Seminar 3: Bayes Formula

Carlos Buitrago & Anastasiya Chikalova

Table of contents

1 Conditional Probability

Let $A, B \in \mathcal{F}$ be events such that $P(B) > 0$. The **conditional probability** of A given B is defined by

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

Intuitively, after learning that the event B has occurred, the sample space is reduced from Ω to B, and probabilities are rescaled accordingly.

Rearranging the definition yields

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

which is often called the **multiplication rule**.

More generally, for events $A_1, \dots, A_n$ satisfying

$$P(A_1 \cap \cdots \cap A_k) > 0, \qquad k = 1, \dots, n-1,$$

we have

$$P(A_1 \cap \cdots \cap A_n) = P(A_1)P(A_2 \mid A_1)P(A_3 \mid A_1 \cap A_2) \cdots P(A_n \mid A_1 \cap \cdots \cap A_{n-1}).$$

2 Partitions of the Sample Space

A collection of events

$$\{B_n\}_{n=1}^{N} \quad \text{or} \quad \{B_n\}_{n=1}^{\infty}$$

is called a **partition** of the sample space Ω if

$$B_i \cap B_j = \emptyset, \qquad i \neq j,$$

and

$$\bigcup_n B_n = \Omega.$$

3 Law of Total Probability

Let $\{B_n\}$ be a partition of Ω such that $P(B_n) > 0$ for all n.

Then for every event $A \in \mathcal{F}$,

$$P(A) = \sum_n P(A \mid B_n) P(B_n).$$

For a finite partition $\{B_1, \ldots, B_m\}$, this becomes

$$P(A) = P(A \mid B_1) P(B_1) + \cdots + P(A \mid B_m) P(B_m).$$

3.1 Example: Exam Questions

Maria is worried about the exam in Probability and would like to maximize her chances of receiving an easy question.

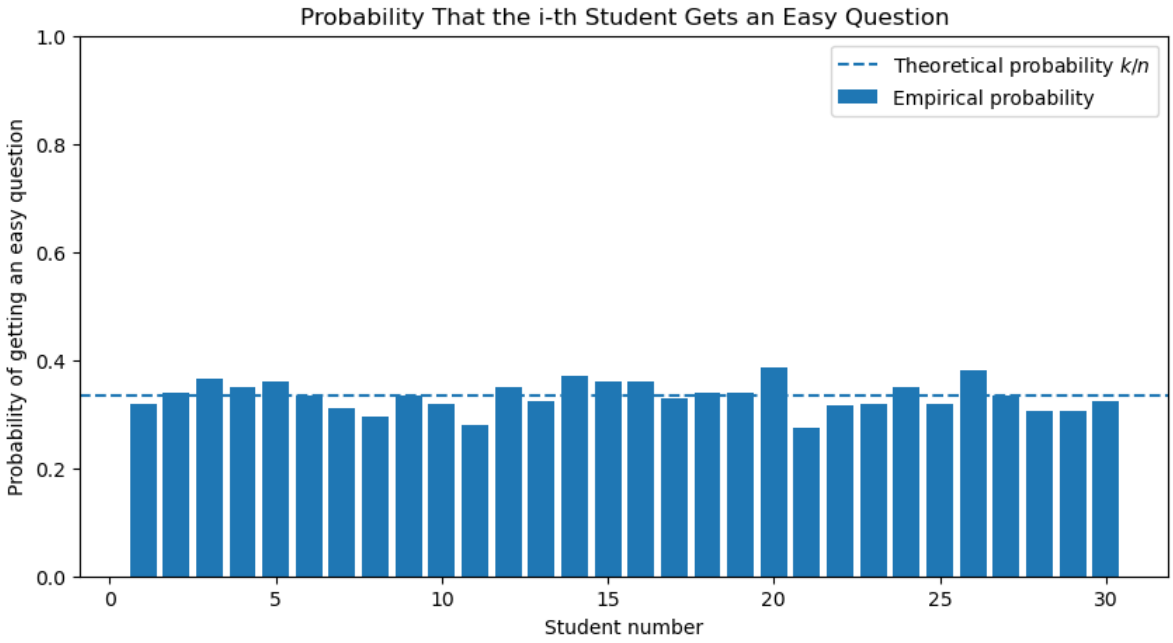
There are 30 students waiting in a line to enter the classroom. On the teacher's desk there are 30 tickets, one question written on each ticket. Among these tickets, 10 contain easy questions and 20 contain difficult questions.

The students enter the classroom one by one. Each student randomly chooses one ticket, takes it with them, and sits down. Thus, after a student chooses a ticket, that ticket is no longer available for the next students.

Maria can choose any position in the line.

Which position should Maria choose? Should she be the first student in the line? The last one? Or perhaps somewhere in the middle?

```
Theoretical probability: 0.3333333333333333
Empirical probability for first student: 0.32
Empirical probability for last student: 0.325
```



3.2 Example: The Drunk Passenger Problem

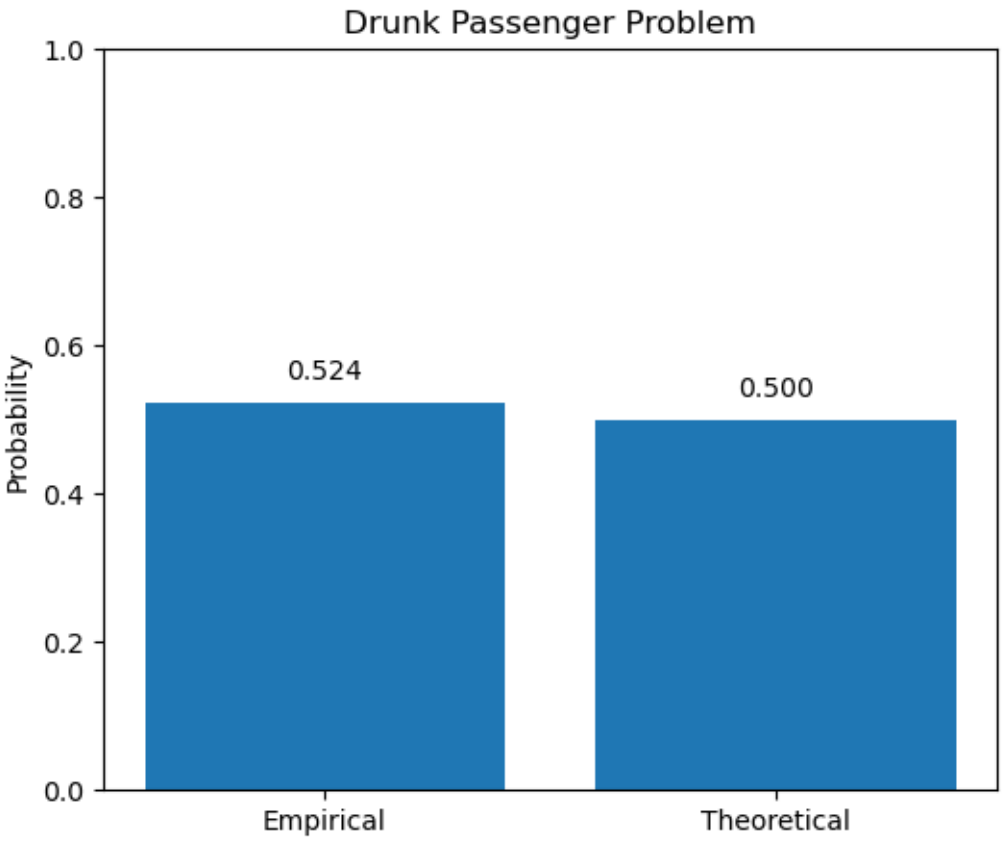
On January 1st, a flight from Moscow to Saint Petersburg with exactly 100 passengers and exactly 100 seats is preparing for boarding. Each passenger has an assigned seat, and all passengers arrive on time.

The first passenger to enter the plane has been celebrating New Year's Eve all night and is still somewhat drunk. As a result, instead of sitting in his assigned seat, he chooses one of the 100 seats uniformly at random.

After that, the remaining passengers board one by one. Each passenger sits in their assigned seat if it is free. If their assigned seat is already occupied, they decide to avoid any arguments and simply choose one of the remaining free seats uniformly at random.

Question: What is the probability that the last passenger to board the plane ends up sitting in their assigned seat?

```
Empirical probability: 0.524
Theoretical probability: 0.5
```



4 Bayes' Formula

Suppose $\{B_n\}$ is a partition of Ω satisfying $P(B_n) > 0$ for all n, and let $A \in \mathcal{F}$ be an event with $P(A) > 0$.

Then for every n,

$$P(B_n \mid A) = \frac{P(A \mid B_n)P(B_n)}{\sum_k P(A \mid B_k)P(B_k)}.$$

This identity is known as **Bayes' formula**.

- $P(B_n)$ is called the **prior probability**,
- $P(A \mid B_n)$ is called the **likelihood**,
- $P(B_n \mid A)$ is called the **posterior probability**.

Bayes' formula describes how probabilities should be updated when new information becomes available.

4.1 Example: Medical Testing

A patient goes to see a doctor. The doctor performs a medical test with **99% reliability**. This means that:

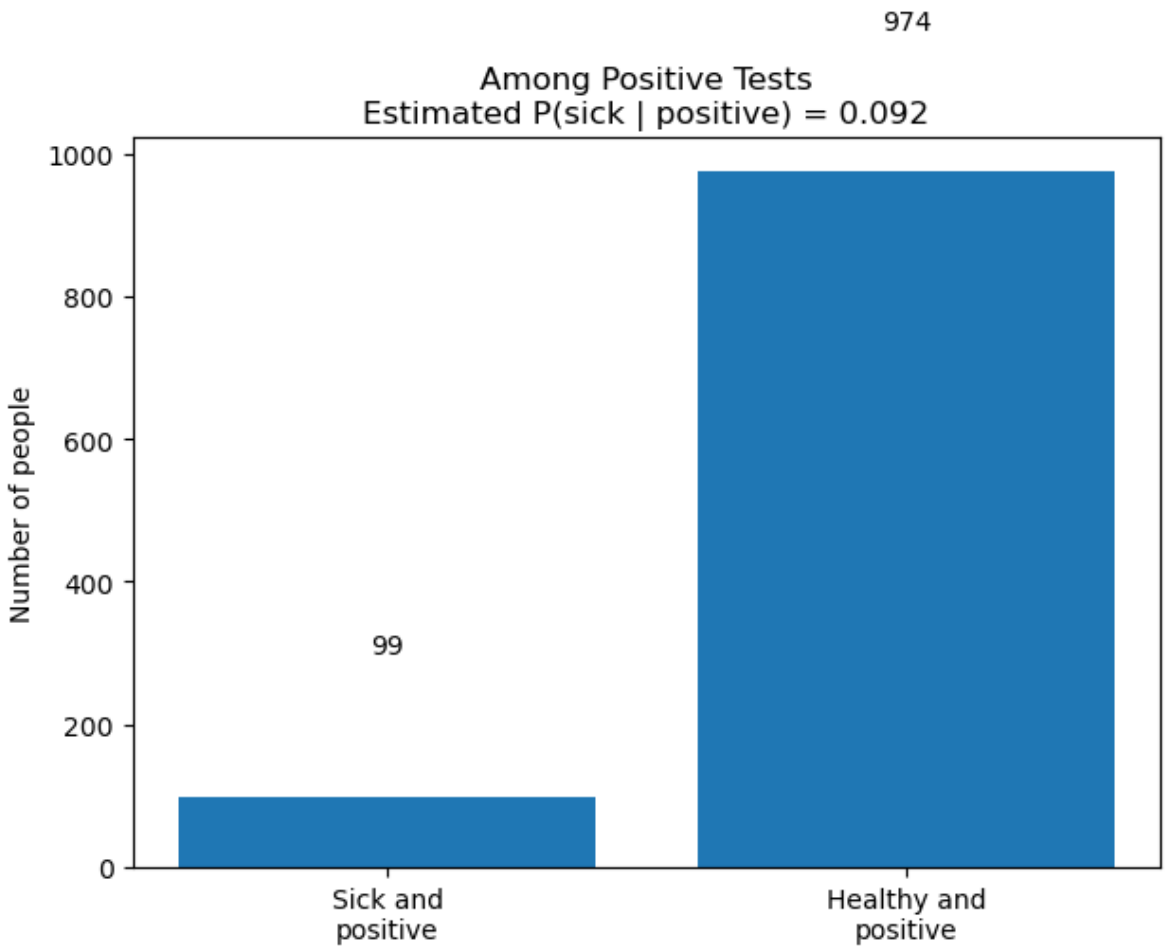
- 99% of sick people test positive;
- 99% of healthy people test negative.

The doctor also knows that only 0.1% of the population is affected by the disease.

A patient takes the test and receives a **positive result**.

Question: What is the probability that the patient is actually sick?

```
Theoretical P(positive): 0.010980000000000009
Theoretical P(sick | positive): 0.09016393442622944

Empirical P(positive): 0.01073
Empirical P(sick | positive): 0.09226467847157502
```



4.2 Example: The Monty Hall Problem

Suppose you're on a game show, and you're given the choice of three doors. Behind one door is a car; behind the others, goats.

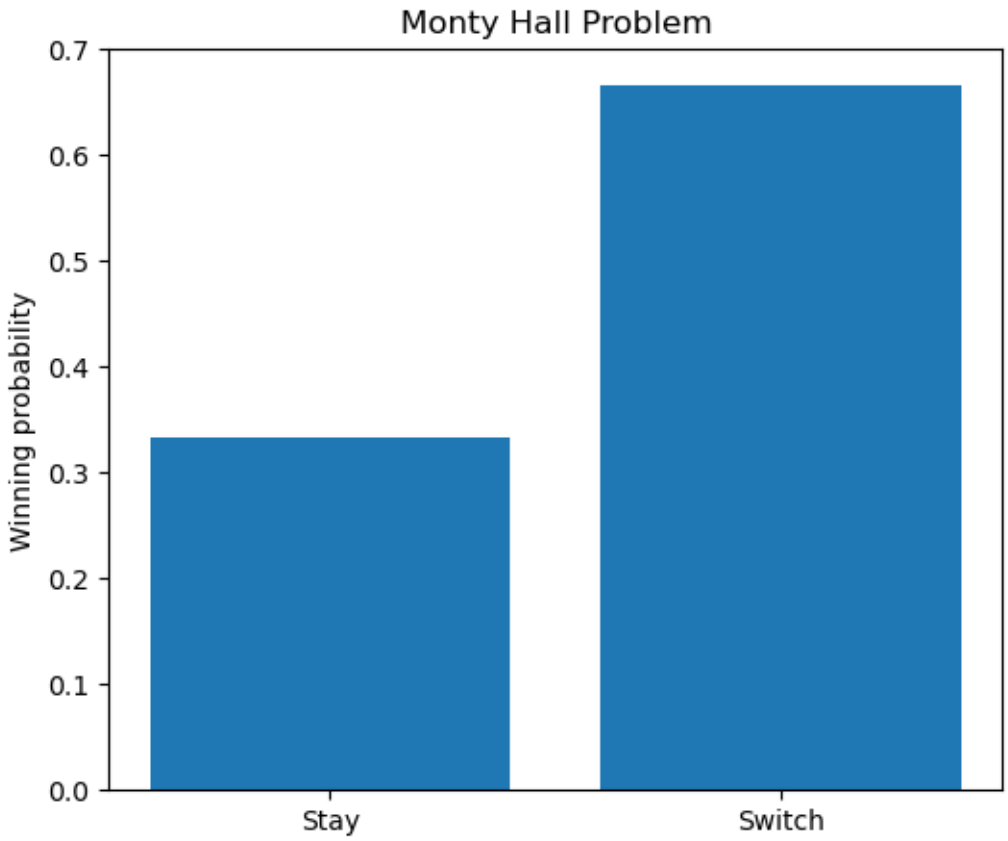
You pick a door, say Door 1. The host, who knows what's behind the doors, opens another door, say Door 3, which contains a goat.

The host then asks:

> "Do you want to switch your choice and pick Door 2 instead?"

Question: Is it to your advantage to switch your choice?

```
Stay: 0.33312
Switch: 0.66688
```



5 Conditional Probability in Large Language Models

Large language models such as ChatGPT are built around conditional probabilities. When generating text, the model repeatedly estimates probabilities of the form

$$P(\text{next word} \mid \text{previous words}).$$

For example, after reading

"The capital of France is"

the model assigns a very high probability to the word

"Paris"

and a very small probability to words such as

"banana".

At every step, the model chooses the next word based on these conditional probabilities.

Although modern AI systems use extremely sophisticated neural networks, conditional probability remains one of the fundamental mathematical ideas behind them.

In this sense, the formulas studied in this lecture are part of the mathematical foundation of modern artificial intelligence.

6 Naive Bayes Classifier

One of the simplest machine learning algorithms is the **Naive Bayes classifier**.

Suppose that an object belongs to one of several classes

$$C_1, C_2, \dots, C_k.$$

After observing some features

$$X = (X_1, \dots, X_n),$$

Bayes' formula gives

$$P(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{P(X)}.$$

The classifier predicts the class with the largest posterior probability.

The difficulty is that computing

$$P(X \mid C_i)$$

can be very complicated.

Naive Bayes assumes that the features are conditionally independent given the class:

$$P(X_1, \dots, X_n \mid C_i) = \prod_{j=1}^{n} P(X_j \mid C_i).$$

Although this assumption is usually false, the resulting classifier often performs surprisingly well.

6.1 Example: Classifying Iris Flowers

Suppose that we are given a flower from one of the following three species:

- **Iris Setosa**
- **Iris Versicolor**
- **Iris Virginica**

For each flower, we measure the following characteristics:

1. Sepal length (cm)
2. Sepal width (cm)
3. Petal length (cm)
4. Petal width (cm)

The goal is to predict the species of the flower using these measurements.

For example, suppose that a flower has the following characteristics:

Feature	Value
Sepal length	5.8 cm
Sepal width	2.7 cm
Petal length	4.1 cm
Petal width	1.0 cm

Question: Given these measurements, what is the probability that the flower belongs to each of the three species?

More generally, if

$$X = (X_1, X_2, X_3, X_4)$$

denotes the vector of observed measurements, we would like to compute

$$P(\text{Setosa} \mid X), \qquad P(\text{Versicolor} \mid X), \qquad P(\text{Virginica} \mid X).$$

Using Bayes' formula,

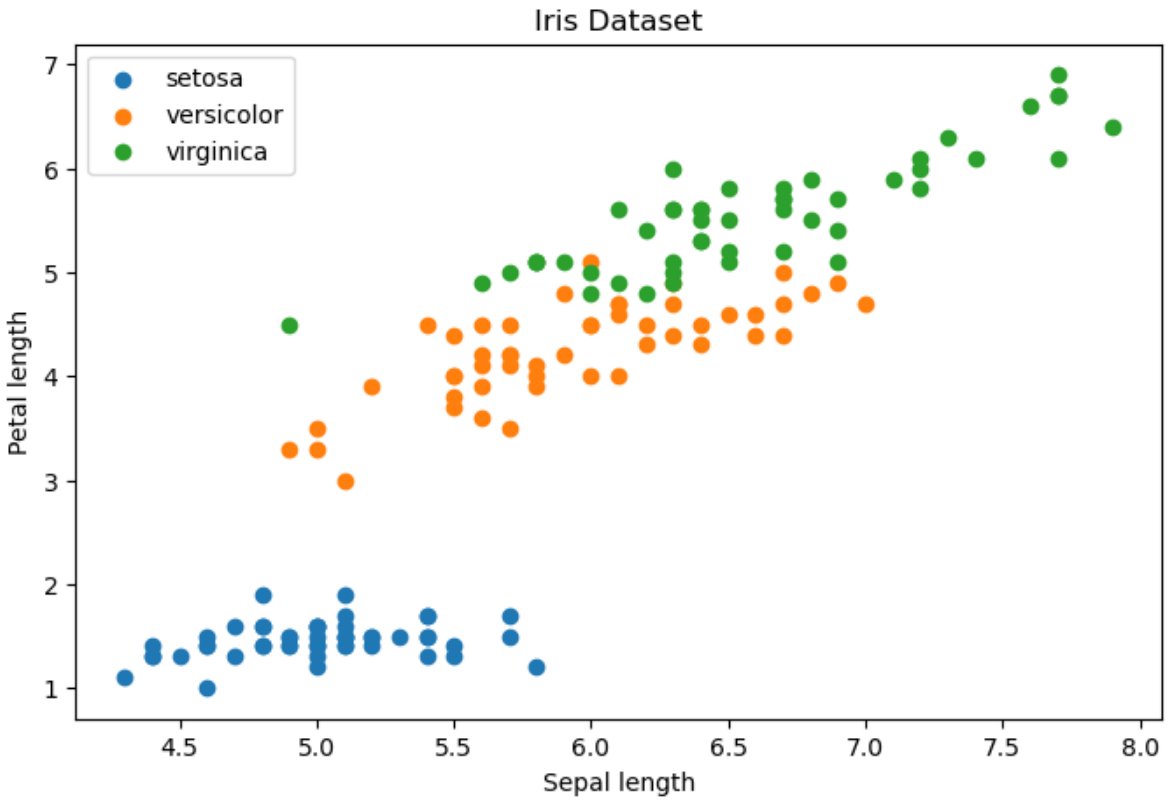
$$P(\text{Species} \mid X) = \frac{P(X \mid \text{Species})P(\text{Species})}{P(X)}.$$

The flower is then classified as belonging to the species with the largest posterior probability.

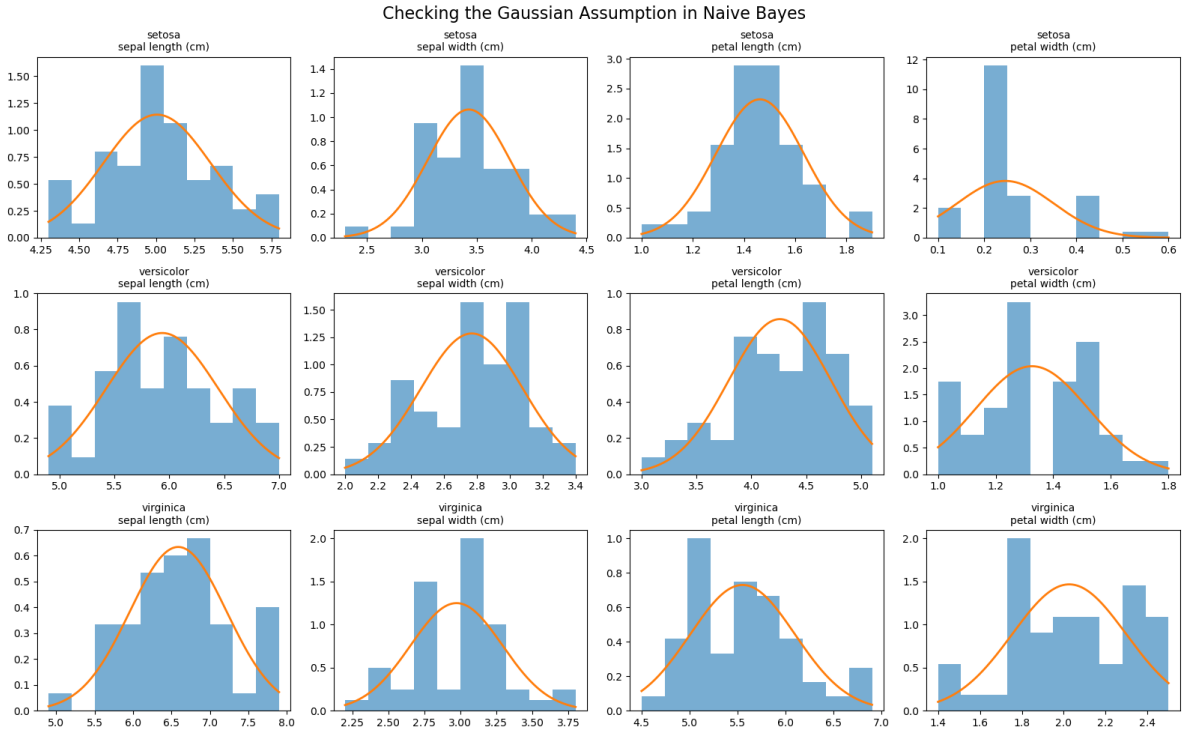
```
.. _iris_dataset:

Iris plants dataset
--------------------

**Data Set Characteristics:**

    :Number of Instances: 150 (50 in each of three classes)
    :Number of Attributes: 4 numeric, predictive attributes and the class
    :Attribute Information:
        - sepal length in cm
        - sepal width in cm
        - petal length in cm
        - petal width in cm
        - class:
                - Iris-Setosa
                - Iris-Versicolour
                - Iris-Virginica

    :Summary Statistics:

    ============== ==== ==== ======= ===== ====================
                    Min  Max   Mean    SD   Class Correlation
    ============== ==== ==== ======= ===== ====================
    sepal length:   4.3  7.9   5.84   0.83    0.7826
    sepal width:    2.0  4.4   3.05   0.43   -0.4194
    petal length:   1.0  6.9   3.76   1.76    0.9490  (high!)
    petal width:    0.1  2.5   1.20   0.76    0.9565  (high!)
    ============== ==== ==== ======= ===== ====================

    :Missing Attribute Values: None
    :Class Distribution: 33.3% for each of 3 classes.
    :Creator: R.A. Fisher
    :Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
    :Date: July, 1988

The famous Iris database, first used by Sir R.A. Fisher. The dataset is taken
from Fisher's paper. Note that it's the same as in R, but not as in the UCI
Machine Learning Repository, which has two wrong data points.

This is perhaps the best known database to be found in the
pattern recognition literature.  Fisher's paper is a classic in the field and
is referenced frequently to this day.  (See Duda & Hart, for example.)  The
```

```
data set contains 3 classes of 50 instances each, where each class refers to a
type of iris plant.  One class is linearly separable from the other 2; the
latter are NOT linearly separable from each other.

.. topic:: References

   - Fisher, R.A. "The use of multiple measurements in taxonomic problems"
     Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to
     Mathematical Statistics" (John Wiley, NY, 1950).
   - Duda, R.O., & Hart, P.E. (1973) Pattern Classification and Scene Analysis.
     (Q327.D83) John Wiley & Sons.  ISBN 0-471-22361-1.  See page 218.
   - Dasarathy, B.V. (1980) "Nosing Around the Neighborhood: A New System
     Structure and Classification Rule for Recognition in Partially Exposed
     Environments".  IEEE Transactions on Pattern Analysis and Machine
     Intelligence, Vol. PAMI-2, No. 1, 67-71.
   - Gates, G.W. (1972) "The Reduced Nearest Neighbor Rule".  IEEE Transactions
     on Information Theory, May 1972, 431-433.
   - See also: 1988 MLC Proceedings, 54-64.  Cheeseman et al"s AUTOCLASS II
     conceptual clustering system finds 3 classes in the data.
   - Many, many more ...
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0



Naive Bayes assumes that, within each class, every feature follows a normal distribution. The orange curve is the normal distribution estimated from the data. How reasonable does this assumption look?



We now implement the Naive Bayes classifier ourselves.

For each class c, we estimate the prior probability

$$P(Y = c),$$

and for each feature X_j, we assume

$$X_j \mid Y = c \sim N(\mu_{cj}, \sigma_{cj}^2).$$

Thus,

$$P(X = x \mid Y = c) = \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_{cj}^2}} \exp\left(-\frac{(x_j - \mu_{cj})^2}{2\sigma_{cj}^2}\right).$$

The classifier predicts the class maximizing

$$P(Y = c \mid X = x) \propto P(Y = c)P(X = x \mid Y = c).$$

In practice, we use logarithms to avoid numerical underflow.

In short, the algorithm does this:

1. For each species, estimate the mean and variance of every feature.
2. For a new flower, compute how likely its measurements are under each species.
3. Multiply by the prior probability of each species.
4. Choose the species with the largest posterior score.

This is exactly Bayes' formula combined with the Gaussian assumption and the naive independence assumption.

```
Accuracy = 0.9777777777777777
```

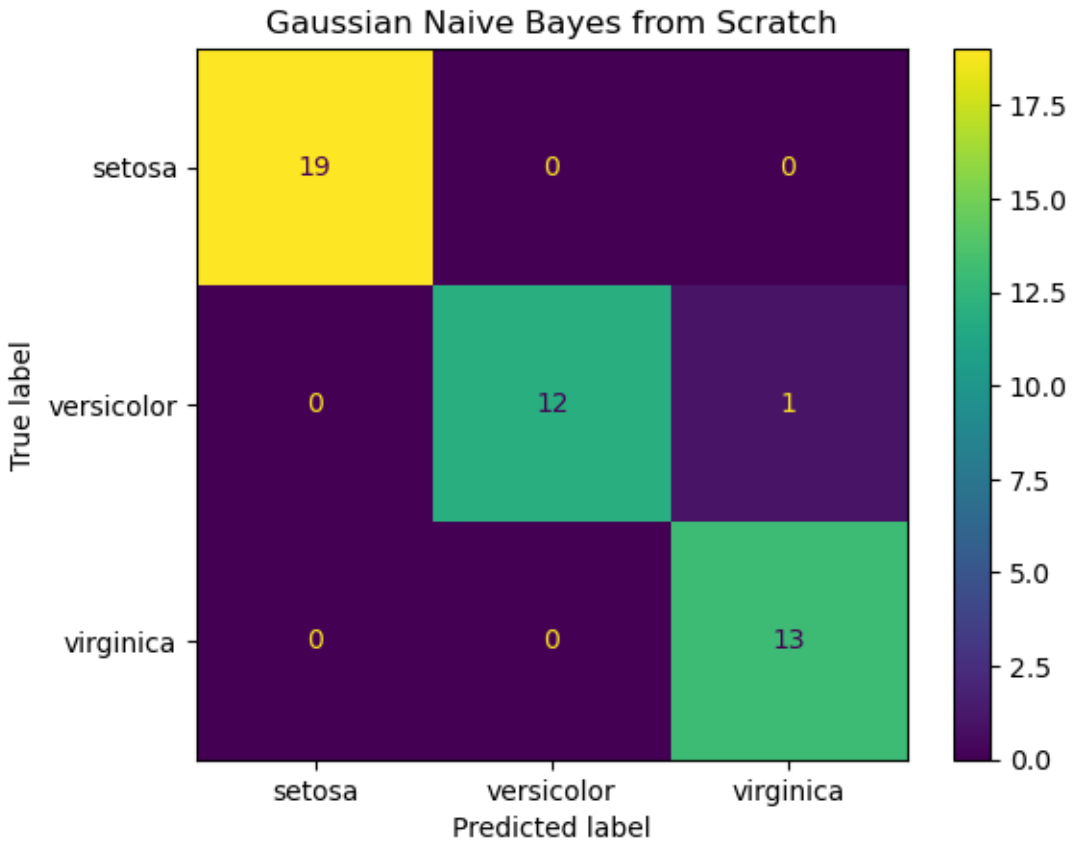
6.1.1 Evaluating the Classifier

To evaluate the performance of our classifier, we compare the true species of each flower in the test set with the species predicted by the model.

A useful tool for this purpose is the **confusion matrix**. The rows correspond to the true species, while the columns correspond to the predicted species.

Ideally, all observations should lie on the main diagonal, indicating correct classifications. Entries outside the diagonal represent classification errors.

The confusion matrix not only tells us how accurate the classifier is, but also reveals which species are most frequently confused with one another.




```
Class: setosa
Prior: 0.29523809523809524
Means: [4.96451613 3.37741935 1.46451613 0.2483871 ]
Variances: [0.1119667  0.13658689 0.03325702 0.01152966]

Class: versicolor
Prior: 0.3523809523809524
Means: [5.86216216 2.72432432 4.21081081 1.3027027 ]
Variances: [0.27532505 0.08724617 0.23934259 0.04134405]

Class: virginica
Prior: 0.3523809523809524
Means: [6.55945946 2.98648649 5.54594595 2.00540541]
Variances: [0.42241052 0.09630387 0.28842951 0.08591673]
```

Since the Iris dataset contains the same number of flowers from each species, the prior probabilities satisfy

$$P(\text{Setosa}) = P(\text{Versicolor}) = P(\text{Virginica}) = \frac{1}{3}.$$

Therefore, in this particular example, the classifier is driven almost entirely by the likelihoods

$$P(X \mid \text{species}).$$

In many real-world applications, however, the classes are highly imbalanced, and the prior probabilities play a crucial role.

7 Machine Learning in Practice

In this lecture, we implemented the Gaussian Naive Bayes classifier from scratch in order to understand the mathematics behind the algorithm. However, in practice, machine learning models are usually already implemented in specialized libraries such as **Scikit-Learn**, **PyTorch**, **TensorFlow**, **XGBoost**, and **LightGBM**.

For example, the entire Naive Bayes classifier can be trained and evaluated with only a few lines of code:

```
model = GaussianNB()

model.fit(X_train, y_train)

predictions = model.predict(X_test)
```

Similarly, many other popular machine learning models are available through simple interfaces:

- Linear Regression
- Logistic Regression
- Naive Bayes
- Decision Trees
- Random Forests
- Gradient Boosting
- XGBoost
- LightGBM
- Support Vector Machines (SVM)
- K-Nearest Neighbors (KNN)
- Neural Networks
- Deep Learning Models

As a result, much of the day-to-day work of a data scientist involves selecting, training, evaluating, and tuning models rather than implementing them from scratch.

Unfortunately, many practitioners learn how to use these tools without fully understanding the underlying mathematics. This can be dangerous: when a model behaves unexpectedly, produces misleading predictions, or violates its assumptions, it is often impossible to diagnose the problem without understanding the theory behind the algorithm.

For this reason, our goal in this course is not only to learn how to use machine learning tools, but also to understand the probabilistic and statistical ideas that make these algorithms work.

```
Accuracy = 0.9777777777777777
```

8 Bayes' Formula in Machine Learning

The Naive Bayes classifier is one of the earliest and most successful machine learning algorithms.

Its entire mathematical foundation is Bayes' formula:

$$P(\text{class} \mid \text{features}) = \frac{P(\text{features} \mid \text{class})P(\text{class})}{P(\text{features})}.$$

Although modern AI systems use much more sophisticated models, probabilistic reasoning and conditional probabilities remain central ideas throughout machine learning and data science.