Insight into Machine Learning

Overview

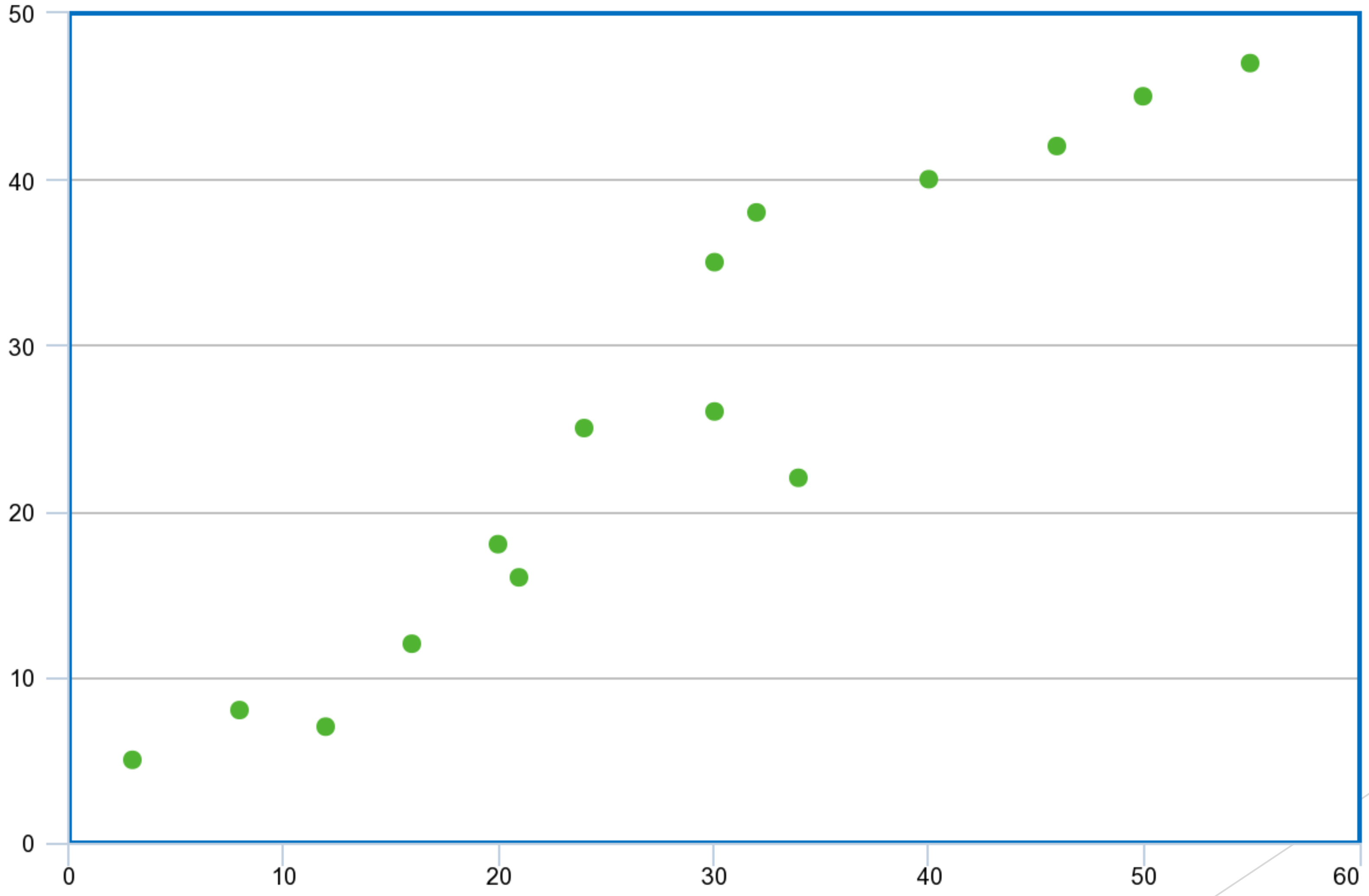
1. What is Machine Learning
   - Linear regression
   - Overfitting
   - What is it good for
   - ML Flowchart
   - Types of Machine Learning
2. Understanding your data
   - Garbage in, garbage out
   - What is my purpose?
3. Algorithms
   - Which one do I need?
   - Decision Trees
   - K-means clustering
4. Evaluation metrics
   - Classification Accuracy
   - ROC Curves and Space
5. Azure Machine Learning Studio

Google

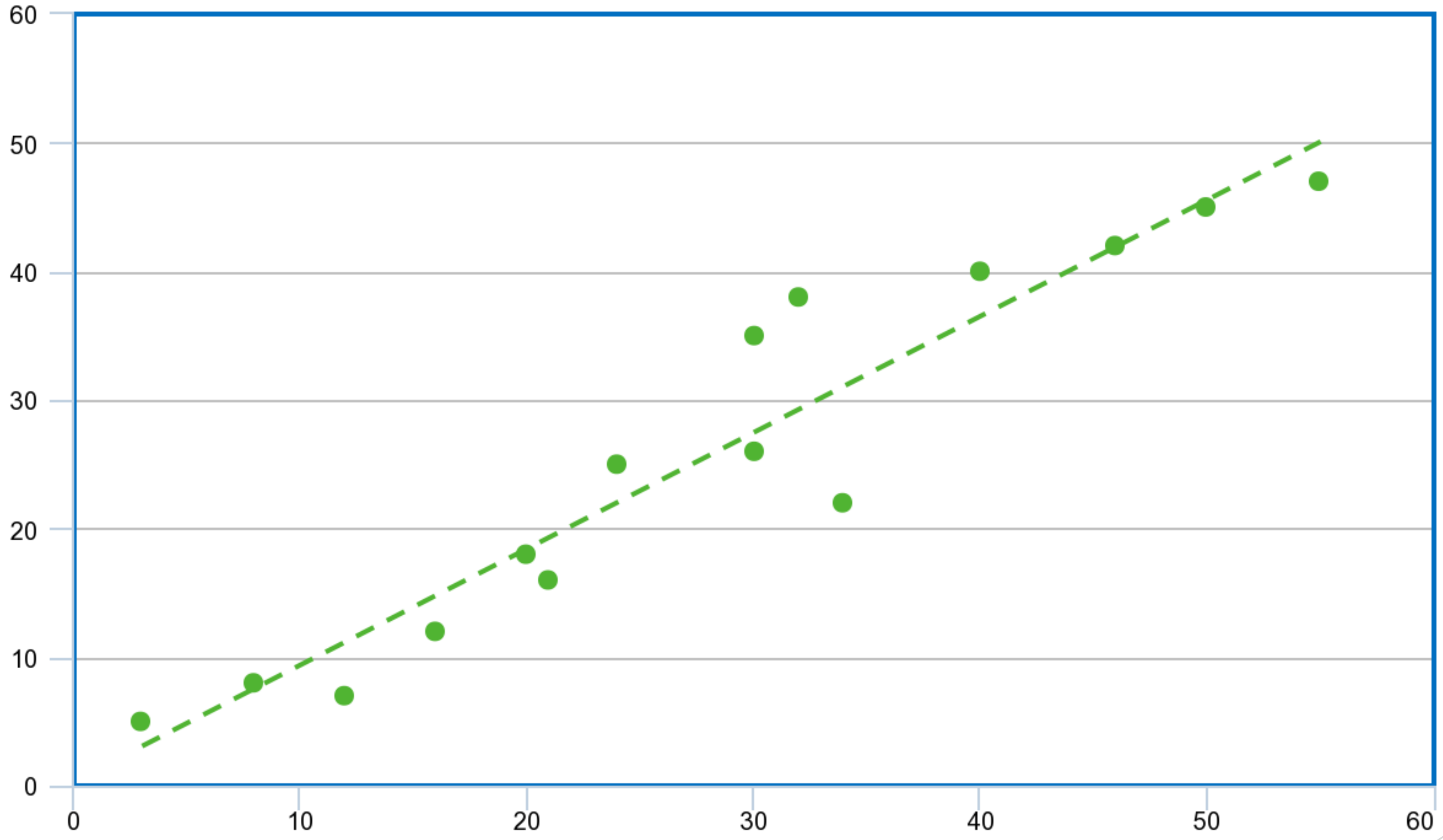
what is machine learning

what is machine learning
what is machine learning **used for**
what is machine learning **quora**
what is machine learning **algorithms**
what is machine learning **and deep learning**
what is machine learning **and artificial intelligence**
what is machine learning **and ai**
what is machine learning **course**
what is machine learning **in python**

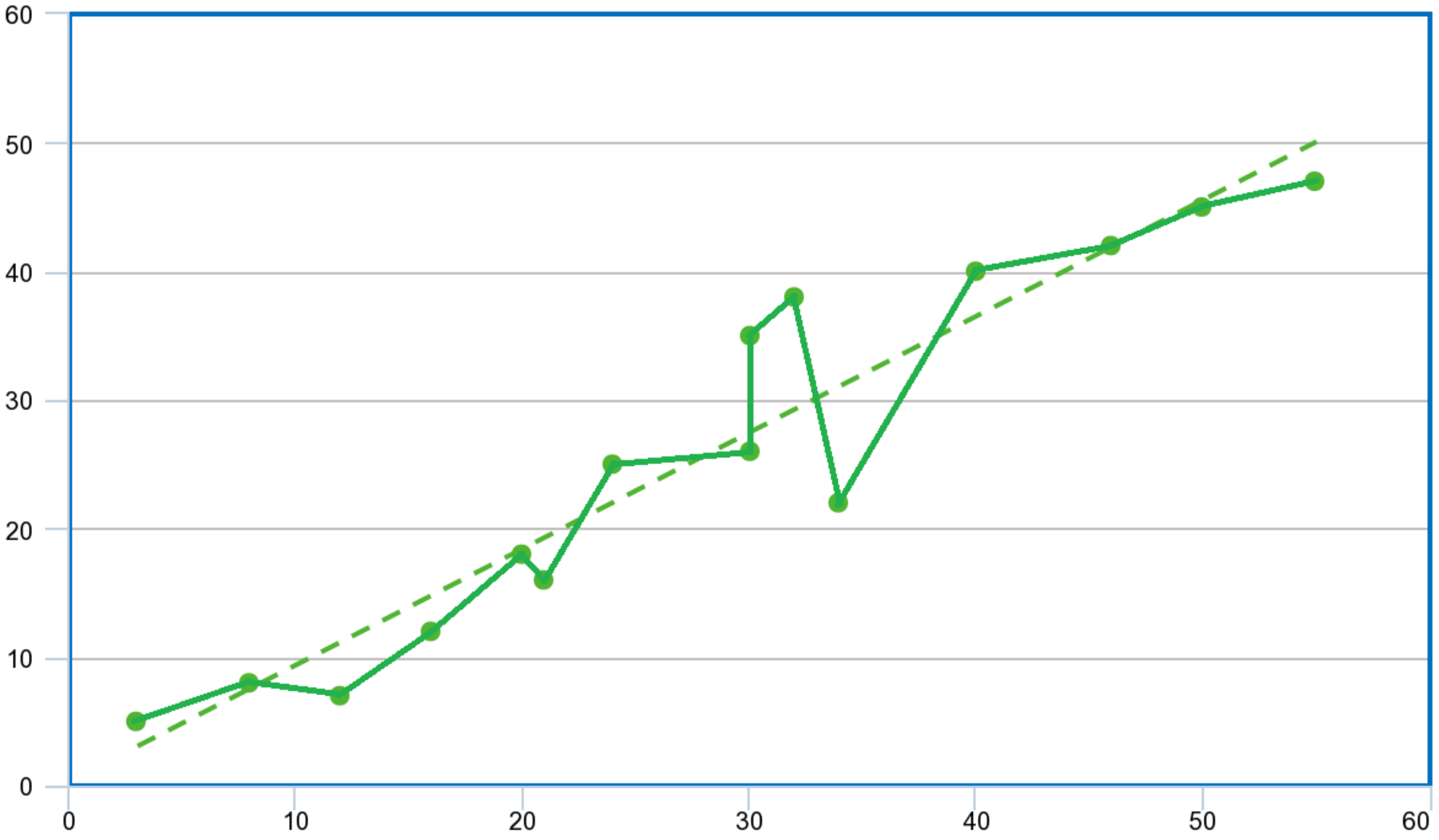
Linear regression



Linear regression



Overfitting



- What is it good for
  - Data security – identifying malware
  - Financial trading – stock markets
  - Fraud detection – money laundering
  - Market Personalization – advertisements
  - Computer vision – image recognition, sentiment analysis
  - Recommendations – Netflix, Amazon

- What is it good for
  - Data security – identifying malware
  - Financial trading – stock markets
  - Fraud detection – money laundering
  - Market Personalization – advertisements
  - Computer vision – image recognition, sentiment analysis
  - Recommendations – Netflix, Amazon

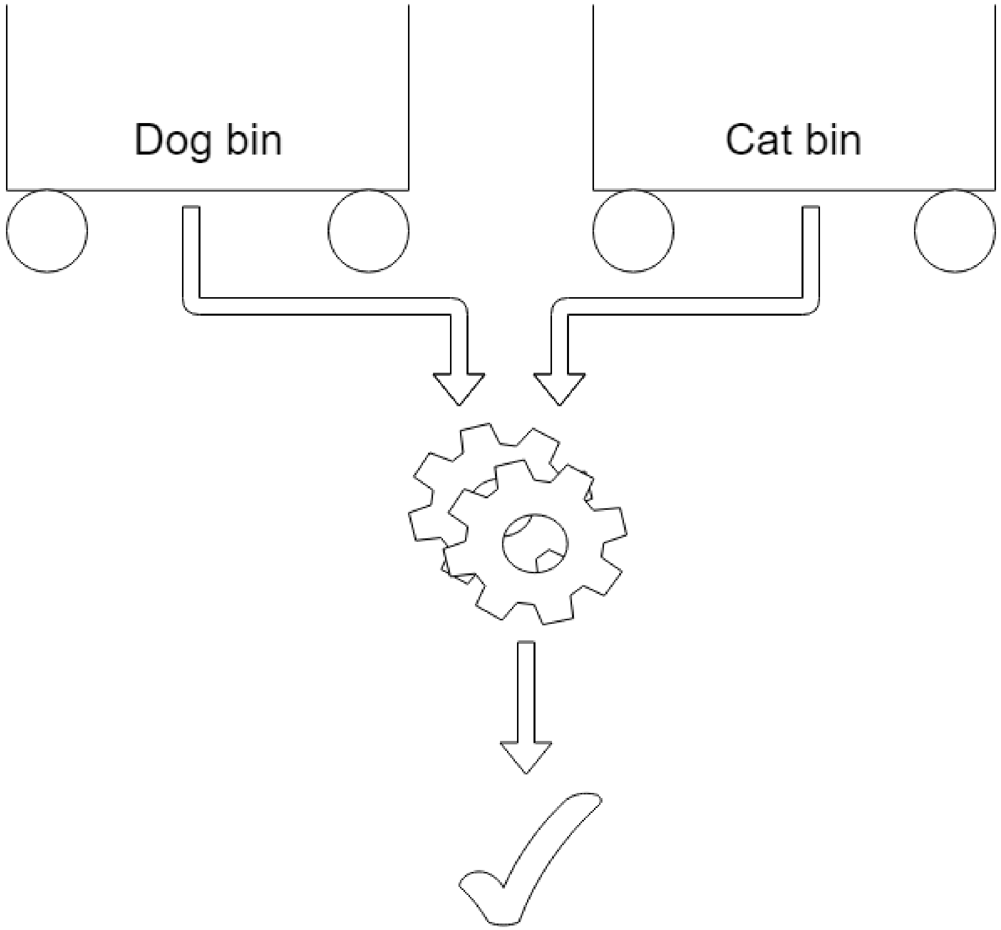
but just as important…

Is it a dog…

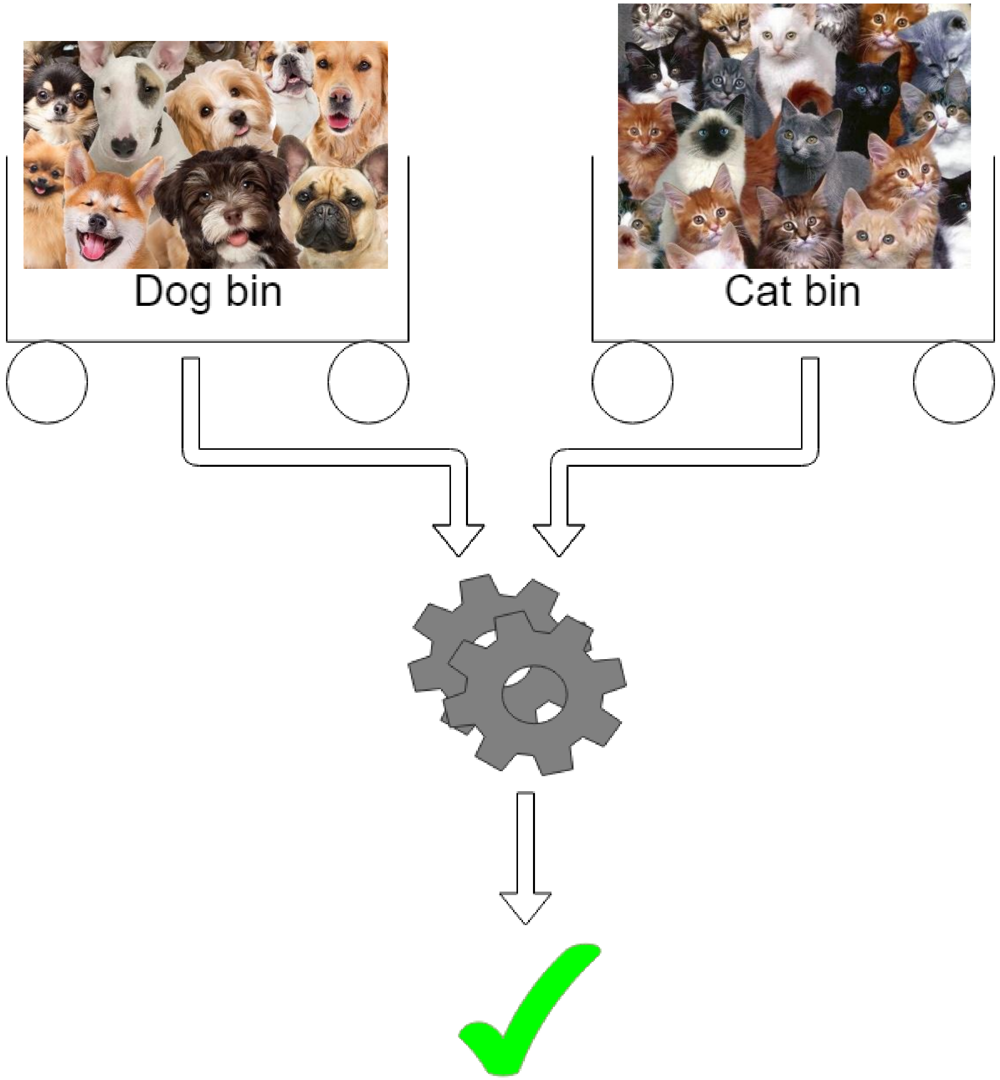
Is it a dog…

… or is it a cat?

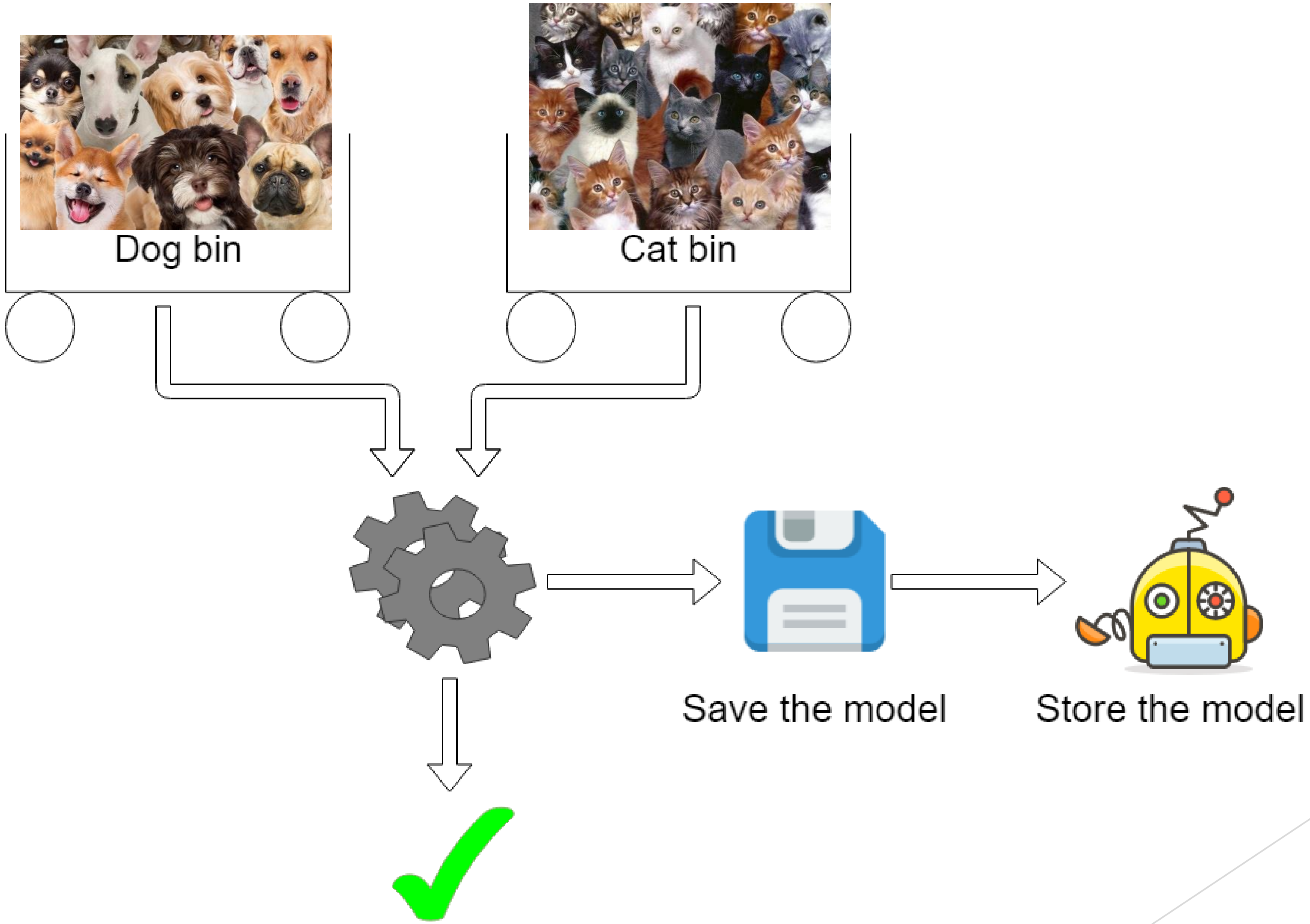
- ML Flowchart



- ML Flowchart



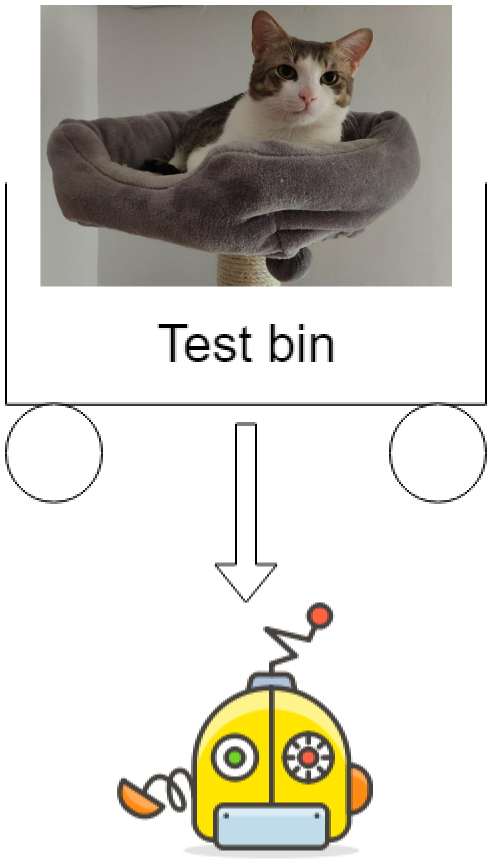
ML Flowchart



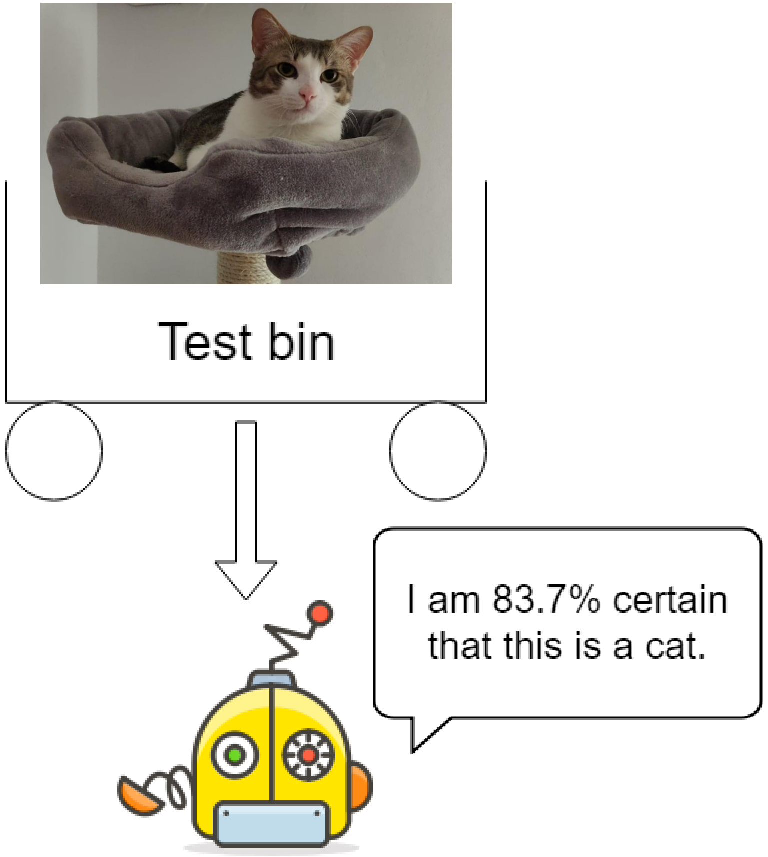
- ML Flowchart

Alright, now what about this fellow?

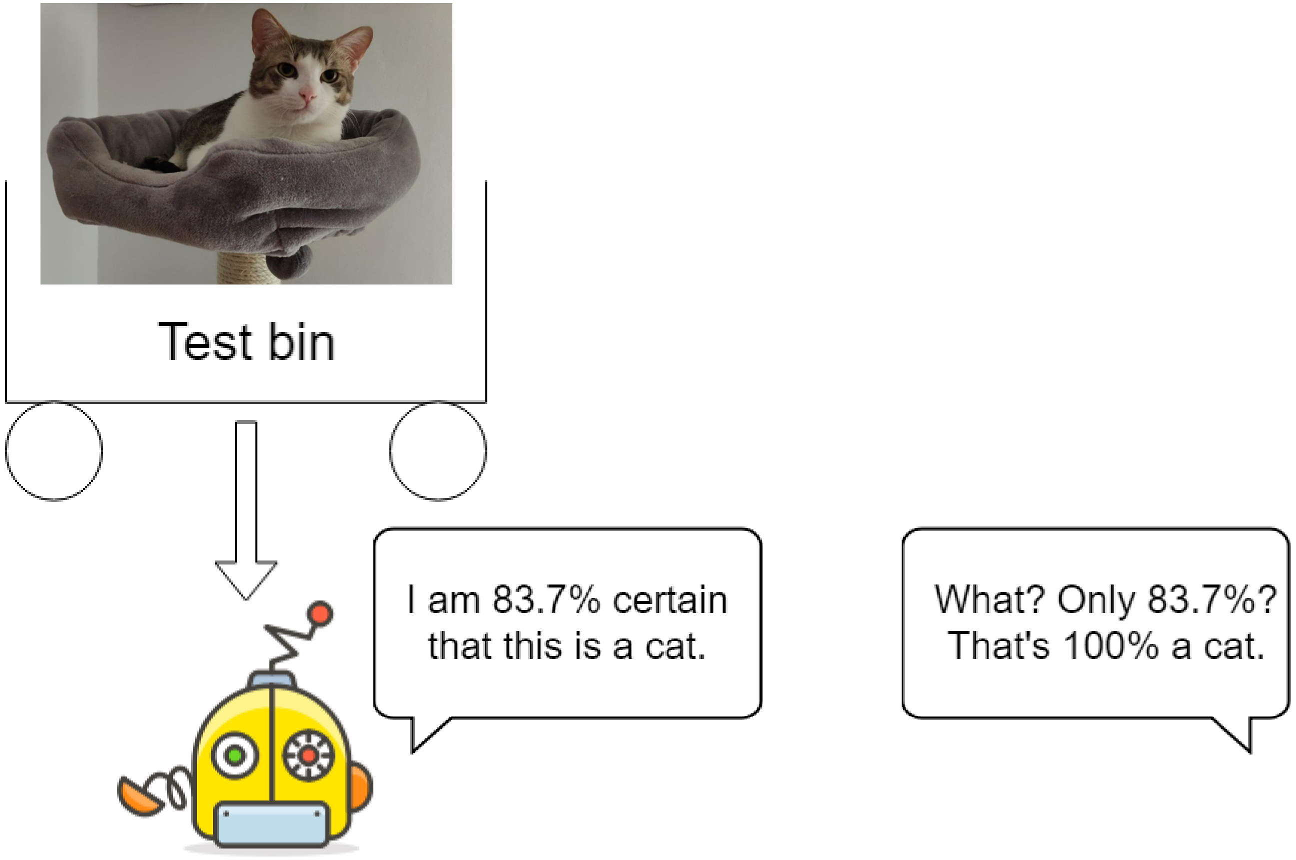
- ML Flowchart



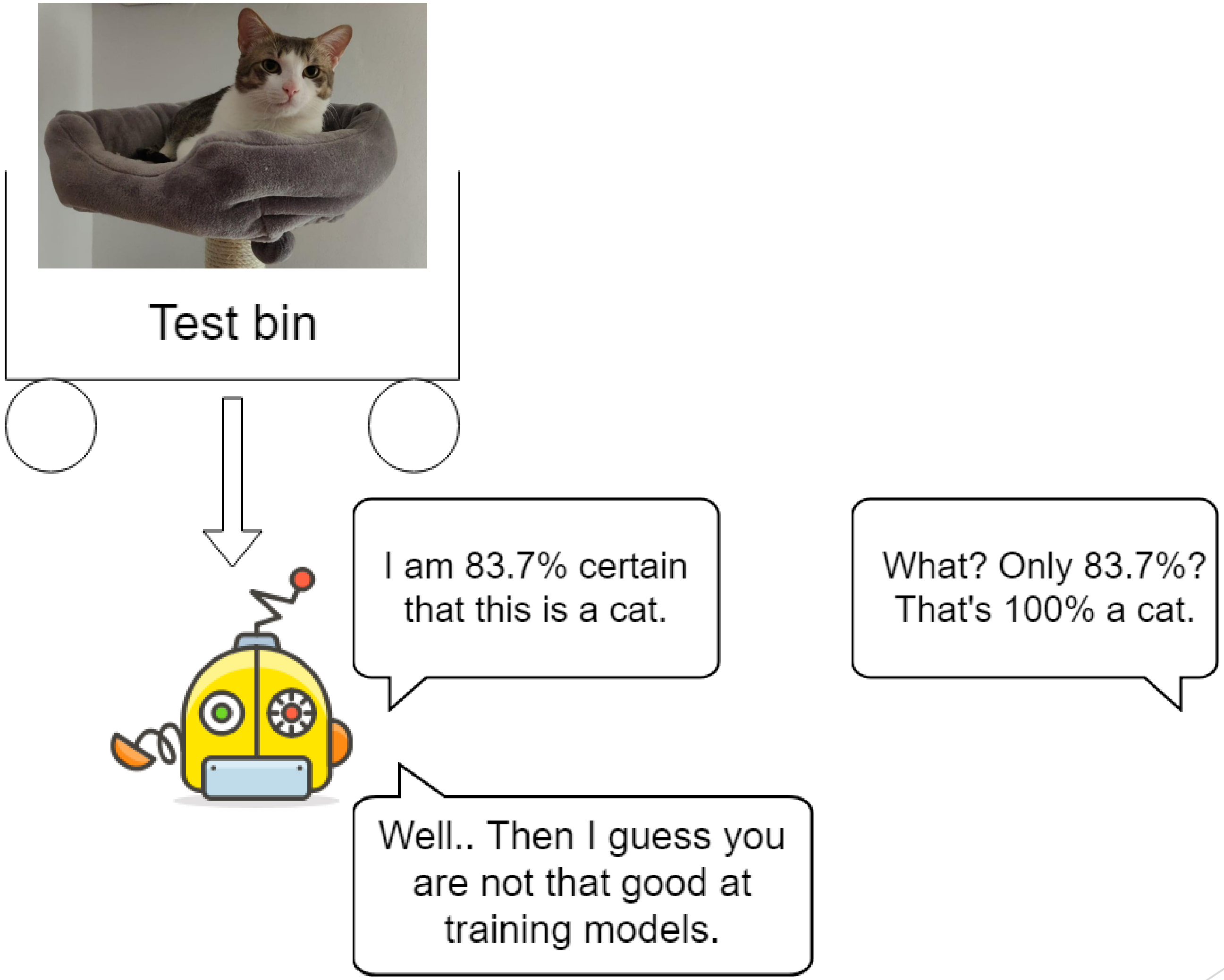
- ML Flowchart



ML Flowchart



ML Flowchart



- Types of Machine Learning

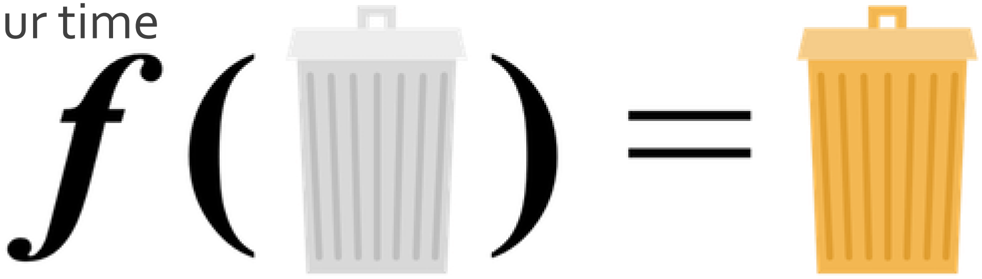
1. Supervised – mail filtering
   i. Has access to a complete dataset for training
   ii. Tries to predict the value of a missing data point in an incomplete dataset
   iii. Labelled data
2. Unsupervised – clustering
   i. Look for patterns inside the data without being offered any help in interpreting the data
   ii. Unlabelled data
3. Reinforcement – robotics
   i. Tries to make a prediction or solve a problem and is given feedback to know if the result was correct.

Understanding your data

Garbage in, garbage out

Things to keep in mind:

- Missing values – remove record or fill missing values
- Duplicates – remove record
- Spelling errors
- Feature relevance
- Normalization
- Aggregations
- Discretization – categories instead of numerical values
- Take your time



https://blog.temboo.com/make-smart-predictions-with-amazon-machine-learning/

What is my purpose?

Categorize:

1. By output:
   i. If the output is a number then it's a regression problem
   ii. If the output is a set of groups then it's a clustering problem
   iii. If the output is a class then it's a classification problem
   iv. Anomaly detection
2. Constraints:
   i. Storage capacity
   ii. Prediction speed (road signs in autonomous driving)
   iii. Learning speed

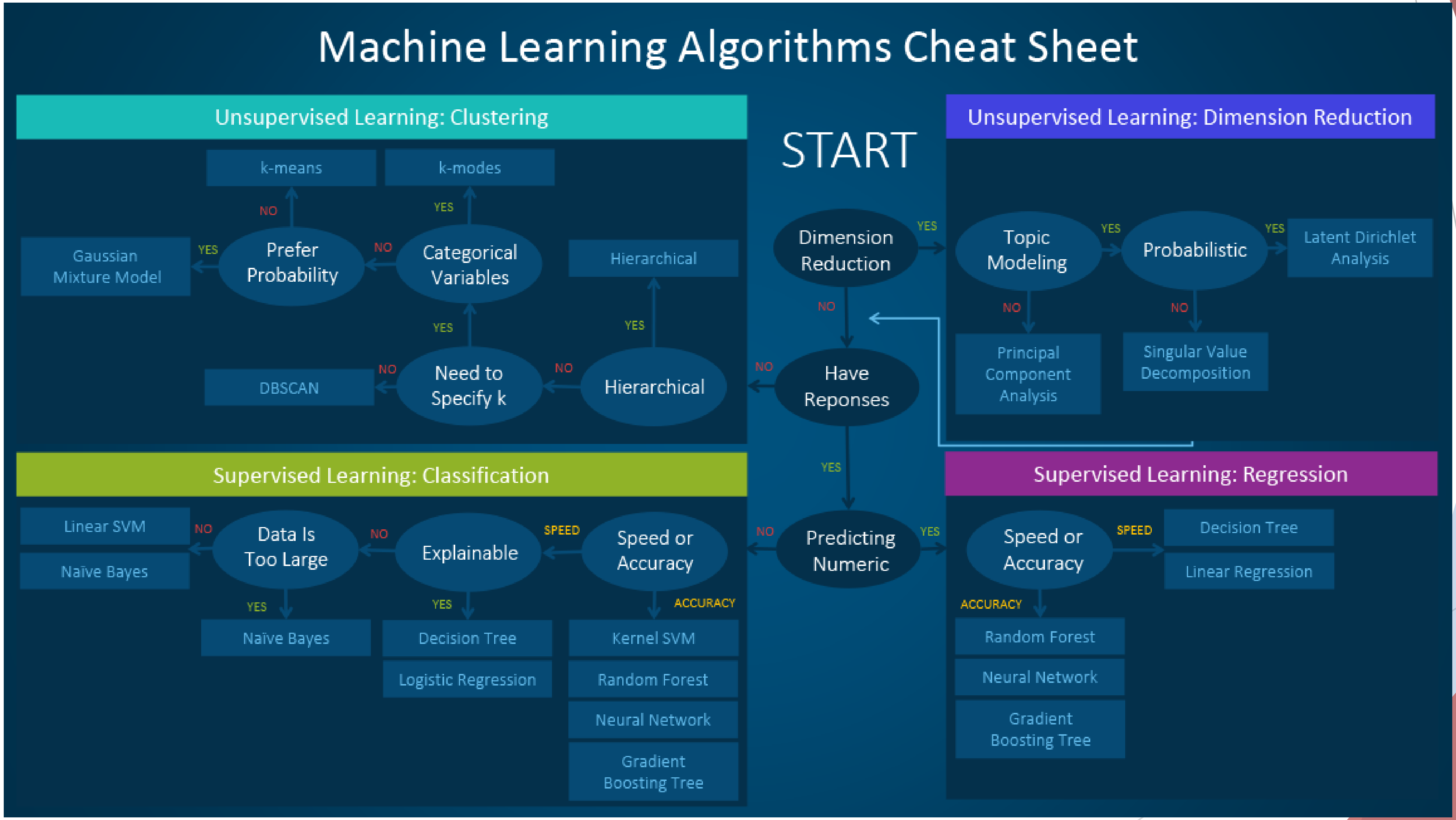
Algorithms

Which one do I need?

Affecting factors:

- Accuracy
- Scalability
- How much pre-processing the model needs
- Business goals
- Model complexity:
  - How many features to learn and predict
  - Computational overhead (single decision tree vs. random forest)
  - Hyperparameters
- Too much complexity can lead to overfitting

- Which one do I need?



https://blogs.sas.com/content/subconsciousmusings/2017/04/12/machine-learning-algorithm-use/

Decision trees

- They try all possible splits using all possible values of each input attribute.
- Used to build classification or regression models in the form of a tree structure.
- Final result represents a tree with decision nodes and leaf nodes.
- Recursive

Decision trees

- They try all possible splits using all possible values of each input attribute.
- Used to build classification or regression models in the form of a tree structure.
- Final result represents a tree with decision nodes and leaf nodes.
- Recursive

9 Yes 5 No

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Overcast	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Overcast	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Overcast	High	Strong	Yes
D13	Overcast	Normal	Weak	Yes
D14	Rain	High	Strong	No

Decision trees

- They try all possible splits using all possible values of each input attribute.

- Used to build classification or regression models in the form of a tree structure.

- Final result represents a tree with decision nodes and leaf nodes.

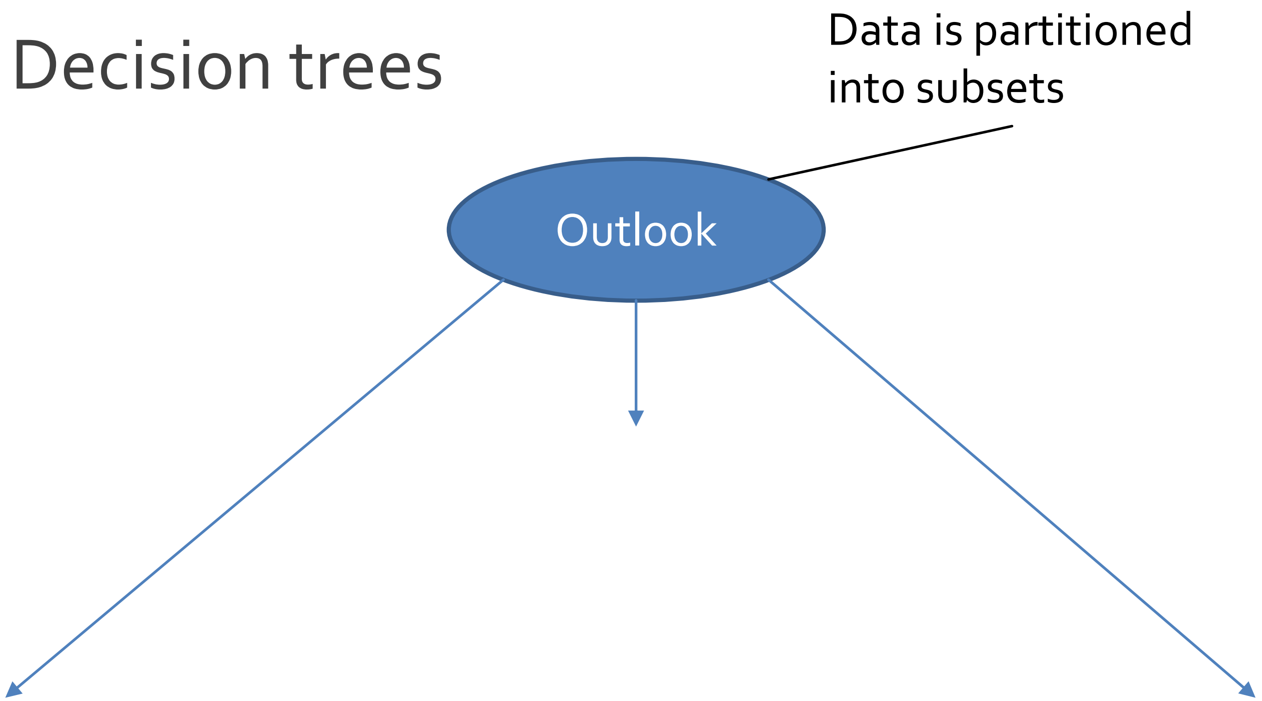
- Recursive

9 Yes 5 No

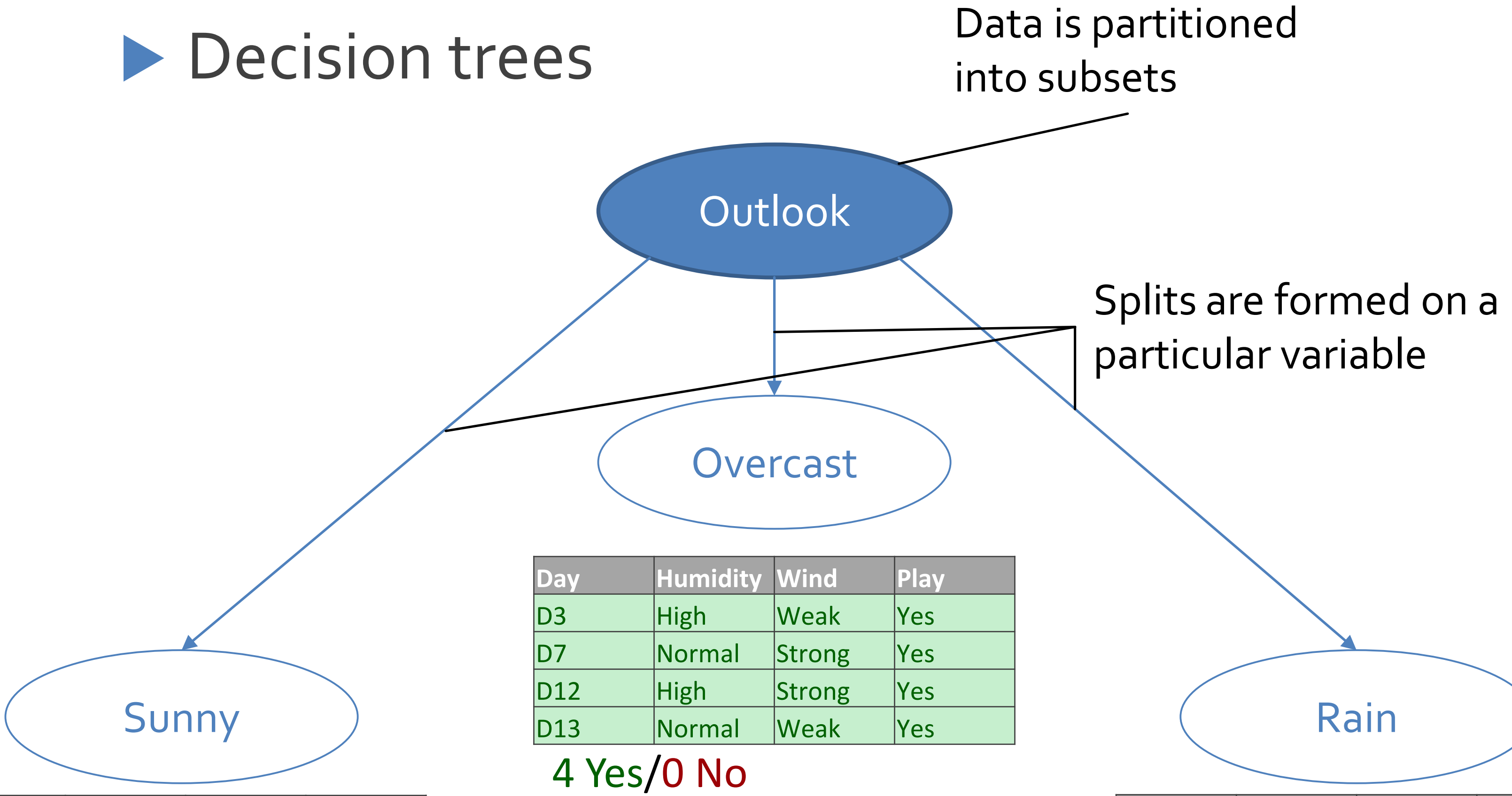
Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Overcast	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Overcast	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Overcast	High	Strong	Yes
D13	Overcast	Normal	Weak	Yes
D14	Rain	High	Strong	No

- Decision trees

- Decision trees



Decision trees



Overcast:

Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Sunny:

Day	Humidity	Wind	Play
D1	High	Weak	No
D2	High	Strong	No
D8	High	Weak	No
D9	Normal	Weak	Yes
D11	Normal	Strong	Yes

2 Yes/3 No

Rain:

Day	Humidity	Wind	Play
D4	High	Weak	Yes
D5	Normal	Weak	Yes
D6	Normal	Strong	No
D10	Normal	Weak	Yes
D14	High	Strong	No

3 Yes/2 No

Decision trees

Data is partitioned into subsets

Outlook

Splits are formed on a particular variable

Overcast

Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Pure

Sunny

Day	Humidity	Wind	Play
D1	High	Weak	No
D2	High	Strong	No
D8	High	Weak	No
D9	Normal	Weak	Yes
D11	Normal	Strong	Yes

2 Yes/3 No

Not pure

Rain

Day	Humidity	Wind	Play
D4	High	Weak	Yes
D5	Normal	Weak	Yes
D6	Normal	Strong	No
D10	Normal	Weak	Yes
D14	High	Strong	No

3 Yes/2 No

Not pure

Is the subset pure (all yes or all no)?
If yes, then stop.
If not, then repeat.

Decision trees

Data is partitioned into subsets

Outlook

Splits are formed on a particular variable

Overcast

Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Pure

Sunny

Day	Humidity	Wind	Play
D1	High	Weak	No
D2	High	Strong	No
D8	High	Weak	No
D9	Normal	Weak	Yes
D11	Normal	Strong	Yes

2 Yes/3 No

Not pure

Rain

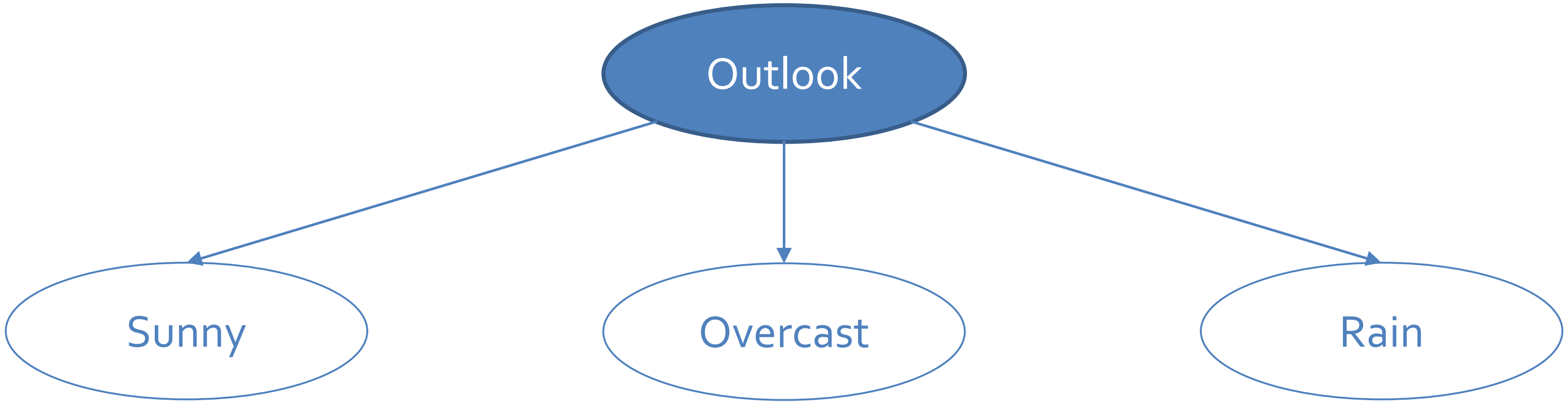
Day	Humidity	Wind	Play
D4	High	Weak	Yes
D5	Normal	Weak	Yes
D6	Normal	Strong	No
D10	Normal	Weak	Yes
D14	High	Strong	No

3 Yes/2 No

Not pure

Is the subset pure (all yes or all no)?
If yes, then stop.
If not, then repeat.

Decision trees

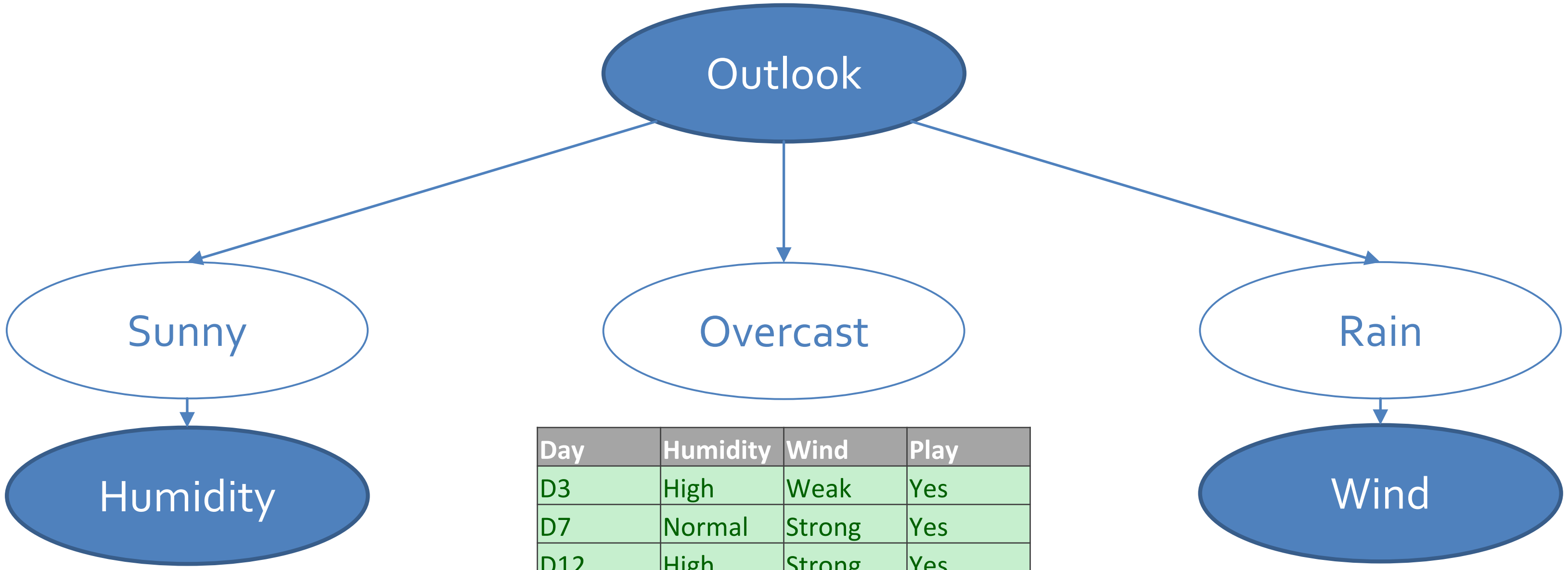


Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Pure

Decision trees

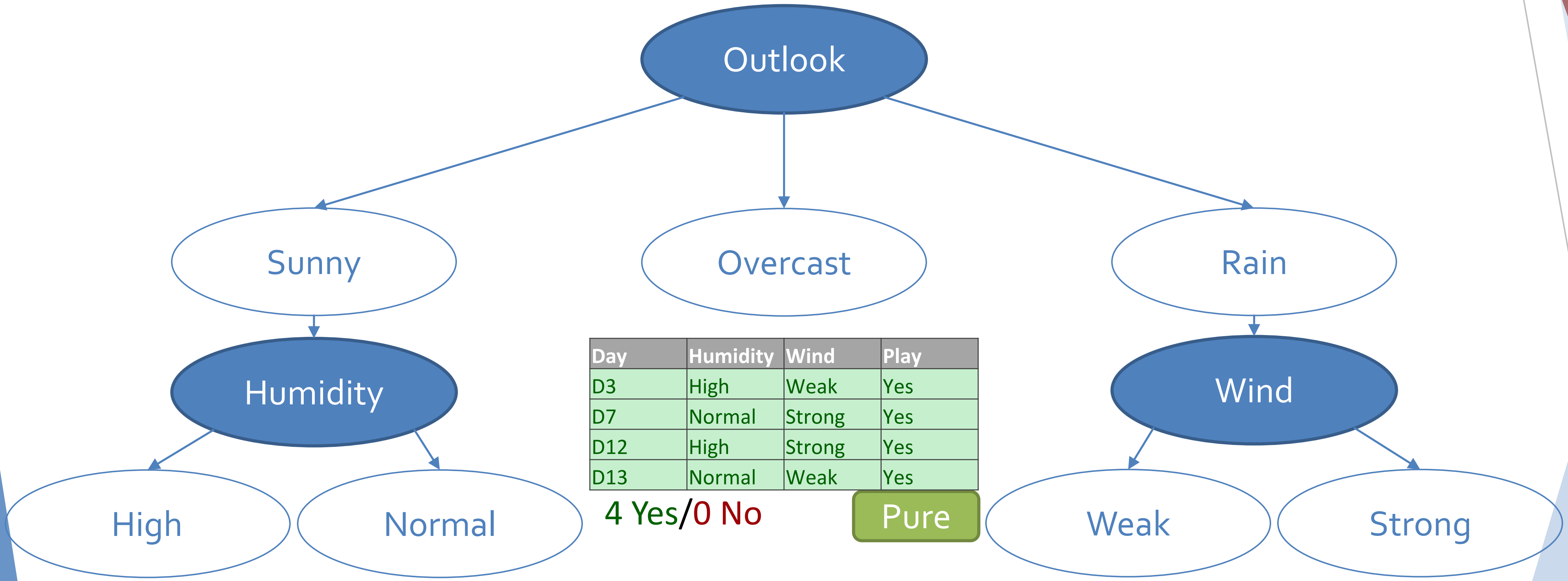


Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Pure

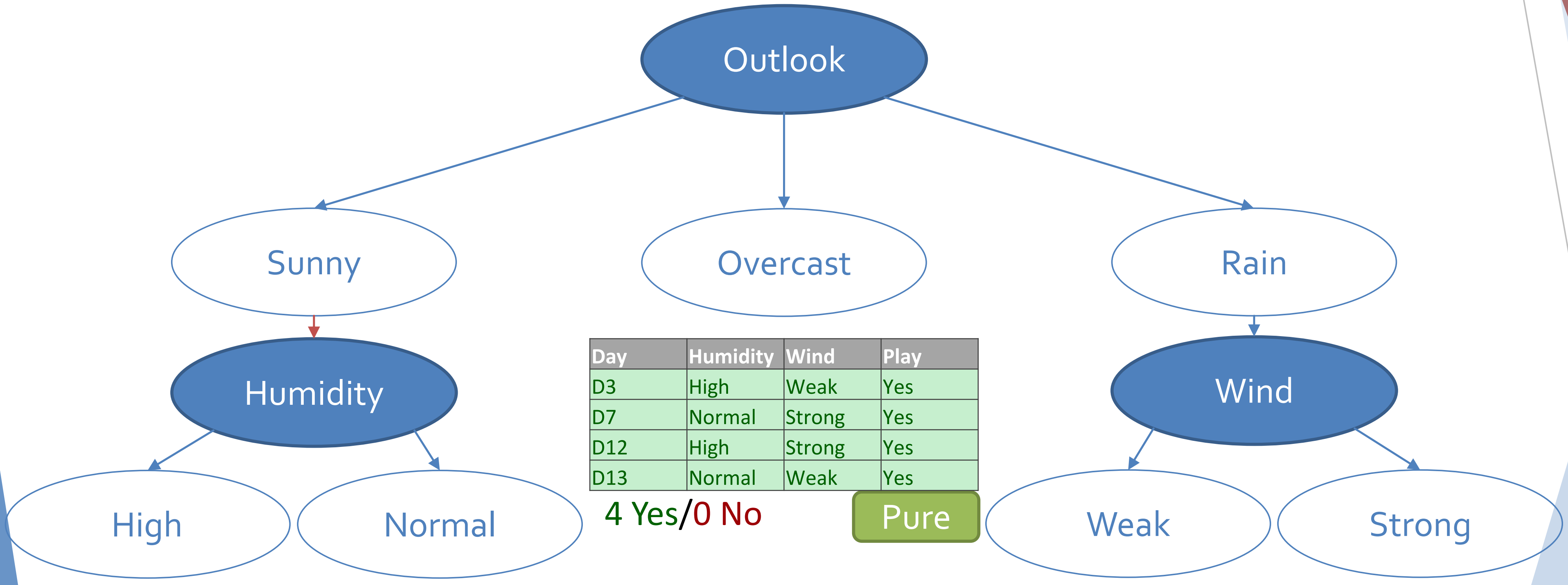
Decision trees



Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

4 Yes/0 No

Decision trees



Day	Humidity	Wind	Play
D3	High	Weak	Yes
D7	Normal	Strong	Yes
D12	High	Strong	Yes
D13	Normal	Weak	Yes

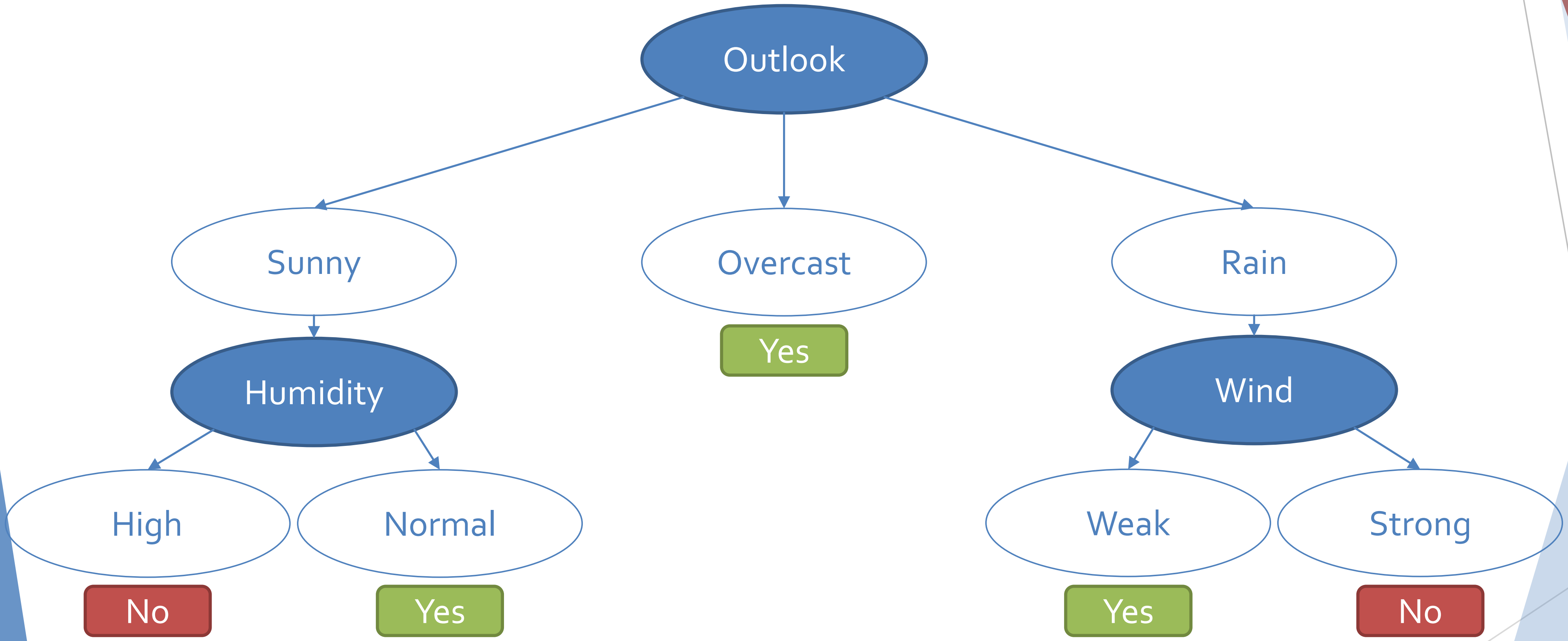
Day	Wind	Play
D1	Weak	No
D2	Strong	No
D8	Weak	No

Day	Wind	Play
D9	Weak	Yes
D11	Strong	Yes

Day	Humidity	Play
D4	High	Yes
D5	Normal	Yes
D10	Normal	Yes

Day	Humidity	Play
D6	Normal	No
D14	High	No

- Decision trees



Decision trees

How to determine on which attribute to split on

- Entropy:
  - The degree of disorder or uncertainty in a system
  - Maximum uncertainty (no prediction probability) – Entropy = 1
    - a coin toss is an example of uniform probability
  - If there is no uncertainty (we can always predict the result) – Entropy = 0

Decision trees

How to determine on which attribute to split on

- Entropy:
  - The degree of disorder or uncertainty in a system
  - Maximum uncertainty (no prediction probability) – Entropy = 1
    - a coin toss is an example of uniform probability
  - If there is no uncertainty (we can always predict the result) – Entropy = 0

$$H(S) = -p_{(+)} log_2 p_{(+)} - p_{(-)} log_2 p_{(-)}$$

  - How many bits do we need to tell if X is positive or negative (X belongs to S)

Decision trees

How to determine on which attribute to split on

- Entropy:
  - The degree of disorder or uncertainty in a system
  - Maximum uncertainty (no prediction probability) – Entropy = 1
    - a coin toss is an example of uniform probability
  - If there is no uncertainty (we can always predict the result) – Entropy = 0

$$H(S) = -p_{(+)} log_2 p_{(+)} - p_{(-)} log_2 p_{(-)}$$

  - How many bits do we need to tell if X is positive or negative (X belongs to S)
- The dataset is split on the different attributes and the entropy for each branch is calculated resulting in the total entropy for the split.
- The result is subtracted from the initial entropy (before the split)
  - The result is the **Information Gain** (decrease in entropy)
  - The attribute with the largest Information Gain is chosen

Decision trees

Pruning and overfitting

Decision trees

Pruning and overfitting

- The algorithm tends to split until all nodes are pure
  - Complexity increases

Decision trees

Pruning and overfitting

- The algorithm tends to split until all nodes are pure
  - Complexity increases
  - Accuracy increases for the train data …

Decision trees

Pruning and overfitting

- The algorithm tends to split until all nodes are pure
  - Complexity increases
  - Accuracy increases for the train data …
  - … but it drops for the test data

Decision trees

Pruning and overfitting

- The algorithm tends to split until all nodes are pure
  - Complexity increases
  - Accuracy increases for the train data …
  - … but it drops for the test data
- Pruning prevents this
  - Grow and then post-prune, using the validation set
  - For each node remove it and its children and revalidate

Decision trees

Pruning and overfitting

- The algorithm tends to split until all nodes are pure
  - Complexity increases
  - Accuracy increases for the train data ...
  - ... but it drops for the test data
- Pruning prevents this
  - Grow and then post-prune, using the validation set
  - For each node remove it and its children and revalidate
  - Remove the node that results in the greatest improvement
  - Repeat until further pruning becomes harmful on the validation set

- K-means clustering

Unsupervised algorithm (the given dataset is unlabeled)

K-means clustering

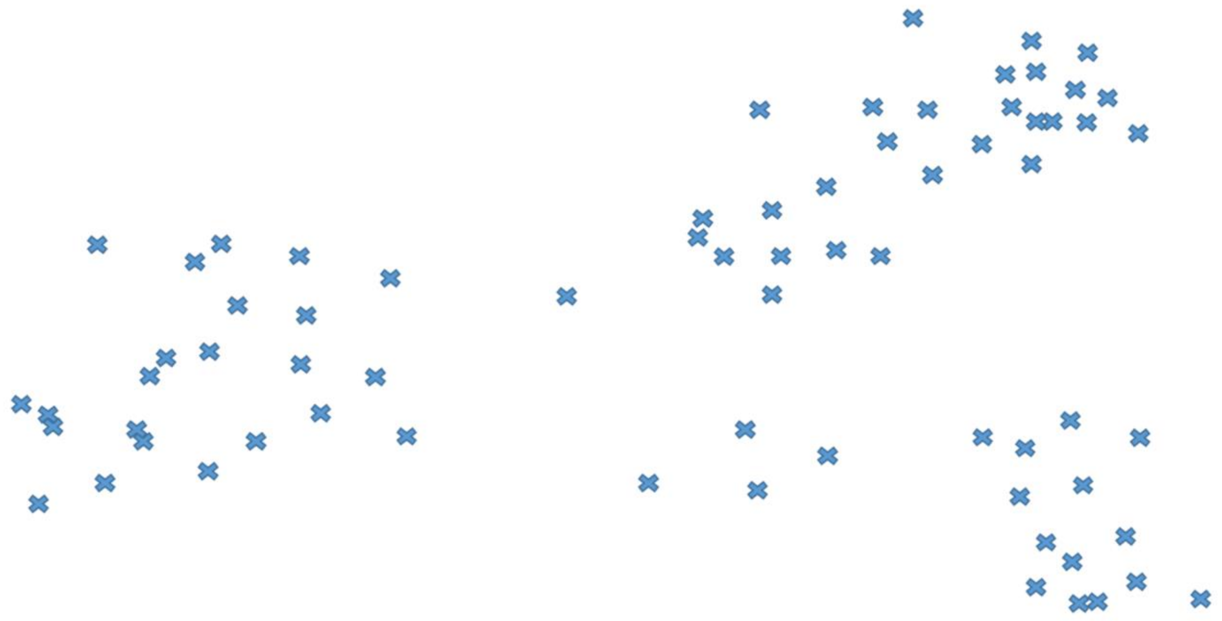
Unsupervised algorithm (the given dataset is unlabeled)

- Used to find clusters with similar characteristics
- There is no outcome to be predicted
- K denotes the number of clusters

K-means clustering

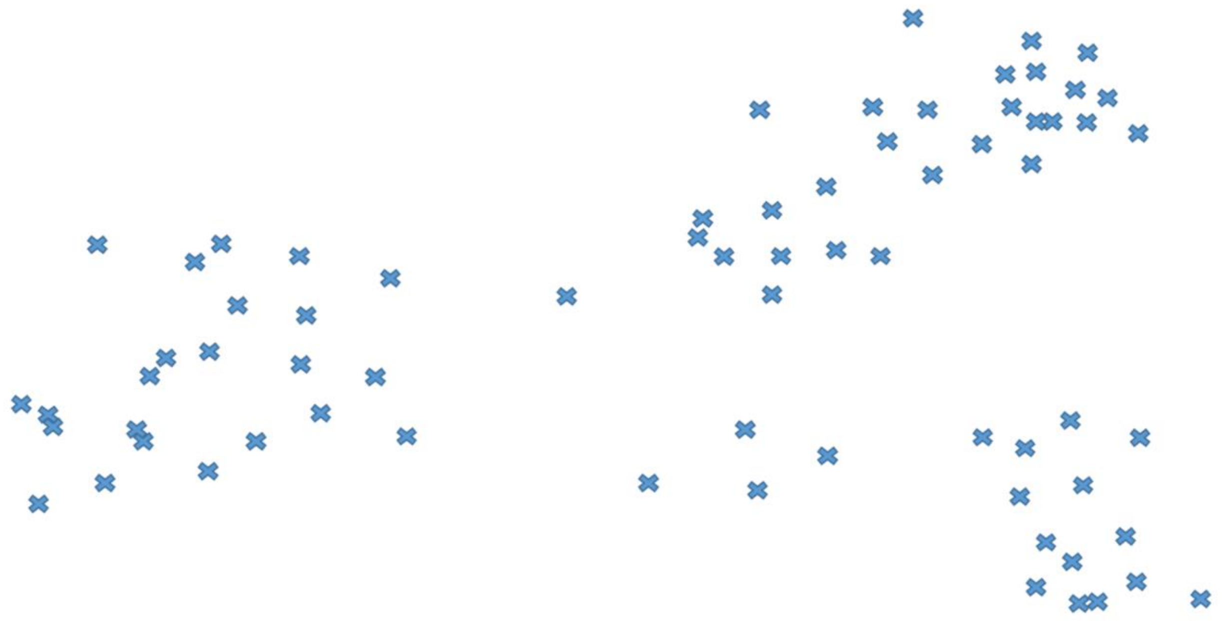
Unsupervised algorithm (the given dataset is unlabeled)

- Used to find clusters with similar characteristics
- There is no outcome to be predicted
- K denotes the number of clusters



Example by matt.willis@adatis.bg

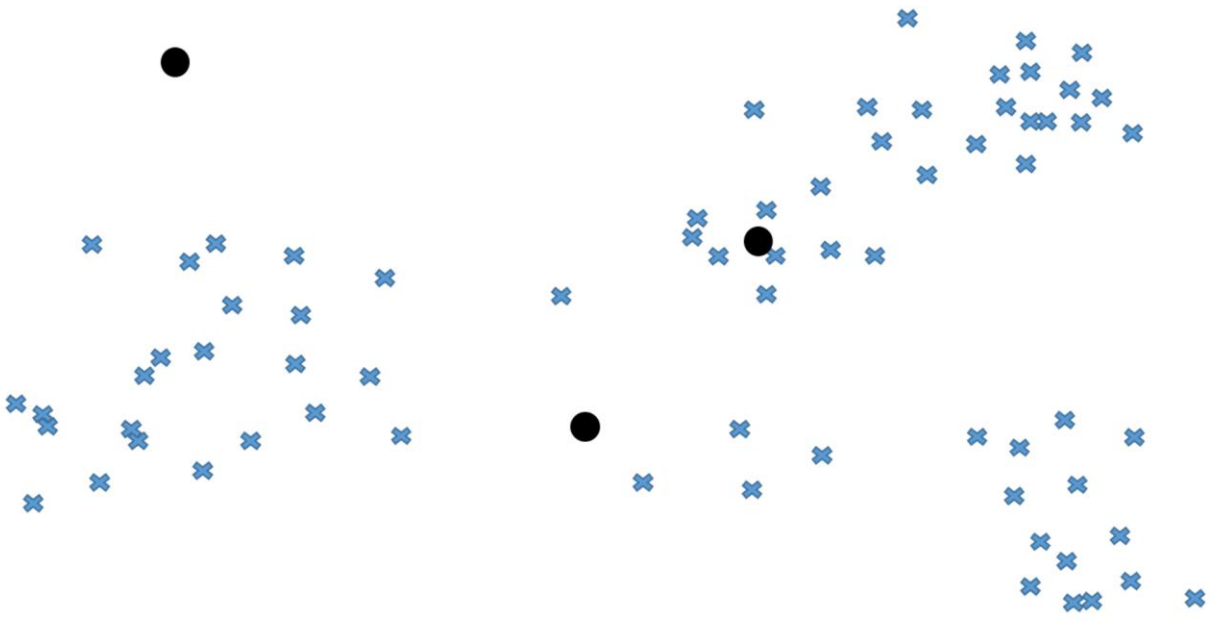
- K-means clustering
  - We decide on 3 clusters



Example by matt.willis@adatis.bg

K-means clustering

- We decide on 3 clusters
- 3 random points are chosen, each assigned to 1 cluster

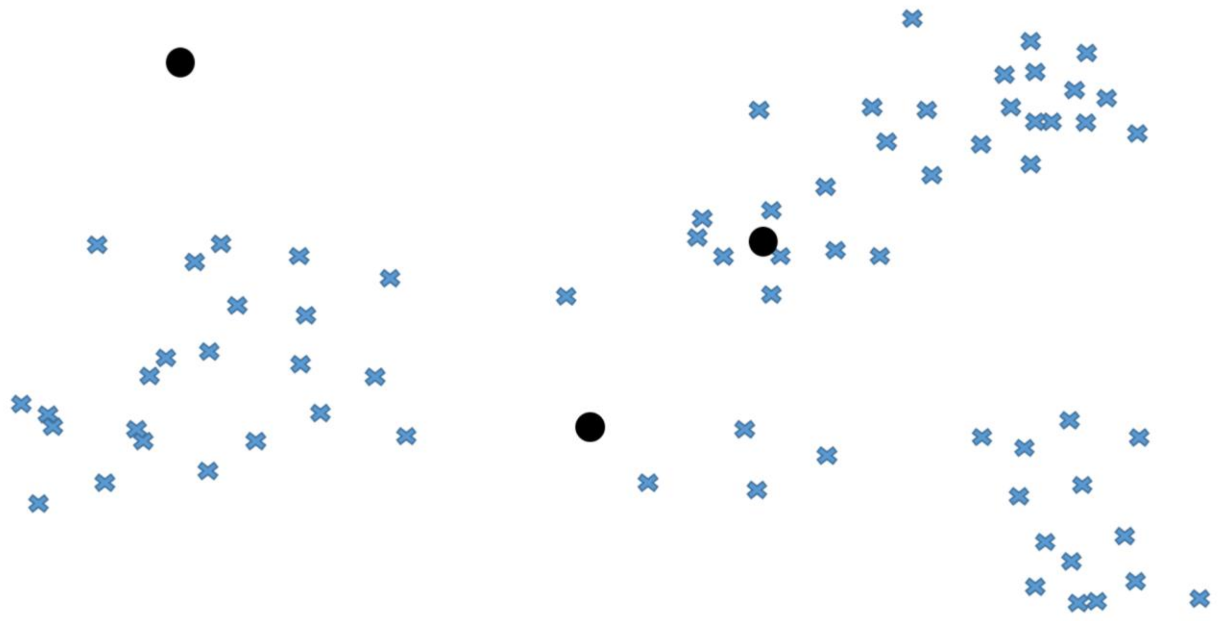


Example by matt.willis@adatis.bg

K-means clustering

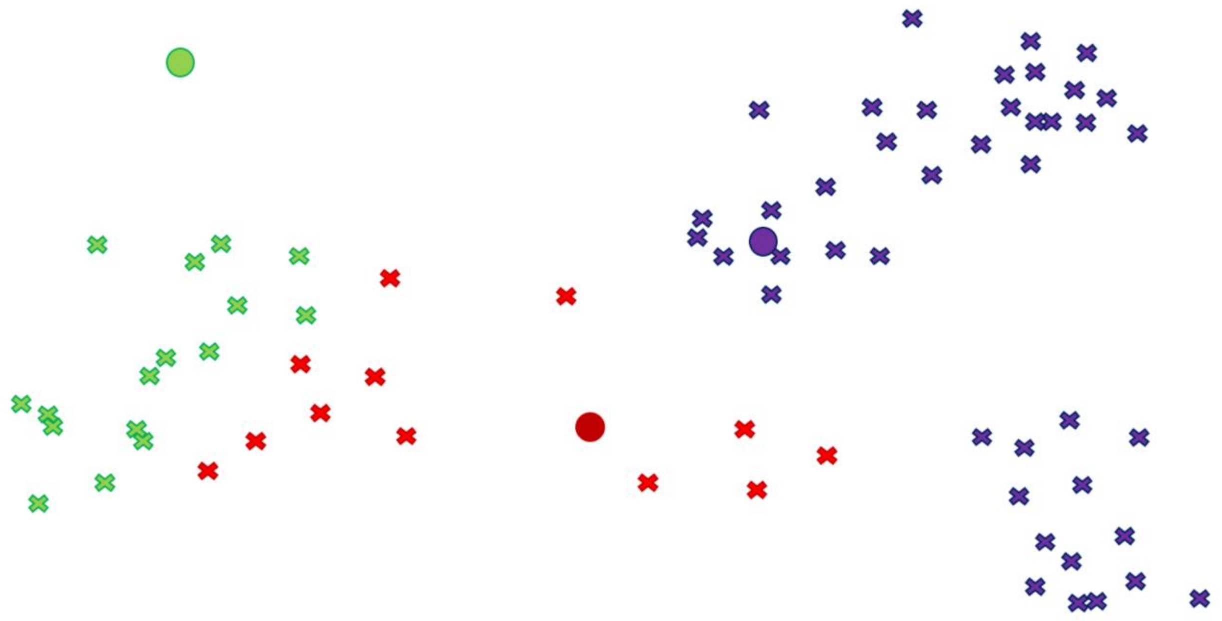
- We decide on 3 clusters
- 3 random points are chosen, each assigned to 1 cluster (centroids)
- The Euclidian distance is measured from each centroid to every other point

$$x = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$



Example by matt.willis@adatis.bg

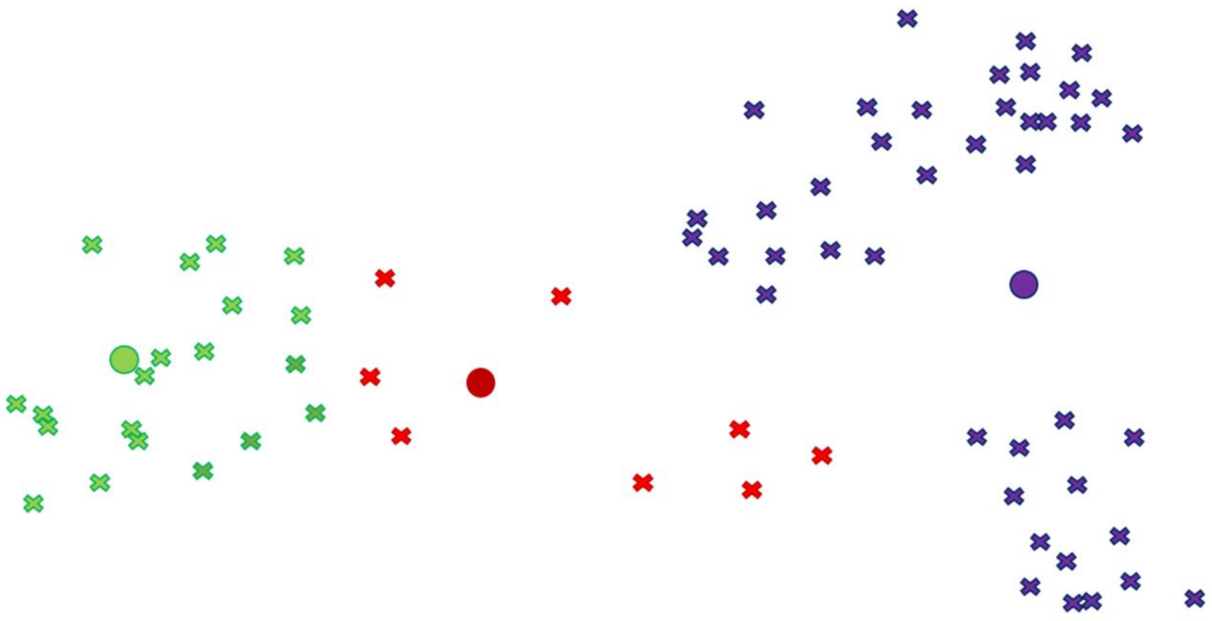
- K-means clustering
  - We assign colors to each cluster



Example by matt.willis@adatis.bg

K-means clustering

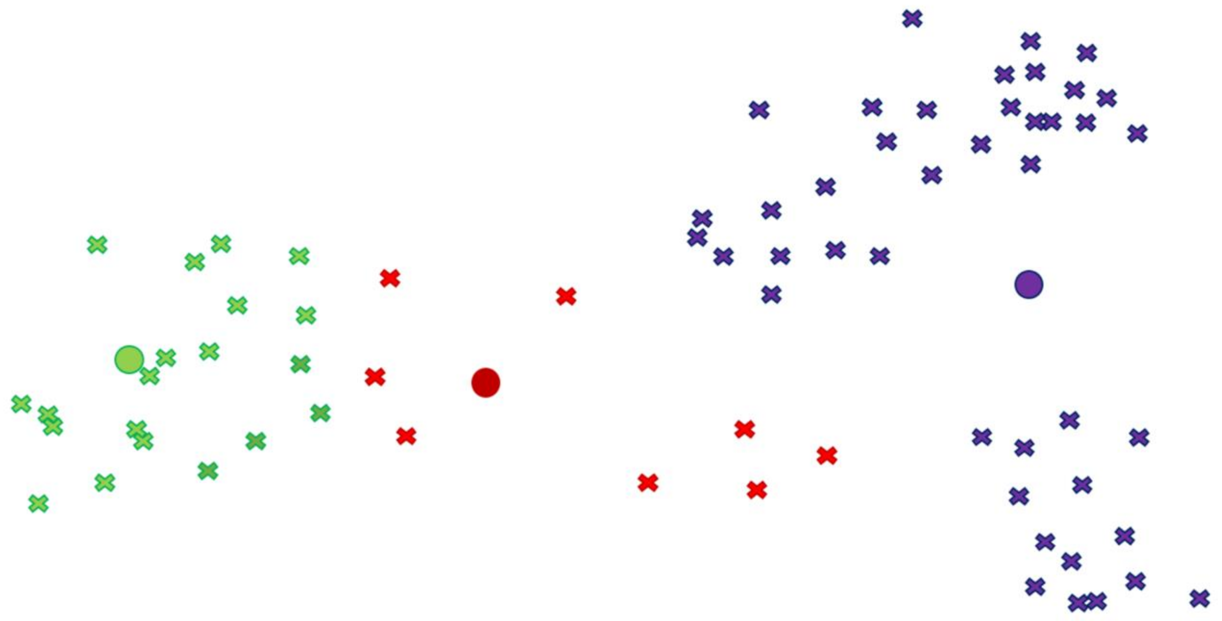
- We assign colors to each cluster
- We update the position for the centroids with the mean value of all the datapoints within that cluster



Example by matt.willis@adatis.bg

K-means clustering

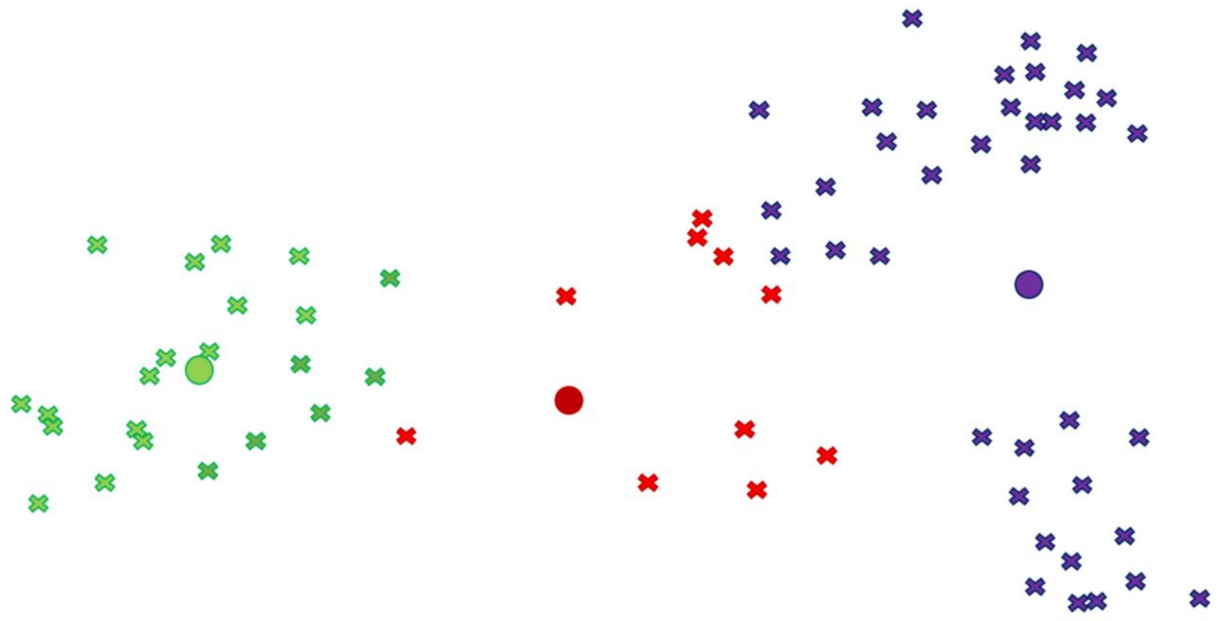
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

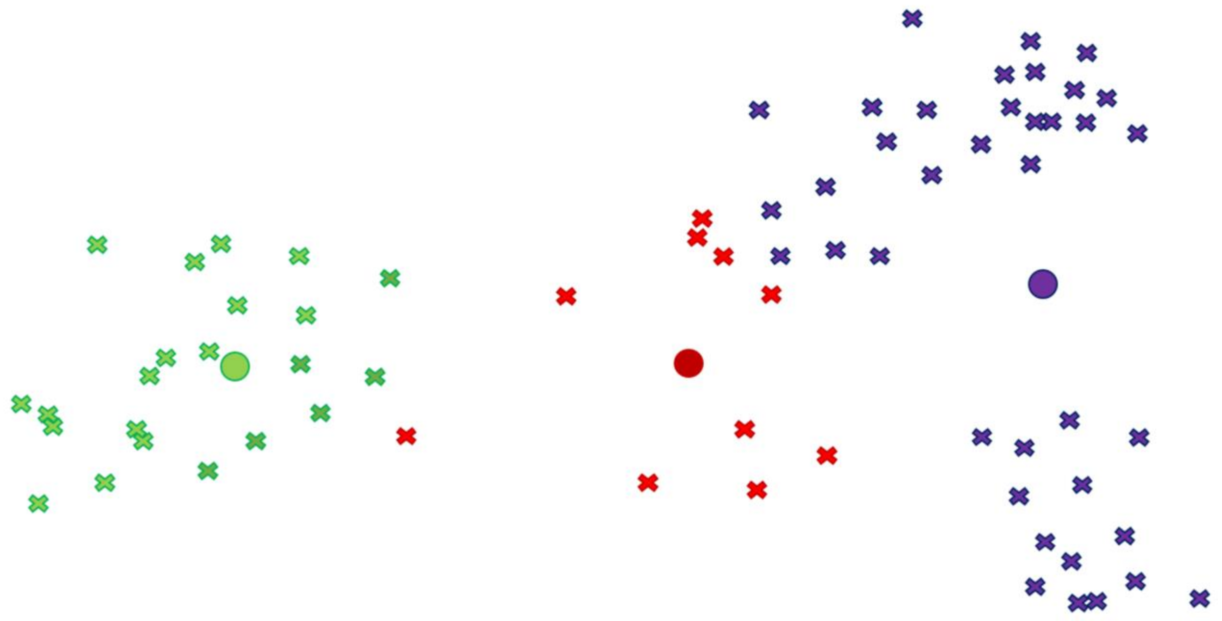
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

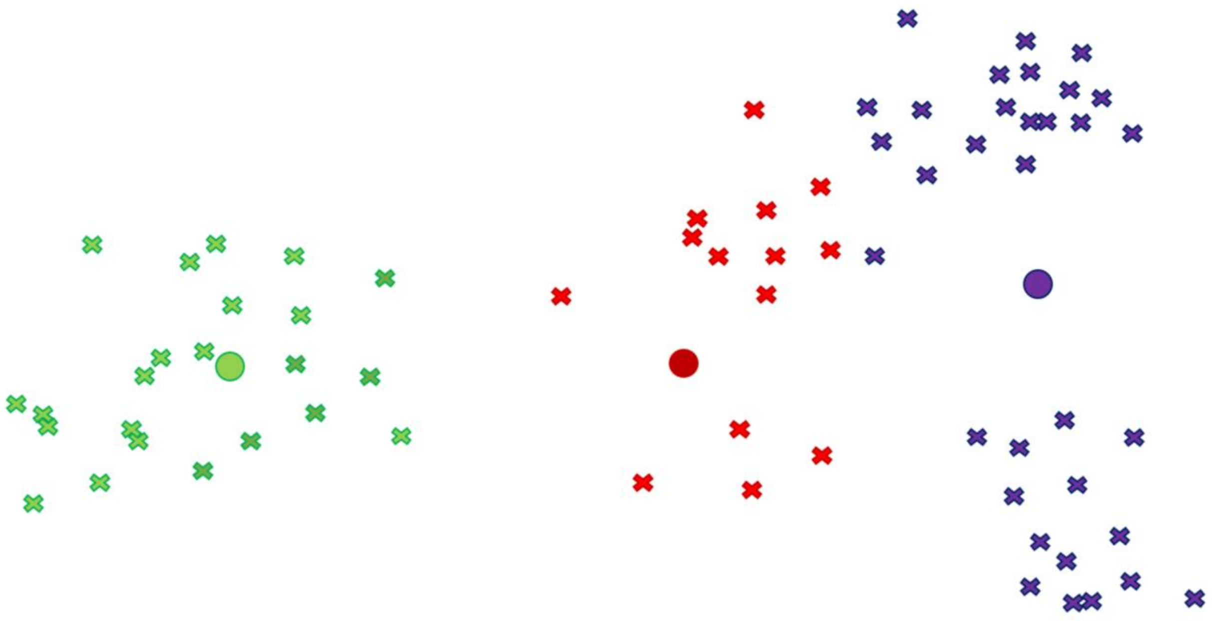
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

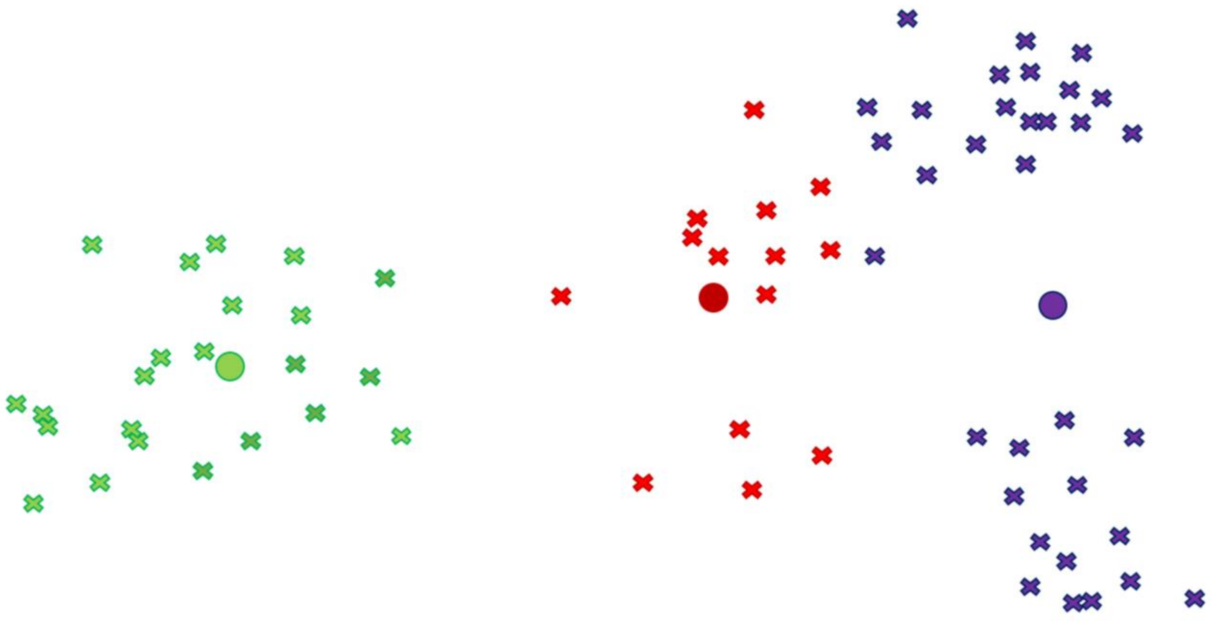
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

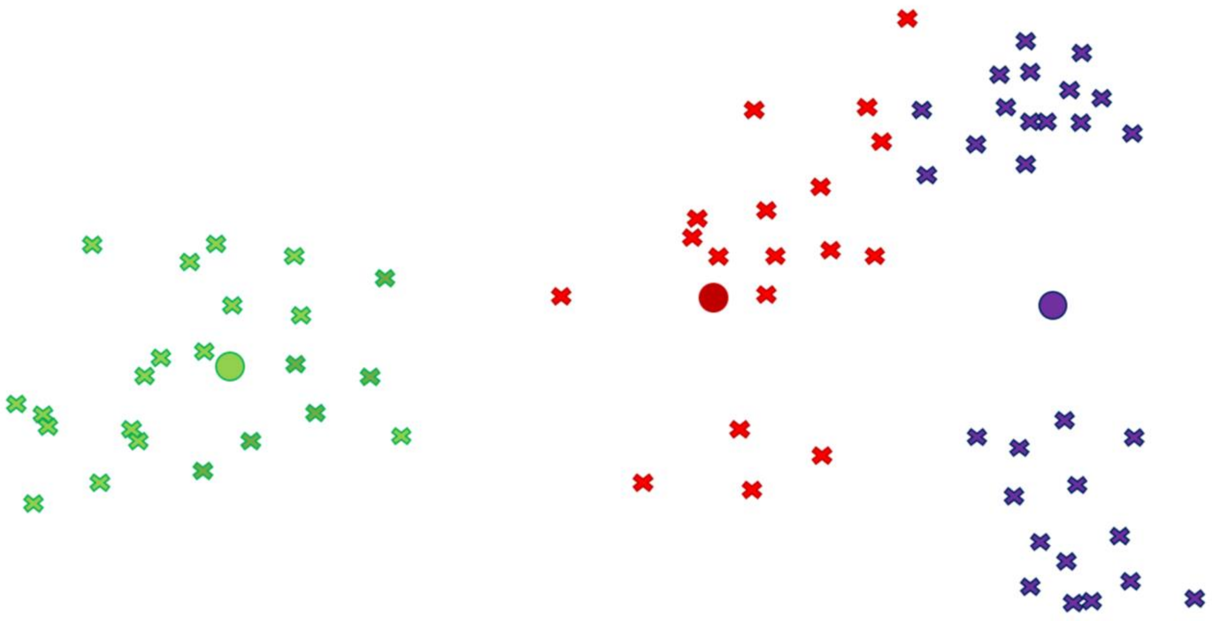
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

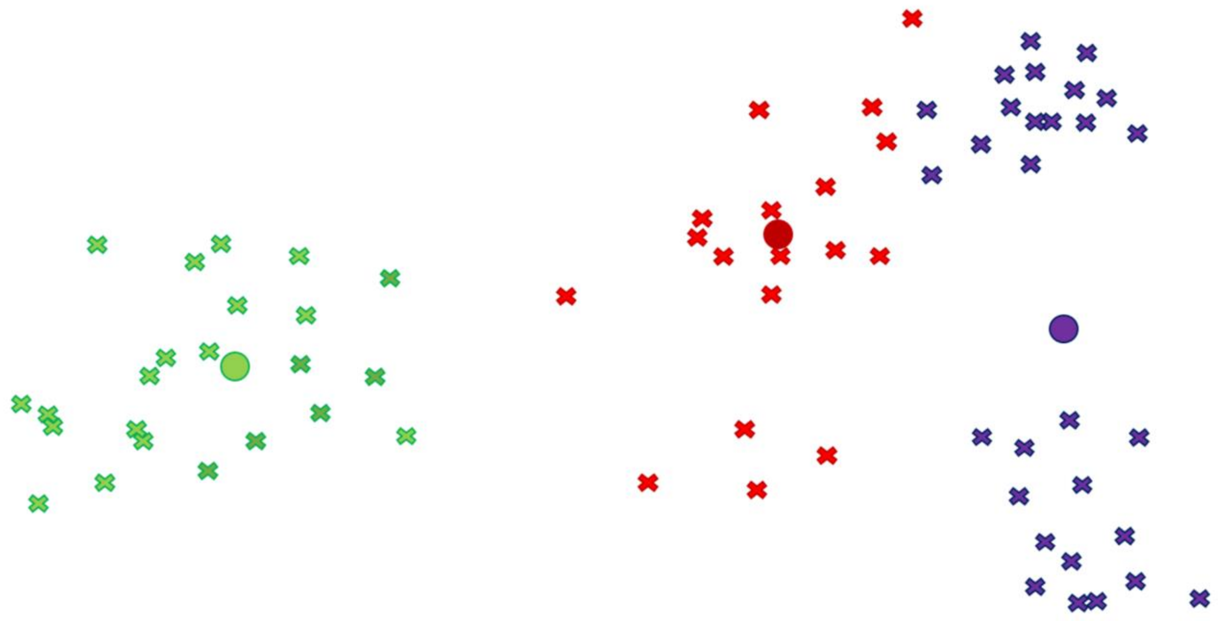
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

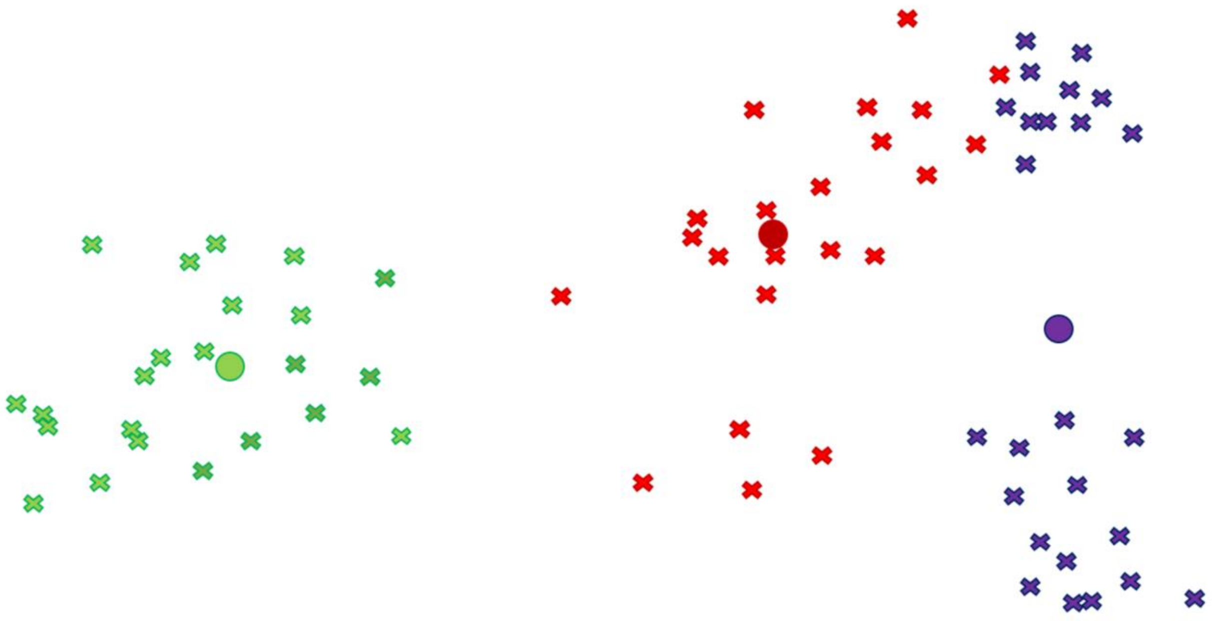
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

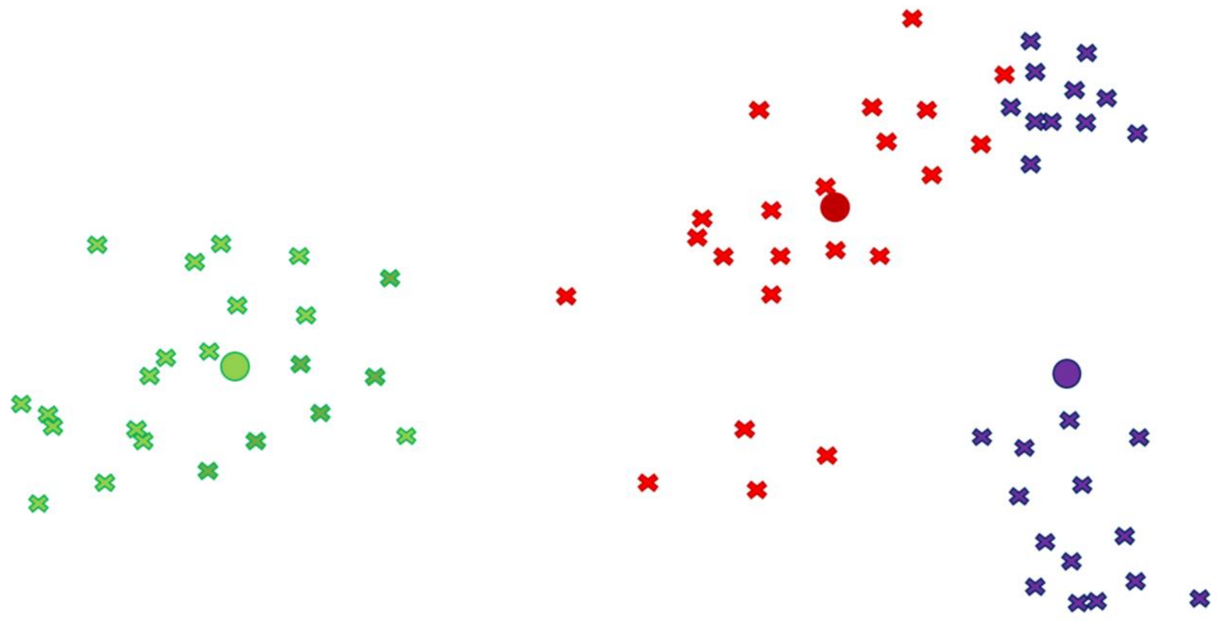
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

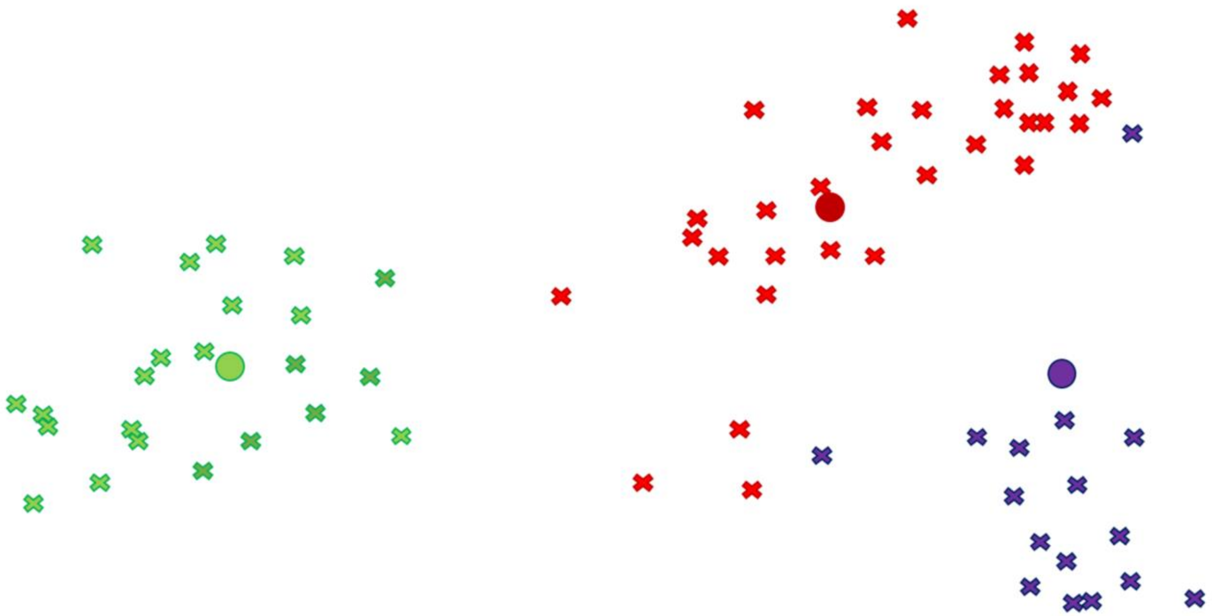
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

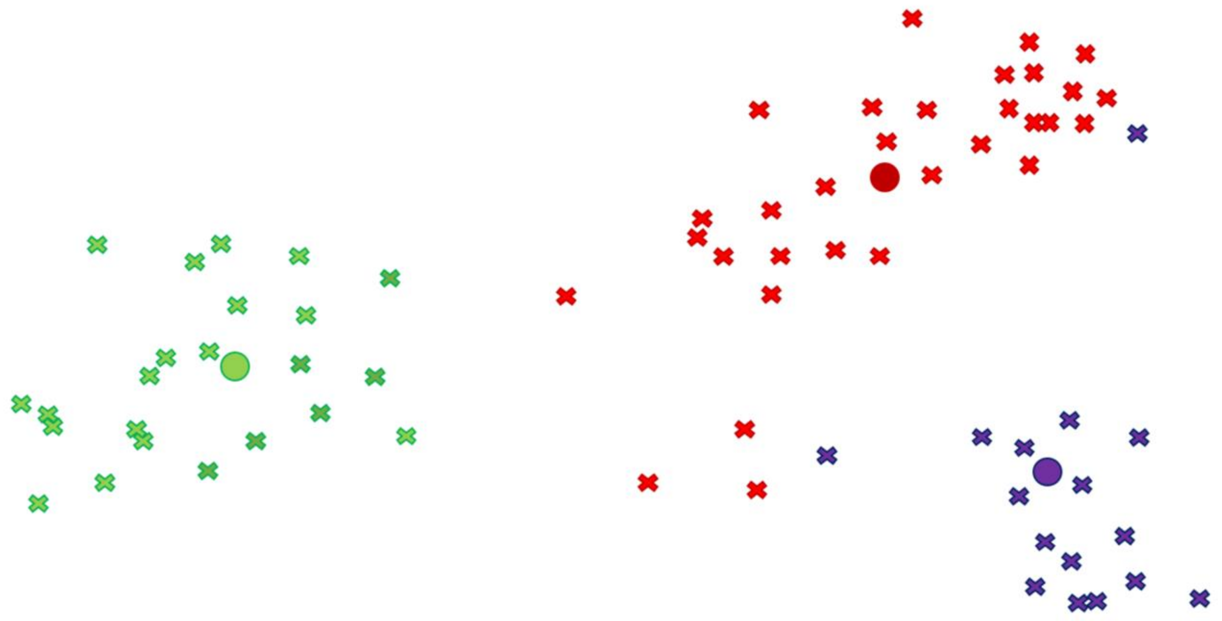
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

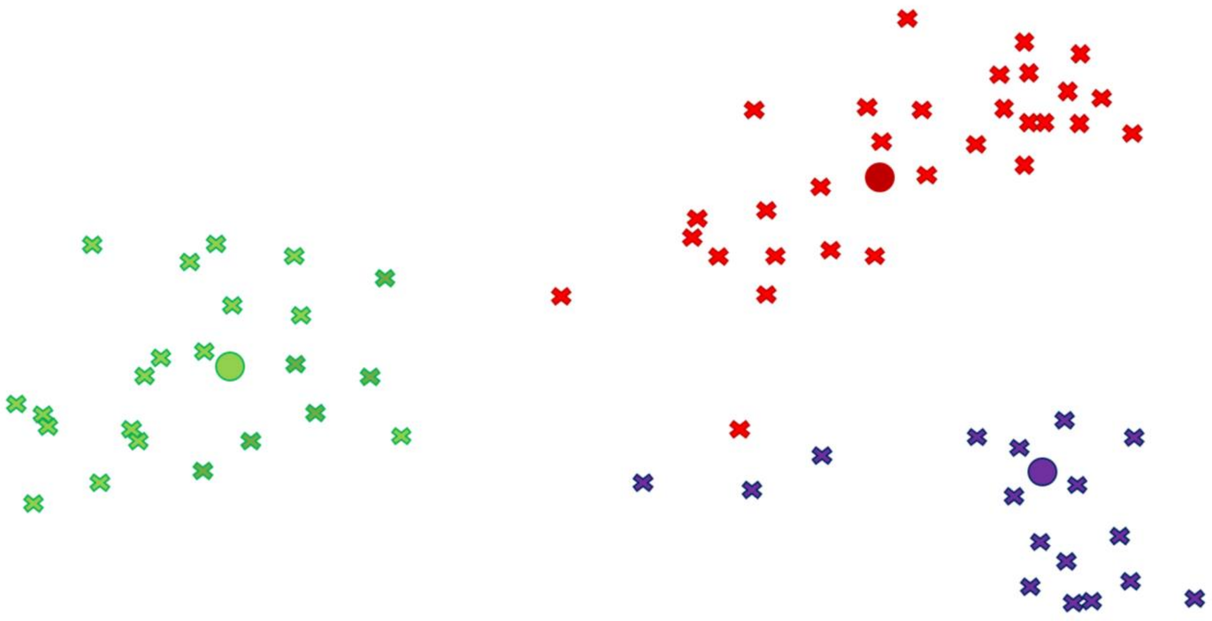
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

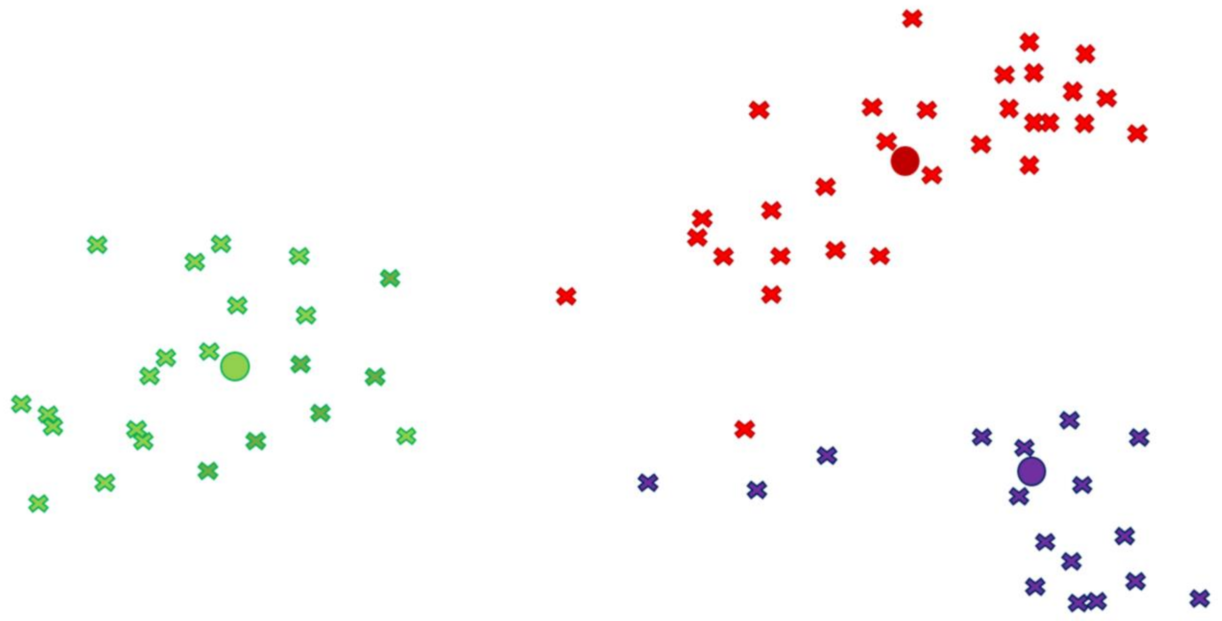
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

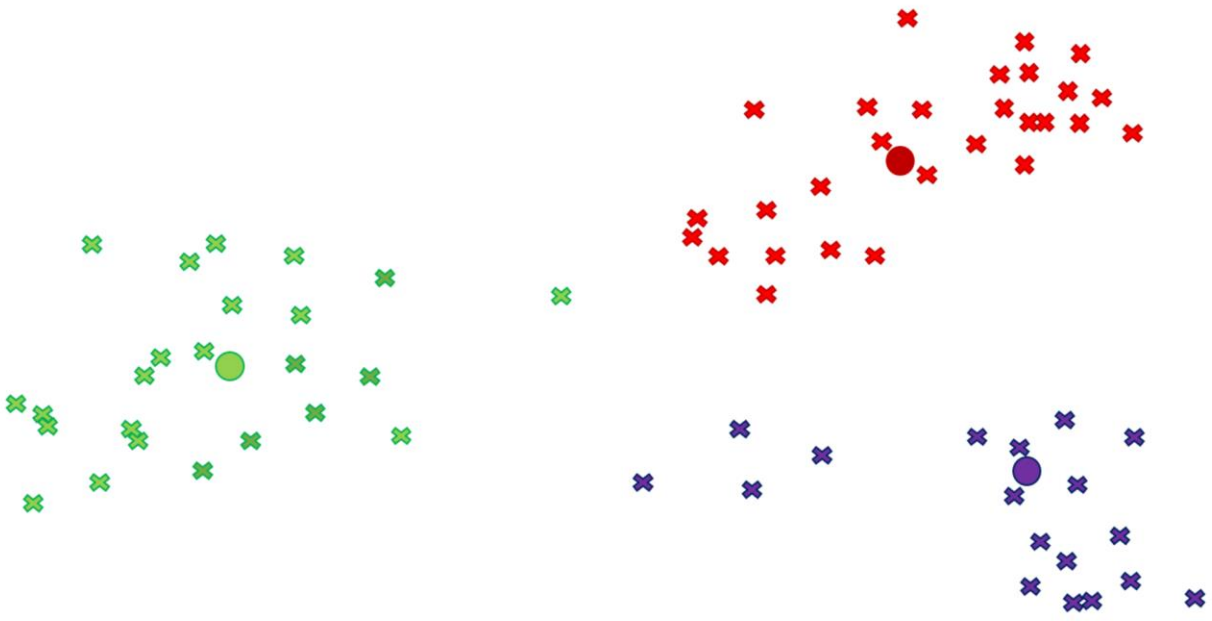
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

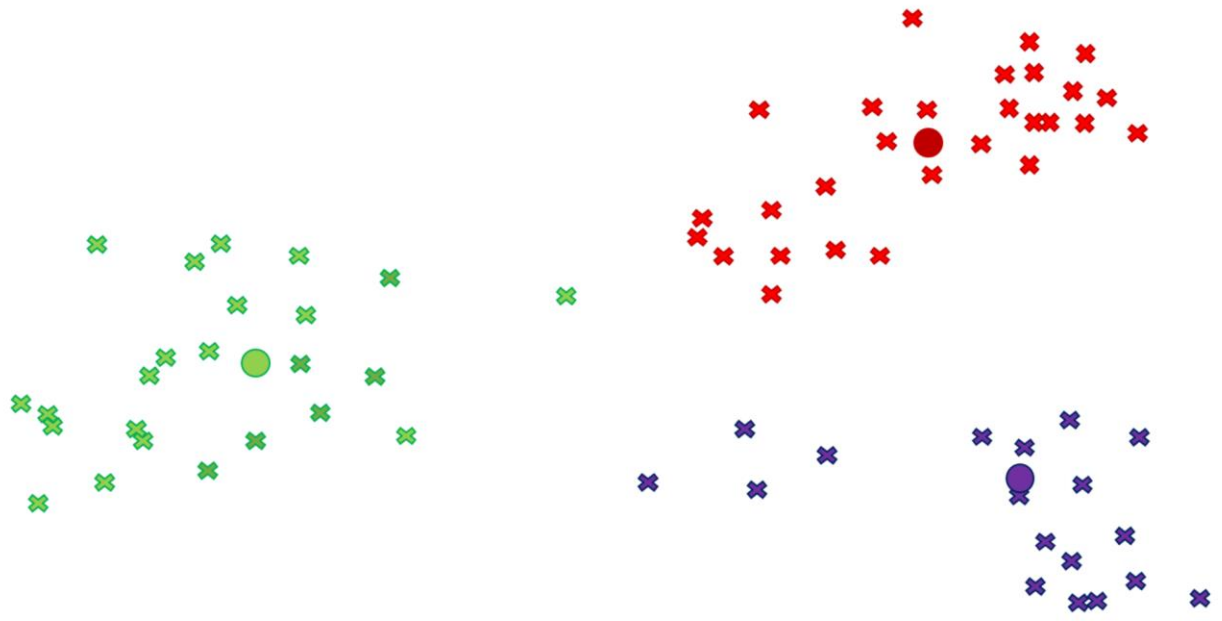
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

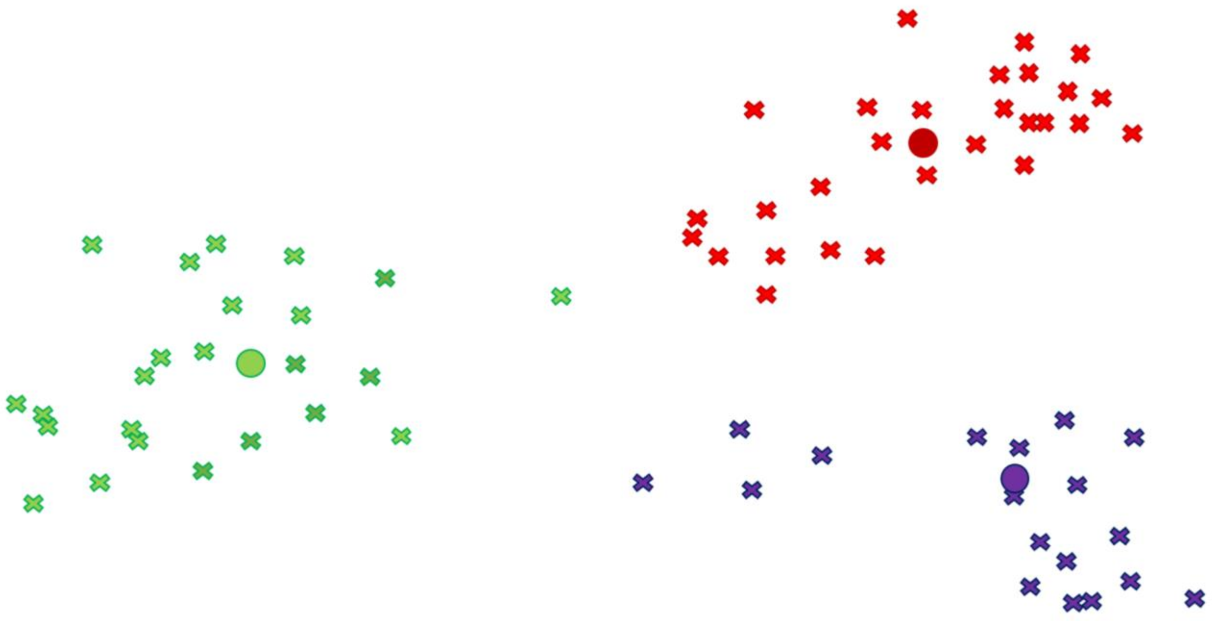
- Repeat until the stopping conditions are met
  - Datapoints assigned to each cluster remain the same
  - Maximum number of iterations was reached
  - Centroids remain the same (our case)



Example by matt.willis@adatis.bg

K-means clustering

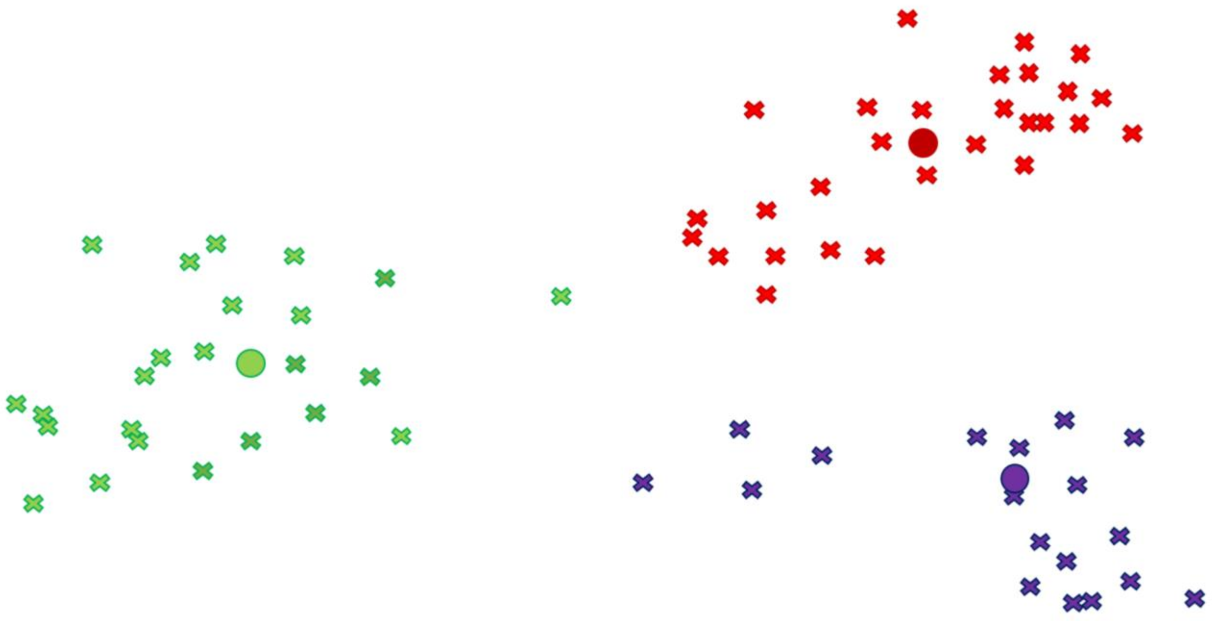
- Looks like we have reached the end



Example by matt.willis@adatis.bg

K-means clustering

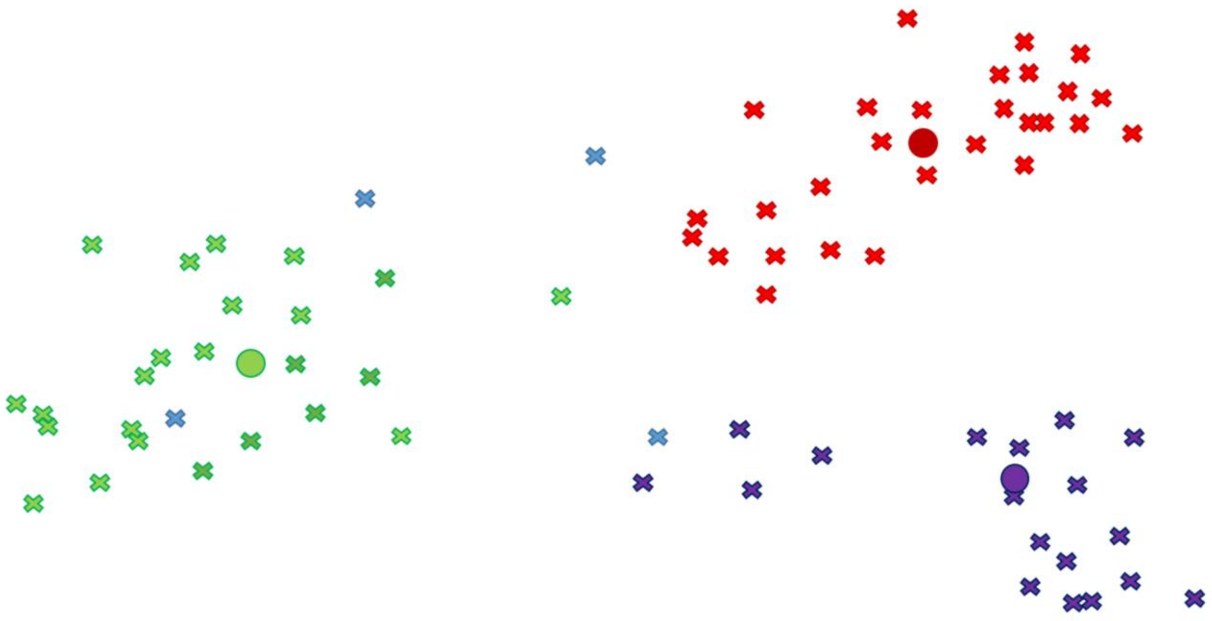
- But what if we add a few more datapoints?



Example by matt.willis@adatis.bg

K-means clustering

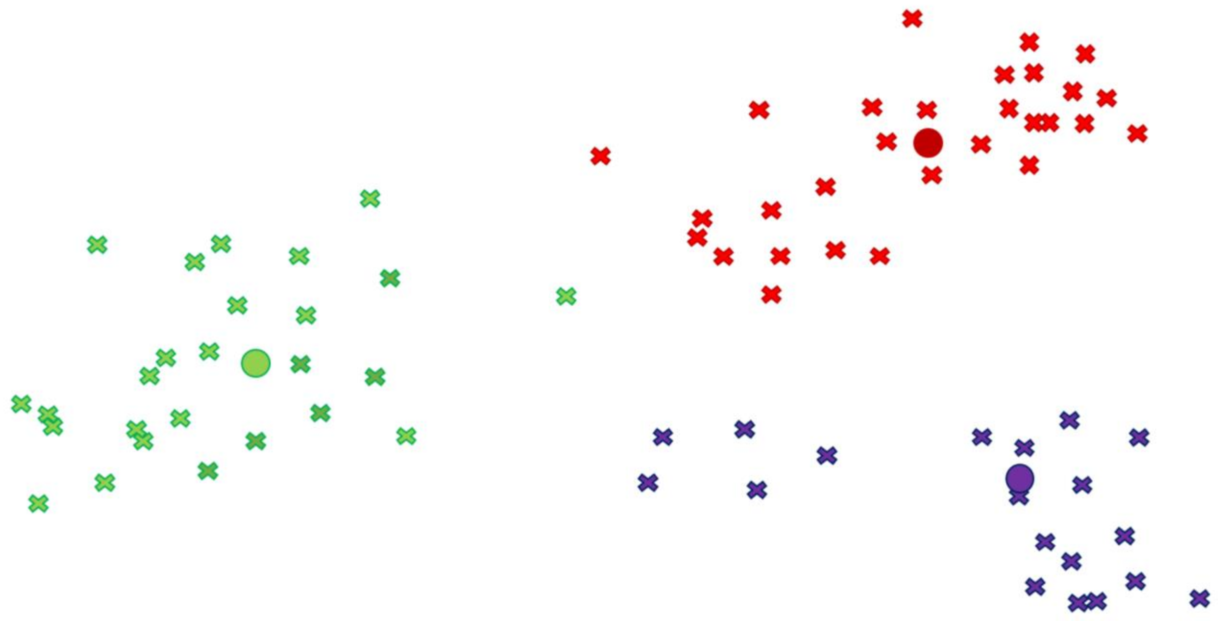
- But what if we add a few more datapoints?



Example by matt.willis@adatis.bg

K-means clustering

- The new points are assigned to their corresponding clusters



Example by matt.willis@adatis.bg

K-means clustering

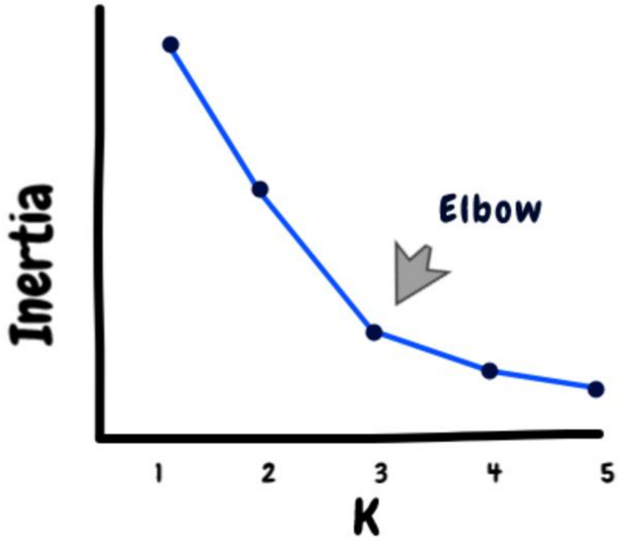
- In order to find the best clusters multiple starting points are chosen and the algorithm for each combination of them
- The best iteration is the one with the least total variation (inertia) of the clusters (distance between the 2 farthest points from the centroid)

K-means clustering

- In order to find the best clusters multiple starting points are chosen and the algorithm for each combination of them
- The best iteration is the one with the least total variation (inertia) of the clusters (distance between the 2 farthest points from the centroid)
- A higher K gives us less total variation but with a lesser reduction
- In order to determine the optimal K we use the elbow graph

K-means clustering

- In order to find the best clusters multiple starting points are chosen and the algorithm for each combination of them
- The best iteration is the one with the least total variation (inertia) of the clusters (distance between the 2 farthest points from the centroid)
- A higher K gives us less total variation but with a lesser reduction
- In order to determine the optimal K we use the elbow graph



https://towardsdatascience.com/k-means-clustering-from-a-to-z-f6242a314e9a

Evaluation metrics

Classification Accuracy

- Most common metric used for model evaluation
- Not necessarily the best in all cases
- Ratio of correct predictions to the total number of input samples

$$Acc = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions\ made}$$

Classification Accuracy

- Suppose we are trying to find whether a person is infected or not with a very dangerous disease
- Out of 100 people only 1 is actually infected (positive)

Classification Accuracy

- Suppose we are trying to find whether a person is infected or not with a very dangerous disease
- Out of 100 people only 1 is actually infected (positive)
- Our model predicts everyone negative

Classification Accuracy

- Suppose we are trying to find whether a person is infected or not with a very dangerous disease
- Out of 100 people only 1 is actually infected (positive)

- Our model predicts everyone negative
- Well.. that means 99% accuracy, awesome!

Classification Accuracy

- Suppose we are trying to find whether a person is infected or not with a very dangerous disease
- Out of 100 people only 1 is actually infected (positive)

- Our model predicts everyone negative
- ~~Well.. that means 99% accuracy, awesome!~~
- Not exactly

Classification Accuracy

- Suppose we are trying to find whether a person is infected or not with a very dangerous disease
- Out of 100 people only 1 is actually infected (positive)
- Our model predicts everyone negative
- ~~Well.. that means 99% accuracy, awesome!~~
- Not exactly
- The cost and danger of having someone infected and not treated outweighs some potential false positives which would require a few people to get more tests

- ROC Curves and Space

(Receiver operating characteristic)

Confusion Matrix

  - True positive – My model said you're infected, and I've got some bad news for you
  - False positive – My model said you're infected, but I've got some good news for you
  - True negative – You're all good, just like my model said
  - False negative – You almost got away.. almost

ROC Curves and Space

Confusion Matrix

- True positive – My model said you're infected, and I've got some bad news for you
- False positive – My model said you're infected, but I've got some good news for you
- True negative – You're all good, just like my model said
- False negative – You almost got away.. almost
- True positive rate: $TPR = \frac{TP}{P}$
- True negative rate: $TNR = \frac{TN}{N}$
- False positive rate: $FPR = 1 - TNR$
- False negative rate: $FNR = 1 - TPR$

ROC Curves and Space

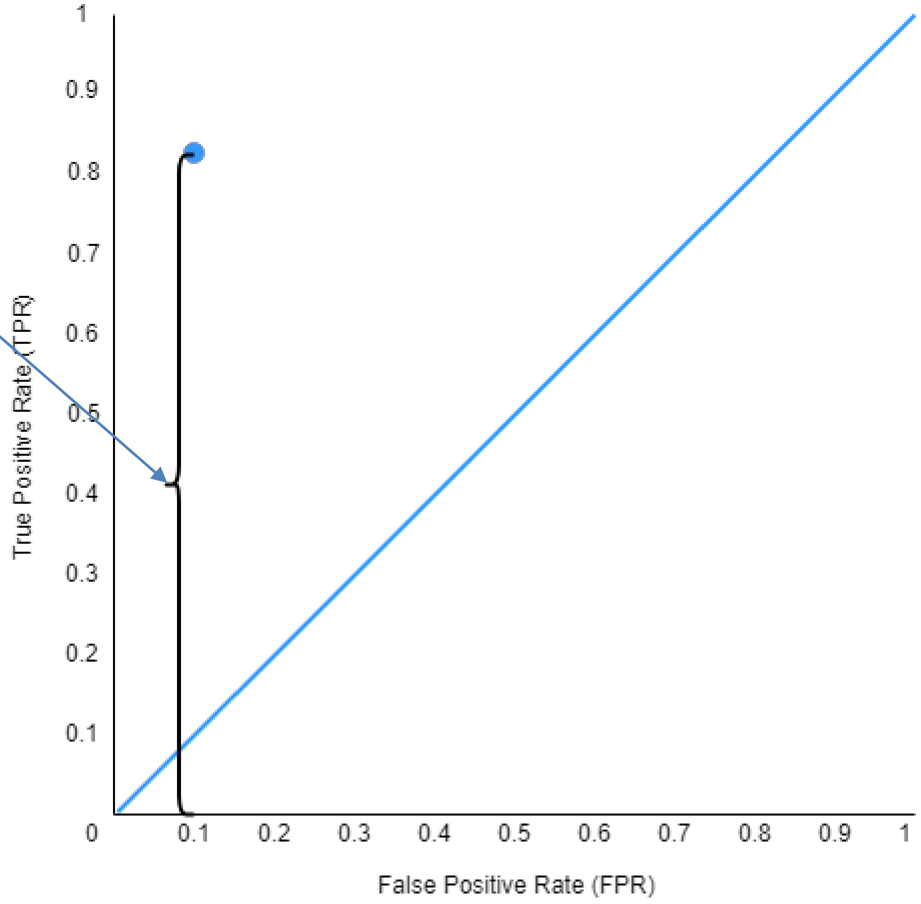
- Let’s modify our example a bit
- Because of our infected patient an epidemic has started
- Many more people have become infected

ROC Curves and Space

- Let's modify our example a bit
- Because of our infected patient an epidemic has started
- Many more people have become infected
- After some more tests, the model results are as follows:
  - Out of 100 people who survived, our model predicted 83.
    - So TPR = 0.83
    - And FNR = 0.17
  - Out of 100 people who died, our model predicted 90.
    - So TNR = 0.9
    - And FPR = 0.1
- ROC Space gives us a way to represent these visually

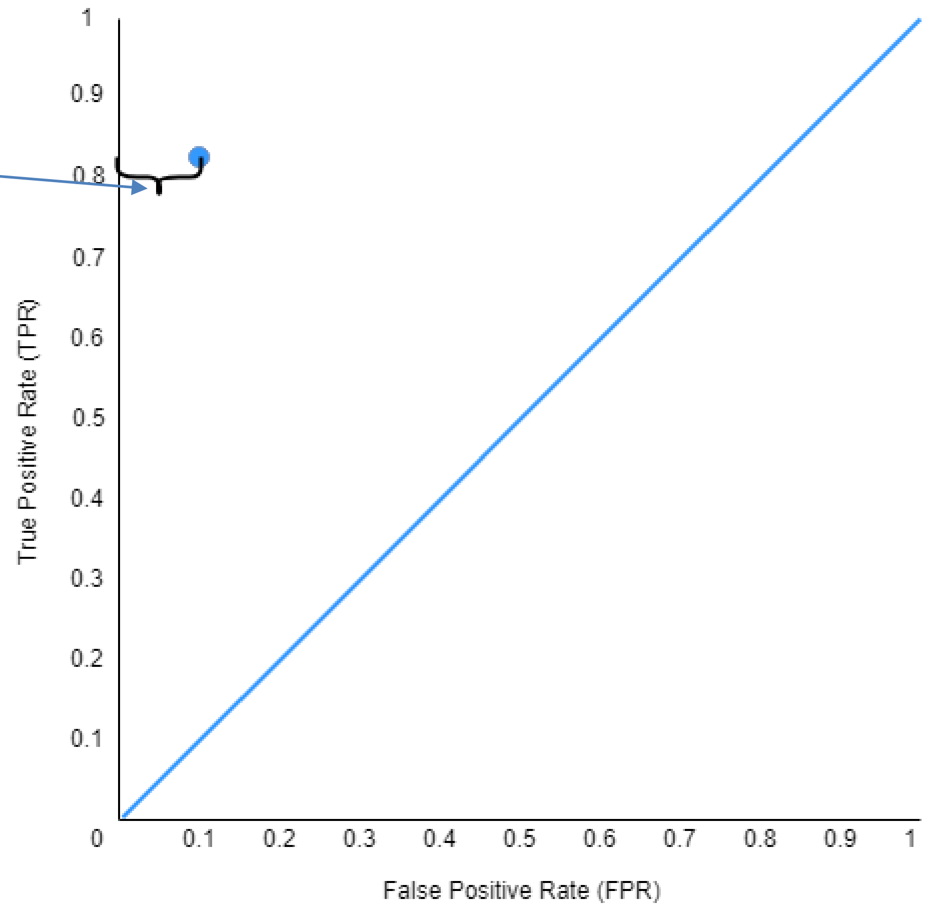
ROC Curves and Space

	Prediction: Survived	Prediction: Died
Survived	TP = 83	FN = 17
Died	FP = 10	TN = 90



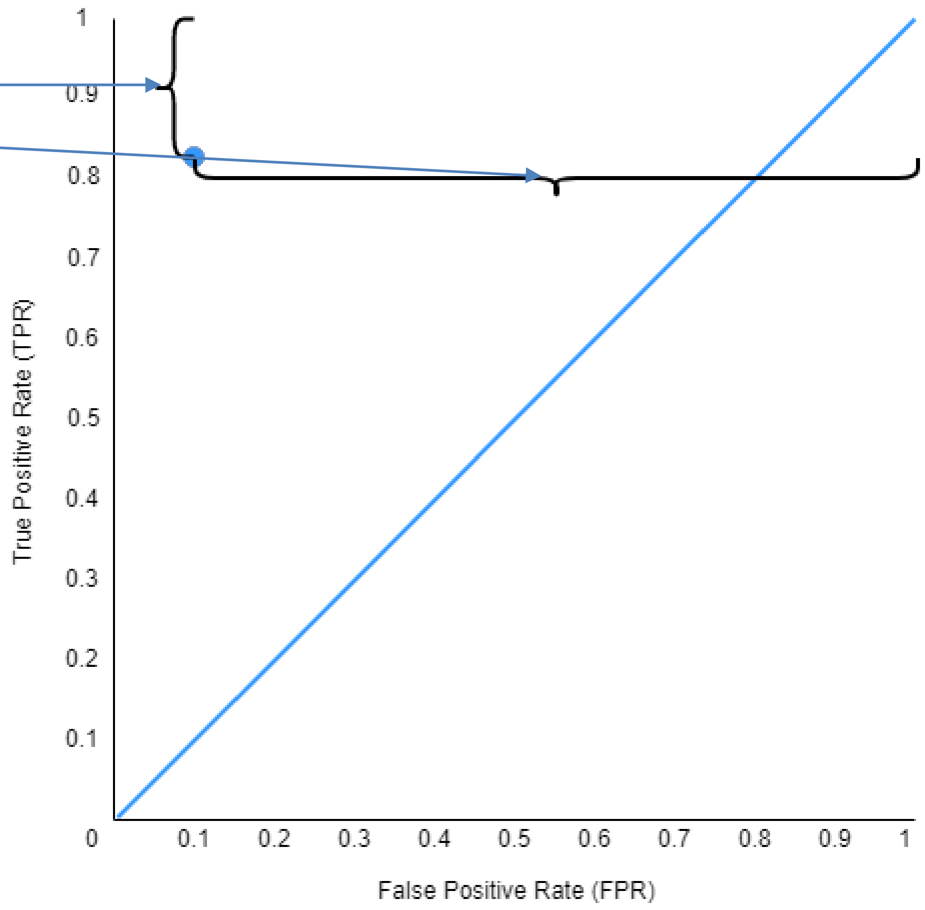
ROC Curves and Space

	Prediction: Survived	Prediction: Died
Survived	TP = 83	FN = 17
Died	FP = 10	TN = 90



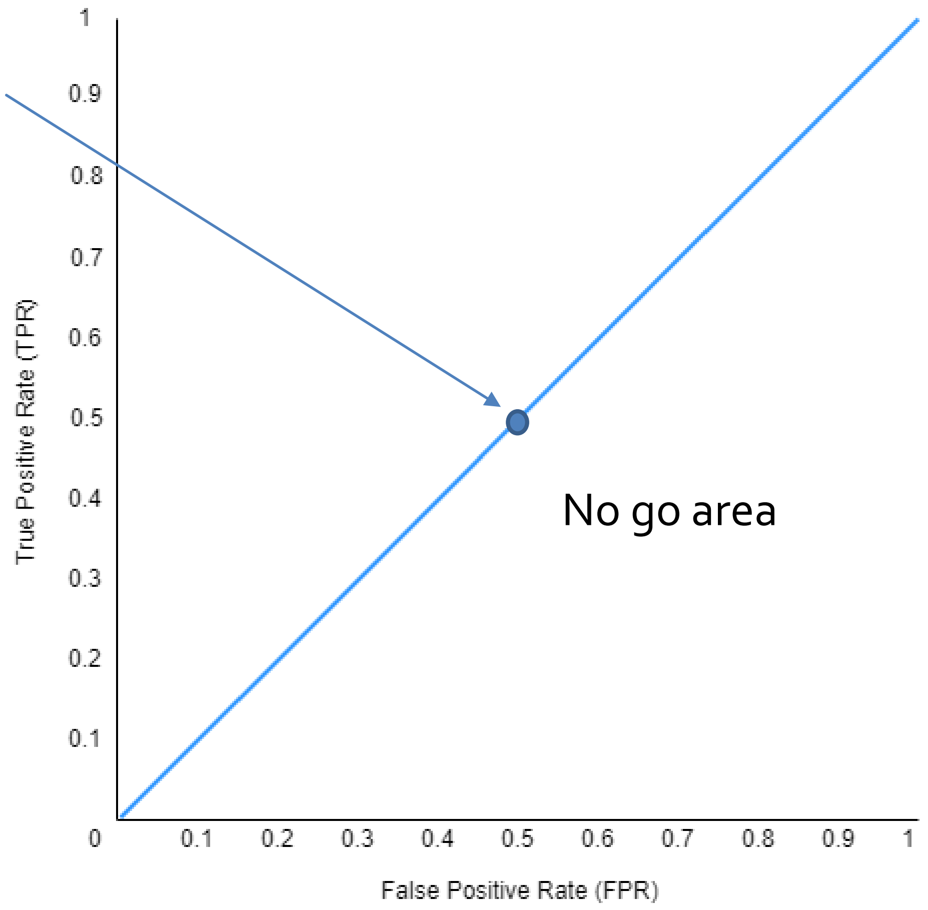
ROC Curves and Space

	Prediction: Survived	Prediction: Died
Survived	TP = 83	FN = 17
Died	FP = 10	TN = 90



ROC Curves and Space

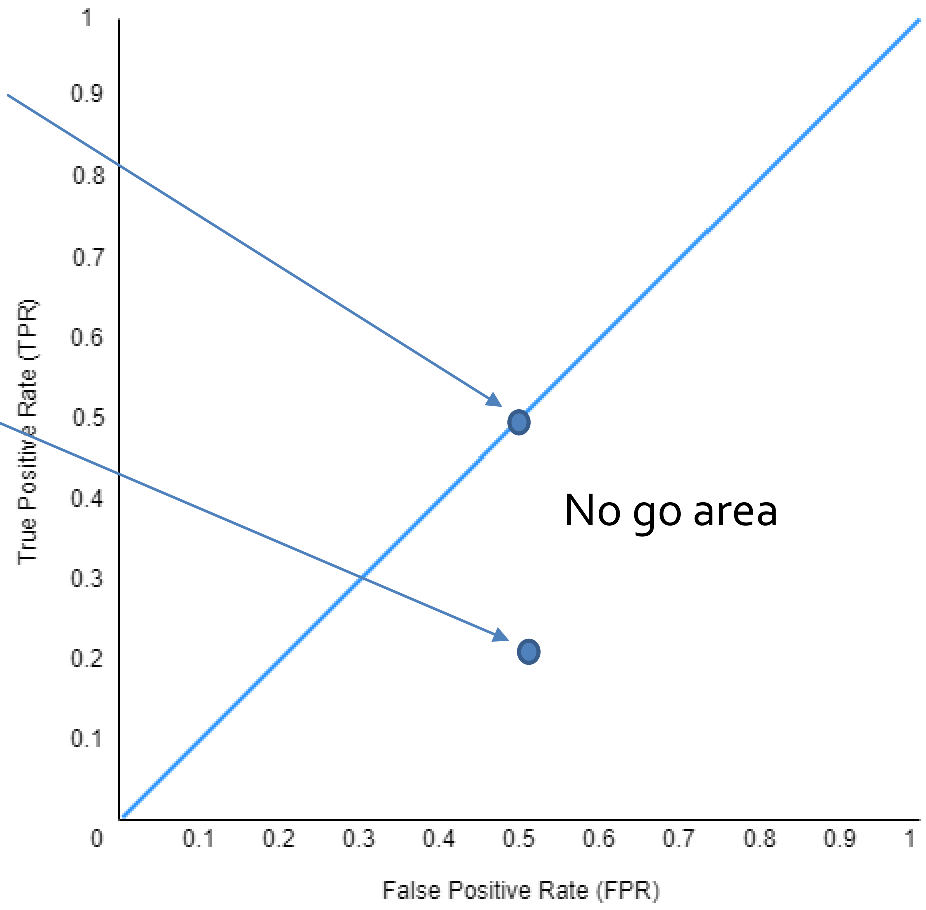
If an algorithm simply guesses it will be represented on the diagonal line (random prediction)



ROC Curves and Space

If an algorithm simply guesses it will be represented on the diagonal line (random prediction).

Anything below the line means that you’ve managed to create a model that makes worse predictions than a random chance.

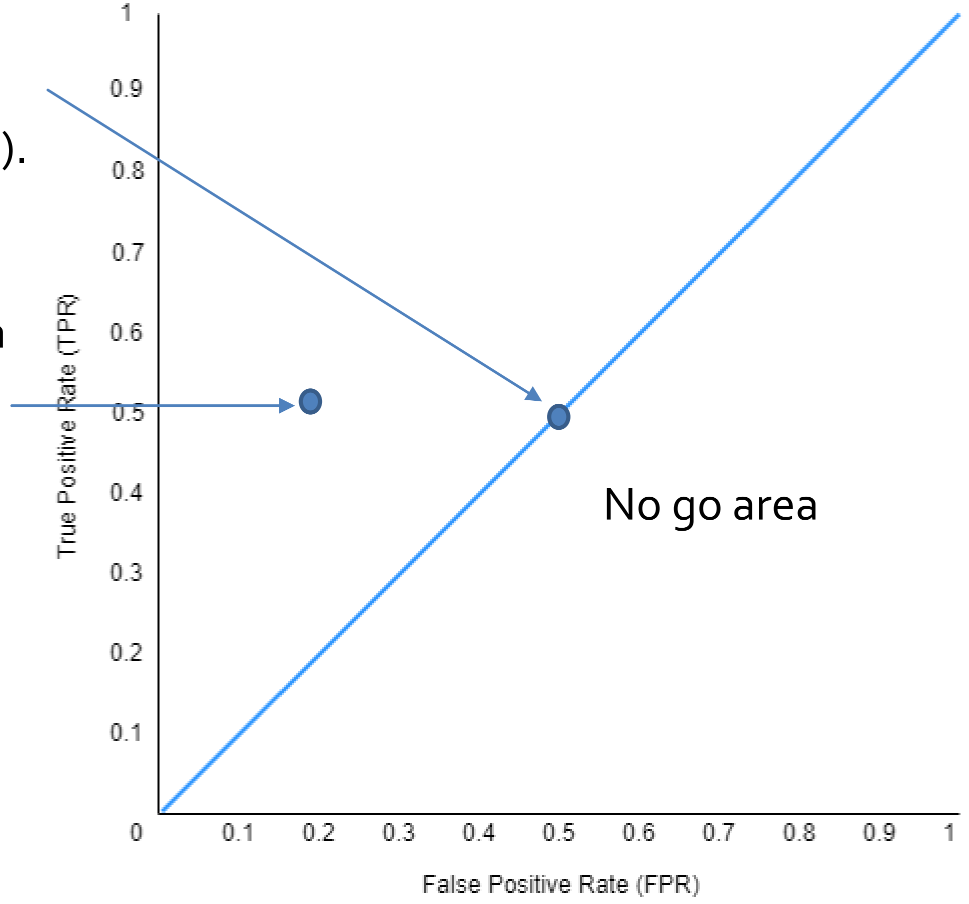


ROC Curves and Space

If an algorithm simply guesses it will be represented on the diagonal line (random prediction).

Anything below the line means that you've managed to create a model that makes worse predictions than a random chance.

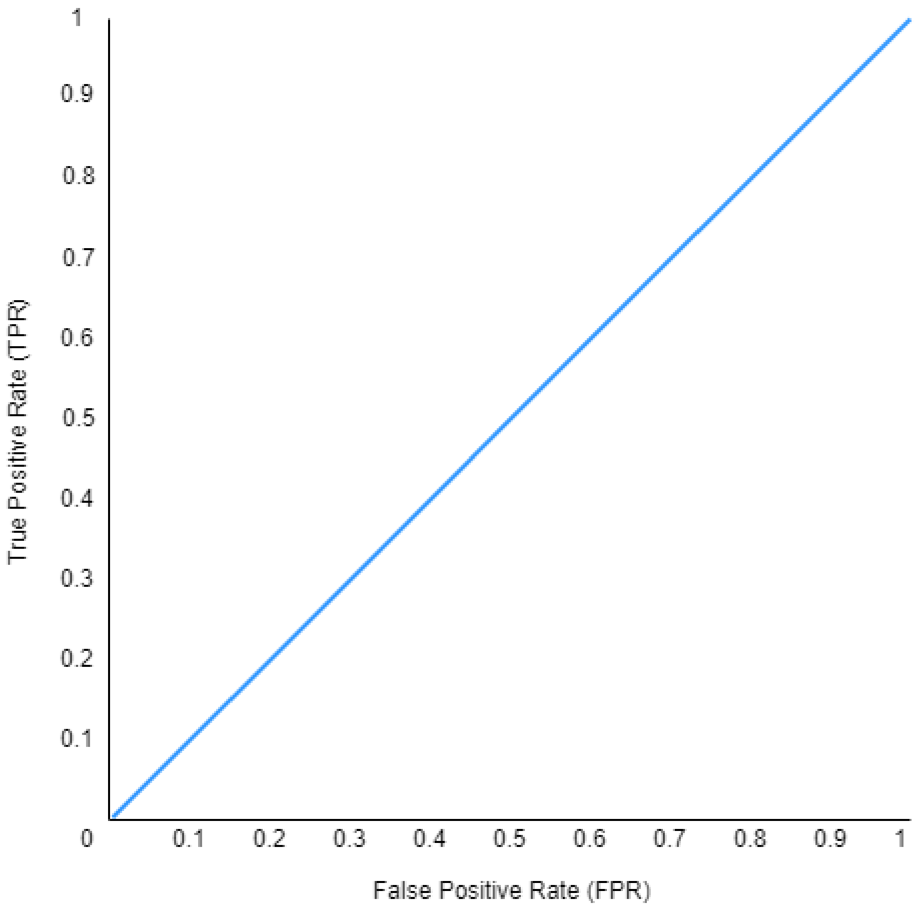
So just reverse it!



ROC Curves and Space

Remember our mistake which doomed the world earlier?

Let’s see what would’ve happened if we knew how to use the ROC Space.

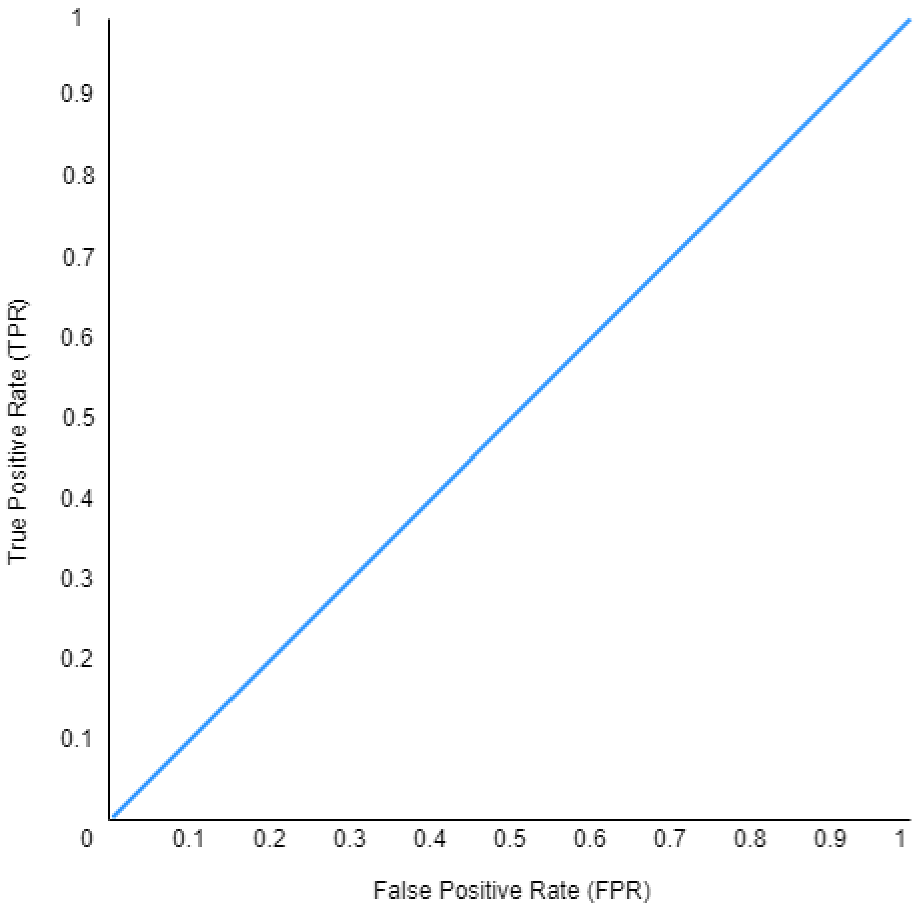


ROC Curves and Space

Remember our mistake which doomed the world earlier?

Let's see what would've happened if we knew how to use the ROC Space.

	Prediction: Positive	Prediction: Negative
Positive	0	1
Negative	0	99



ROC Curves and Space

Remember our mistake which doomed the world earlier?

Let's see what would've happened if we knew how to use the ROC Space.

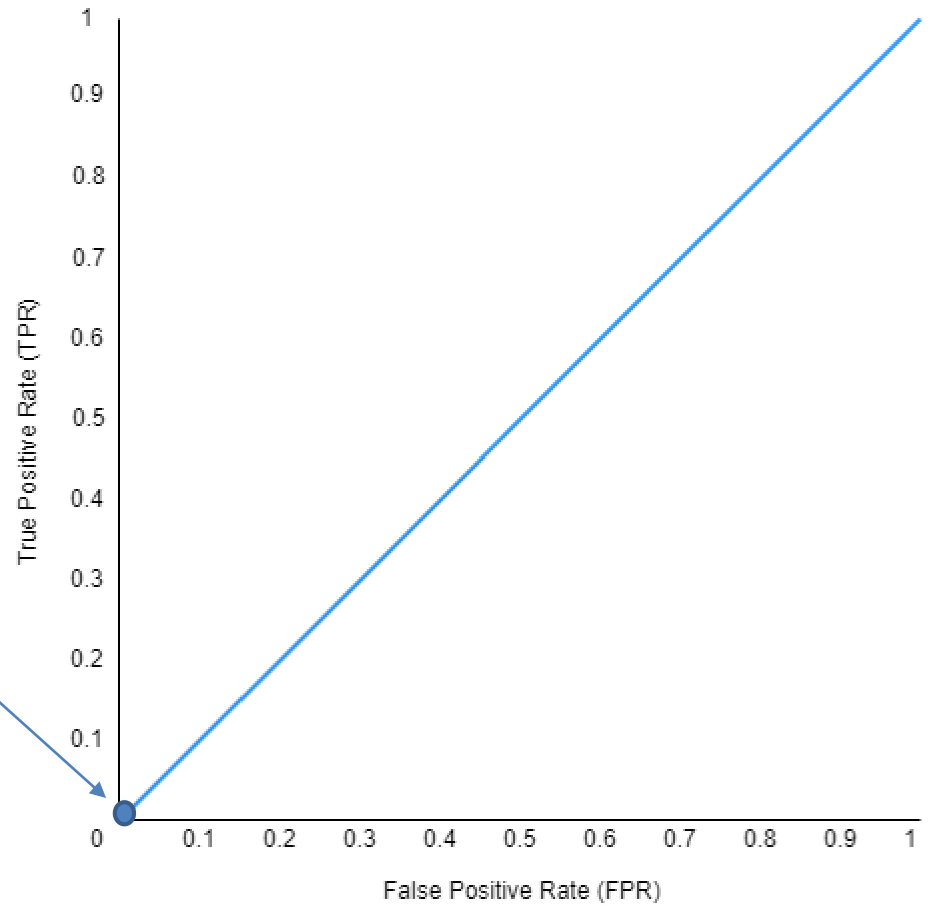
	Prediction: Positive	Prediction: Negative
Positive	0	1
Negative	0	99

TPR = 0

FNR = 1

TNR = 0.99

FPR = 0.01



ROC Curves and Space

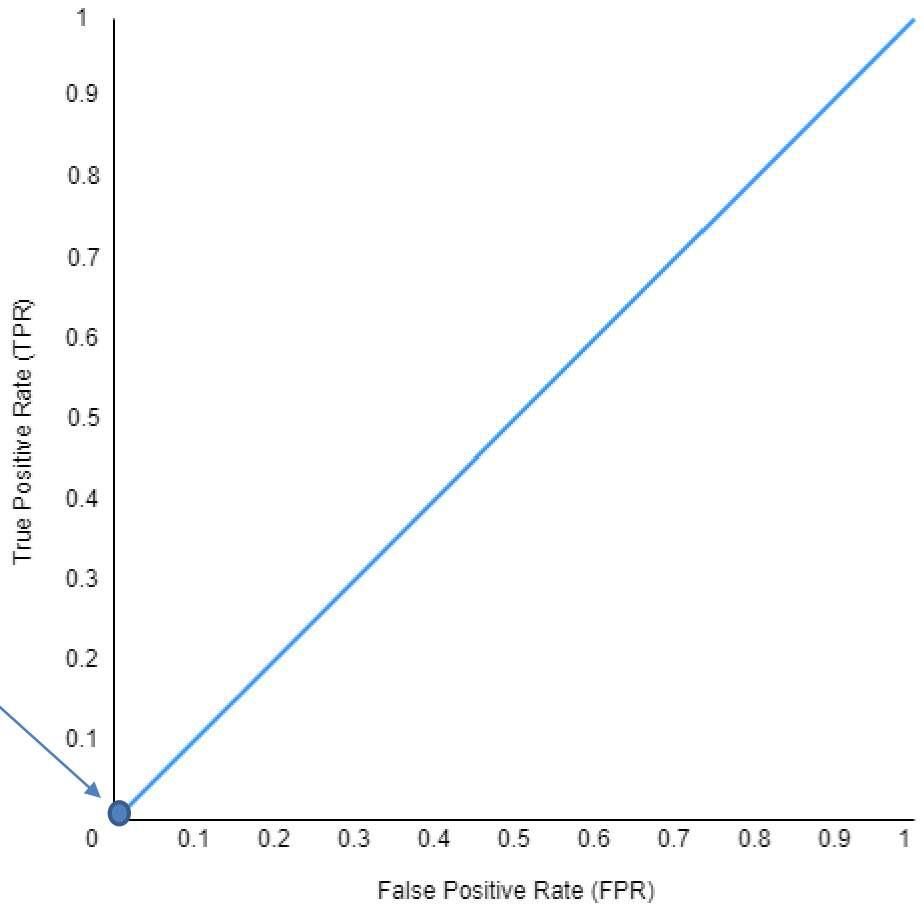
Remember our mistake which doomed the world earlier?

Let's see what would've happened if we knew how to use the ROC Space.

	Prediction: Positive	Prediction: Negative
Positive	0	1
Negative	0	99

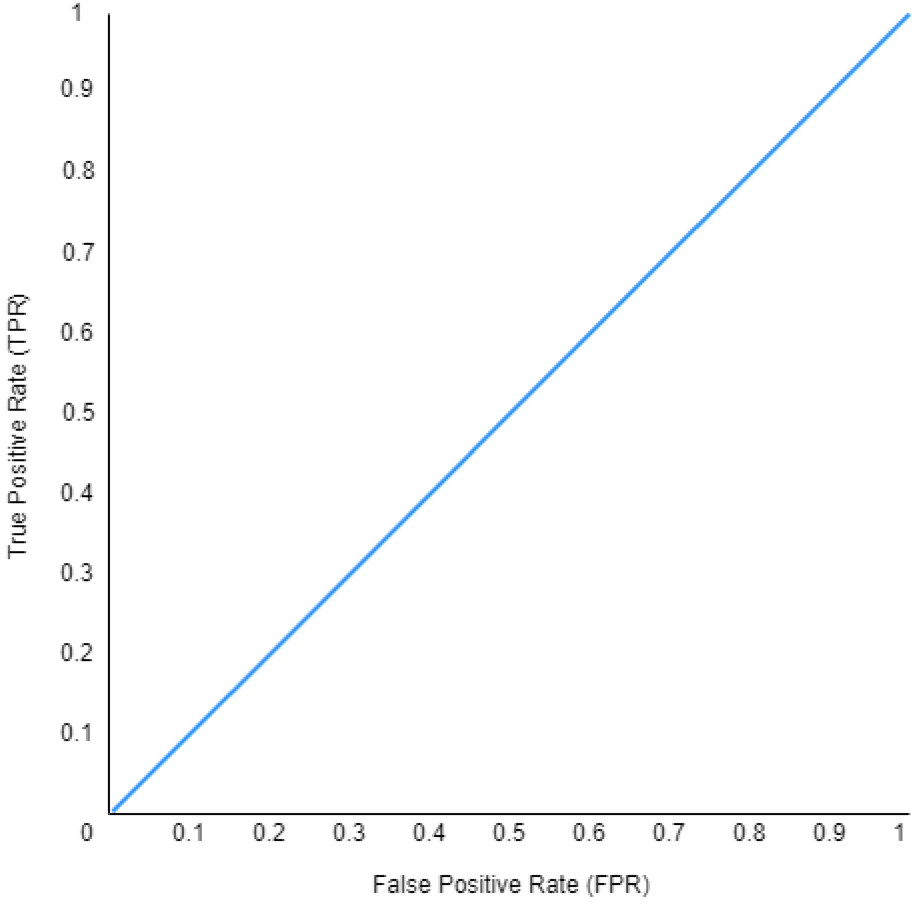
TPR = 0
FNR = 1
TNR = 0.99
FPR = 0.01

Well.. This is terrible!



ROC Curves and Space

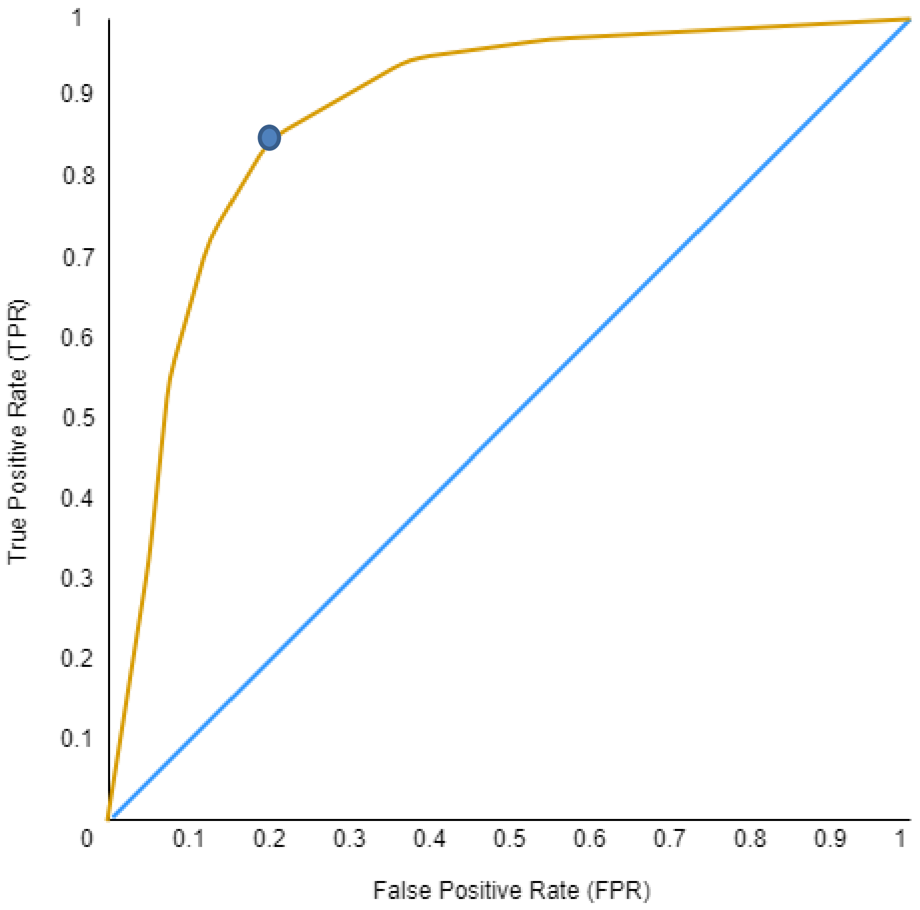
As the efficiency increases, the point on the ROC Space gets closer to the upper left corner



ROC Curves and Space

As the efficiency increases, the point on the ROC Space gets closer to the upper left corner

Depending on the model and the accuracy, the point will describe a curve trajectory



ROC Curves and Space

- Does it give us all the answers?
  - Yes

ROC Curves and Space

- Does it give us all the answers?
  - ~~Yes~~
  - Well, actually no.

ROC Curves and Space

- Does it give us all the answers?
  - ~~Yes~~
  - Well, actually no.
  - But it helps out a lot!

ROC Curves and Space

- Does it give us all the answers?
  - ~~Yes~~
  - Well, actually no.
  - But it helps out a lot!
  - We still have to take into consideration the other constraints:
    - Cost to business
    - Time to run
    - Computational overhead

Azure Machine Learning Studio

Azure ML Studio

- In the words of Microsoft "Azure Machine Learning Studio is a collaborative, drag-and-drop tool you can use to build, test and deploy predictive analytics solutions on your data. Machine Learning Studio publishes models as web services that can easily be consumed by custom apps or BI tools such as Excel"
- Now, on to the demo! (assuming the demo is not in the beginning of the presentation)

https://docs.microsoft.com/en-us/azure/machine-learning/studio/what-is-ml-studio

Thank you!

Questions?