Naive Bayes Classifier using Python

Tushar B. Kute,
http://tusharkute.com

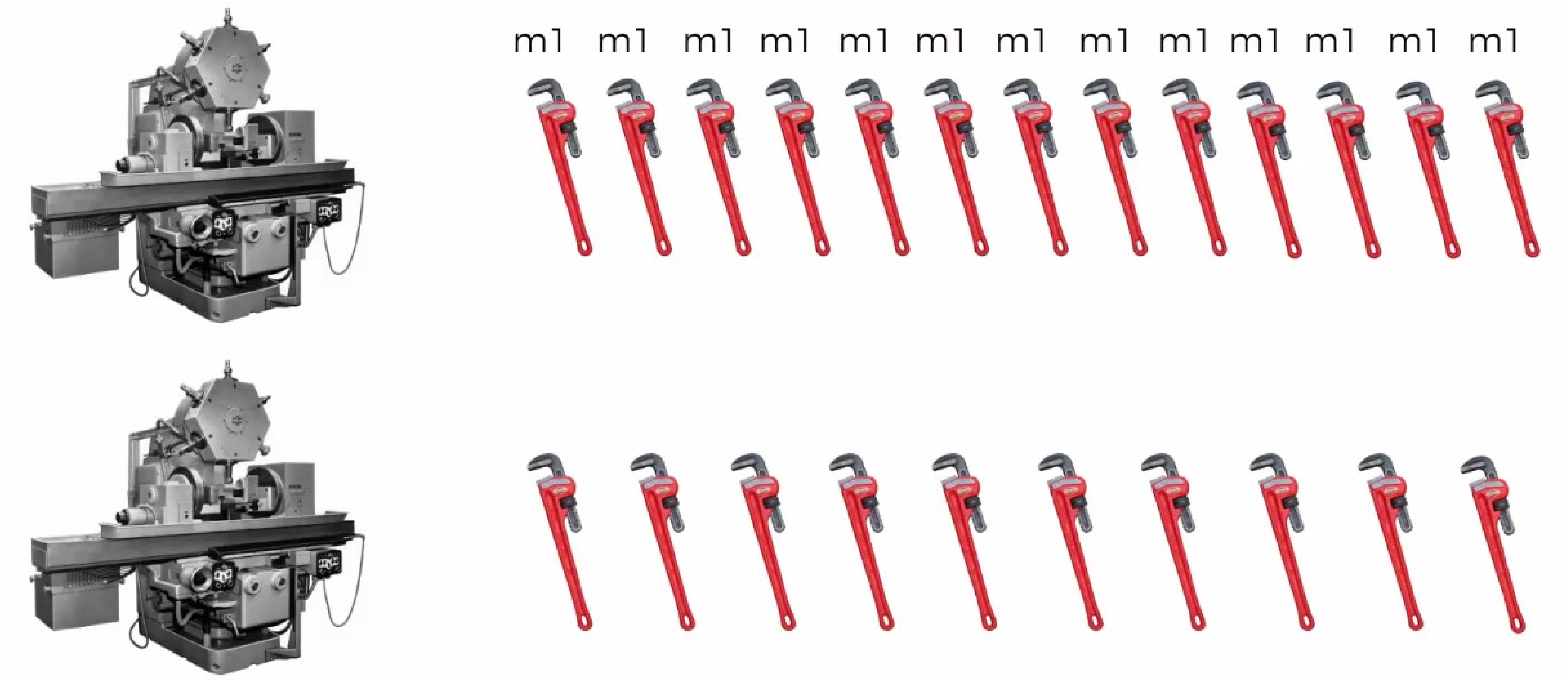
$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

Thomas Bayes
1702 - 1761

Naive Bayes Classifier

- Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem.
- It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Bayes Theorem



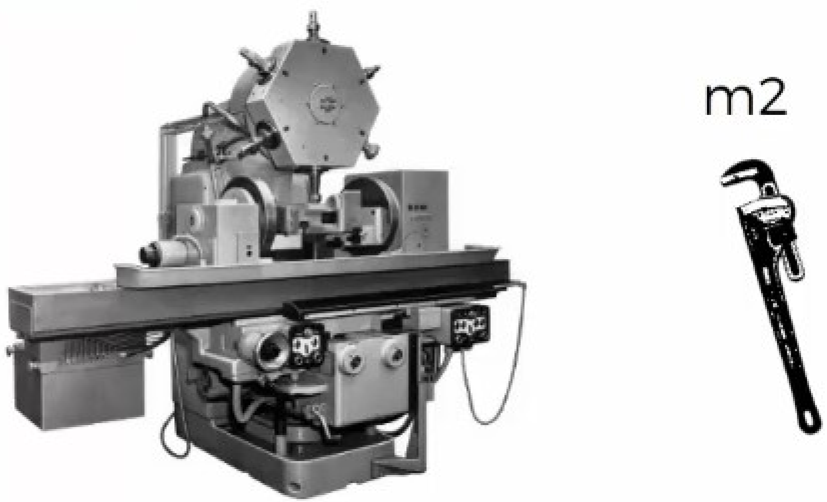
Example Reference: Super Data Science

Bayes Theorem

Defective Spanners

Bayes Theorem

What’s the probability?



Bayes Theorem

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Bayes Theorem

Mach1: 30 wrenches / hr -> P(Mach1) = 30/50 = 0.6
Mach2: 20 wrenches / hr -> P(Mach2) = 20/50 = 0.4

Out of all produced parts:
We can SEE that 1% are defective -> P(Defect) = 1%

Out of all defective parts:
We can SEE that 50% came from mach1 -> P(Mach1 | Defect) = 50%
And 50% came from mach2 -> P(Mach2 | Defect) = 50%

Question:
What is the probability that a part produced by mach2 is defective = ? -> P(Defect | Mach2) = ?

Bayes Theorem

$$P(\text{Defect} \mid \text{Mach2}) = \frac{P(\text{Mach2} \mid \text{Defect}) * P(\text{Defect})}{P(\text{Mach2})}$$

$$P(\text{Defect} \mid \text{Mach2}) = \frac{0.5 \quad * \quad 0.01}{0.4} = 0.0125$$

That's intuitive

$$P(\text{Defect} \mid \text{Mach2}) = \frac{P(\text{Mach2} \mid \text{Defect}) \; * \; P(\text{Defect})}{P(\text{Mach2})} = 1.25\%$$

Let's look at an example:

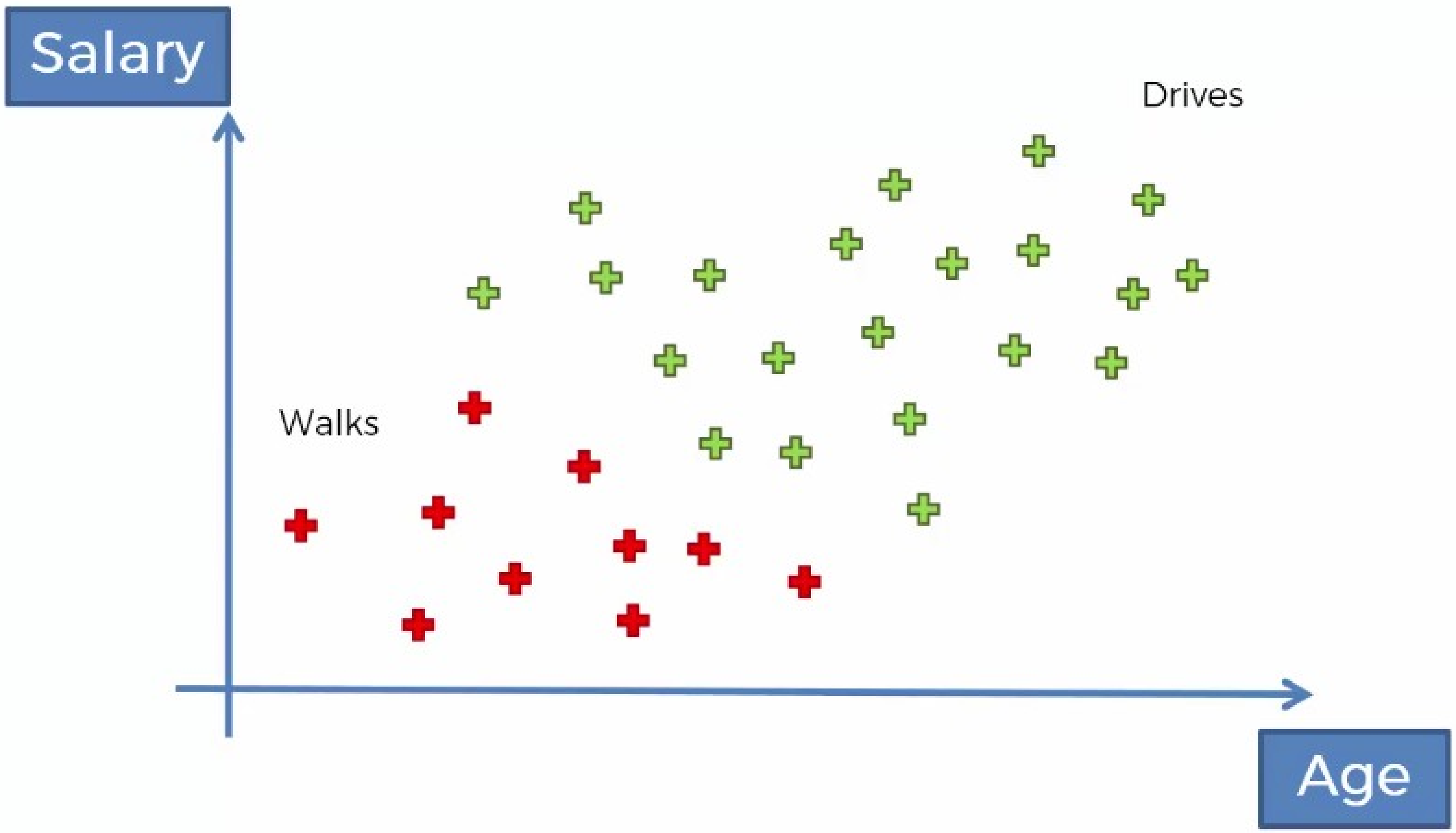
- 1000 wrenches
- 400 came from Mach2
- 1% have a defect = 10
- of them 50% came from Mach2 = 5
- % defective parts from Mach2 = 5/400 = 1.25%

Exercise

Quick exercise:

P(Defect | Mach1) = ?

Example:

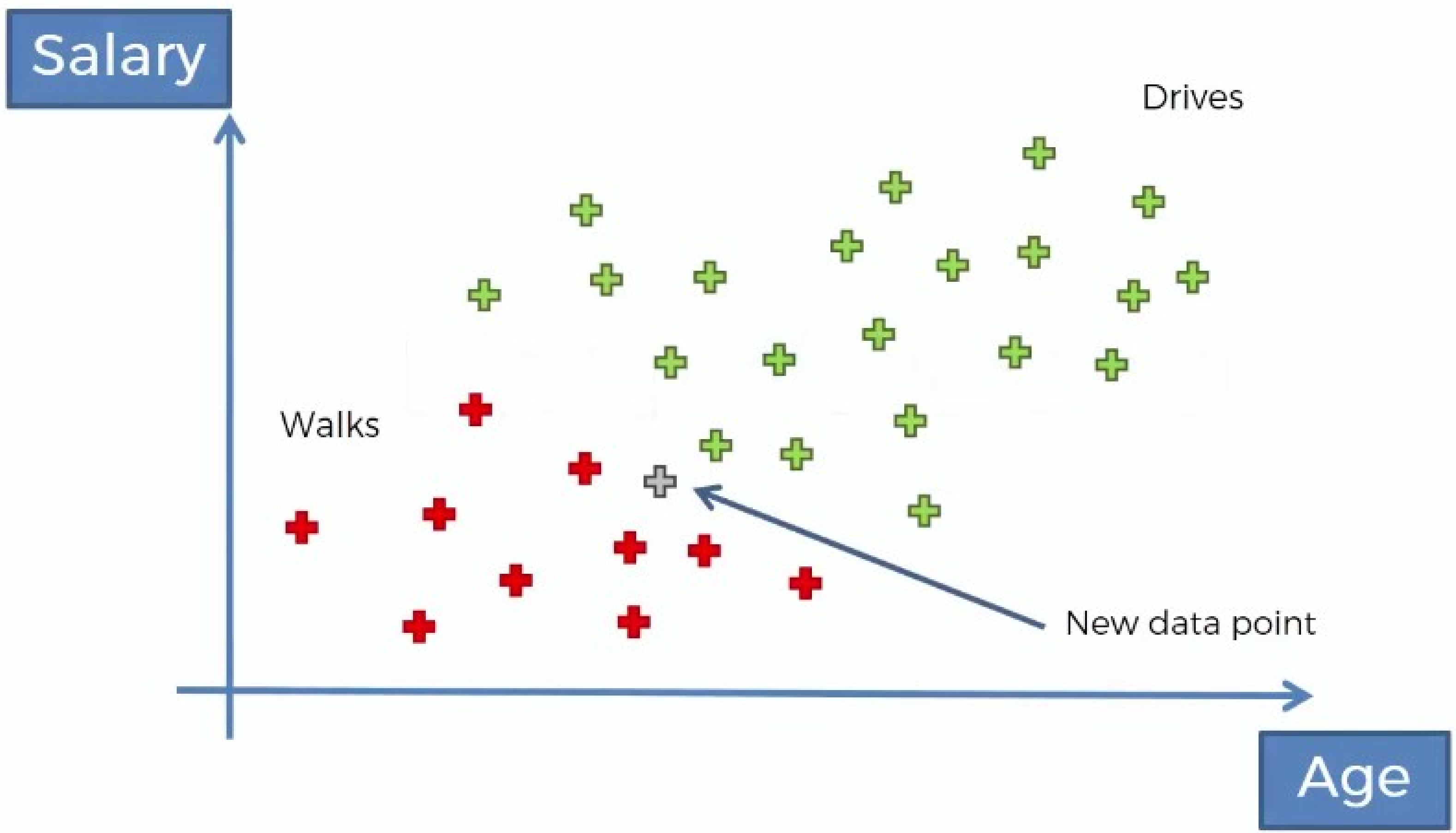


Step-1

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

$$P(Walks|X) = \frac{P(X|Walks) * P(Walks)}{P(X)}$$

Step-1



Step-1

#4 Posterior Probability

#3 Likelihood

#1 Prior Probability

$$P(Walks|X) = \frac{P(X|Walks) * P(Walks)}{P(X)}$$

#2 Marginal Likelihood

Step-2

#4 Posterior Probability

#3 Likelihood

#1 Prior Probability

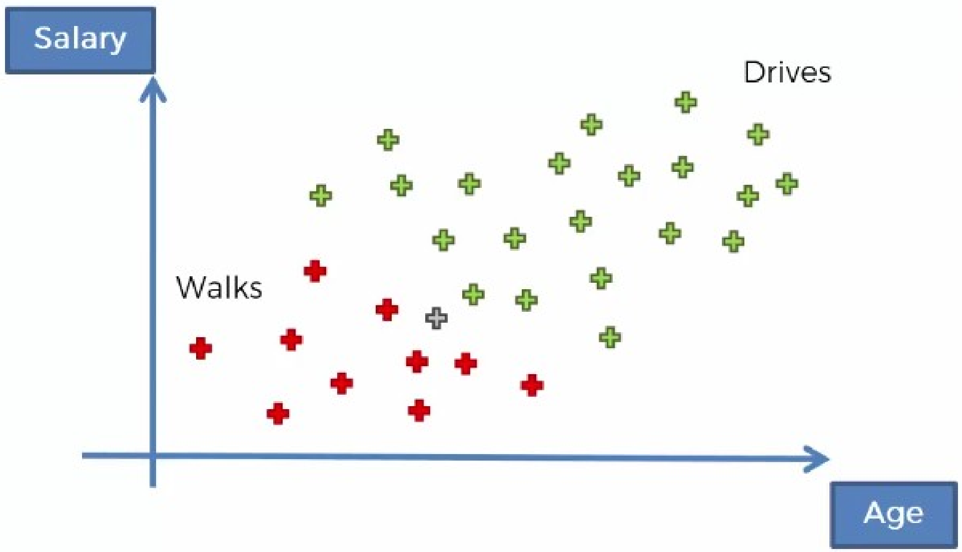
$$P(Drives|X) = \frac{P(X|Drives) * P(Drives)}{P(X)}$$

#2 Marginal Likelihood

Step-3

$$P(Walks|X)\ v.s.\ P(Drives|X)$$

Naive Bayes – Step-1

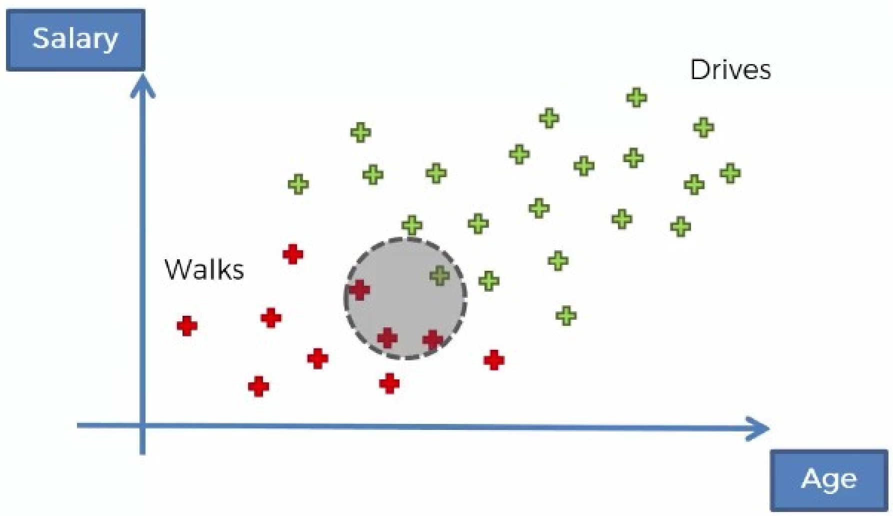


#1. P(Walks)

$$P(Walks) = \frac{Number\ of\ Walkers}{Total\ Observations}$$

$$P(Walks) = \frac{10}{30}$$

Naive Bayes – Step-2

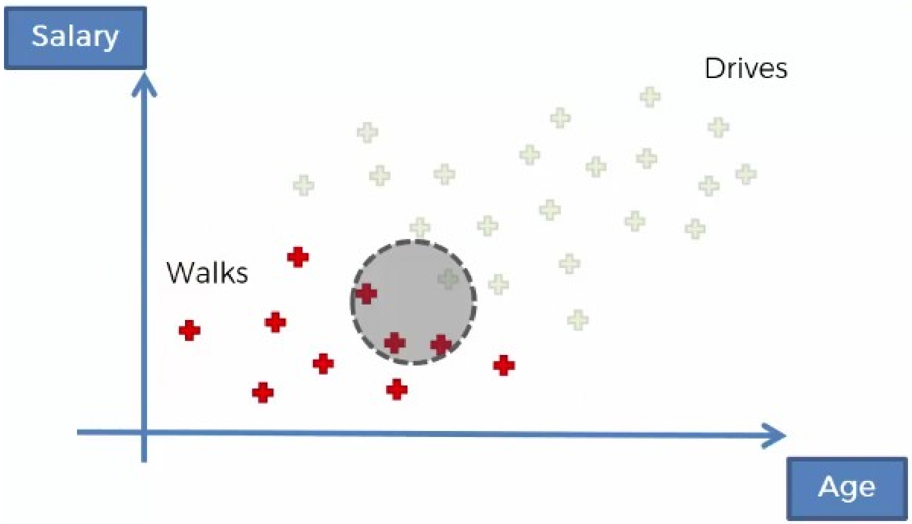


#2. P(X)

$$P(X) = \frac{Number\ of\ Similar\ Observations}{Total\ Observations}$$

$$P(X) = \frac{4}{30}$$

Naive Bayes – Step-3



#3. P(X|Walks)

$$P(X|Walks) = \frac{\substack{Number\ of\ Similar \\ Observations \\ Among\ those\ who\ Walk}}{Total\ number\ of\ Walkers}$$

$$P(X|Walks) = \frac{3}{10}$$

Combining altogether

#4 Posterior Probability

#3 Likelihood

#1 Prior Probability

$$P(Walks|X) = \frac{\frac{3}{10} * \frac{10}{30}}{\frac{4}{30}} = 0.75$$

#2 Marginal Likelihood

Naive Bayes – Step-4

#4 Posterior Probability

#3 Likelihood

#1 Prior Probability

$$P(Drives|X) = \frac{P(X|Drives) * P(Drives)}{P(X)}$$

#2 Marginal Likelihood

Naive Bayes – Step-5

#4 Posterior Probability

#3 Likelihood

#1 Prior Probability

$$P(Drives|X) = \frac{\frac{1}{20} * \frac{20}{30}}{\frac{4}{30}} = 0.25$$

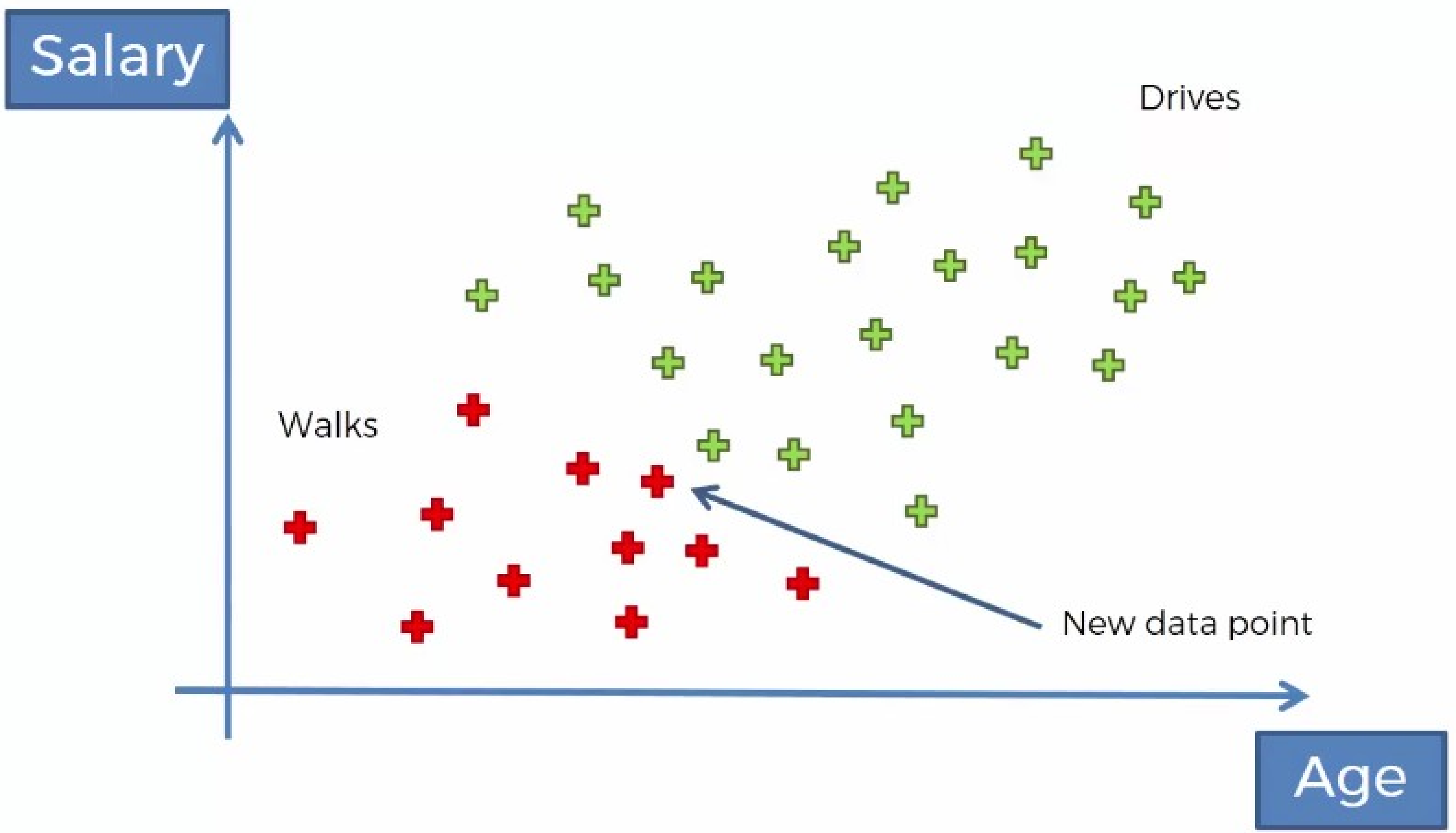
#2 Marginal Likelihood

Types of model

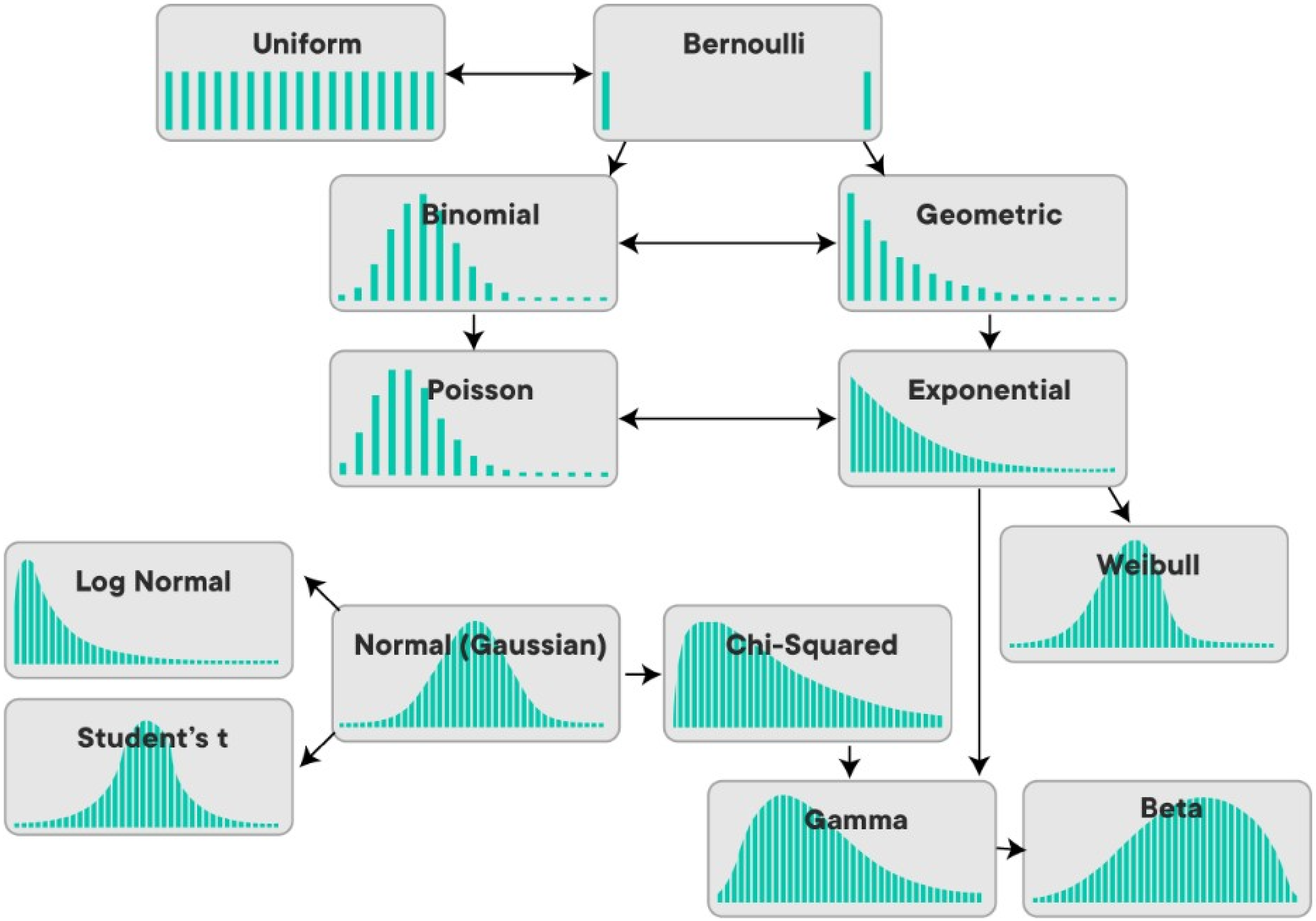
$$P(Walks|X)\ v.s.\ P(Drives|X)$$

$$0.75\ v.s.\ 0.25$$

Final Classification



Probability Distribution



Advantages

- When assumption of independent predictors holds true, a Naive Bayes classifier performs better as compared to other models.
- Naive Bayes requires a small amount of training data to estimate the test data. So, the training period is less.
- Naive Bayes is also easy to implement.

Disadvantages

- Main imitation of Naive Bayes is the assumption of independent predictors. Naive Bayes implicitly assumes that all the attributes are mutually independent. In real life, it is almost impossible that we get a set of predictors which are completely independent.
- If categorical variable has a category in test data set, which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as Zero Frequency. To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

Useful resources

- www.datacamp.com
- www.scikit-learn.org
- www.towardsdatascience.com
- www.medium.com
- www.analyticsvidhya.com
- www.kaggle.com
- www.stephacking.com
- www.github.com

Thank you

@mitu_skillologies

/mITuSkillologies

@mitu_group

/company/mitu-skillologies

MITUSkillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

contact@mitu.co.in

tushar@tusharkute.com