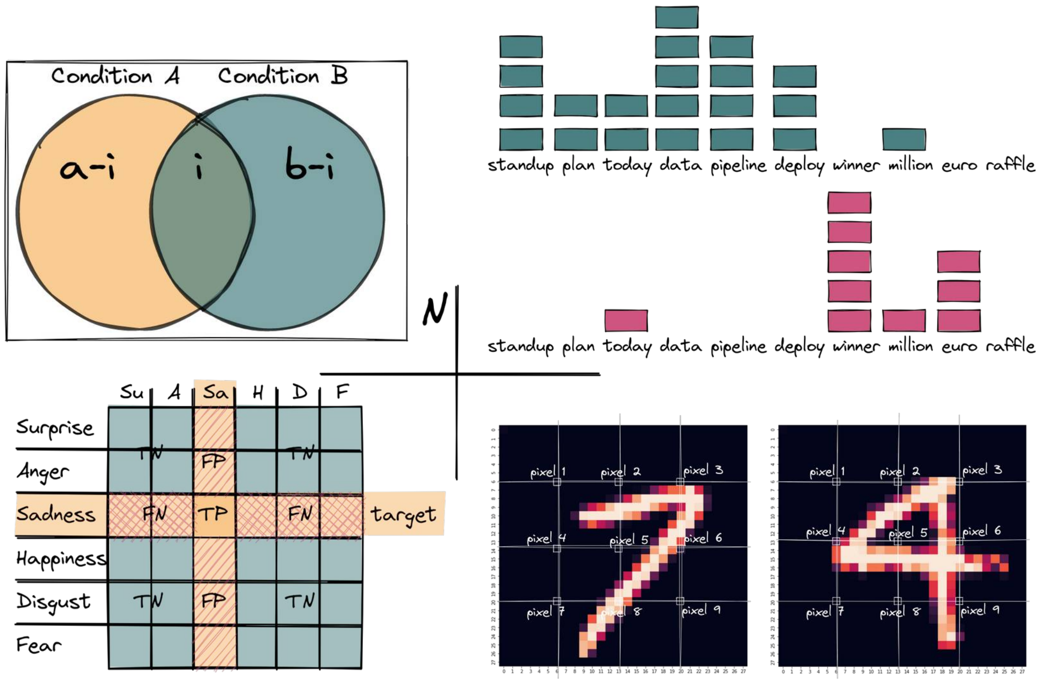


Naive Bayes

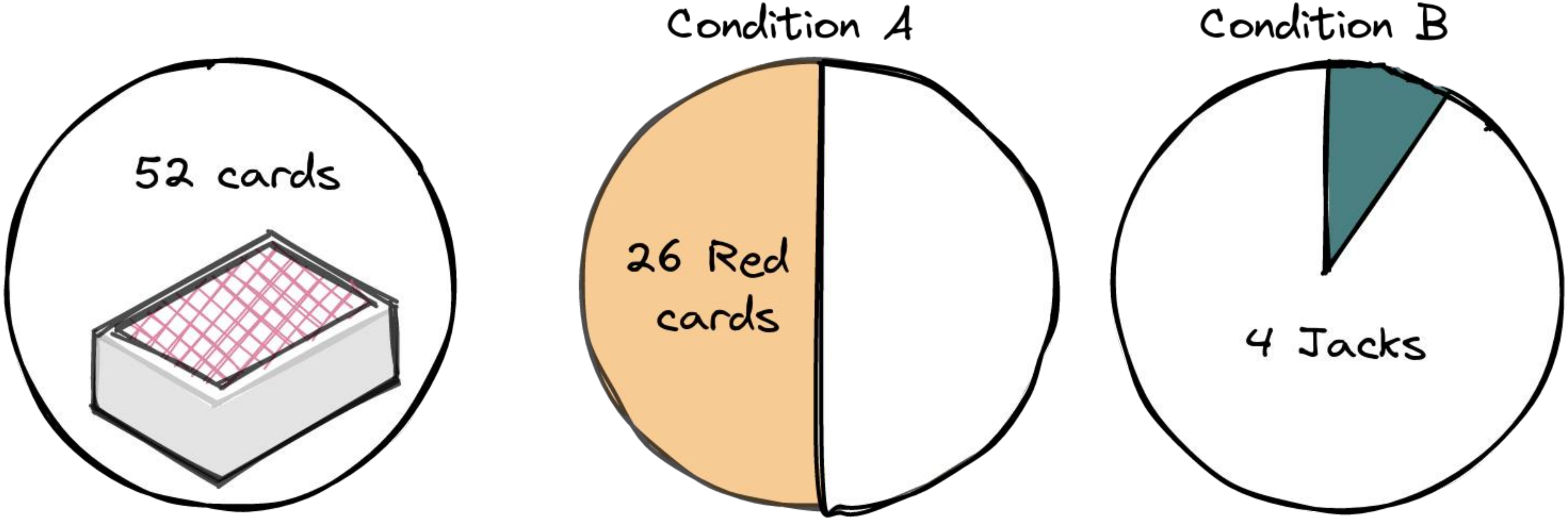
Week 14

Middlesex University Dubai;
CST4050; Instructor: Ivan Reznikov

Plan

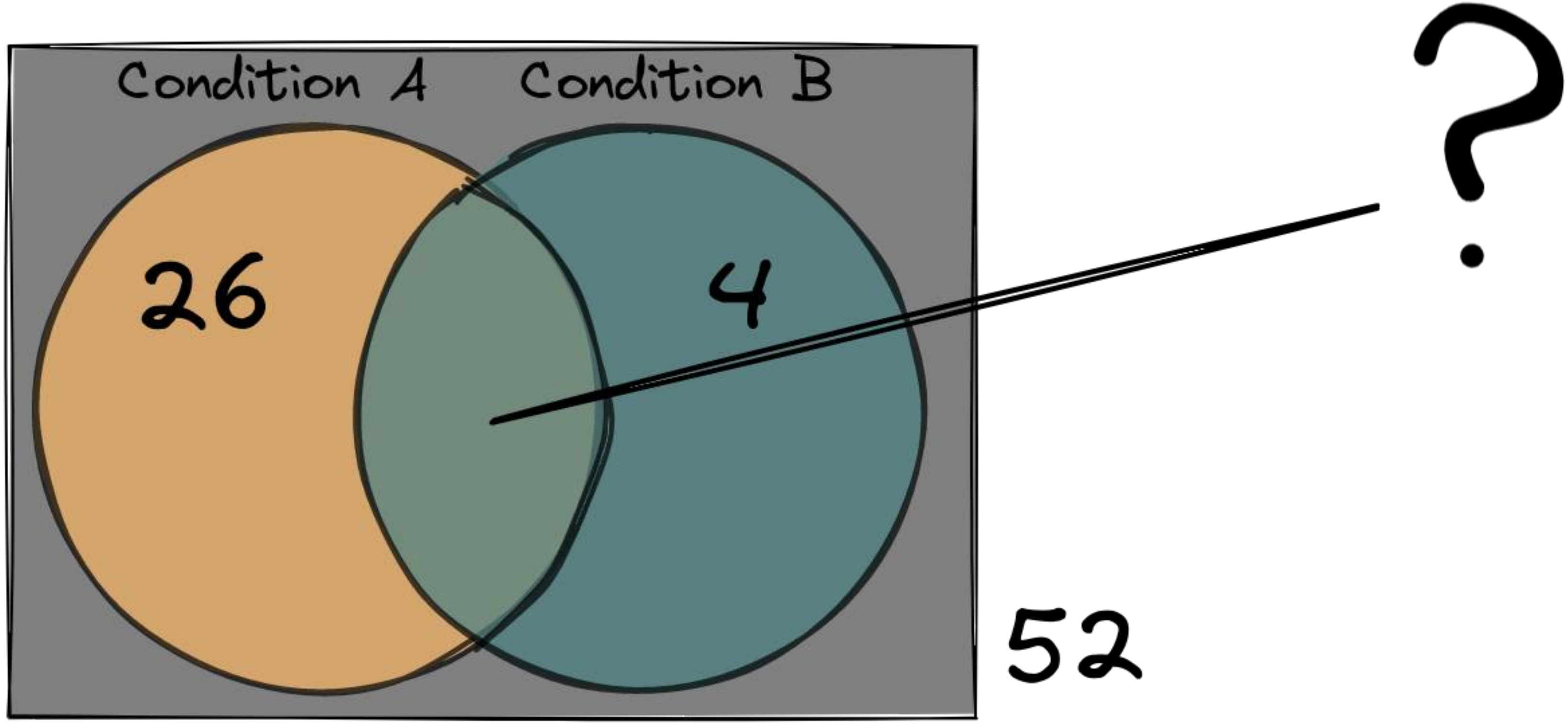
- Bayes' Theorem:
  - Card example
  - Derive Bayes' Theorem
  - Probability Tree
  - Independence of events
- Classification metrics
  - Accuracy, confusion matrix, recall, precision, f1-score
  - Error types
- Naive Bayes
  - Naive assumption
  - Practical cases

Card example: conditions

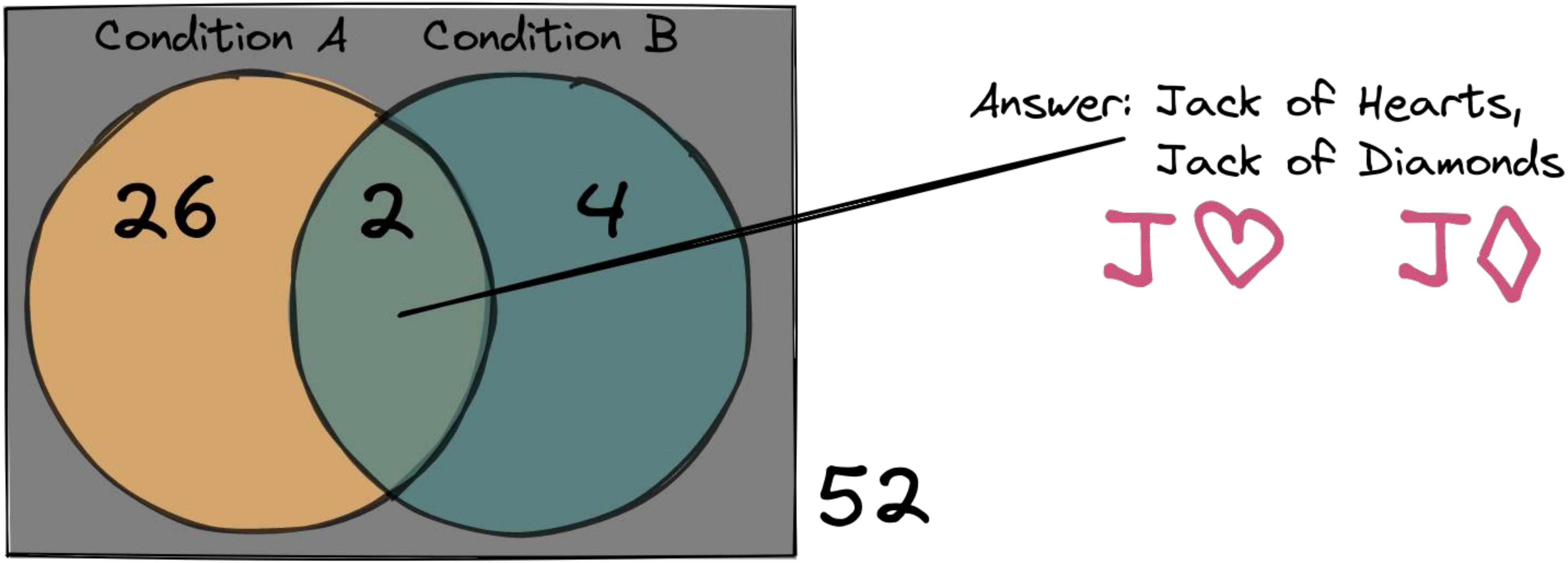


Condition A: Card is Red
Condition B: Card is Jack

Card example: intersection



Card example: intersection



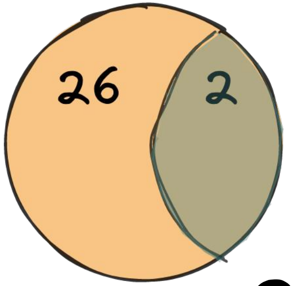
Probabilities

Marginal probability:

$$P(\text{Red}) = \frac{26}{52} = \frac{1}{2} \qquad P(\text{Jack}) = \frac{4}{52} = \frac{1}{13}$$

Conditional probability:

What is the probability of Jack if (*condition*) card is Red?



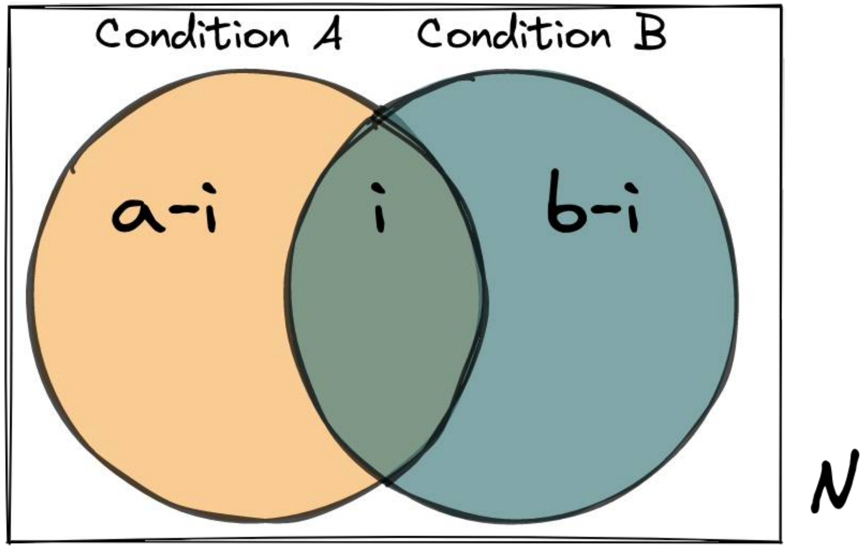
$$P\underbrace{(\text{Jack} \mid \text{Red})}_{\text{condition}} = \frac{2}{26} = \frac{1}{13}$$

Joint probability:

What is the probability of Jack and Red?

$$P\underbrace{(\text{Red} \cap \text{Jack})}_{\text{intersection}} = \frac{2}{52} = \frac{1}{26}$$

Bayes' Theorem



What is the probability of A if B happened?

$$P(A \mid B) = \frac{i}{b}$$

What is the probability of A and B happening?

$$P(A \cap B) = \frac{i}{N}$$

What is the probability of B happening?

$$P(B) = \frac{b}{N} \quad \Rightarrow \quad \boxed{P(A \mid B) = \frac{P(A \cap B)}{P(B)}}$$

Bayes' Theorem

The probability of A if B happened:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

More often we calculate the probability of both A and B happening:

$$P(A \cap B) = \boxed{P(B) * P(A \mid B) = P(A) * P(B \mid A)}$$

The probability of A being True (Prior probability)

The probability of B being true given that A is True (Likelihood)

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

The probability of A being true given that B is True (Posterior probability)

The probability of B being True (Evidence)

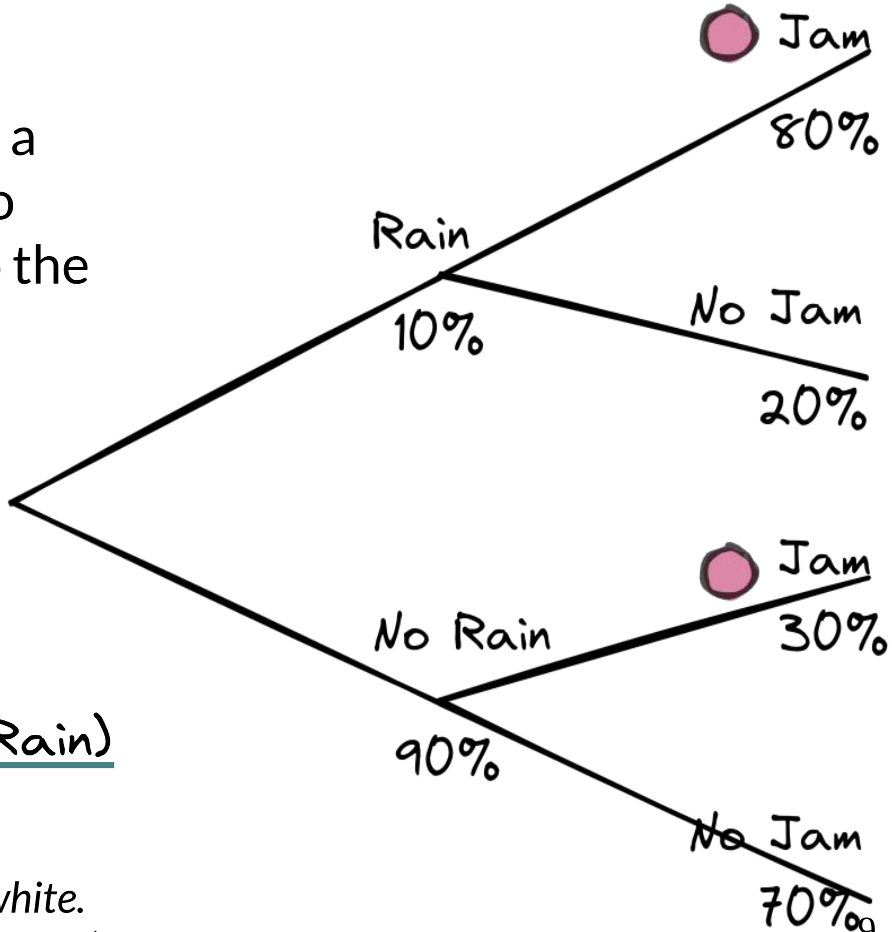
Probability Tree

What is the probability of us getting stuck in a jam today? Intuition suggests that in order to calculate this probability, we need to sum up the following multiplication results:

$$P\ (Jam) = 0.1 * 0.8 + 0.9 * 0.3 = 0.35$$

In fact, we are applying Bayes' theorem:

$$P\ (Jam) = \underline{P\ (Rain) * P\ (Jam \mid Rain)} + \\ + \underline{P\ (No\ Rain) * P\ (Jam \mid No\ Rain)}$$



DS Interview question: you have 12 beads – 7 black and 5 white. What is the probability of taking 1b+1w simultaneously? (35/66)

Independence of events

What if one would like to learn if two events are independent? To answer it, we need to understand what is independent? Intuition hints to us that events A and B are independent if the probability that event B will occur remains the same, regardless of whether event A has occurred or not:

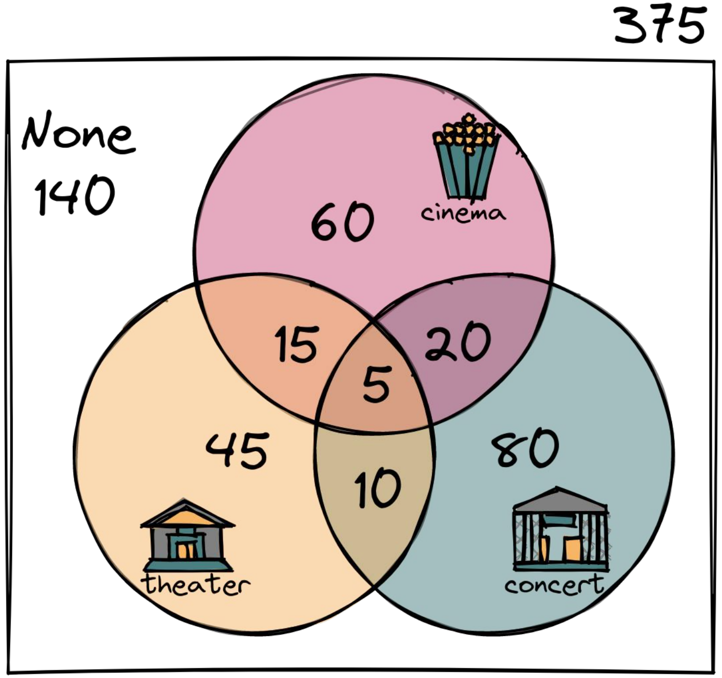
$$P(B) = P(B \mid A)$$

We could've achieved the above formula from the following ones:

independent events: $P(A \cap B) = P(A) * P(B)$
conditional events: $P(A \cap B) = P(A) * P(B \mid A)$

Independence of events: Case

Suppose we took a survey from 375 people and asked if they've visited theater, cinema or concert in the last year at least once:



Let's check if visiting cinema and concert are independent events:

P (cinema) * P (concert) = 100/375 *
* 115/375 = 0.8177

P (cinema ∩ concert) = 25/375 = 0.666

Events are dependent, as 0.8177 ≠ 0.666

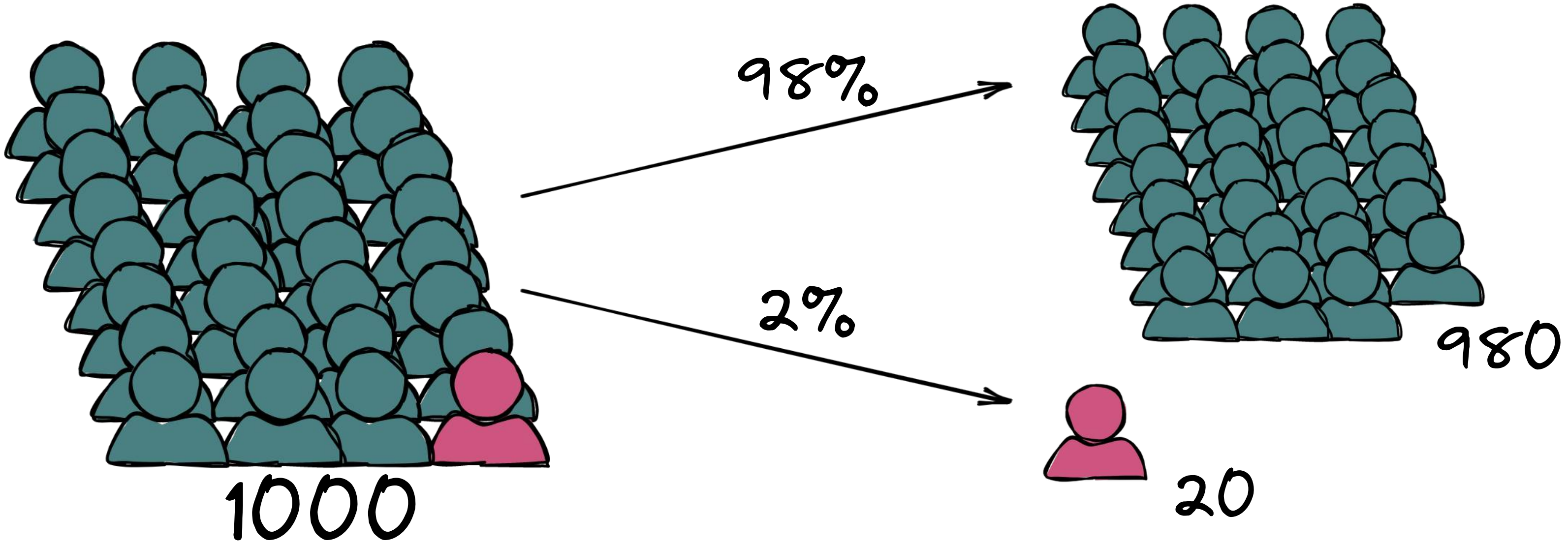
Now let's check if visiting theater and cinema are independent events:

P (theater) * P (cinema) = 75/375 * 100/375 = 0.5333

P (cinema ∩ concert) = 20/375 = 0.5333

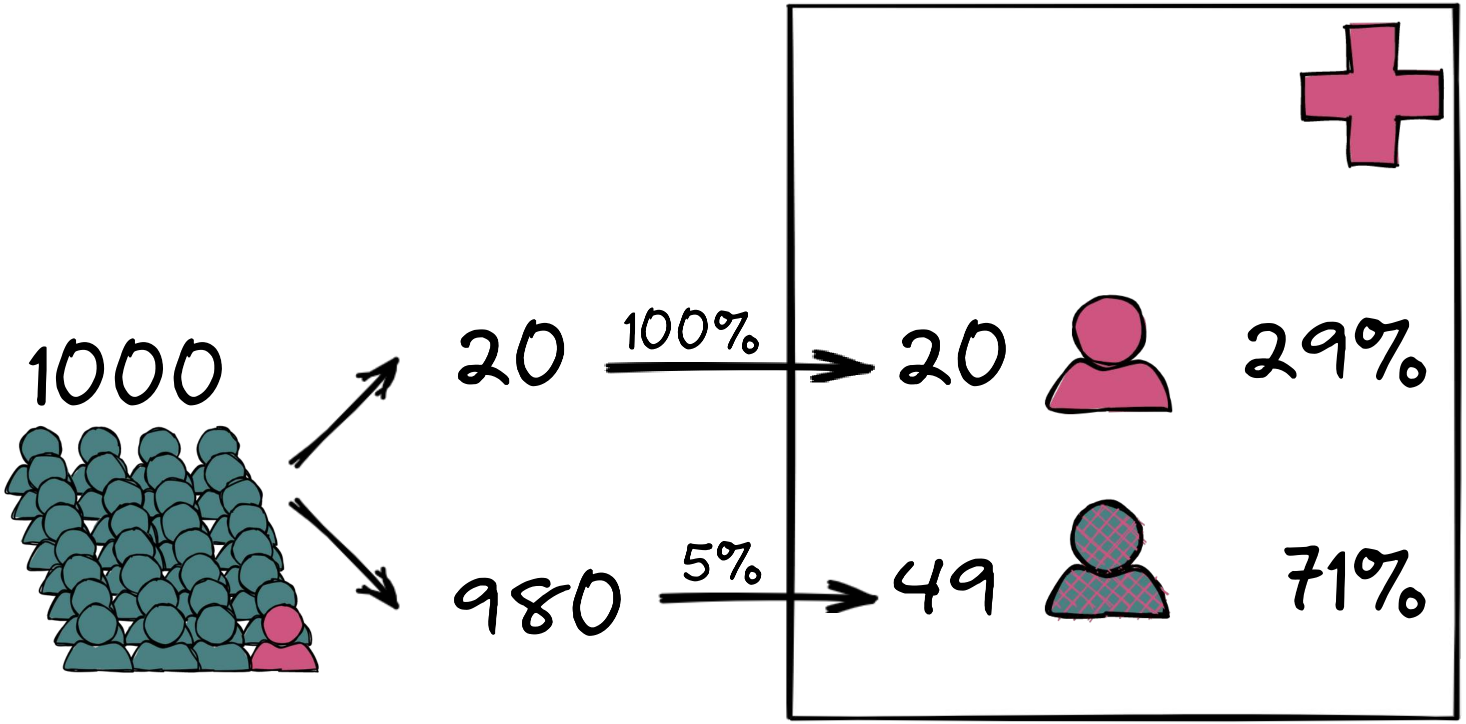
Events are independent, as 0.5333 == 0.5333

Classification metrics: Case1



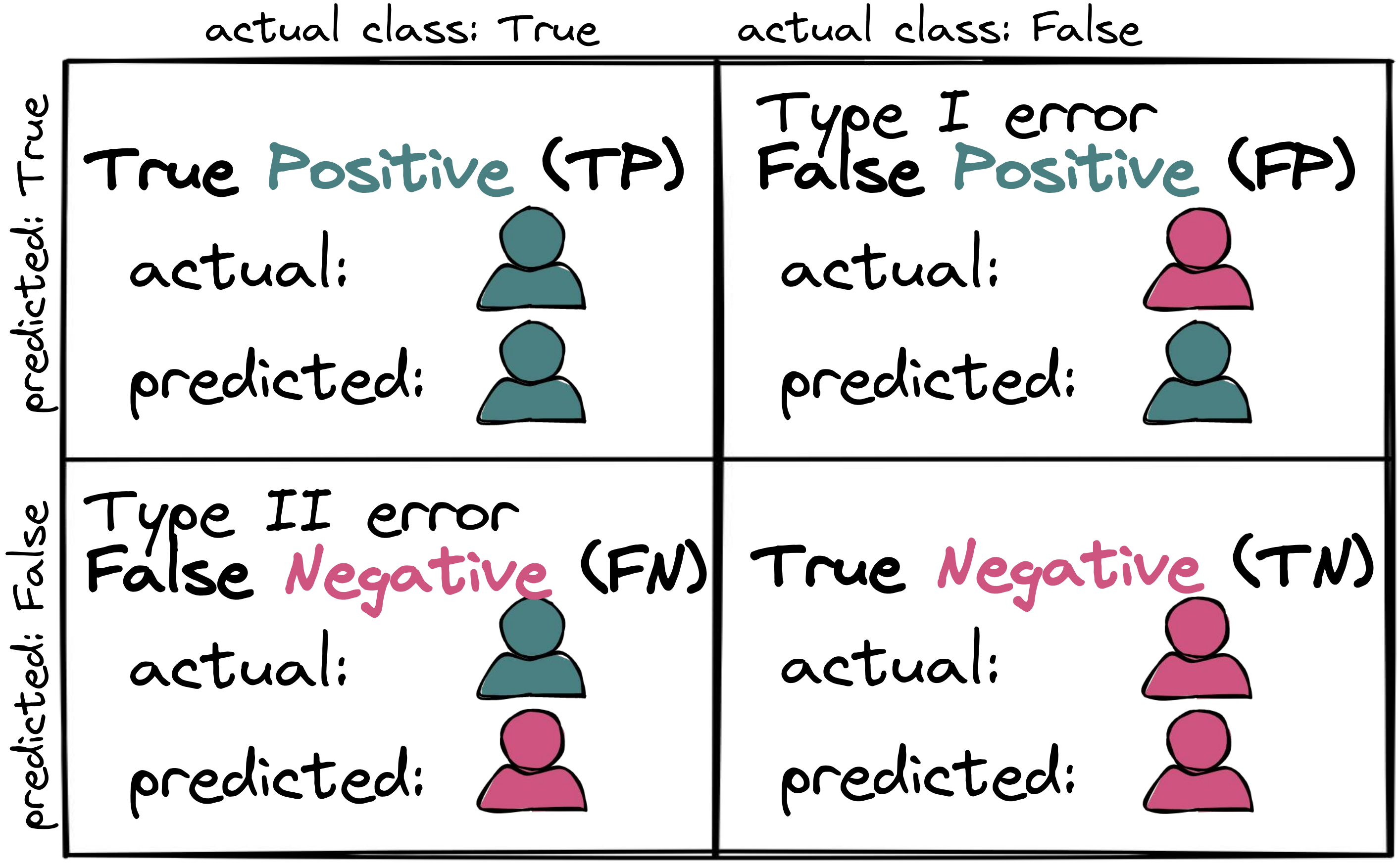
Disease Test: 100% correct if ill
95% correct if not ill

Classification metrics: Case1



71% of people in the hospital aren't ill.
The error is much higher than 5%

Confusion matrix



$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Sensitivity (Recall)} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

$$NPV^{*} = \frac{TN}{TN+FN}$$

$$F1 = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

*NPV - negative predicted value

Confusion matrix: metrics

Precision relates to the amount of false positive results (type I errors): high precision means small amount of type I errors.

Recall relates to the amount of false negative results (type II errors): high recall means small amount of type II errors

One may use the following observation to remember:
precision – **p**ositive, r**e**call – n**e**gative

F1-score is a metric, taking both precision and recall into account. One of the main reasons, why we multiple precision and recall instead of taking, say, their average, is to deal with extremely low values.

Confusion matrix: Case1

actual class

predicted class

True Positive (TP) act: pred: 931	False Positive (FP) act: pred: 0
False Negative (FN) act: pred: 49	True Negative (TN) act: pred: 20

$$\text{Precision} = \frac{931}{931+0} = 1$$

$$\text{Sensitivity (Recall)} = \frac{931}{931+49} = 0.95$$

$$\text{Specificity} = \frac{20}{20+0} = 1$$

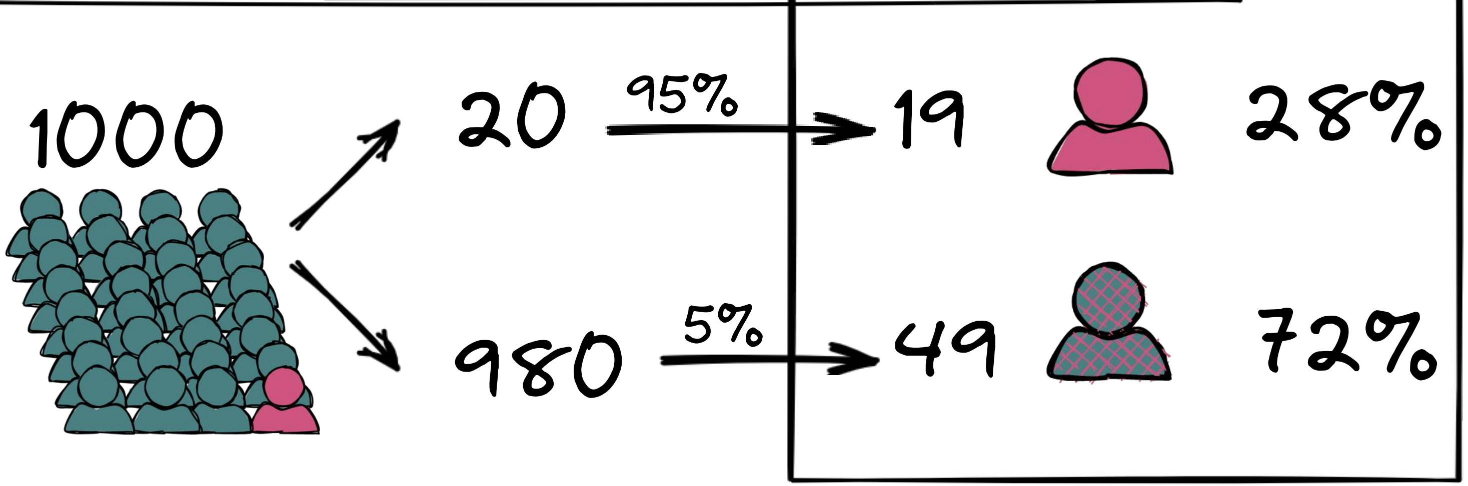
$$\text{NPV} = \frac{20}{20+49} = 0.29$$

$$\text{F1} = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

$= 0.97$

Classification metrics: Case2

Disease Test2: 95% correct if ill
95% correct if not ill



72% of people in the hospital aren't ill

Confusion matrix: Case2

Disease Test2:	95% correct if ill
	95% correct if not ill

The mentioned TN, TP, FN, FP are easily calculated using <u>Bayes' Theorem</u>:

TP:

P (ill if test says ill) = P (B) * P (A | B) = 0.98*0.95 = 0.931 = 93.1%

FN:

P (ill if test says not ill) = 0.98*0.05 = 0.049 = 4.9%

TN:

P (not ill if test says not ill) = 0.02*0.95 = 0.019 = 1.9%

FP:

P (not ill if test says ill) = 0.02*0.05 = 0.001 = 0.1%

Confusion matrix: Case2

predicted class \ actual class		
	True Positive (TP) act: pred: 931	False Positive (FP) act: pred: 1
	False Negative (FN) act: pred: 49	True Negative (TN) act: pred: 19

$$\text{Precision} = \frac{931}{931+1} = 0.99$$

$$\text{Sensitivity (Recall)} = \frac{931}{931+49} = 0.95$$

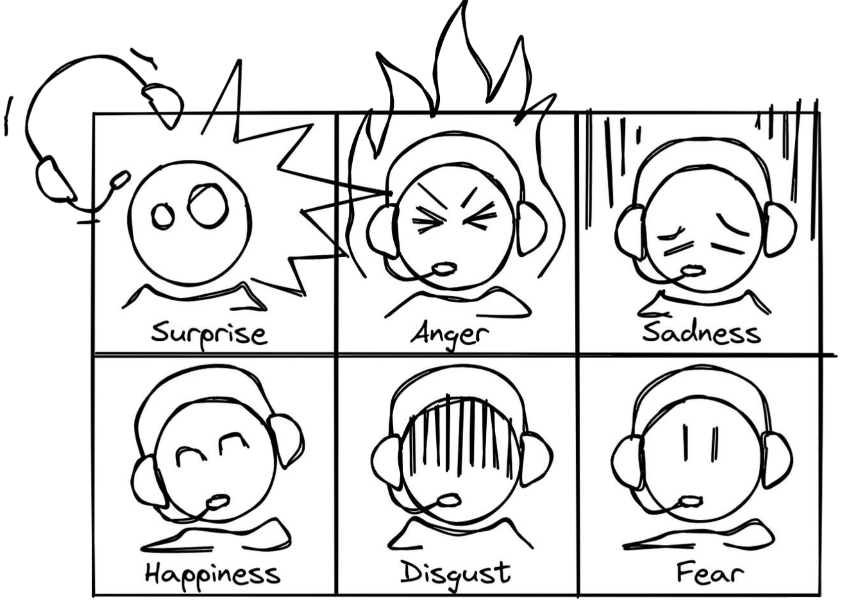
$$\text{Specificity} = \frac{19}{19+1} = 0.95$$

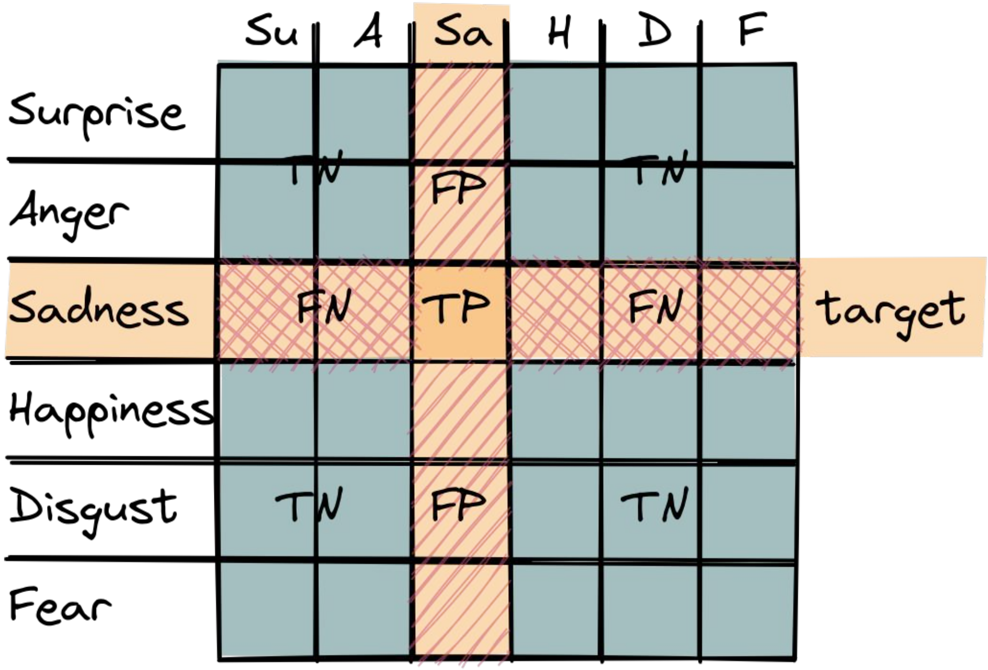
$$\text{NPV} = \frac{19}{19+49} = 0.28$$

$$\text{F1} = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = 0.97$$

Confusion matrix: Case3

Imagine you're building a system to detect emotions via audio.
Among six classes of emotions, you need to detect <u>sadness</u>. The confusion matrix won't change much:





TN – True Negative, TP – True Positive
FN – False Negative, FP – False Positive

Confusion matrix: errors

H_0 – patient doesn't have COVID-19.

Type I error (first kind error; *false positive*) – your prediction/observation is *positive*, and it's *false*. Type I error involves rejecting a true null hypothesis – the patient has no COVID-19, but a PCR test falsely returned "positive." Another example is a fire alarm going on, but there is no fire ("false alarm").

Type II error (second kind error; *false negative*) – your prediction/observation is *negative*, and it's *false*. Type II error deals with failing to reject a false null hypothesis – the patient has COVID-19, but the PCR test fails to detect it. Another example: fire alarm, silent, whereas fire, is all around the building.

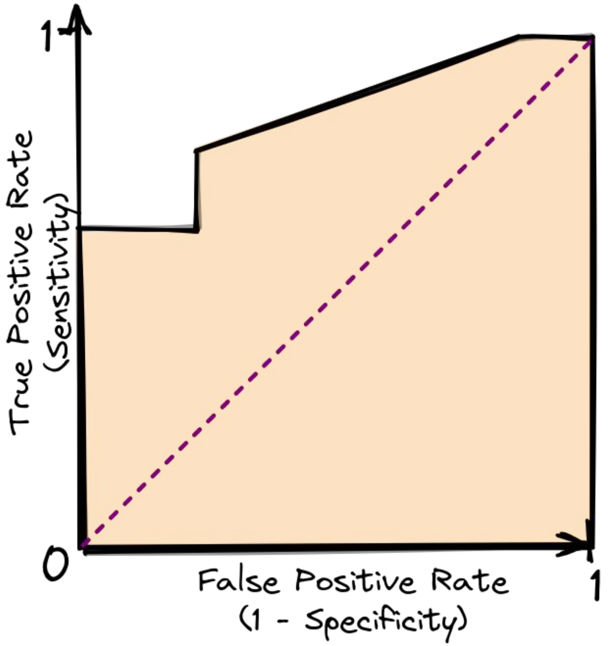
Usually, type II errors are far worse than type I errors, as the consequences of type II errors – not detecting the disease or fire is more harmful than setting up a false alarm.

Confusion matrix: metrics

Of course, besides the discussed metrics we can use basic accuracy:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

or ROC-AUC: The Area Under the Curve (AUC) of the Receiver Operator Characteristic (ROC) probability curve. The ROC curve displays represent all confusion matrixes for a given model. The closer the AUC to 1 – the better the model.

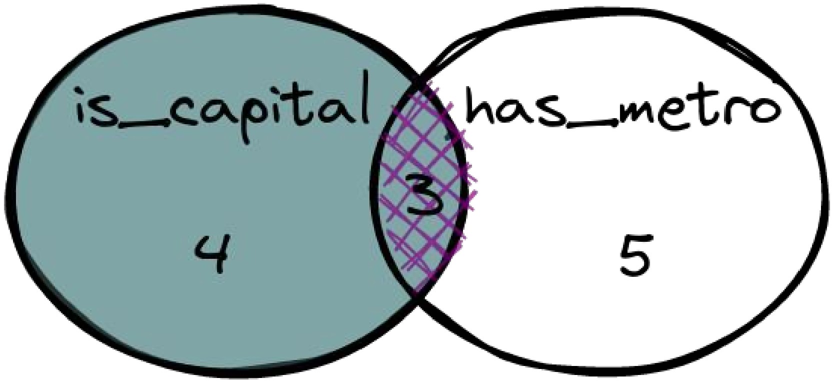


We'll cover ROC-AUC deeper in one of the following lectures.

Naive Bayes

p(has_metro | is_capital) =
=3/4 = 0.75

4 samples

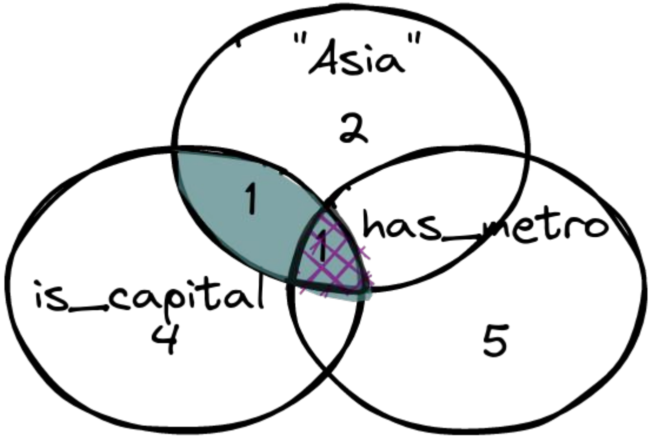


Let's consider we have a small dataset of several world cities with attributes as **is_capital**, **continent**, and **population**. The target column is has_metro – whether the city has a metro.
Using only one feature – **is_capital**, one may conclude that the probability of having a metro is ¾, based on the 4 matching values.

id	is_capital	continent	population	has_metro
1	True	Europe	2.000.000	True
2	True	Europe	1.500.000	False
3	False	Europe	1.000.000	False
4	True	Asia	5.000.000	True
5	False	Asia	7.000.000	True
6	True	America	4.000.000	True
7	False	America	3.000.000	True
X	True	Asia	2.500.000	?

Naive Bayes

$$p(\text{has\_metro} \mid (\text{is\_capital} \,\&\, \&\, \text{continent} == \text{"Asia"})) = \underbrace{1/1}_{\text{1 sample}} = 1$$



If we take two features: **is_capital** and **continent**, there will only be a single city that matches our target row.

If we use all three features, there will be no matches.

id	is_capital	continent	population	has_metro
1	True	Europe	2.000.000	True
2	True	Europe	1.500.000	False
3	False	Europe	1.000.000	False
4	True	Asia	5.000.000	True
5	False	Asia	7.000.000	True
6	True	America	4.000.000	True
7	False	America	3.000.000	True
X	True	Asia	2.500.000	?

Naive Bayes

$$\boxed{P(y \mid x) = \frac{P(y) * P(x \mid y)}{P(x)}}$$

Alternatively, we can use Bayes Theorem to calculate $P(y \mid x)$.

$P(y)$ is quite easy to estimate. For example, if Y takes on discrete binary values estimating $P(y)$ can be reduced to coin tossing. $P(x)$ doesn't depend on y, so we don't really care about it.

Estimating $P(x \mid y)$ is the real challenge. The trick is to make an assumption, that **all features are independent**. In this case, $P(x \mid y)$ can be estimated as the product of multiplication all $P(x_a \mid y)$, where x_a is the value for feature a.

As as result Naive Bayes predicts the probabilities of each group. The group with the highest probability is selected as the most likely group.

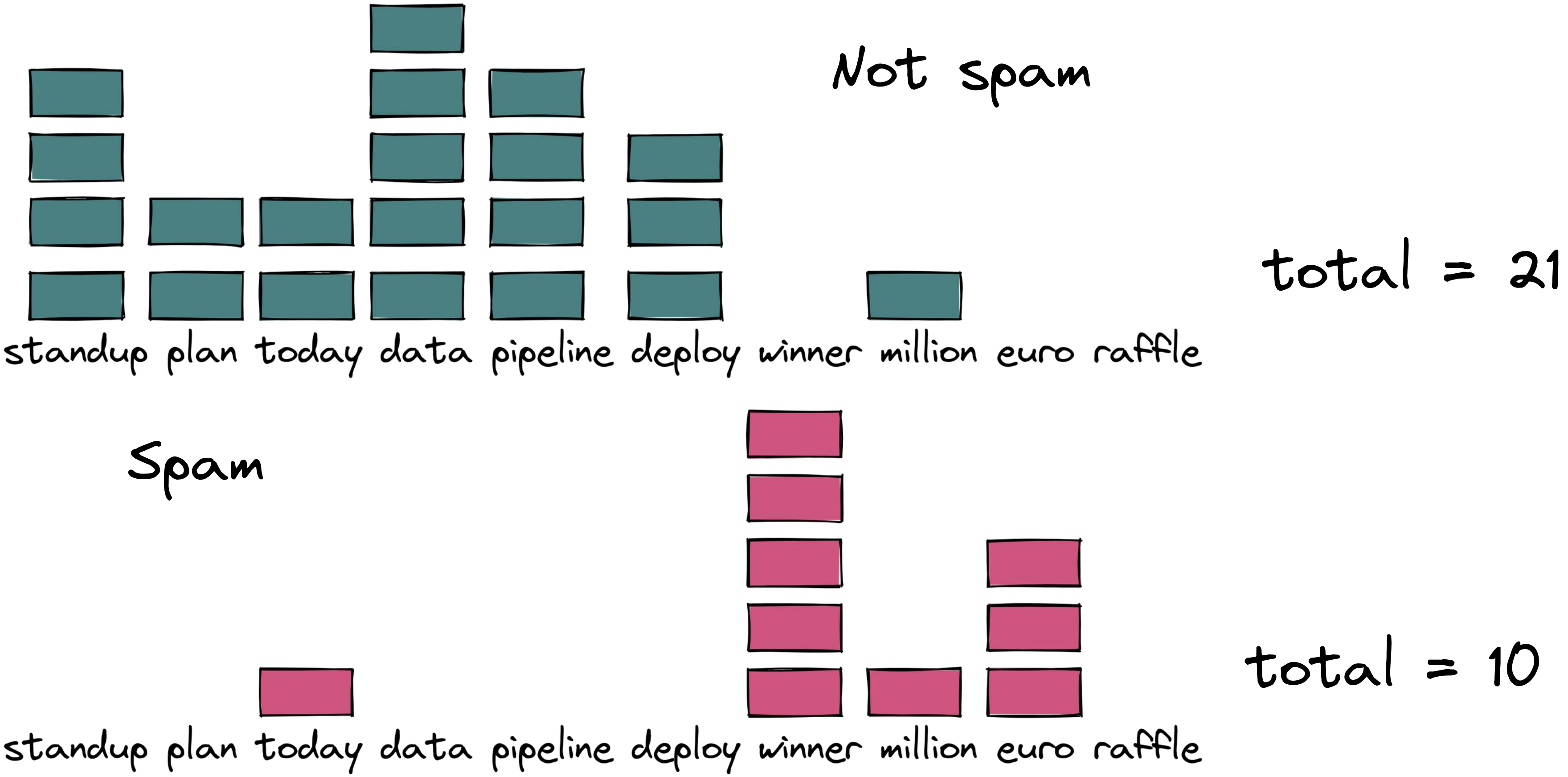
Naive Bayes: classic spam mail

Hi all,
The Daily Standup planned for today is canceled, due to vacation in UAE.
Regards,

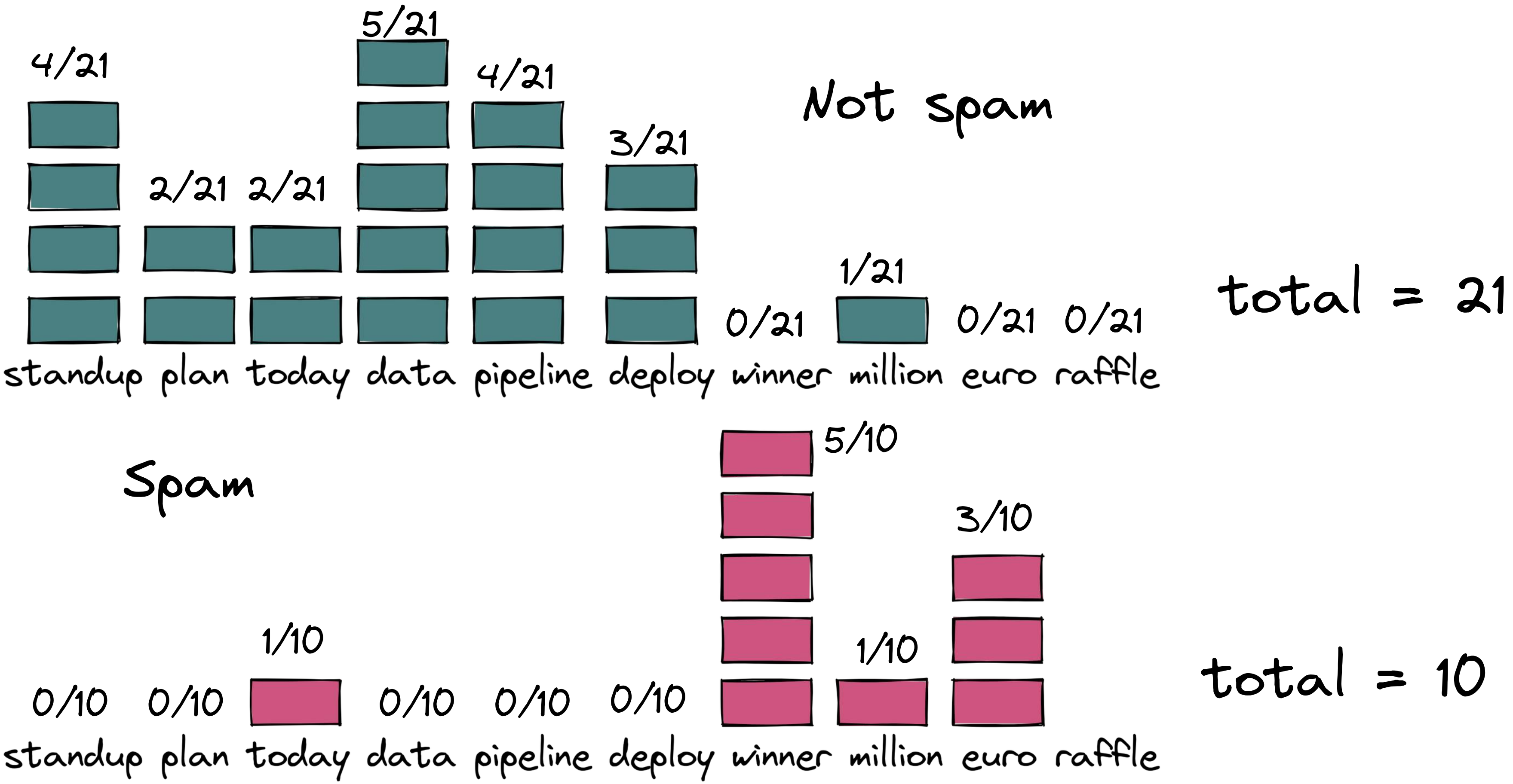
Ivan, hi.
Can we discuss the latest changes in the data pipeline we plan to deploy on prod next week over lunch?

Good Day,
Your Email rolled among many others was confirmed as the winner of 2-Million-Euro Raffle Held in 2022 in Europe.
<bla-bla-bla>.
Congratulations once more.
Collect your money today!

Naive Bayes: draw frequencies



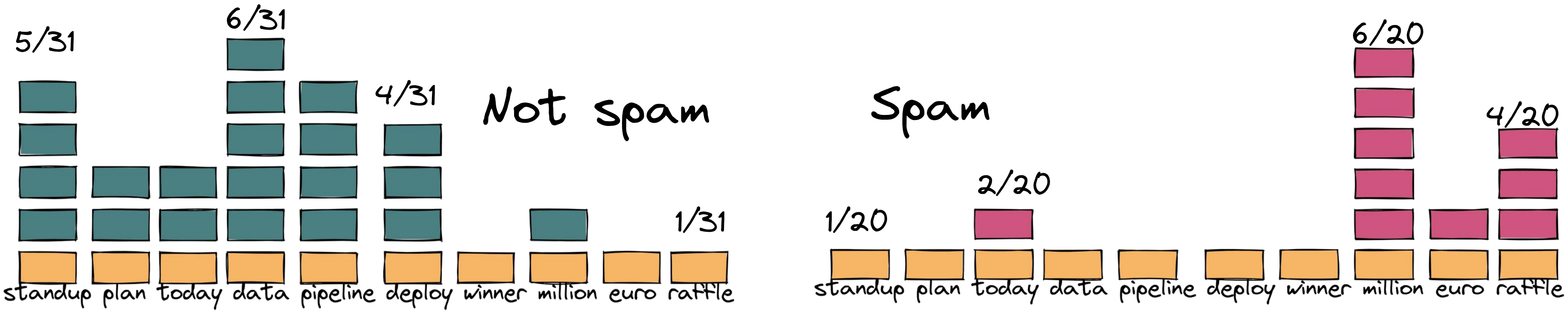
Naive Bayes: get probabilities



Naive Bayes: tweeks

1. Prior probability ($P(y)$) of letter being spam is 10/(10+21) = 10/31
not being spam = 21/(10+21) = 21/31

2. As probabilities ($P(x_a \mid y)$) will be multiplied, it makes sense to add an additional block for every word to avoid multiplication by zero



Naive Bayes: Case1a

	spam	not spam
standup	1/20	5/31
plan	1/20	3/31
today	2/20	3/31
data	1/20	6/31
pipeline	1/20	5/31
deploy	1/20	4/31
winner	1/20	1/31
million	6/20	2/31
euro	2/20	1/31
raffle	4/20	1/31

Text: We need to finish the update for
data pipeline today

Not Spam:
Prior probability (PP): 21/31
Score: PP × p(data) × p(pipeline) × p (today) =
= 21/31 × 6/31 × 5/31 × 3/31 = **2.047** × 10^{-3}

Spam:
Prior probability (PP): 10/31
Score: PP × p(data) × p(pipeline) × p (today) =
= 10/31 × 1/20 × 1/20 × 2/20 = **0.0807** × 10^{-3}

Naive Bayes: Case1b

	spam	not spam
standup	1/20	5/31
plan	1/20	3/31
today	2/20	3/31
data	1/20	6/31
pipeline	1/20	5/31
deploy	1/20	4/31
winner	1/20	1/31
million	6/20	2/31
euro	2/20	1/31
raffle	4/20	1/31

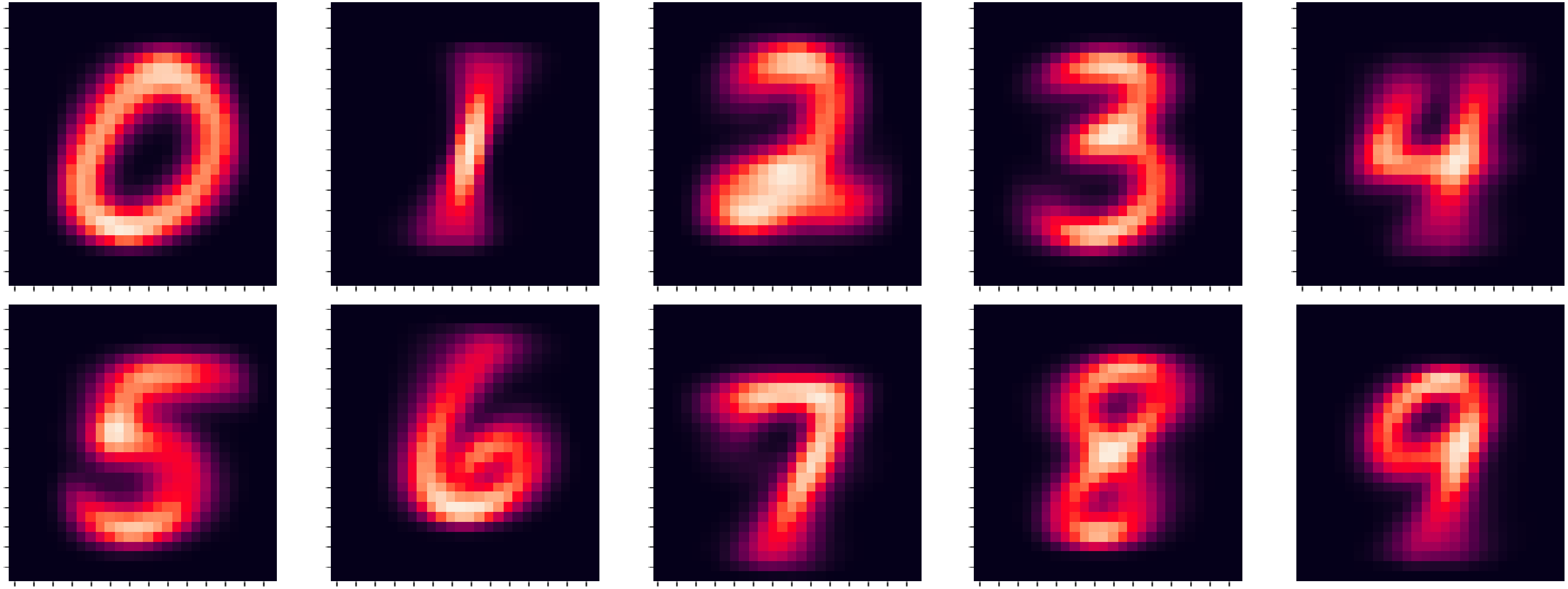
Text: You've just won the McDuck lottery! Collect your 6 million dollars prize now!

Not Spam:
Prior probability (PP): 21/31
Score: PP × p(winner) × p(million) =
= 21/31 × 1/31 × 2/31 = **1.41** × 10^{-3}

Spam:
Prior probability (PP): 10/31
Score: PP × p(winner) × p(million) =
= 10/31 × 1/20 × 6/20 = **4.83** × 10^{-3}

Case2: mini-MNIST heatmaps

MNIST is a dataset of hand-written digits. In our case, we'll use only 1250 matrixes for each label. Heatmaps are shown below:



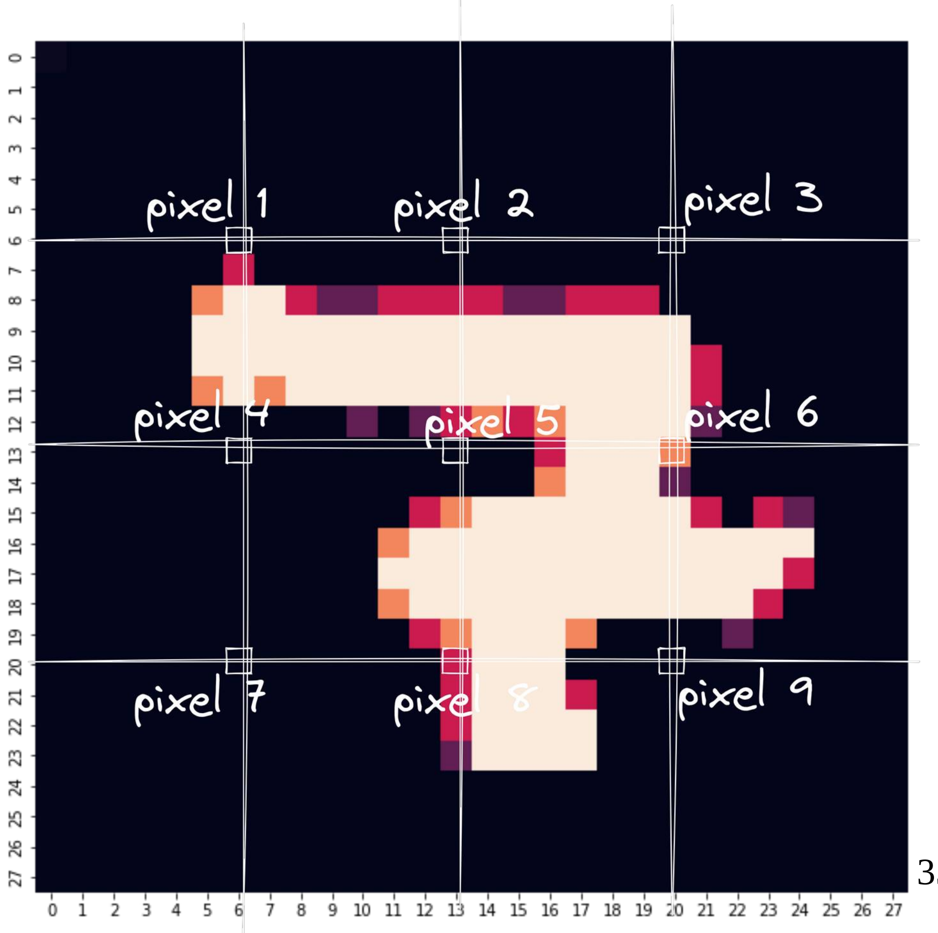
mini-MNIST heatmap

Input: 784 pixels (28x28) with values from 0 to 255 – shades of black
1250 images of each number
Output: Labels 0-9

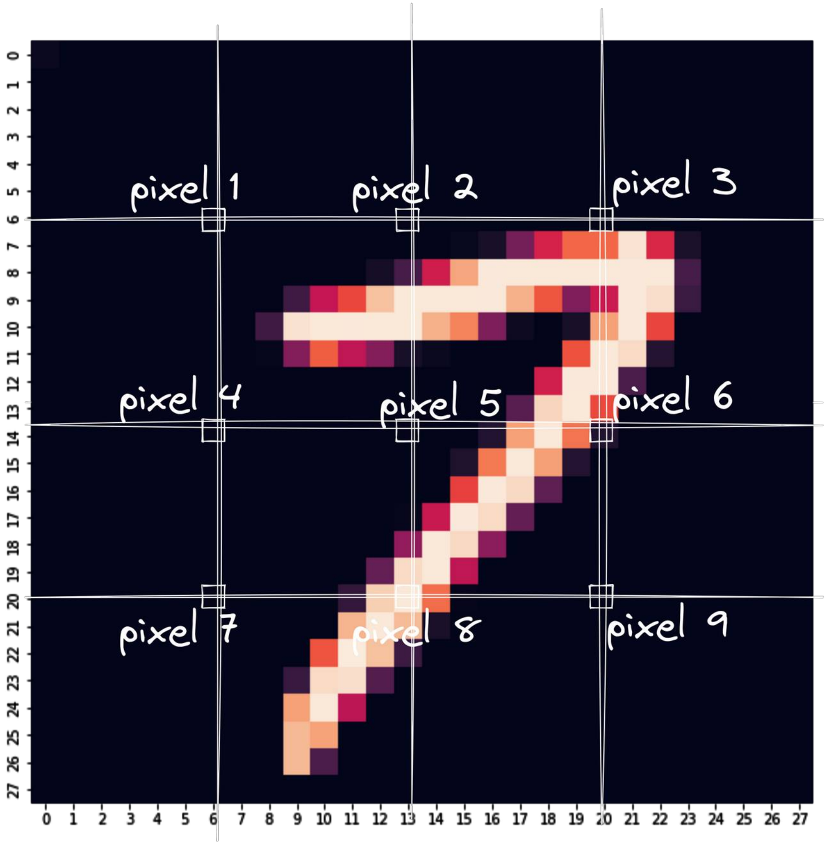
For illustrative purposes, we'll use only 9 pixels from the grid on the right:

For each of 9 pixels we'll calculate the probability of each label. If the value is greater than zero, we'll add that score, else, we substract it.

We'll classify the number based on the highest score.



Naive Bayes intuition: Case2a



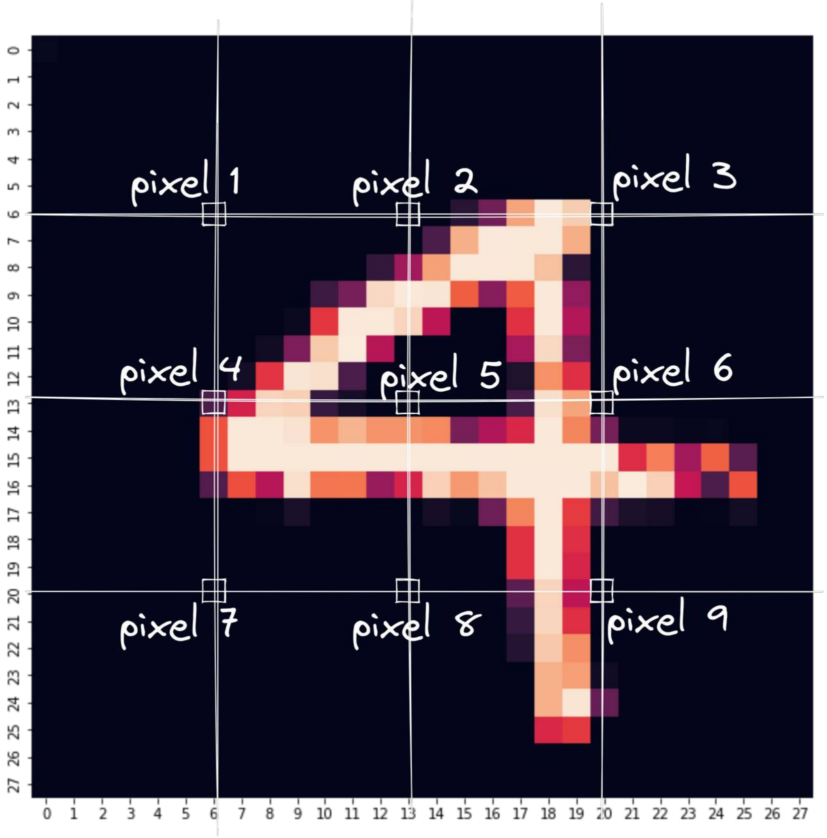
Label probability if pixel not zero

label	pixel 1	pixel 2	pixel 3	pixel 4	pixel 5	pixel 6	pixel 7	pixel 8	pixel 9
0	0.031076	0.139281	0.173121	0.316681	0.020525	0.243015	0.307356	0.105269	0.187705
1	0.003389	0.060456	0.061956	0.000789	0.169887	0.000222	0.007525	0.125108	0.003412
2	0.268259	0.144354	0.116400	0.014922	0.058575	0.098633	0.272539	0.131285	0.235566
3	0.417618	0.169296	0.094322	0.014095	0.202727	0.057936	0.155603	0.053954	0.170488
4	0.086999	0.031018	0.113692	0.211661	0.048830	0.110856	0.007707	0.068244	0.040989
5	0.049409	0.107068	0.186167	0.070696	0.144866	0.039419	0.150952	0.074231	0.124342
6	0.034938	0.101496	0.026729	0.075727	0.078471	0.189557	0.027943	0.187085	0.116258
7	0.067279	0.030190	0.019830	0.096564	0.008866	0.109324	0.000852	0.116264	0.009083
8	0.039923	0.150040	0.170224	0.028812	0.187809	0.065174	0.067424	0.066099	0.080942
9	0.001111	0.066802	0.037559	0.170053	0.079444	0.085864	0.002099	0.072462	0.031215

mask:

pixel 1	pixel 2	pixel 3	pixel 4	pixel 5	pixel 6	pixel 7	pixel 8	pixel 9
-1	-1	-1	-1	-1	-1	-1	1	-1

Naive Bayes intuition: Case2b



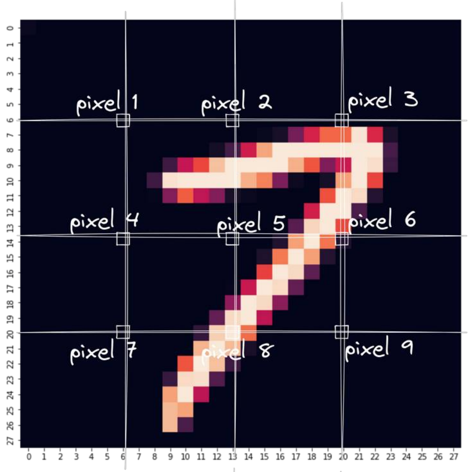
Label probability if pixel not zero

label	pixel 1	pixel 2	pixel 3	pixel 4	pixel 5	pixel 6	pixel 7	pixel 8	pixel 9
0	0.031076	0.139281	0.173121	0.316681	0.020525	0.243015	0.307356	0.105269	0.187705
1	0.003389	0.060456	0.061956	0.000789	0.169887	0.000222	0.007525	0.125108	0.003412
2	0.268259	0.144354	0.116400	0.014922	0.058575	0.098633	0.272539	0.131285	0.235566
3	0.417618	0.169296	0.094322	0.014095	0.202727	0.057936	0.155603	0.053954	0.170488
4	0.086999	0.031018	0.113692	0.211661	0.048830	0.110856	0.007707	0.068244	0.040989
5	0.049409	0.107068	0.186167	0.070696	0.144866	0.039419	0.150952	0.074231	0.124342
6	0.034938	0.101496	0.026729	0.075727	0.078471	0.189557	0.027943	0.187085	0.116258
7	0.067279	0.030190	0.019830	0.096564	0.008866	0.109324	0.000852	0.116264	0.009083
8	0.039923	0.150040	0.170224	0.028812	0.187809	0.065174	0.067424	0.066099	0.080942
9	0.001111	0.066802	0.037559	0.170053	0.079444	0.085864	0.002099	0.072462	0.031215

mask:

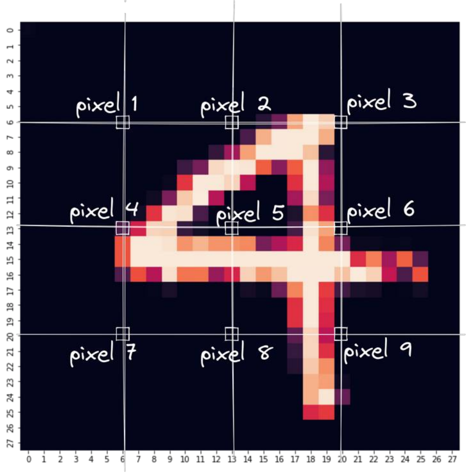
pixel 1	pixel 2	pixel 3	pixel 4	pixel 5	pixel 6	pixel 7	pixel 8	pixel 9
-1	-1	-1	1	-1	-1	-1	-1	-1

Naive Bayes intuition



label	score
0	0.015795
1	0.171025
2	0.048123
3	0.027511
4	0.115989
5	0.086455
6	0.132388
7	0.165097
8	0.096672
9	0.140945

The idea is to calculate the probability for membership of all data points to each class. As a result, we achieve 1 for case2a (actual result 7) and 9 for case2b (actual result 4) as these classes have the highest scores.



label	score
0	0.073830
1	0.136899
2	0.016180
3	0.016570
4	0.155359
5	0.085484
6	0.101819
7	0.159689
8	0.086436
9	0.167735

Naive Bayes is one of the fastest and simple classification algorithms and is usually used as a baseline for various classification problems.