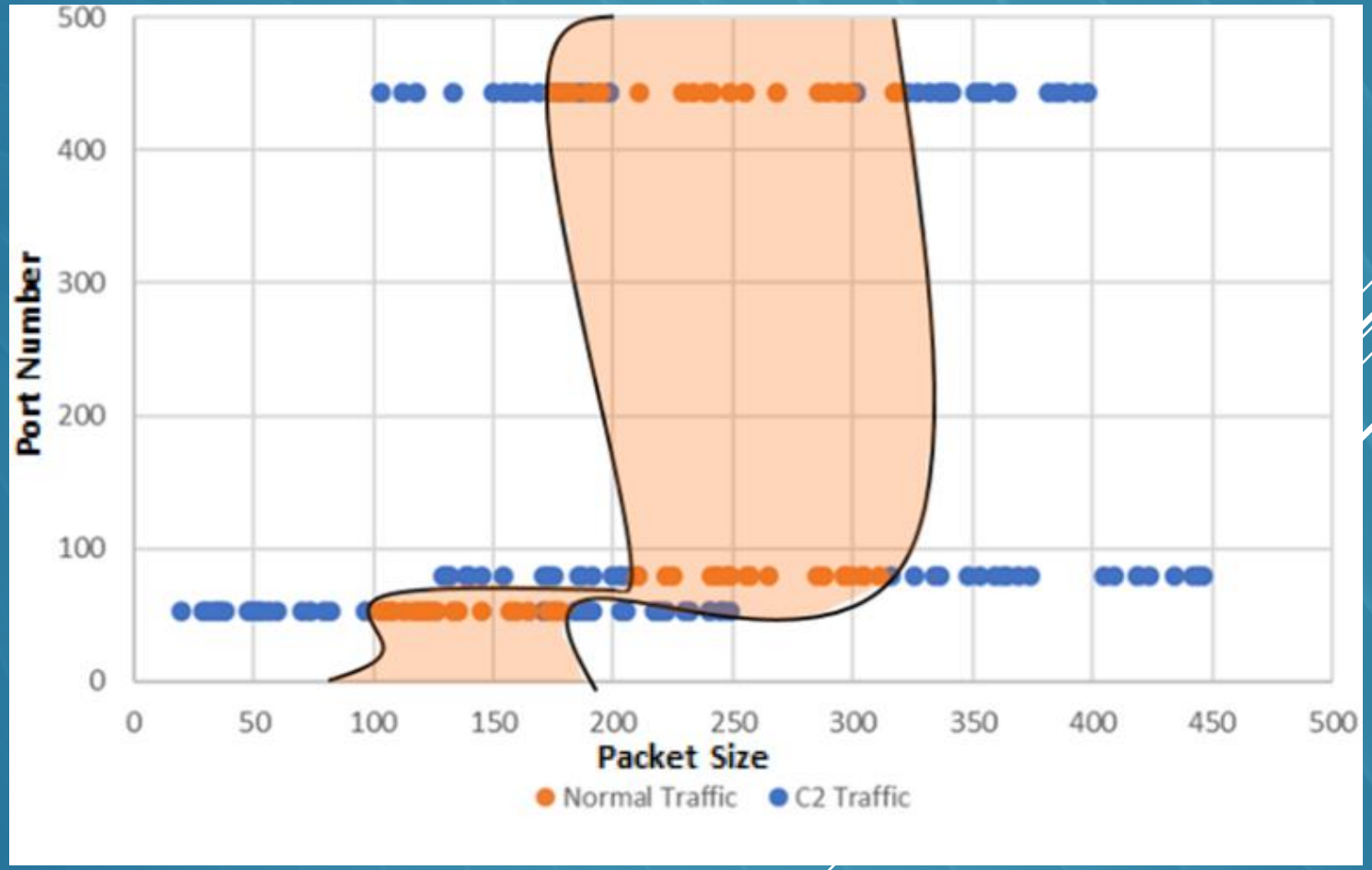
Case study in bad supervised learning

- A client has decided they don't like any of the current network monitoring tools in the market, so they decided to build their own, implementing machine learning to do so.
- The goal being to classify network traffic into either normal or malicious
- They grab a bunch of network logs as their dataset from HTTP, HTTPS, and DNS
- They classify each network packet as either malicious or non malicious
- They decide that packet size is a good metric to use for this classifier

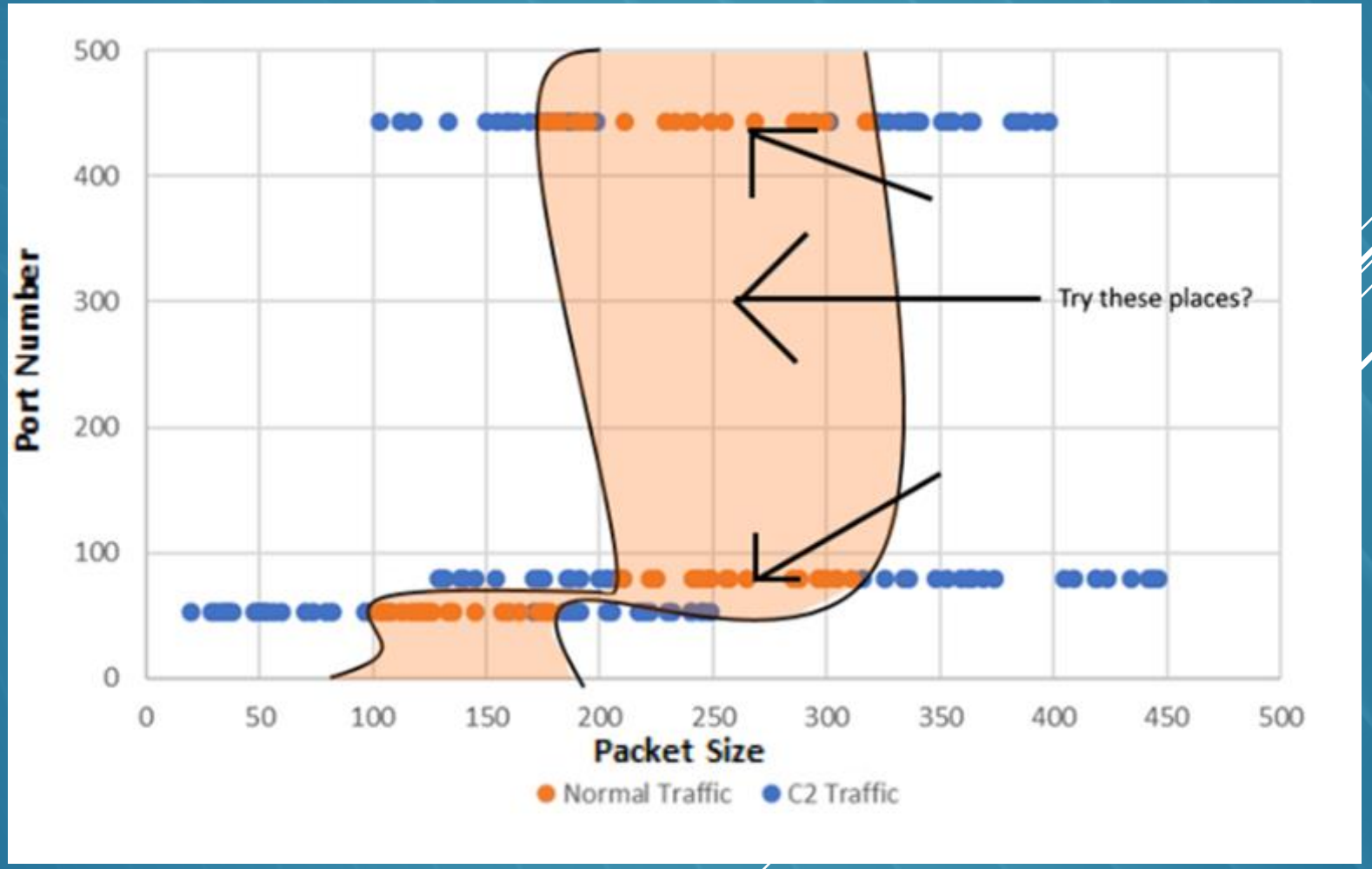
The Training Data



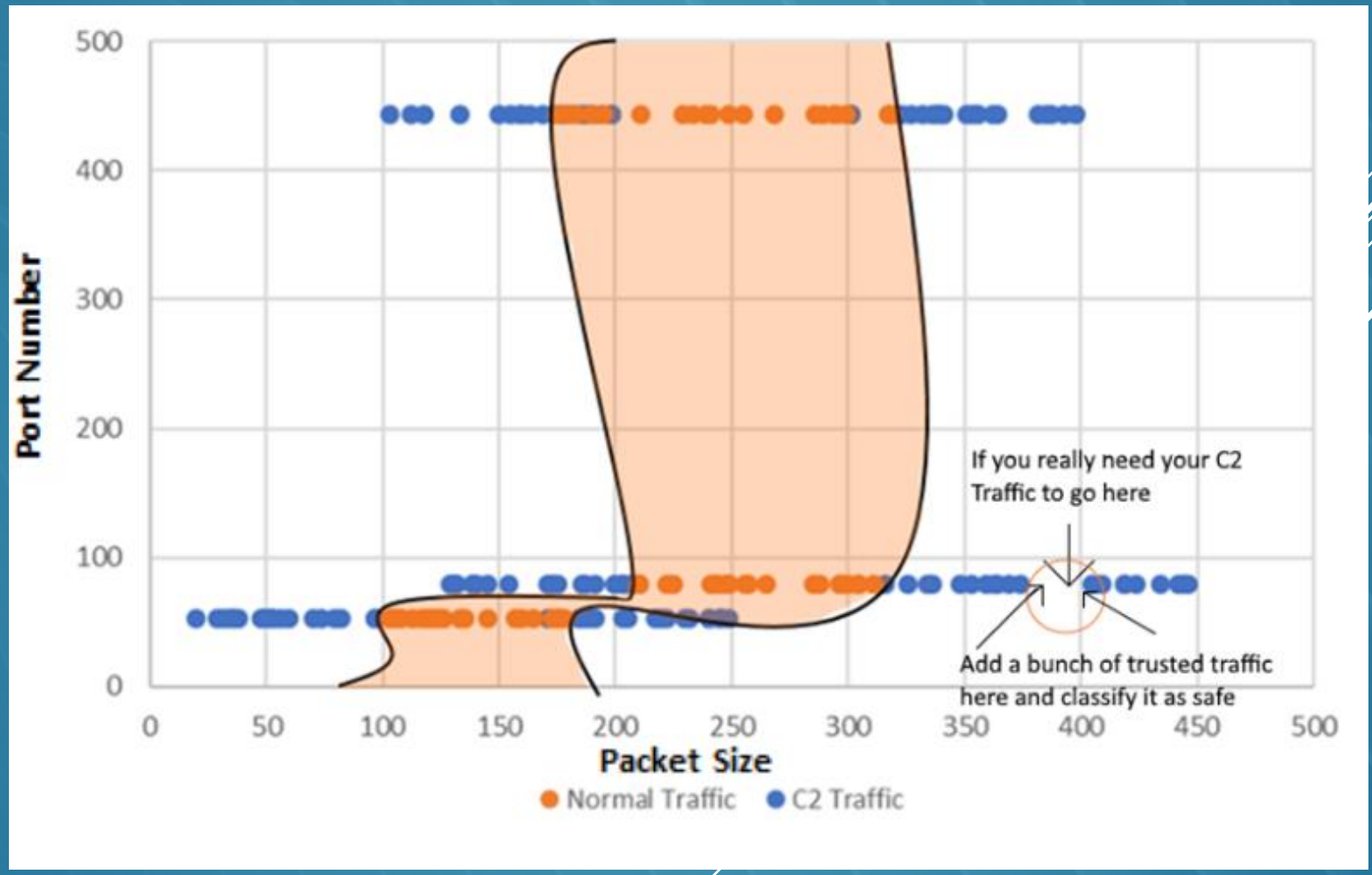
Predicting the model



Exploiting blind spots



Backdooring a dynamic network



Issues with supervised learning

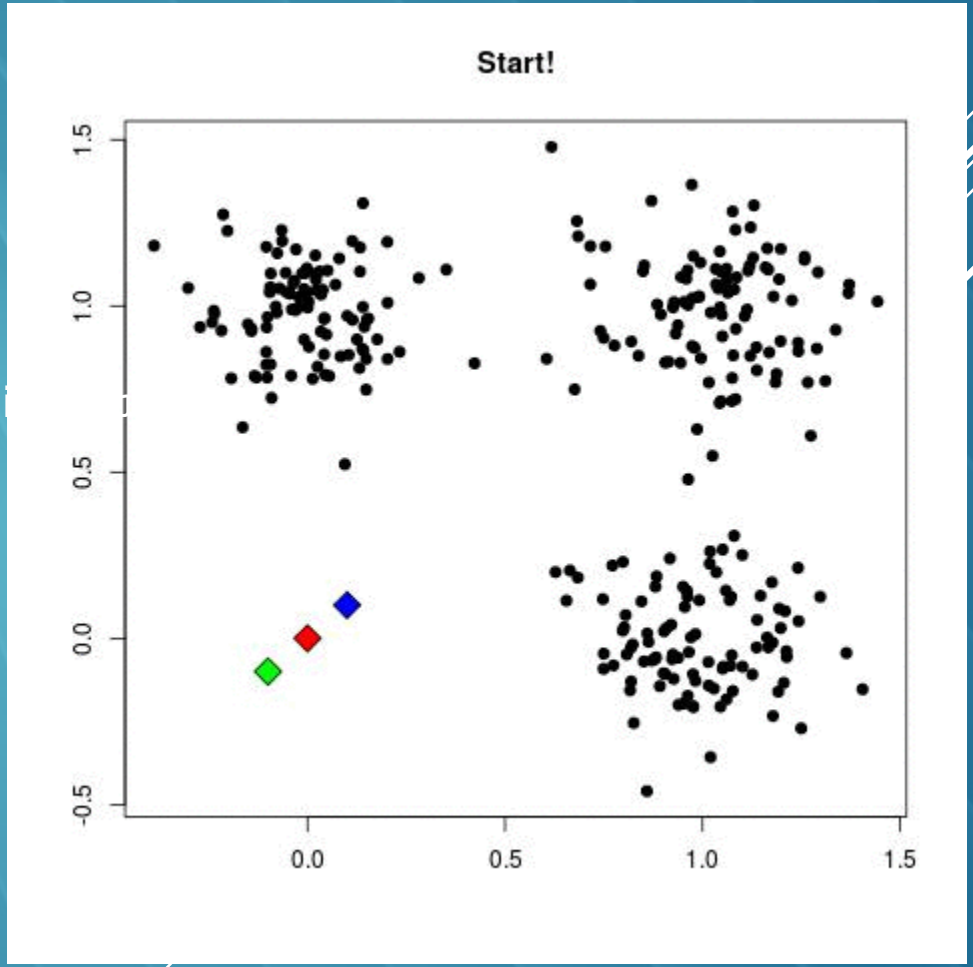
- By default all data is equally valued.
- If you have control over any of the learned data, you can backdoor a network
- If you have access to the trained model you can find blind spots in the neural network
- If you can control how much the model trains, you can over-train the network to backdoor it

Unsupervised learning

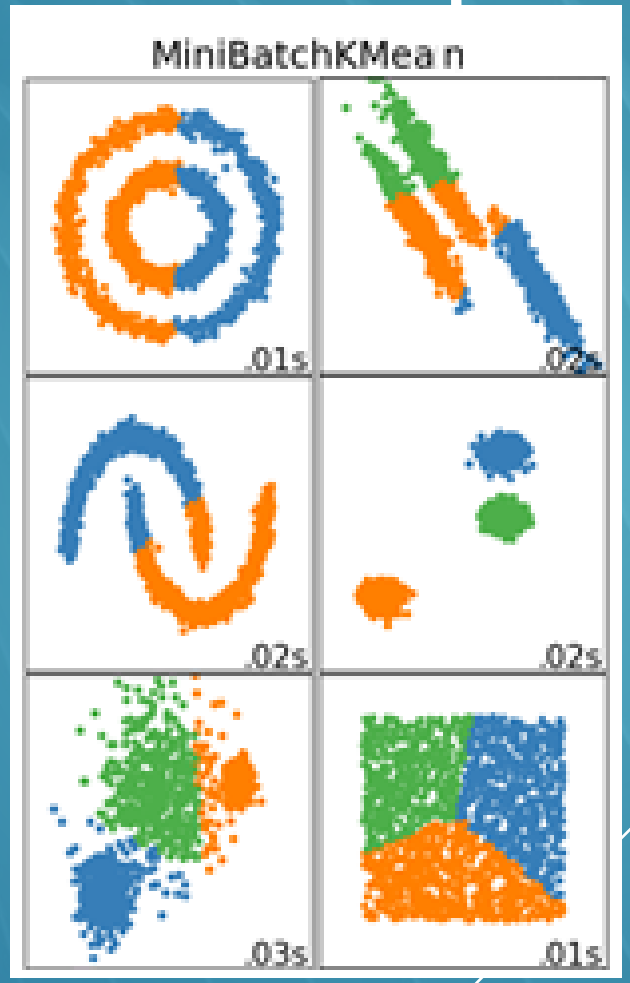
- This works by giving all the data to an algorithm and getting it to figure out what classes are what
- They don't require someone to manually classify every data point in a data set, making it significantly faster and cheaper to implement
- Used for several classification problems with reasonable results
- Works as a good heuristic to verify if you are querying the right values for a problem
- An easy example of this is K-Means clustering

K-Means clustering works by:

1. Defining K number of nodes, and randomly give them values
2. Giving each data point the class title of the nearest node
3. Moving the node to the center of that cluster
4. Repeating steps 2 and 3 until nothing moves

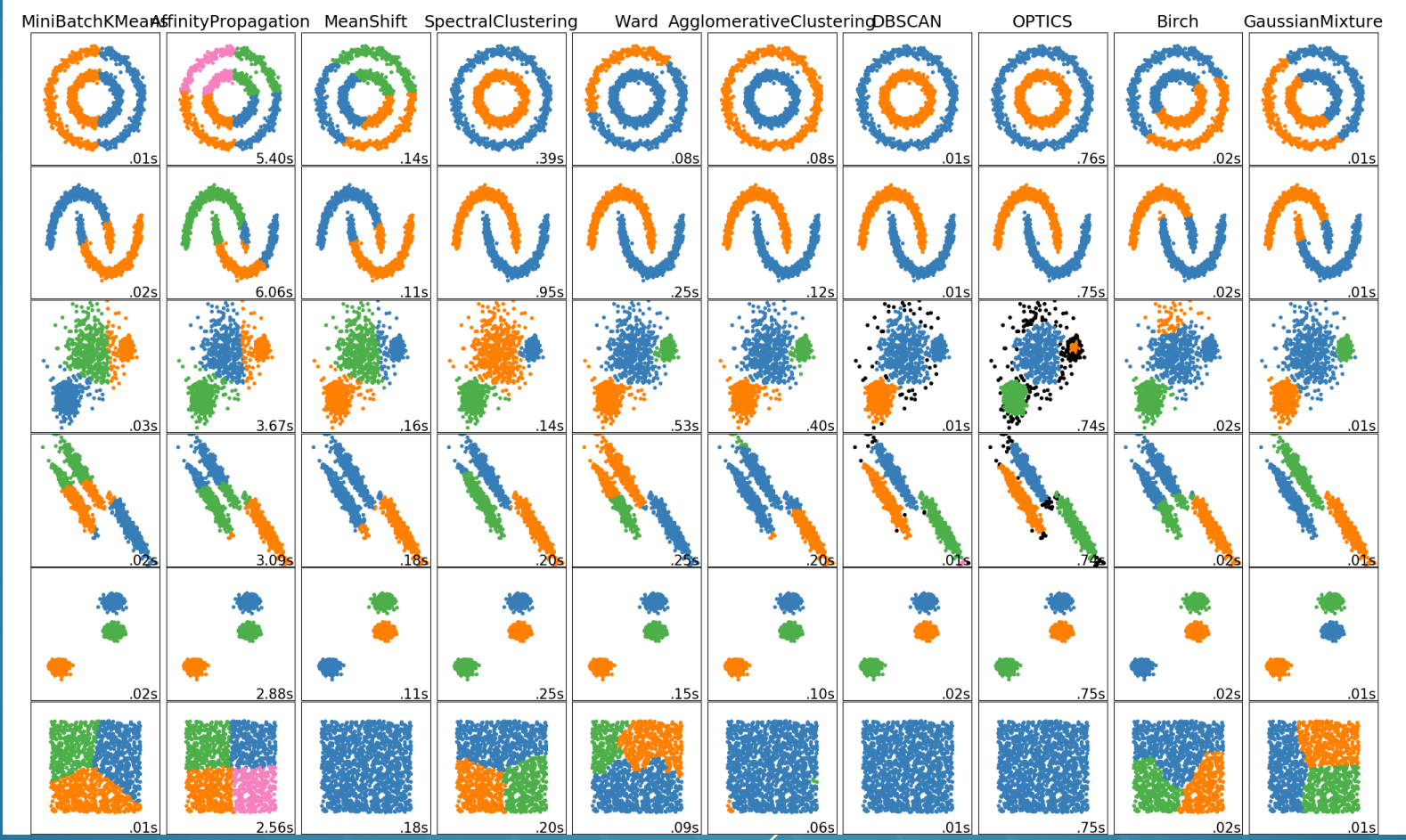


Issues with unsupervised learning



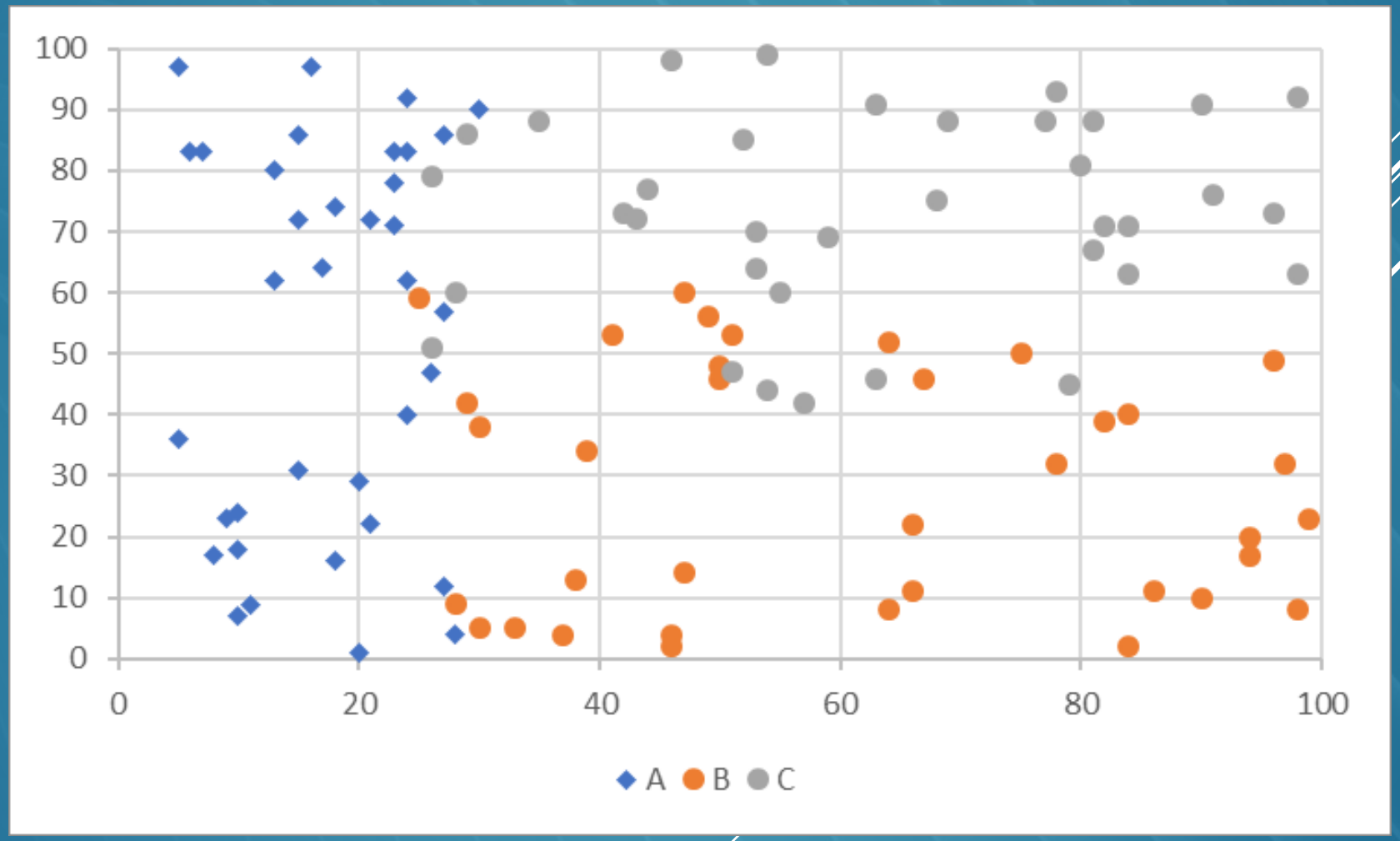
https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Issues with unsupervised learning

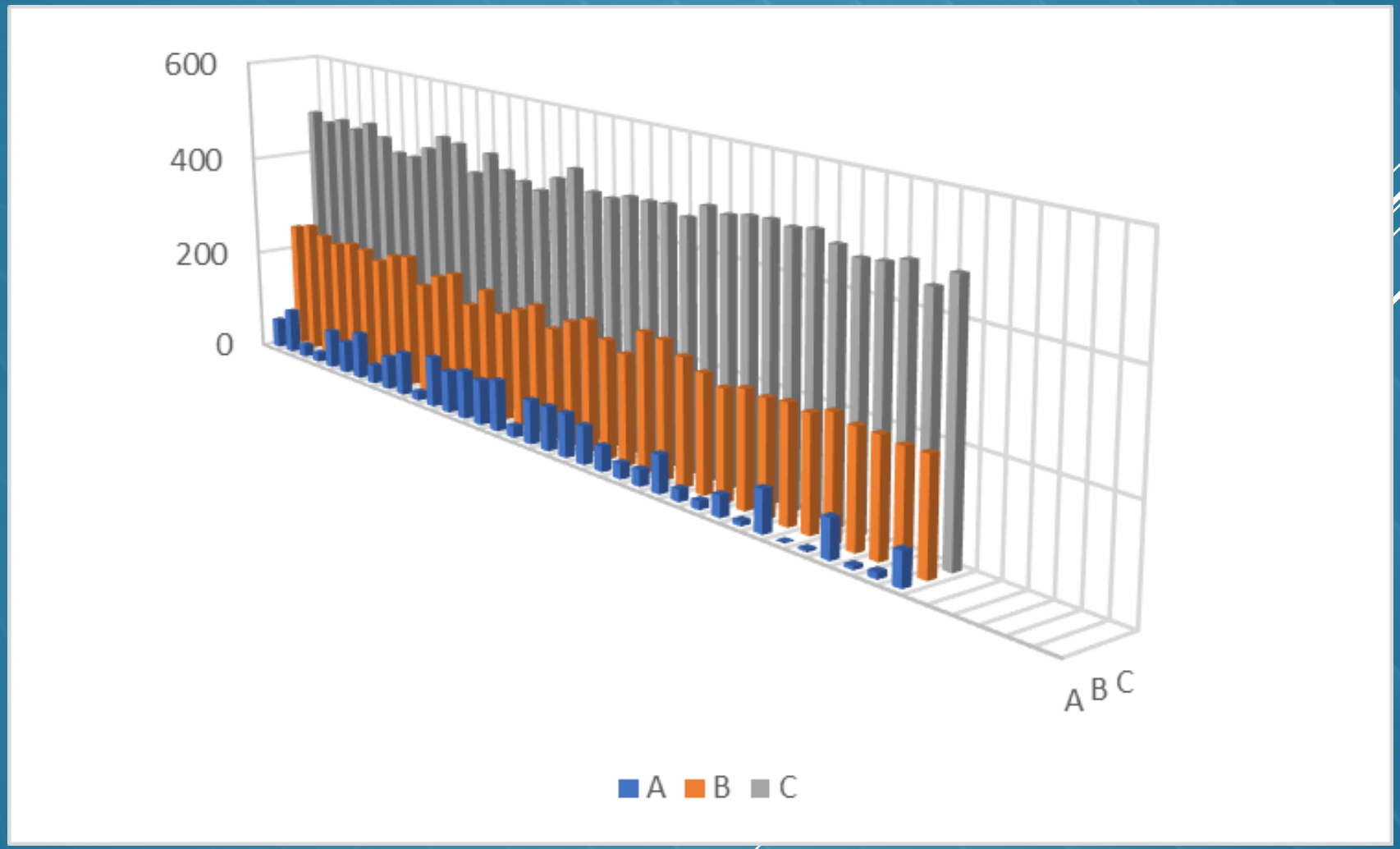


https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Exploiting a third dimension



Exploiting a third dimension

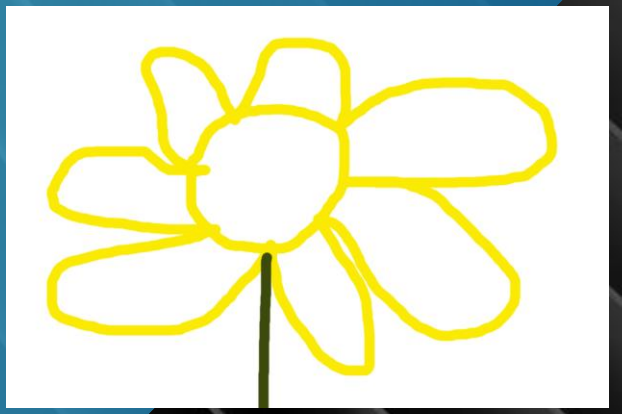


Issues with unsupervised learning

- By default all data is equally valued
- As with supervised learning, if you control the input data, you can control how the classes are formed. By extension allowing you to backdoor a classifier
- If no one is manually classifying the data, you open yourself up to supply chain attacks
- If you know the underlying algorithm or the dataset used for training, you can cater your attacks to abuse biases
- Generally they are less precise than supervised learning, making them more vulnerable to brute force

Why Images Are Hard

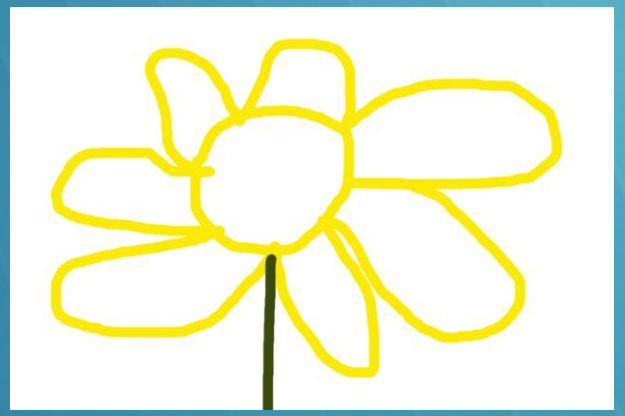
In your head, order these images in order from least to most changed



https://pxhere.com/en/photo/1092972

Pixelwise Comparison

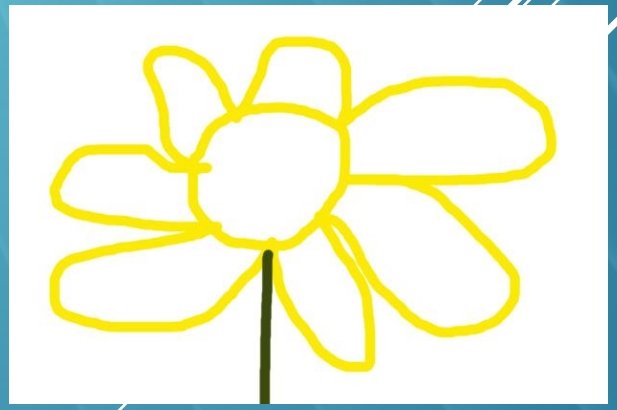
Least ---------- Most



https://pxhere.com/en/photo/1092972

Bitwise comparison (Hamming Distance)

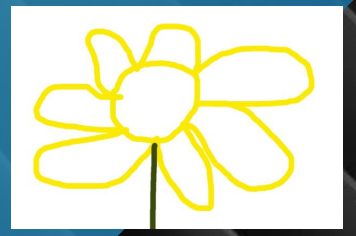
Least ---------- Most



https://pxhere.com/en/photo/1092972

Perpetual Hash (P-Hash)

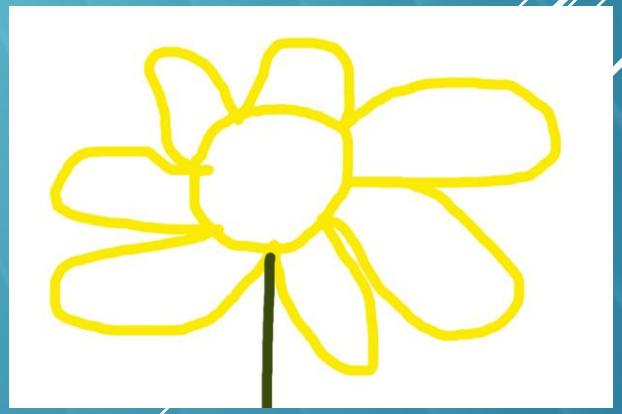
Least ---------- Most



https://pxhere.com/en/photo/1092972

Euclidian (Sum of pixelwise) distance between images

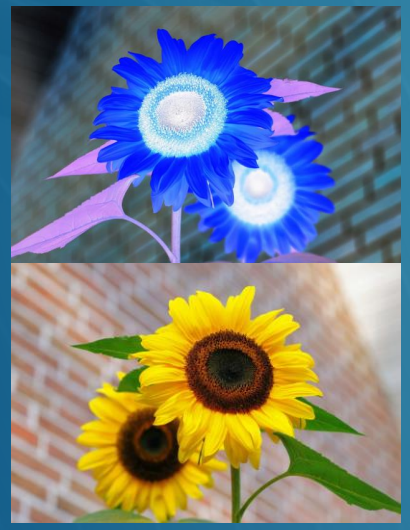
Least ---------- Most

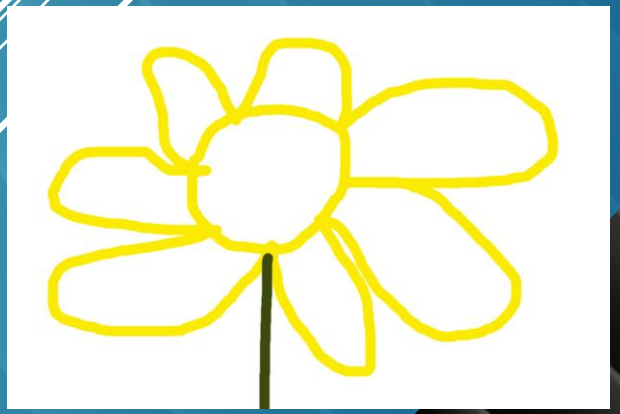


https://pxhere.com/en/photo/1092972

Image Loss (Reversibility)

Least ---------- Most

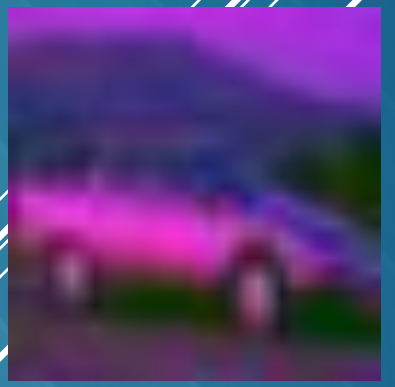




https://pxhere.com/en/photo/1092972

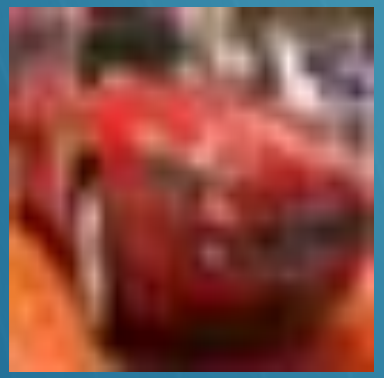
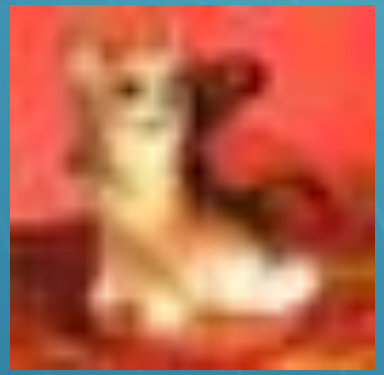
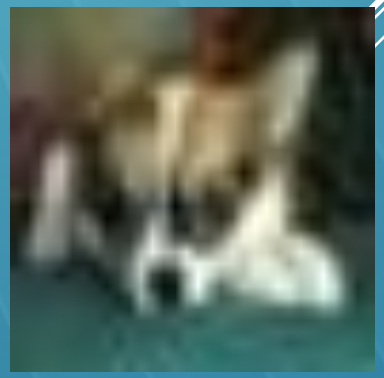
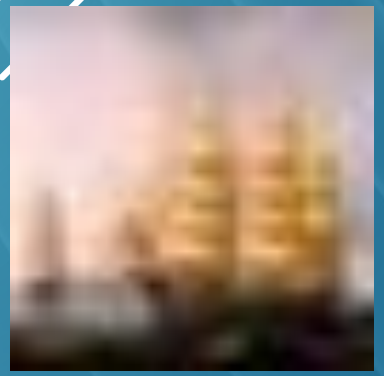
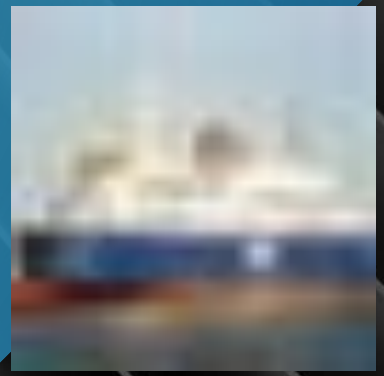
Challenge #1: P-Hack K-means

A website allows you to upload files to it to add to the database.
The website attempts to classify the images as either dogs, boats, or cars.
It doesn't do particularly well off the starting 3,000 images, because the sample size is small and the images aren't the best quality.



The challenge for this is to modify the dataset to be artificially successful or unsuccessful by exploiting quirks in the K-means algorithm, and the fact that the dataset is user controlled.

Hint: You can p-hack the dataset as whole, or you can tamper with the individual data.

Challenge #2: P-Hack A CNN

This is a challenge from PicoCTF 2018, so all credit for the challenge goes to them.
The website allows you to classify an image as one of 1000 classes.
Your goal is to take an image of a dog, and modify it so that it classifies as an image of a tree frog with +85% confidence while maintaining the integrity of the image (Checked via a P-Hash)

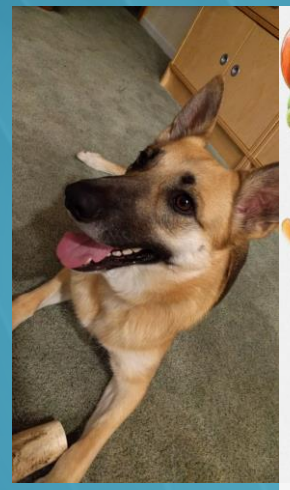
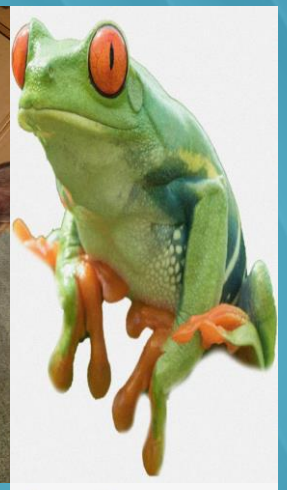
You are given a copy of the CNN model.

The key things to google to help you out:
Adversarial Noise
Gaussian Noise
P-Hashing

Challenge #2: P-Hack A CNN

Hint: This is my table of dogs and frogs

I have provided starting code for 3 difficulties.
I would start with the harder ones because the easy ones may spoil it for you

CNN Classification \ My Classification	Dog	Frog
Dog		
Frog		

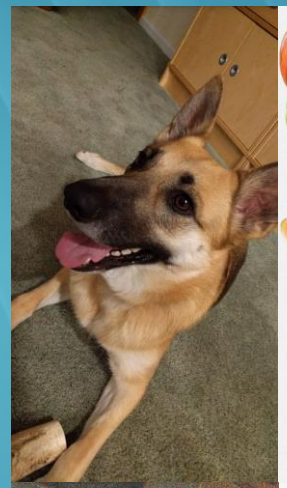
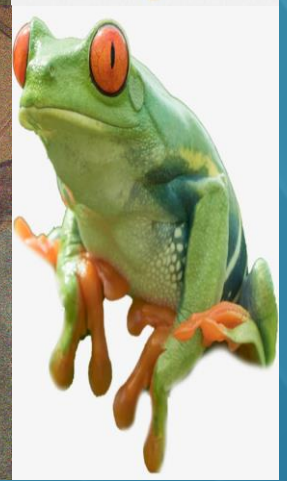
Challenge #1: P-Hack K-Means

P-hack the dataset:
You can K-means the dataset, then remove anything that doesn't fit, then K-means it again, and repeat.
Pros: Every data point is valid and untouched, and can pass a visual inspection.
Cons: You lose a large portion of the dataset in the process.

Modify the dataset:
This can be done significantly more subtly, however for my example I just add a coloured border to each image to denote its class, making the dataset look like this:

Challenge #2: P-Hack A CNN

My finished table of dogs and frogs

CNN Classification \ My Classification	Dog	Frog
Dog		
Frog		

Steps:

1. Add gaussian noise to the image
2. Check the image confidence against the CNN
3. If it has higher tree frog confidence it updates current image
4. Repeat steps 2 and 3 until confidence is +85%

Take Away Points

- Treat datasets as untrusted data.
- Don't have user-controlled datasets
- Be aware of the limitations of the tech.
- Protect your neural net models the way you would secret keys.

GANs: The Future of Breaking ML

- They are essentially an AI trained to beat another AI
- If you want to learn how to really destroy some modern machine learning, instead of just backdooring a dataset or exploiting blind spots, these are the thing to research
- GANs require a starting model, but once you have that, its game over

GANs: The Future of Breaking ML

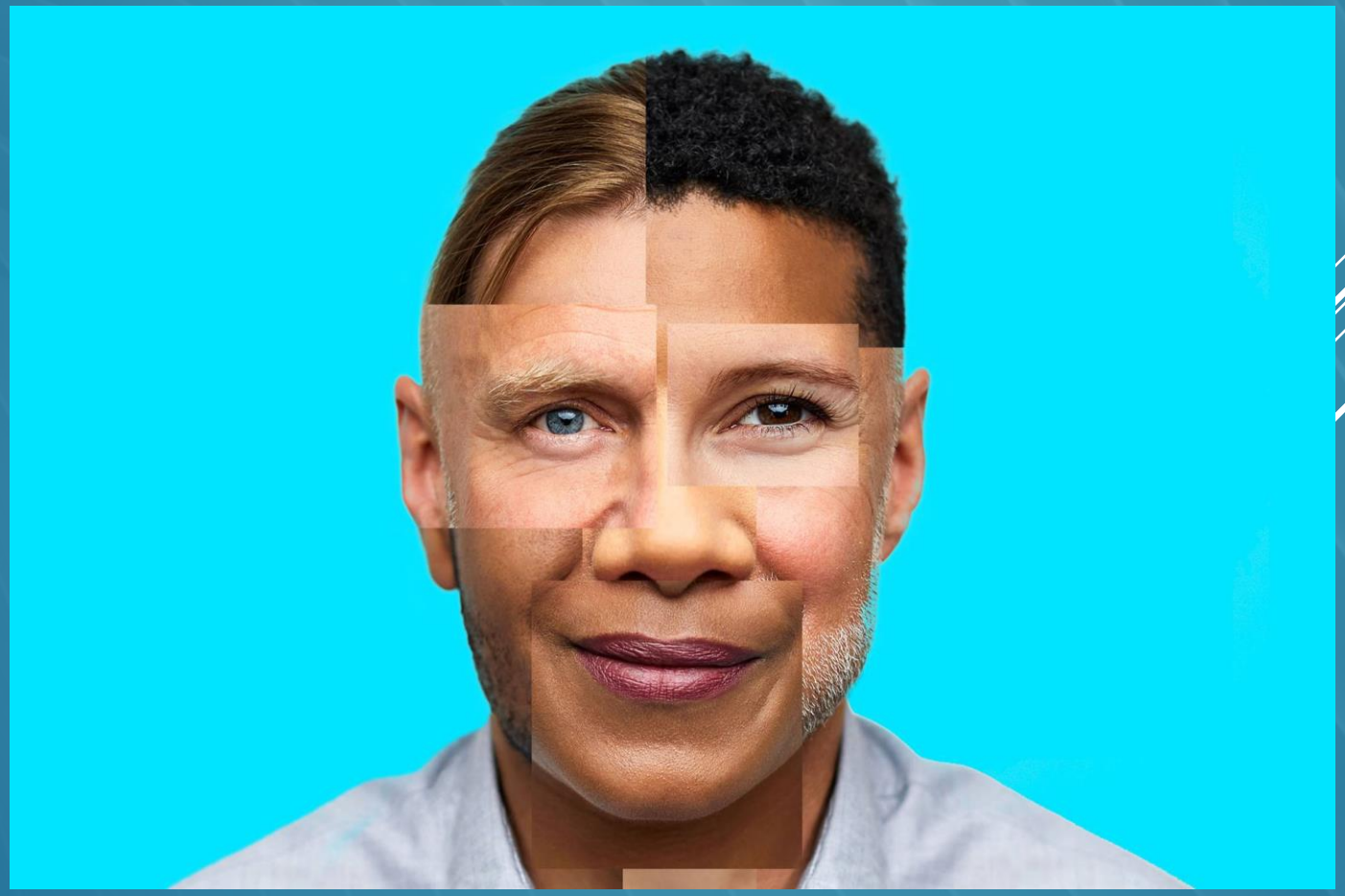
- They are essentially an AI trained to beat another AI
- If you want to learn how to really destroy some modern machine learning, instead of just backdooring a dataset or exploiting blind spots, these are the thing to research
- GANs require a starting model, but once you have that, its game over
- Generative Adversarial Networks are terrifying

One final thought exercise

What is a face?

One final thought exercise

Is this a face?



One final thought exercise

Is this a face?

https://thispersondoesnotexist.com/

Thanks for listening

The slides will be on my GitHub after this. https://Github.com/JankhJankh

I hope you all learned something, or gained some new perspective