Decision Tree Classifier

Nazgul Rakhimzhanova

COURSE SCHEDULE

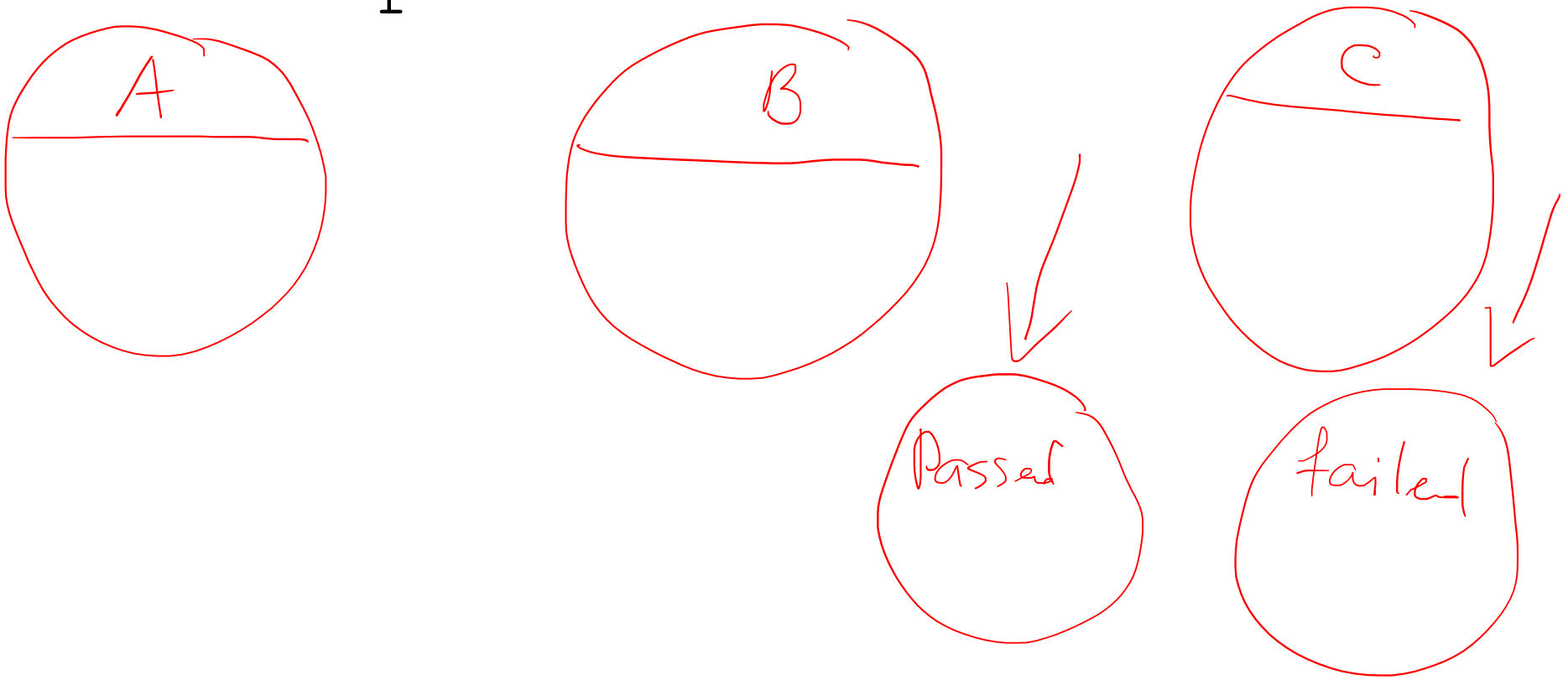
week	Mid Term (weeks 01-07)	End Term (weeks 08-14)	week
01	Intro: Data Science Area and open source tools for Data Science	Statistics: Distribution – Lognormal, Exponential	08
02	NumPy package for data science	Sampling and Estimation	09
03	Pandas package for data science	Visualization II. Correlation and Covariance	10
04	Visualization with matplotlib	Hypothesis testing	11
05	Statistics: Distribution – Normal	Decision Tree Classifier	12
06	Exploratory Data Analysis (EDA)	Linear Regression	13
07	**Summary for 6 weeks QA session**	**Summary for 6 weeks QA session**	**14**
15	**Course summary**		

AGENDA

- Classification task
- Decision Tree Classification

IDEA

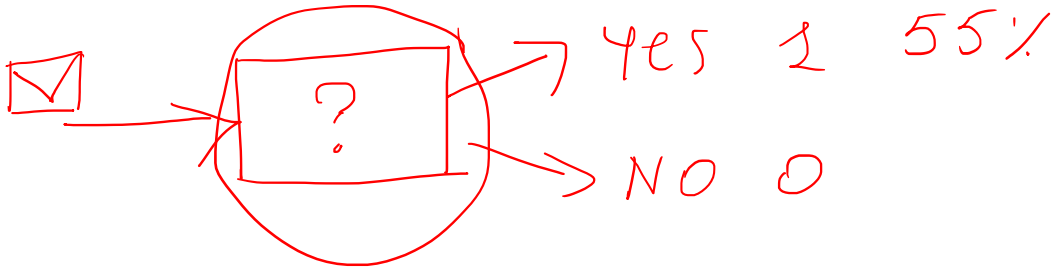
- "Can I guess which students will get A, B and C grades using some known parameters"



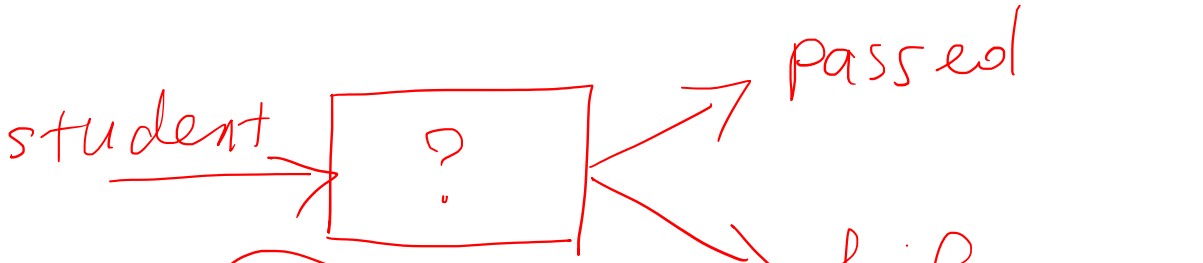
CLASSIFICATION TASK

fraud detection

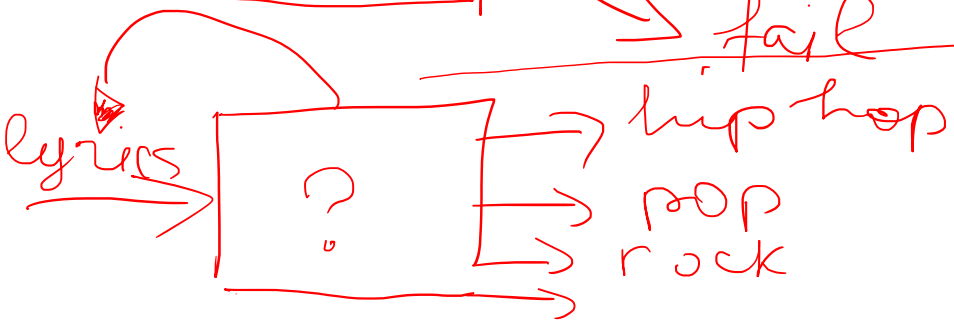
spam detection



student perf.

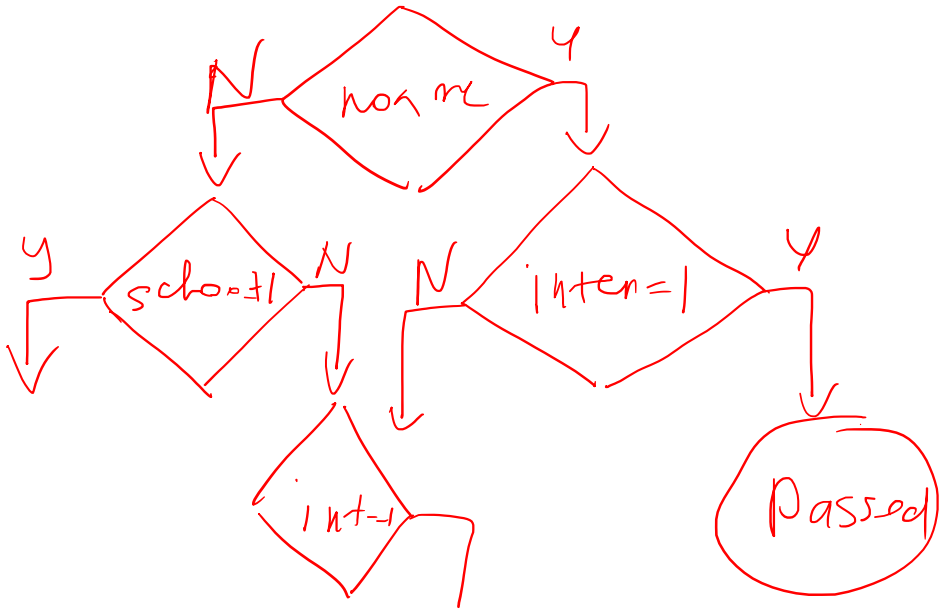


Song lyric genre detect.



IDEA

Student	nom	age	location	school	interes
	mc	20	Almaty	2	1

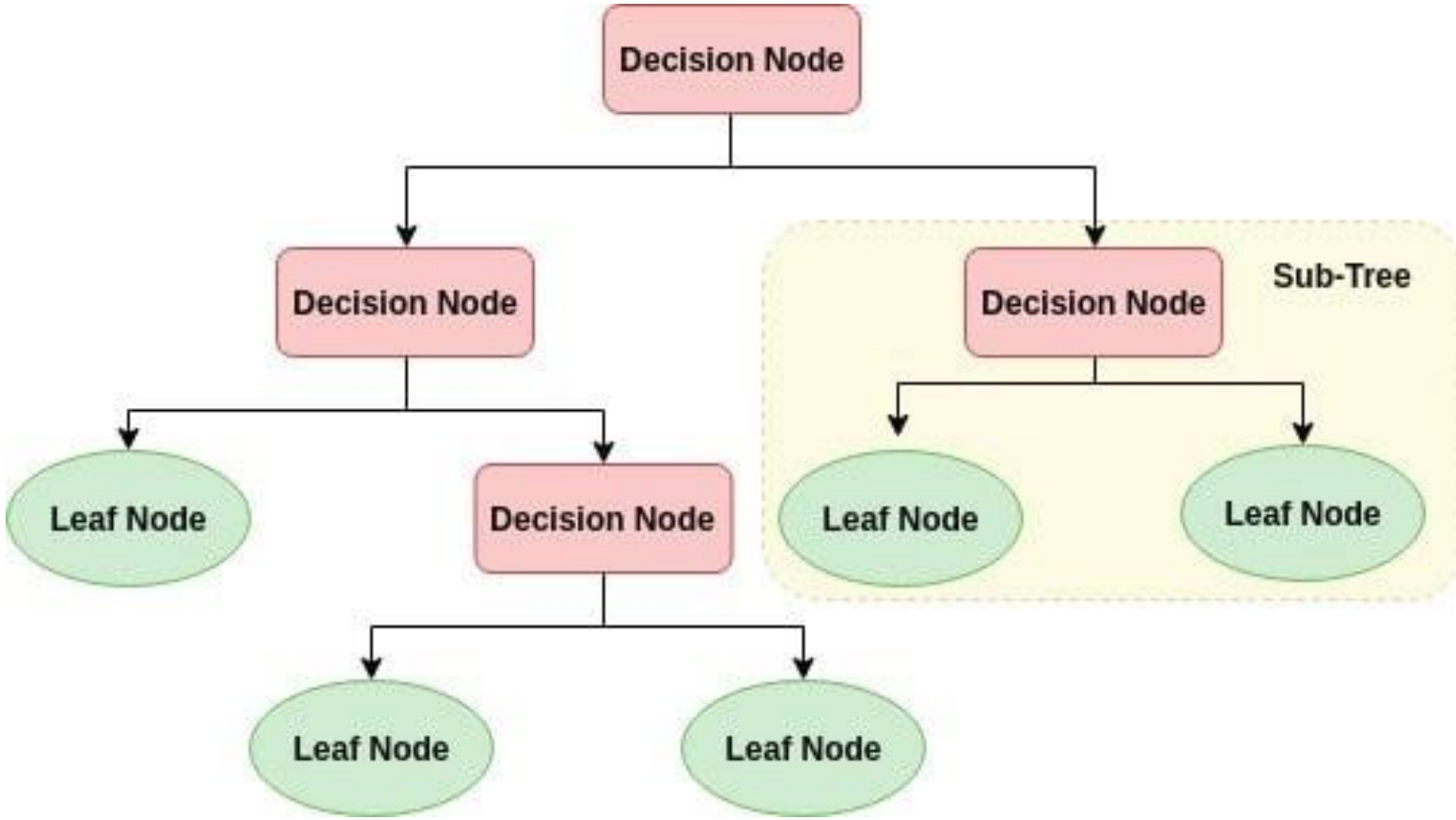


0 –
1 –
2 –

0 –
1 –

DECISION TREE

DECISION TREE ALGORITHM



DECISION TREE

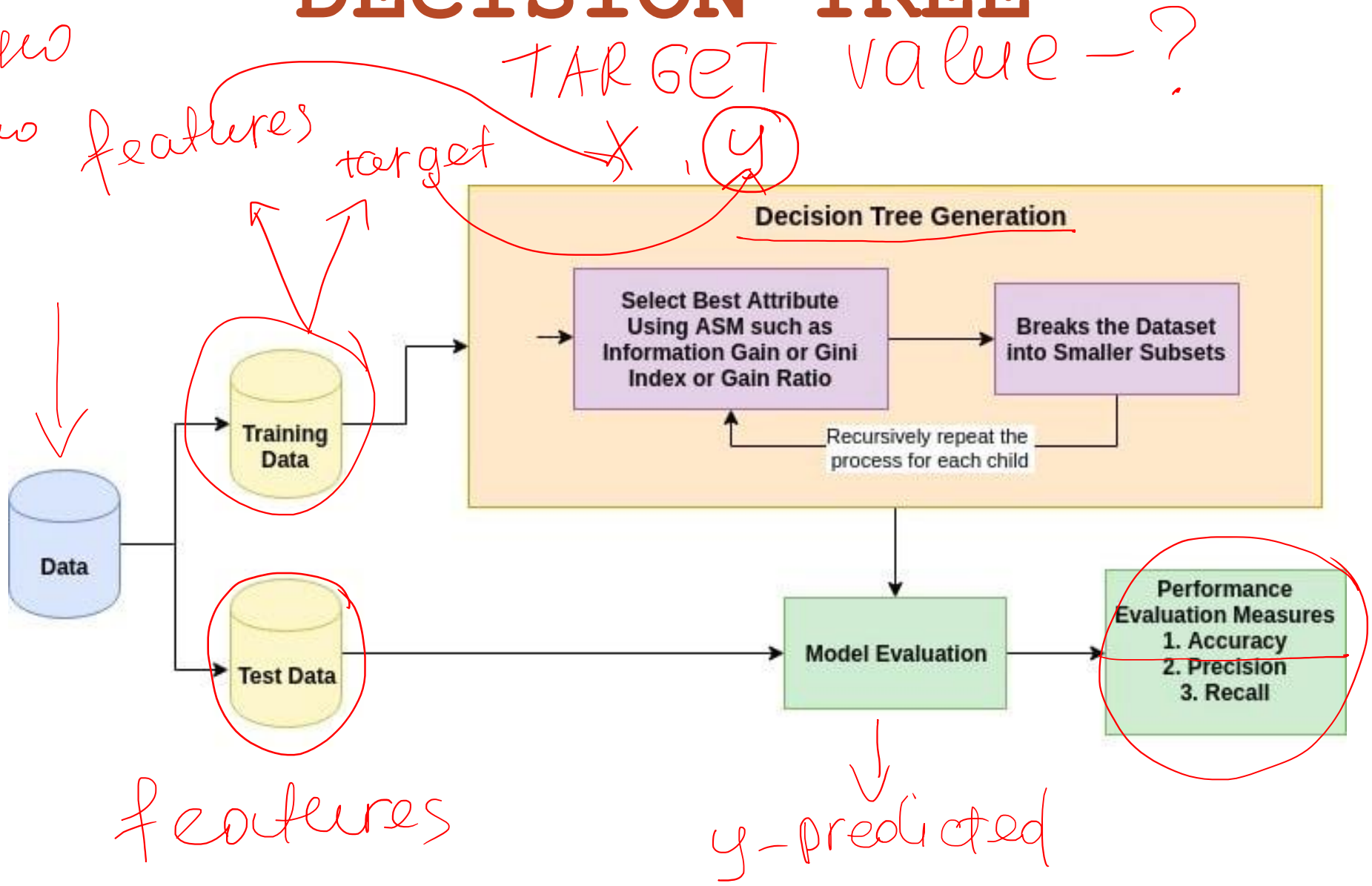
- Decision Trees are one of the best-known supervised classification methods
- A tree is composed of nodes, and those nodes are chosen looking for the optimum split of the features

DECISION TREE

- Decision Tree is a white box type of ML algorithm
- It shares internal decision-making logic, which is not available in the black box type of algorithms such as Neural Network
- The decision tree is a distribution-free or non-parametric method, which does not depend upon probability distribution assumptions
- The algorithm is designed to find the optimal point of the most predictive feature in order to split 1 dataset into 2

A = 90-100
отлично
хорошо

DECISION TREE



DECISION TREE

DECISION TREE MODEL

`sklearn.tree.DecisionTreeClassifier`

pip install sklearn

```
>>> from sklearn.datasets import load_iris
>>> from sklearn.model_selection import cross_val_score
>>> from sklearn.tree import DecisionTreeClassifier
```

https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

PURITY

- Pure when some class is homogenous
- Decision tree algorithms use **information gain** to split a node.
- **Gini index** and **entropy** are the criteria for calculating information gain.
- A node having multiple classes is impure whereas a node having only one class is pure.

INFORMATION GAIN

- Information Gain, or **IG** for short, measures the reduction in entropy or surprise by splitting a dataset according to a given value of a random variable.
- Lower probability events have more information, higher probability events have less information. Entropy quantifies how much information there is in a random variable, or more specifically its probability distribution. A skewed distribution has a low entropy, whereas a distribution where events have equal probability has a larger entropy.

INFORMATION GAIN

- It is commonly used in the construction of decision trees from a training dataset, by evaluating the information gain for each variable, and selecting the variable that maximizes the information gain, which in turn minimizes the entropy and best splits the dataset into groups for effective classification.

GINI

- **The gini impurity** measures the frequency at which any element of the dataset will be **mislabelled** when it is randomly labeled.

$$GiniIndex = 1 - \sum_j p_j^2$$

GINI

- The minimum value of the Gini Index **is 0.**
- **Gini Index = 0** when node is pure. Therefore, this node will not be split again.
- Thus, **the optimum split** is chosen by the features with less Gini Index.

$$Gini_{min} = 1 - (1^2) = 0$$

$$Gini_{max} = 1 - (0.5^2 + 0.5^2) = 0.5$$

ENTROPY

- **Entropy is a measure** of information that indicates the **disorder** of the features with the target.

$$Entropy = -\sum_j p_j \cdot log_2 \cdot p_j$$

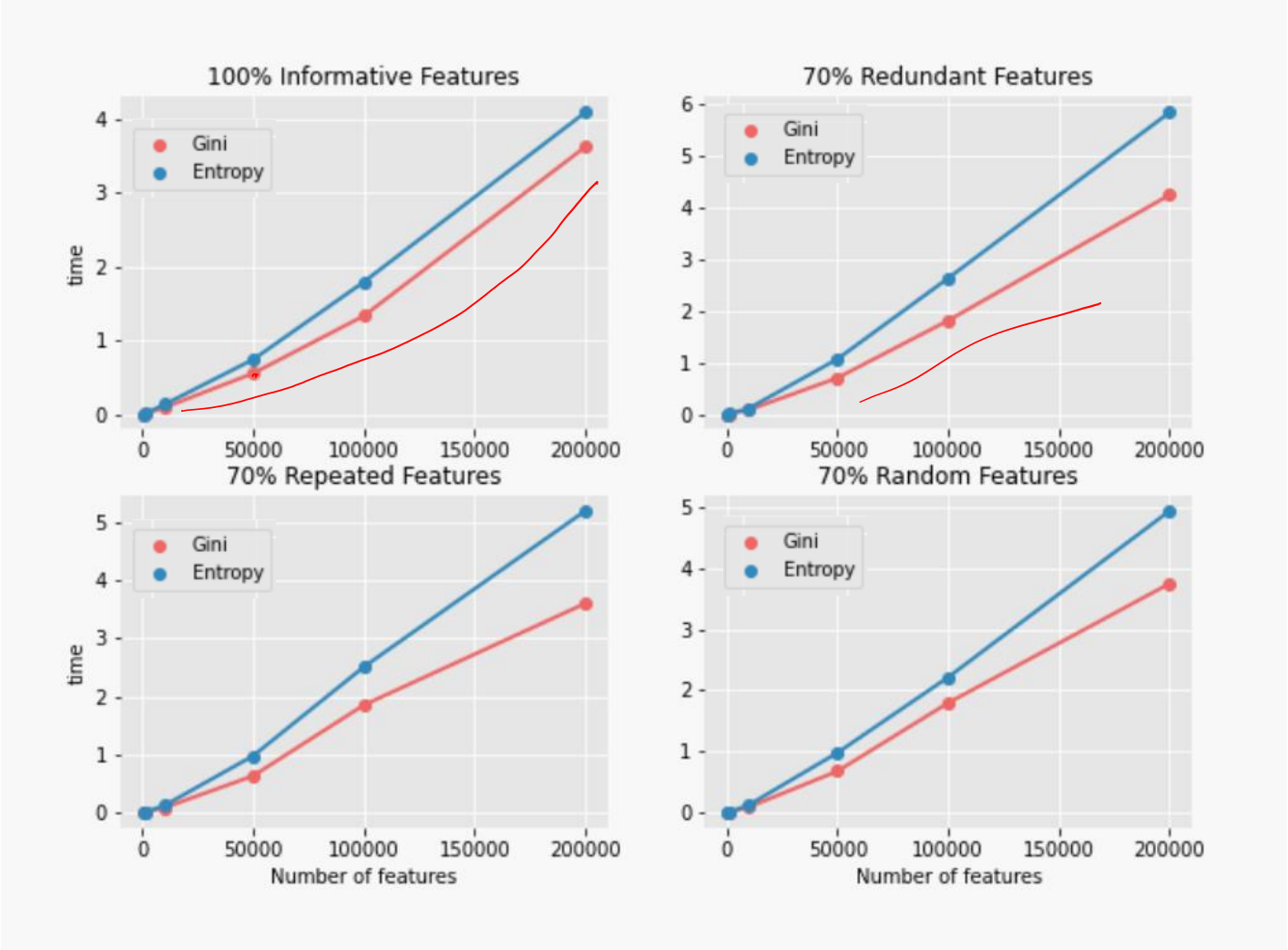
ENTROPY

- Entropy minimum value is 0
- Entropy maximum value is 1

$$Entropy_{min} = -1 \cdot log_2(1) = 0$$

$$Entropy_{max} = -0.5 \cdot log_2(0.5) - 0.5 \cdot log_2(0.5) = 1$$

GINI vs. ENTROPY



READINGS

- https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html
- https://machinelearningmastery.com/what-is-information-entropy/
- https://scikit-learn.org/stable/modules/tree.html