République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche
scientifique Université 8Mai 1945 – Guelma
Faculté des sciences et de la Technologie
Département d'Electronique et Télécommunications

Mémoire de fin d'étude
Pour l'obtention du diplôme de Master Académique

Domaine : **Sciences et Technologie**
Filière : **Electronique**
Spécialité : **Instrumentation**

ECG Signal Classification Using Convolutional Neural Networks (CNN)

Présenté par :

HAOUALA Mouadh

Sous la direction de :

Dr. BOUKAACHE Abdelnour

Juin 2022

ABSTRACT

Cardiovascular disease (CVD) is the main cause of death in the United States today. Analyzing the Electrocardiogram (ECG), a medical monitoring device that records heart activity, is the current method of disease detection. Unfortunately, finding professionals to examine a big volume of ECG data takes up far too much medical time and money. As a result, machine learning-based methods for recognizing ECG characteristics have steadily gained traction.
These traditional methods, on the other hand, have several downsides, such as the need for manual feature recognition, complex models, and a long learning curve. The five micro-classes of heartbeat types in the MIT-BIH Arrhythmia database are classified using a robust and efficient 12-layer deep one-dimensional convolutional neural network the five types of heartbeat features are classified, and wavelet self-adaptive threshold denoising method is used in the experiments the results reveal that the experimental model performs better in terms of accuracy, sensitivity, robustness, and anti-noise capabilities. Its precise classification effectively conserves medical resources, which benefits clinical practice.

RESUMÉ

Les maladies cardiovasculaires (MCV) sont la principale cause de décès aux États-Unis aujourd'hui. L'analyse de l'électrocardiogramme (ECG), un appareil de surveillance médicale qui enregistre l'activité cardiaque, est la méthode actuelle de détection de la maladie. Malheureusement, trouver des professionnels pour examiner un grand volume de données ECG prend beaucoup trop de temps et d'argent. En conséquence, les méthodes basées sur l'apprentissage automatique pour reconnaître les caractéristiques de l'ECG n'ont cessé de gagner du terrain.
Ces méthodes traditionnelles, en revanche, présentent plusieurs inconvénients, tels que la nécessité d'une reconnaissance manuelle des caractéristiques, des modèles complexes et une longue courbe d'apprentissage. Les cinq micro-classes de types de battements cardiaques dans la base de données d'arythmie MIT-BIH sont classées à l'aide d'un réseau de neurones convolutifs unidimensionnels profonds à 12 couches robustes et efficaces. Les cinq types de caractéristiques de battements cardiaques sont classés et une méthode de débruitage en ondelettes avec seuil auto-adaptatif est utilisée dans les expériences. Les résultats révèlent que le modèle expérimental fonctionne mieux en termes de précision, de sensibilité, de robustesse et de capacités anti-bruit. Sa classification précise préserve efficacement les ressources médicales, ce qui profite à la pratique clinique.

ملخص

أمراض القلب والأوعية الدموية (الأمراض القلبية الوعائية) هو السبب الرئيسي للوفاة في الولايات المتحدة اليوم. يعد تحليل مخطط كهربية القلب، وهو جهاز مراقبة طبي يسجل نشاط القلب، الطريقة الحالية للكشف عن المرض. لسوء الحظ، فإن العثور على محترفين لفحص حجم كبير من بيانات تخطيط القلب يستغرق الكثير من الوقت والمال الطبي. ونتيجة لذلك، اكتسبت الأساليب القائمة على التعلم الآلي للتعرف على خصائص تخطيط القلب قوة دفع ثابتة.

هذه الأساليب التقليدية، من ناحية أخرى، لها العديد من الجوانب السلبية، مثل الحاجة إلى التعرف اليدوي على الميزات والنماذج المعقدة ومنحنى التعلم الطويل. يتم تصنيف الفئات الصغرى الخمس لأنواع ضربات القلب في قاعدة بيانات عدم انتظام ضربات القلب في معهد ماساتشوستس للتكنولوجيا باستخدام شبكة عصبية تلافيفيه عميقة ذات بعد واحد قوية وفعالة من 12 طبقة يتم تصنيف الأنواع الخمسة من ميزات ضربات القلب، ويتم استخدام طريقة تقليل الضوضاء عتبة التكيف الذاتي المويجات في التجارب تكشف النتائج أن النموذج التجريبي يعمل بشكل أفضل من حيث الدقة والحساسية والمتانة وقدرات مكافحة الضوضاء. يحافظ تصنيفها الدقيق على الموارد الطبية بشكل فعال، مما يفيد الممارسة السريرية.

Acknowledgements

First and foremost, I'd want to express my gratitude to my Electronic Teacher, **Dr. Boukaache Abdelnour,** for guiding me through these project. He gave me important guidance and assisted me in finishing this project. His enthusiasm and assistance made a significant contribution to the project's success.

Furthermore, I would like to thank **Ng. Andrew,** a Stanford University professor, for teaching me machine learning and deep learning. His work and instruction have had a significant impact on my learning experience.

I'd also like to express my gratitude to my family and friends for their support. Without that help, I would not have been able to finish this project.

Last but not least, I want to express our gratitude to everyone who assisted and encouraged me to work on this project.

Contents

List of figures

List of tables

Introduction 1

Chapter 1 Electrocardiogram (ECG) signals

1. Anatomy and Physiology of the Heart 3
1.1 Overview of the heart 3
1.2 Location, size, and the shape of the heart 3
1.3 Surfaces of the heart 4
2. The electrocardiogram (ECG) 7
2.1 What the ECG is about 7
2.2 The electricity of the heart 7
2.2.1 The wiring diagram of the heart 7
2.2.2 The rhythm of the heart 8
2.3 The different parts of the ECG 8
2.3.1 Times and Speeds 10
2.4 The ECG electrical pictures 12
2.4.1 The 12 leads ECG 13
2.5 Disorders affecting a 12 leads ECG 16
2.5.1 Angina 16
2.5.2 Bundle branch block 17
2.5.2.1 Right bundle branch block 17
2.5.2.2 Left bundle branch block 19
2.5.3 Myocardial infarction 21
3. Conclusion 22

Chapter 2 Deep learning in sequence data

1. Introduction 24
2. From everything to machine learning 24
3. What is intelligence 25
4. Artificial intelligence, machine learning, deep learning 28
4.1 Artificial intelligence 29
4.2 Machine learning 29

4.3 The “deep’’ in deep learning 31
4.4 Understanding how deep learning works, in three figures 33
4.5 Gradient Descent 35
4.6 The Backpropagation algorithm 39
5. Convolutional Neural Networks 39
5.1 Architecture of ConvNet 39
5.2 Convolution layer 42
5.3 Pooling layer 45
5.4 Sequence processing with convnets 46
5.5 ConvNets building blocks for sequence data 46
5.5.1 1D convolutional for sequence data 47
5.5.2 Activation layers 47
5.5.3 Pooling layer for sequence data 48
5.5.4 Fully connected layers (Dense) 48
5.5.5 Dropout layer 49
5.5.6 Softmax 49
5.6 Training algorithm for sequence data 49
6. Conclusion 50

Chapter 3 ECG Classifictaion Using Convolutional Neural Network

1. Introduction 52
2. ECG Data Processing 52
2.1 ECG Dataset 52
2.2 Preprocessing 53
2.3 Data segmentation 54
2.4 Data enhancement 56
3. The Architecture of the convolutional neural network 56
4. Experiment setup 59
5. Evaluation index 59
6. Results and discussion 60
7. Examining the confusion matrix 62
8. Conclusion 64
Conclusion 66
References

List of figures

1.1 Anterior view of the chest wall showing skeletal structures and the surface projection of the heart 4
1.2 Appearance of the heart this photograph shows a living human heart prepared from transplantation into a patient 4
1.3 The base of the heart 5
1.4 The anterior surface of the heart 5
1.5 The inferior surface of the heart the inferior part of the fibrous pericardium has been removed with the diaphragm 6
1.6 The wiring diagram of the heart 8
1.7 Shape of the normal ECG, including a U wave 9
1.8 Parts of the QRS complex 9
1.9 Relationship between the square on ECG paper and time 10
1.10 The components of the ECG complex 11
1.11 Normal PR interval and QRS complex 12
1.12 Normal PR interval and prolonged QRS complex 12
1.13 Why do we need 12 leads 14
1.14 The ECG patterns recorded by the six 'standard' leads 14
1.15 The relationship between the six chest leads and the heart 15
1.16 The ECG patterns recorded by the chest leads 16
1.17 ECG changes associated with angina 17
1.18 How RBBB occurs 18
1.19 Recognizing RBBB 19
1.20 How LBBB occurs 20
1.21 Recognizing LBBB 21
1.22 Changes in an MI 22
2.1 From everything to machine learning 25
2.2 Intelligence defined as ability to accomplish complex goals 27
2.3 Artificial intelligence, machine learning, deep learning 29
2.4 Machine learning: a new programming paradigm 30
2.5 A deep neural network for digit classification 32
2.6 Deep representations learned by a digit classification model 33

2.7 A neural network is parameterized by its weight 34
2.8 A loss function measures the quality of the network's output 34
2.9 The loss score is used as feedback signal to adjust the weight 35
2.10 Gradient Descent 36
2.11 Learning rate too small 37
2.12 Learning rate too large 37
2.13 Gradient Descent pitfalls 38
2.14 Gradient Descent with and Without Feature scaling 38
2.15 Feature extractor used to be independent of Machine learning 40
2.16 ConvNet's feature extractor is composed of special kinds of neural network 41
2.17 Typical architecture of ConvNet 41
2.18 The convolution layer process 43
2.19 4x4 pixel image 43
2.20 The convolution operation starts at the upper left corner 44
2.21 The second convolution operation 44
2.22 The convolution operation starts over from the left 44
2.23 The feature map of the given filter has been completed 45
2.24 The 4x4 pixel input image 45
2.25 The results of pooling operation using two different methods 46
2.26 How 1D convolution work: each output timestep is obtained from a temporal patch in the input sequence 47
3.1 ECG signal before and after filtering 53
3.2 The architecture for the proposed CNN model 58
3.3 The training process 60
3.4 Zoom in of training progress 61
3.5 The confusion matrix 62

List of tables

1.1 Relationship between the number of large squares between successive R waves and the heart rate 11

1.2 ECG leads 13

2.1 ReLU, Leaky ReLU and ELU activations 48

3.1 Table of beats type in MIT – BIH records 54

3.2 Dataset size of the different types 54

3.3 A summary table of the proposed CNN model for this work 57

3.4 Selected best used parameters 59

3.5 Accuracy of the four different cases 63

Introduction

Introduction

Cardiovascular disease is a widespread condition that poses a major threat to human health, particularly among the middle-aged and elderly. It is marked by a high prevalence, impairment, and fatality rate. The globe is currently dealing with an aging population. The worsening of cardiovascular disease has turned into a huge public health issue. ECG analysis is a useful tool for determining heart health. As a result, cardiovascular disorders require the identification and classification of ECG signals. The early prevention and rapid discovery and treatment makes the classification of related ECG signals extremely important to research.

The ECG is a real-time visual time series (Sequence data) that records the electrical activity generated by each cardiac cycle of the heart. It is currently frequently utilized in heart rate detection.

This non-invasive detection technology is simple to use and has become a valuable tool for assisting doctors with pathology analysis. At this point, the doctor's evaluation of cardiovascular illness is mostly based on his or her experience. There are many different forms of cardiac disorders, and long-term manual detection makes it easier to make a mistake.

A new dilemma has arisen: how to swiftly and accurately analyze certain diseases. Furthermore, the characteristics of ECG signals include unpredictable, low frequency, and vulnerable, leading in unreliable diagnosis outcomes to increase the need of the efficiency and accuracy of ECG recognition. Intelligent automatic recognition and classification of ECG signals has become an unavoidable option.

In recent years, machine learning and deep learning networks have not only achieved remarkable achievements in the fields of image processing, audio recognition, and a variety of other fields, but they have also become widely used in the assisted diagnosis of heart disease based on ECG signals. For example, *M. Wu et al.* (2020) proposed an optimization-based deep convolutional neural network to classify five different heartbeats. Deep learning networks, as opposed to typical neural networks, can extract features automatically, recognize detailed data patterns, and remove complex signal preparation.

The deep learning networks also has a better nonlinear fitting ability, which helps it recognize single-lead, multi-class, and imbalanced ECG datasets more effectively.

The convolutional neural network (CNN) is a feedforward neural network that has been extensively studied and utilized in deep learning and has been effectively applied to the categorization of arrhythmias.

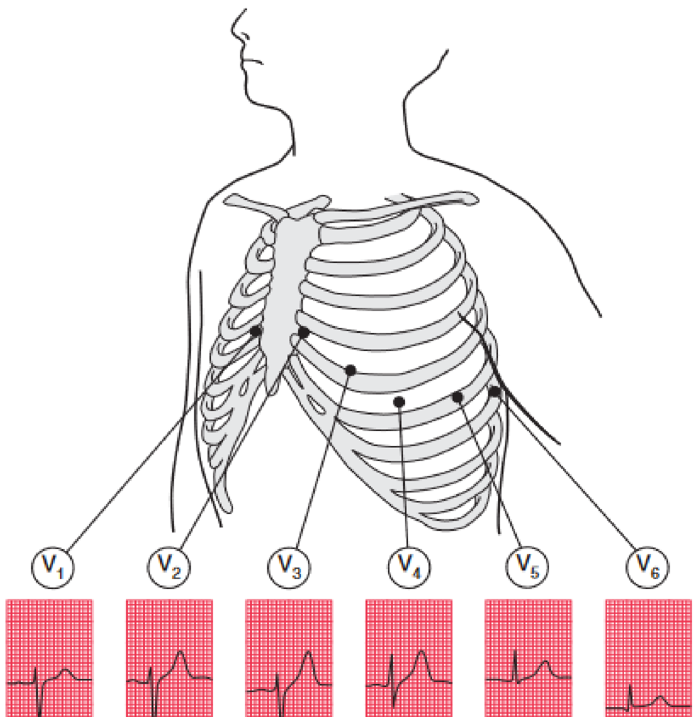
There is very little effort devoted to classifying the micro-classes of the ECG signal, which is our main motivation to study the micro-classification heartbeats, which are divided into five types: Normal (NOR), Left Bundle Branch Block (LBBB), Right Bundle Branch Block (RBBB), Atrial Premature (AP), and Premature Ventricular Contraction (PVC).

The used algorithm is endowed with the capacity to effectively process the non-filtered dataset with its potential anti-noise features, and the findings results can be used as a valuable source of benchmark literature for other researchers in the same field for future study work.

The experiment primarily describes the ECG dataset utilized in this study and provides a full description of data segmentation and preparation in the ECG Data Processing section. The architecture of the used algorithm and the specific experiment design are detailed in the last chapter.

The proposed network's performance and resilience are assessed using the MIT-BIH Arrhythmia database in chapter 3. Finally, a conclusion is given.

Electrocardiogram (ECG)



1. Anatomy and Physiology of the Heart

1.1 Overview of the heart

The heart is a muscular organ that acts as a pump to send blood continuously throughout your body. The heart is at the center of the circulatory system. This system consists of a network of blood vessels, such as arteries, veins, and capillaries. These blood vessels carry blood to and from all areas of the body. [1]

An electrical system regulates the heart and uses electrical signals to contract the heart's walls. When the walls contract, blood is pumped into the circulatory system. A system of inlet and outlet valves in the heart chambers works to ensure that blood flows in the right direction. The heart is vital to your health and nearly everything that goes on in the body. Without the heart's pumping action, blood can't circulate within the body. [1]

Blood carries the oxygen and nutrients that your organs need to work normally. Blood also carries carbon dioxide, a waste product, to your lungs to be passed out of the body and into the air. [1]

A healthy heart supplies the areas of the body with the right amount of blood at the rate needed to work normally. If disease or injury weakens the heart, the body's organs won't receive enough blood to work normally. [1]

1.2 Location, size, and shape of the heart

The heart is a hollow muscular organ that lies in the space between the lungs (i.e., the mediastinum) in the middle of the chest (Figure. 1.1). It sits behind the sternum and just above the diaphragm. About two-thirds of the heart lies to the left of the midline of the sternum, and the remaining third lies to the right of the sternum. [2]

The adult heart is about 5 inches (12 cm) long, 3.5 inches (9 cm) wide, and 2.5 inches (6 cm) thick (Figure. 1.2). It weighs typically between 250 and 350 g (about 11 oz) and is about the size of its owner's fist. The weight of the heart is about 0.45% of a man's body weight and about 0.40% of a woman's, a person's heart size and weight are influenced by his or her age, body weight and build frequency of physical exercise, and heart disease. [2]

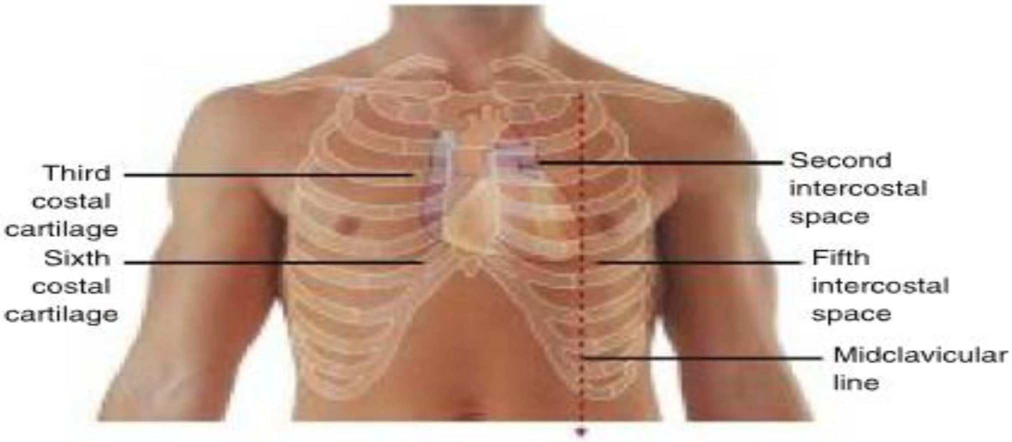


Figure 1.1 Anterior view of the chest wall showing skeletal structures and the surface projection of the heart. [2]

Figure 1.2 Appearance of the heart. This photograph shows a living human heart prepared from transplantation into a patient. [2]

1.3 Surfaces of the heart

The base or posterior surface of the heart is formed by the left atrium, a small portion of the right atrium, and proximal portions of the superior and inferior venae cave and the pulmonary veins (Figure. 1.3). [2]

The front (anterior) surface of the heart lies behind the sternum and costal cartilages. It is formed by portions of the right atrium and the left and right ventricles (Figure. 1.4). However, because the heart is tilted slightly toward the left in the chest, the right ventricle is the area of the heart that lies almost directly behind the sternum the apex, or lower portion of the heart, is formed by the tip of the left ventricle. The apex lies just above the diaphragm at about the level of the fifth intercostal space in the midclavicular line. [2]

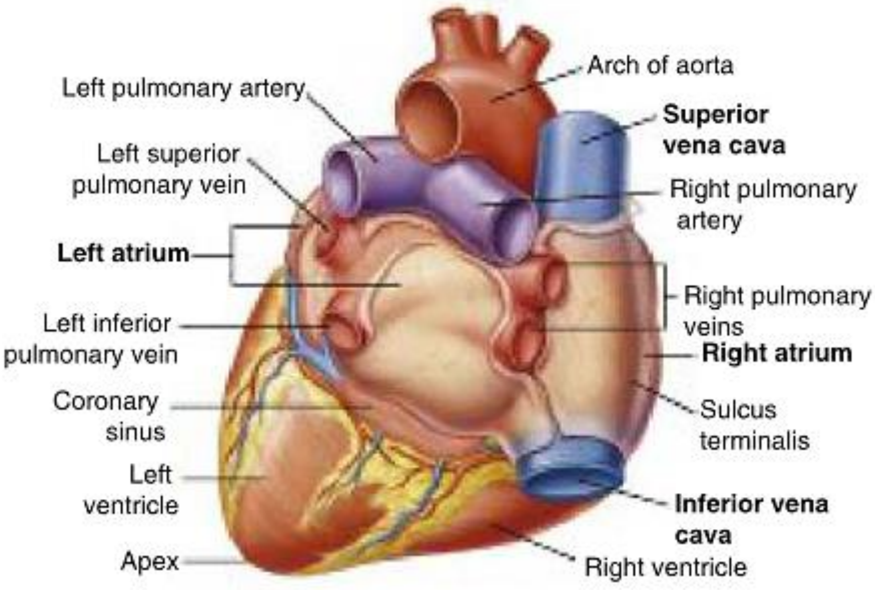


Figure 1.3 The base of the heart. [2]

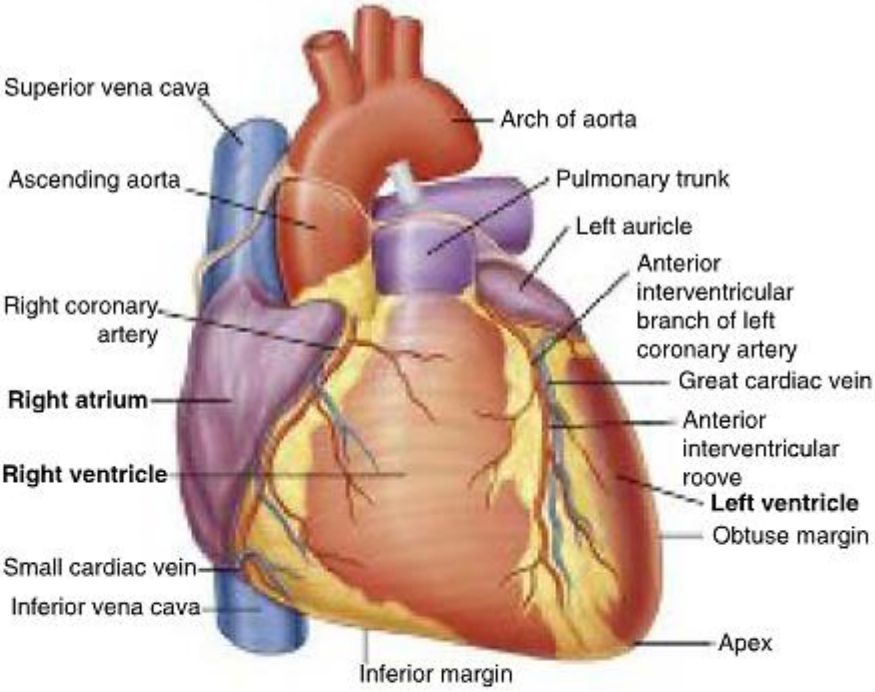


Figure 1.4 The anterior surface of the heart. [2]

The heart's left side (i.e., left lateral surface) faces the left lung and is made up mostly of the left ventricle and a portion of the left atrium. The right lateral surface faces the right lung and consists of the right atrium. The heart's bottom (i.e., inferior) surface is formed primarily by the left ventricle, with small portions of the right ventricle and right atrium. The right and left ventricles are separated by a groove containing the posterior interventricular vessels. Because the inferior surface of the heart rests on the diaphragm, it is also called the diaphragmatic surface (Figure. 1.5). [2]

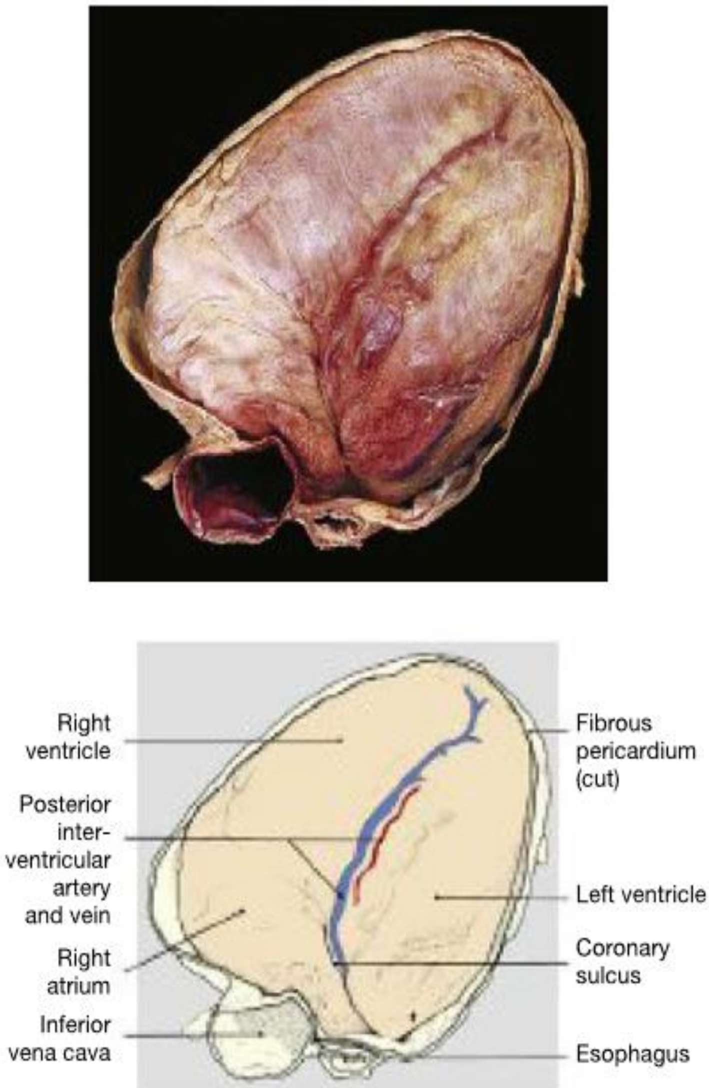


Figure 1.5 The inferior surface of the heart. The inferior part of the fibrous pericardium has been removed with the diaphragm. [2]

2. The electrocardiogram (ECG)

2.1 What the ECG is about

ECG stands for electrocardiogram or electrocardiograph. In some countries, the abbreviation used is ‘EKG’. The ECG can provide evidence to support a diagnosis, and in some cases, it is crucial for patient management. It is, however important to see the ECG as a tool, and not as an end in itself. Clinical diagnosis depends mainly on a patient’s history and to a lesser extent on the physical examination. [3]

The ECG is essential for the diagnosis, and therefore the management, of abnormal cardiac rhythms. It helps with the diagnosis of the cause of chest pain, and the proper use of early intervention in myocardial infarction depends upon it. It can help with the diagnosis of the cause of dizziness, syncope, and breathlessness with practice, interpreting the ECG is a matter of pattern recognition. However, the ECG can be analyzed from the first principles if a few simple rules and basic facts are remembered. [3]

2.2 The electricity of the heart

The contraction of any muscle is associated with electrical changes called depolarization. These changes can be detected by electrodes attached to the surface of the body. Since all muscular contraction will be detected, the electrical changes associated with contraction of the heart muscle will only be clear if the patient is fully relaxed and no skeletal muscles are contracting. Although the heart has four chambers, from the electrical point of view it can be thought of as having only two, because the two atria contract together (depolarization), and then the two ventricles contract together. [3]

2.2.1 The wiring diagram of the heart

The electrical discharge for each cardiac cycle normally starts in a special area of the right atrium called the ‘sinoatrial (SA) node’ (Figure.1.6). Depolarization then spreads through the atrial muscle fibers. There is a delay while depolarization spreads through another special area in the atrium, the ‘atrioventricular node’ (also called the ‘AV node’, or sometimes just the node’). [3]

Thereafter, the depolarization of the wave travels very rapidly down specialized conduction tissue, the 'bundle of His', which divides in the septum between the ventricles into right and left bundle branches. The left bundle branch itself divides into two. Within the mass of ventricular muscle, conduction spreads somewhat more slowly, through specialized tissue called 'Purkinje fibers'. [3]

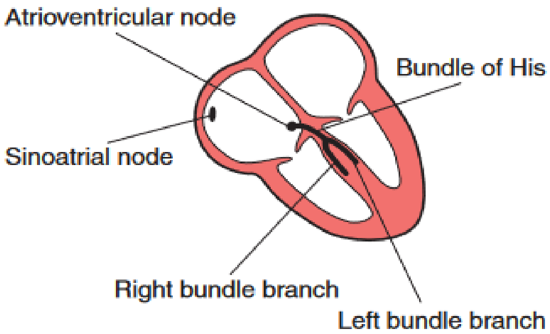


Figure 1.6 The wiring diagram of the heart. [3]

2.2.2 The rhythm of the heart

As we shall see later, electrical activation of the heart can sometimes begin in places other than the SA node. The word 'rhythm' is used to refer to the part of the heart which is controlling the activation sequence. The normal heart rhythm, with electrical activation beginning in the SA node, is called 'sinus rhythm'. [3]

2.3 The different parts of the ECG

The muscle mass of the atria is small compared with that of the ventricles, and so the electrical change accompanying the contraction of the atria is small, contraction of the atria is associated with the ECG wave called 'P' (Figure. 1.7). The ventricular mass is large, and so there is a large deflection of the ECG when the ventricles are depolarized: this is called the 'QRS' complex, The 'T' wave of the ECG is associated with the return of the ventricular mass to its resting electrical state ('repolarization'). [3]

The letters P, Q, R, S, and T were selected in the early days of ECG history and were chosen arbitrarily. The P Q, R, S, and T deflections are all called waves; the Q, R, and S waves together make up a complex; and the interval between the S wave and the beginning of the T wave is called the ST 'segment'. [3]

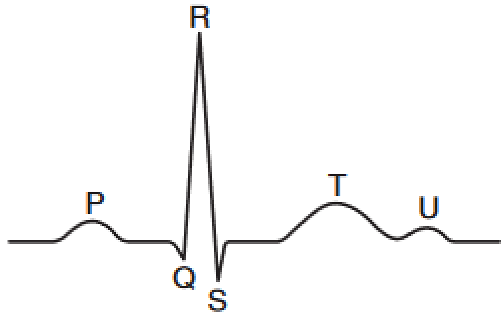


Figure 1.7 Shape of the normal ECG, Including a U wave. [3]

In some ECGs, an extra wave can be seen on the end of the T wave, and this is called a U wave. Its origin is uncertain, though it may represent repolarization of the papillary muscles. If a U wave follows a normally shaped T wave, it can be assumed to be normal. If it follows a flattened T wave, it may be pathological. [3]

The different parts of the QRS complex are labeled as shown in Figure 1.8. If the first deflection is downward, it is called a Q wave (Fig. 1.8a). An upward deflection is called an R wave, regardless of whether it is preceded by a Q wave or not (Figs 1.8b and 1.8c). [3]

Any deflection below the baseline following an R wave is called an S wave, regardless of whether there is a preceding Q wave (Figs 1.8d and 1.8e). [3]

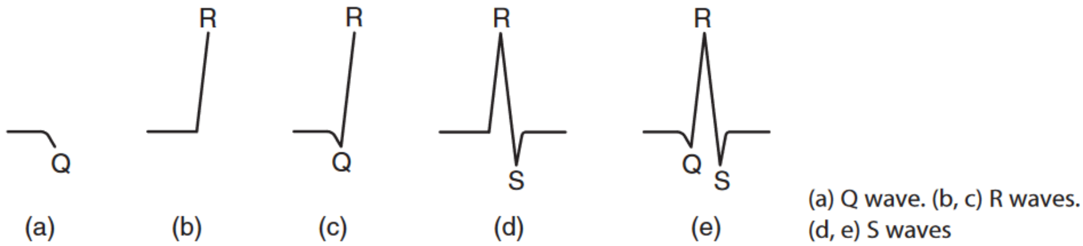


Figure 1.8 Parts of the QRS complex. [3]

2.3.1 Times and Speeds

ECG machines record changes in electrical activity by drawing a trace on a moving paper strip.

ECG machines run at a standard rate of 25 mm/s and use paper with standard-sized squares, each large square (5 mm) represents 0.2 second (s), i.e. 200 milliseconds (ms) (Figure.1.9). Therefore, there are five large squares per second, and 300 per minute. So an ECG event, such as a QRS complex, occurs once per a large square is occurring at a rate of 300/min, the heart rate can be calculated rapidly by remembering the sequence in Table 1.1. [3]

Just as the length of paper between R waves gives the heart rate, so the distance between the different parts of the P–QRS–T complex shows the time taken for conduction of the electrical discharge to spread through the different parts of the heart. [3]

The PR interval is measured from the beginning of the P wave to the beginning of the QRS complex. It is the time taken for excitation to spread from the SA node, through the atrial muscle and the AV node, down the bundle of His and into the ventricular muscle logically, it should be called the PQ interval, but common usage is 'PR interval' (Figure. 1.10). The normal PR interval is 120–220 ms, represented by 3–5 small squares. Most of this time is taken up by delays in the AV node (Figure. 1.11). [3]

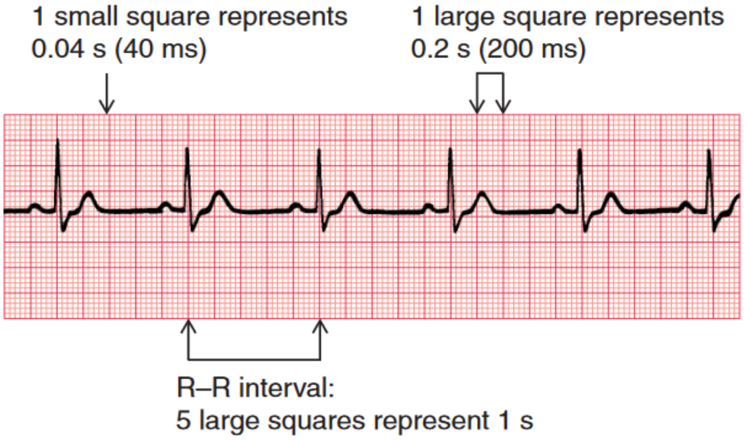


Figure 1.9 Relationship between the squares on ECG paper and time. [3]

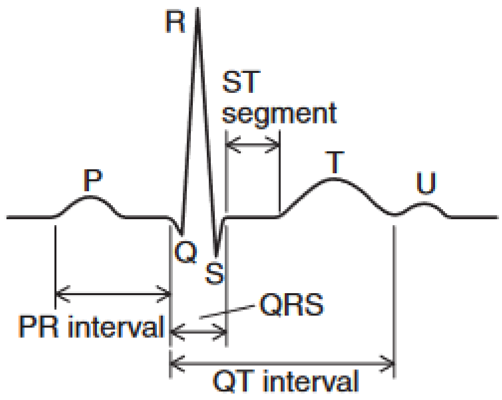


Figure 1.10 The components of the ECG complex. [3]

Table 1.1 Relationship between the number of large squares between successive R waves and the heart rate. [3]

R–R interval (large squares)	Heart rate (beats/min)
1	300
2	150
3	100
4	75
5	60
6	50

If the PR interval is very short, either the atria have been depolarized from close to the AV node, or there is abnormally fast conduction from the atria to the ventricles. [3]

The duration of the QRS complex shows how long excitation takes to spread through the ventricles. The QRS complex duration is normally 120 ms (represented by three small squares) or less, but any abnormality of conduction takes longer, and causes widened QRS complexes (Figure. 1.12). Remember that the QRS complex represents depolarization, not a contraction, of the ventricles contraction, is proceeding during the ECG's ST segment. [3]

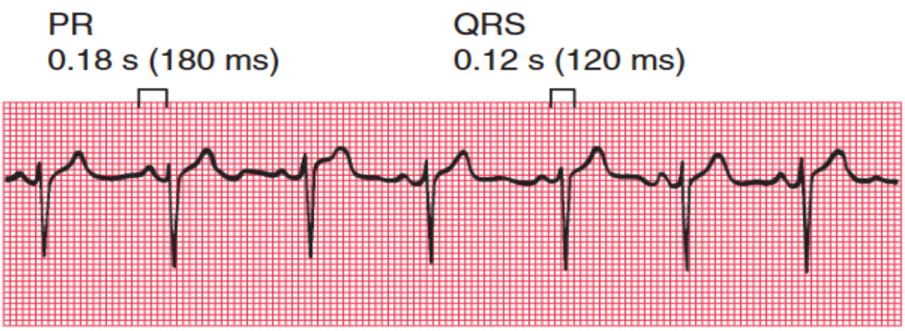


Figure 1.11 Normal PR interval and QRS complex. [3]

The QT interval varies with the heart rate. It is prolonged in patients with some electrolyte abnormalities, and more importantly, it is prolonged by some drugs. A prolonged QT interval (greater than 450 ms) may lead to ventricular tachycardia. [3]

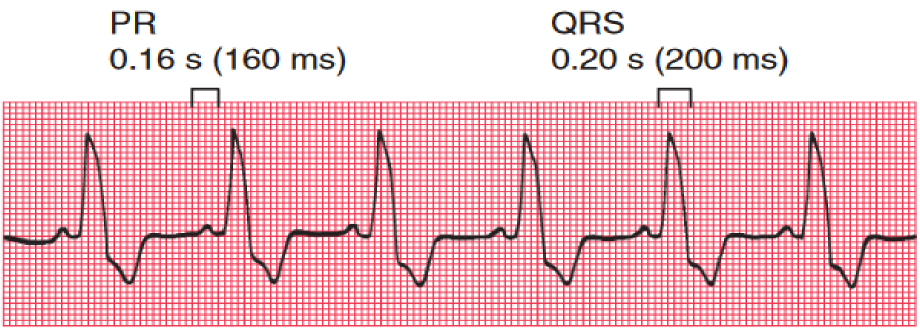


Figure 1.12 Normal PR interval and prolonged QRS complex. [3]

2.4 The ECG electrical pictures

The word 'lead' sometimes confuses sometimes it is used to mean the pieces of wire that connect the patient to the ECG recorder. [3]

Properly, a lead is an electrical picture of the heart. The electrical signal from the heart is detected at the surface of the body through electrodes, which are joined to the ECG recorder by wires one electrode is attached to each limb, and six to the front of the chest. The ECG recorder compares the electrical activity detected in the different electrodes, and the electrical picture so obtained is called a 'lead'. The different comparisons 'look at the heart from different directions. [3]

For example, when the recorder is set to 'lead I' it is comparing the electrical events detected by the electrodes attached to the right and left arms, each lead gives a different view of the electrical activity of the heart, and so a different ECG pattern. Strictly, each ECG pattern should be called 'lead ...', but often the word 'lead' is omitted. [3]

The ECG is made up of 12 characteristic views of the heart, six obtained from the 'limb' leads (I, II, III, VR, VL, VF) and six from the 'chest' leads (V 1–V6). It is not necessary to remember how the leads (or views of the heart) are derived by the recorder, but for those who like to know how it works, see Table 1.2. The electrode attached to the right leg is used as earth and does not contribute to any lead. [3]

Table 1.2 ECG leads. [3]

Lead	Comparison of electrical activity
I	LA and RA
II	LL and RA
III	LL and LA
VR	RA and average of (LA + LL)
VL	LA and average of (RA + LL)
VF	LL and average of (LA + RA)
V_1	V_1 and average of (LA + RA + LL)
V_2	V_2 and average of (LA + RA + LL)
V_3	V_3 and average of (LA + RA + LL)
V_4	V_4 and average of (LA + RA + LL)
V_5	V_5 and average of (LA + RA + LL)
V_6	V_6 and average of (LA + RA + LL)

Key: LA, left arm; RA, right arm; LL, left leg.

2.4.1 The 12 leads ECG

The 12-lead ECG provides 12 different views of the electrical activity of the heart, each view looking from the outside of the chest toward the reference point within the heart. [4]

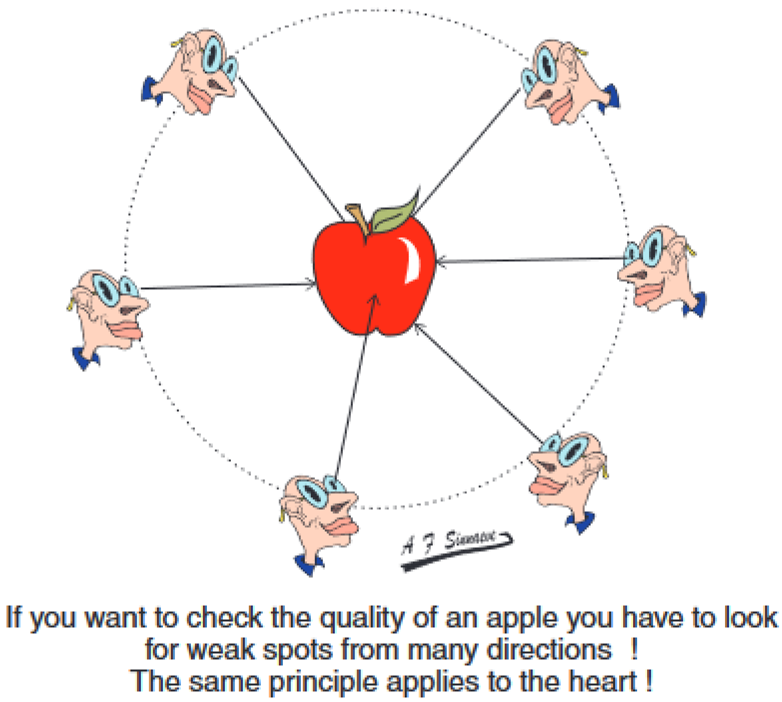


Figure 1.13 Why do we need 12 leads. [5]

ECG interpretation is easy if you remember the directions from which the various leads look at the heart. The six 'standard' leads, which are recorded from the electrodes attached to the limbs, can be thought of as looking at the heart in a vertical plane (i.e. from the sides of the feet) (Figure. 1.14). [3]

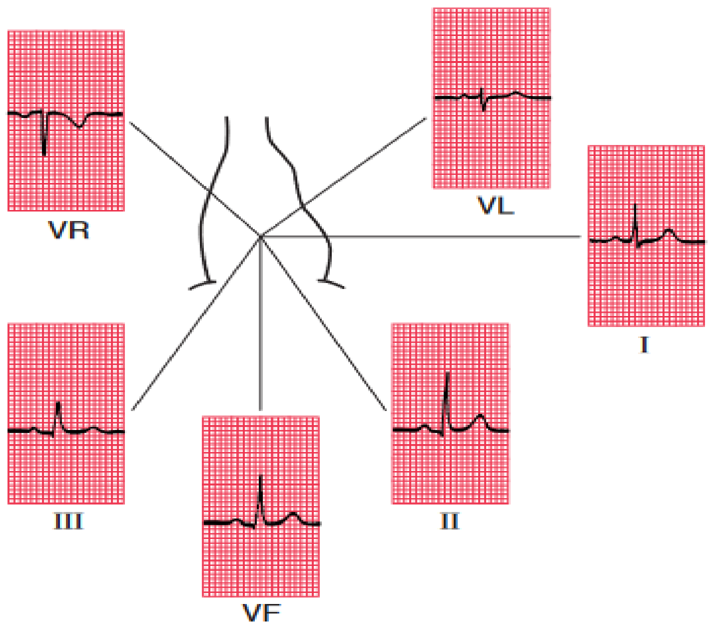


Figure 1.14 The ECG patterns recorded by the six 'standard' leads. [3]

Leads I, II, and VL look at the left lateral surface of the heart, leads III and VF at the inferior surface, and lead VR looks at the right atrium. The six V leads (V 1–V6) look at the heart in a horizontal plane, from the front and the left side. Thus, leads V 1 and V 2 look at the right ventricle, V 3 and V 4 look at the septum between the ventricles and the anterior wall of the left ventricle, and V 5 and V 6 look at the anterior and lateral walls of the left ventricle (Figure. 1.15). [3]

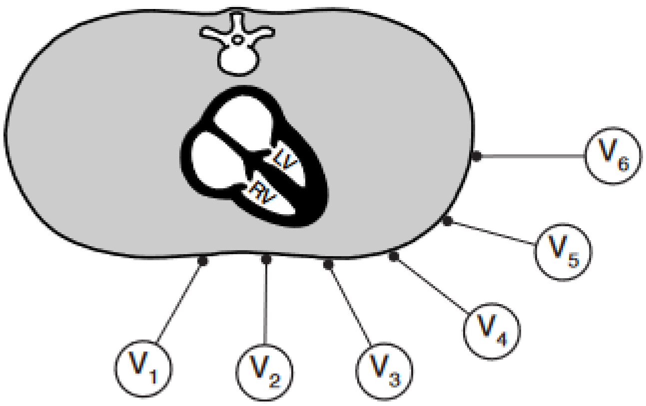


Figure 1.15 The relationship between the six chest leads and the heart. [3]

As with the limb leads, the chest leads each shows a different ECG pattern (Figure. 1.18). In each lead, the pattern is characteristic, being similar to individuals who have normal hearts.

The cardiac rhythm is identified from whichever lead shows the P wave most clearly usually lead II. When a single lead is recorded simply to show the rhythm, it is called a ‘rhythm strip’, but it is important not to make any diagnosis from a single lead, other than identifying the cardiac rhythm. [3]

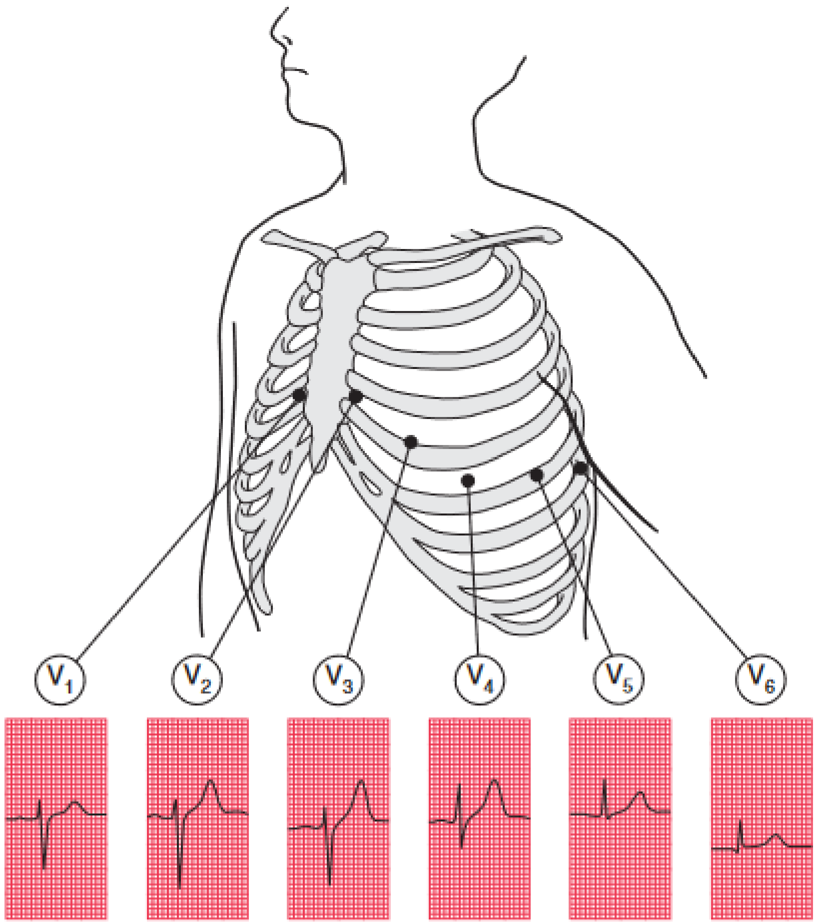


Figure 1.16 The ECG patterns recorded by the chest leads. [3]

2.5 Disorders affecting a 12 lead ECG

A 12-lead ECG is used to diagnose such conditions as angina, bundle-branch block, and myocardial infarction (MI). By reviewing sample ECGs, you'll know what classic signs to look for. Here's a rundown on these three common cardiac conditions and what 12-lead ECG signs to look for. [4]

2.5.1 Angina

During an episode of angina, the myocardium demands more oxygen than the coronary arteries can deliver. The arteries can't deliver enough blood, commonly as a result of a narrowing of the arteries from coronary artery disease (CAD), a condition that may be complicated by platelet clumping, thrombus formation, or vasospasm. [4]

An episode of angina usually lasts between 2 and 10 minutes. The closer to 30 minutes the pain lasts, the more likely the pain is from an MI rather than angina. [4]

A patient with either form of angina typically shows ischemic changes on the ECG only during the angina attack. (See ECG changes associated with angina (Figure1.17.) Because these changes may be fleeting, always obtain an order for, and perform, a 12-lead ECG as soon as the patient reports chest pain. [4]

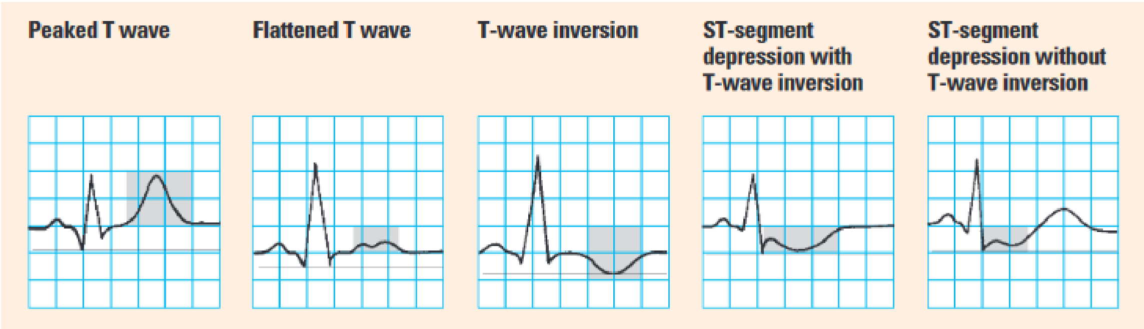


Figure 1.17 ECG changes associated with angina. [4]

2.5.2 Bundle branch block

One potential complication of an MI is a bundle-branch block. In this disorder, either the left or the right bundle branch fails to conduct impulses. A bundle-branch block that occurs farther down the left bundle, in the posterior or anterior fasciculus, is called a *hemiblock*. [4]

Some blocks require treatment with a temporary pacemaker. Others are monitored only to detect whether they progress to a more complete block. [4]

2.5.2.1 Right bundle branch block

RBBB occurs with such conditions as anterior wall MI, CAD, cardiomyopathy, cor-pulmonale, and pulmonary embolism. It may also occur without cardiac disease. If this block develops as the heart rate increases, it's called rate-related RBBB. (See How RBBB occurs Figure 1.18). [4]

In this disorder, the QRS complex is greater than 0.12 seconds and has a different configuration, sometimes resembling rabbit ears or the letter "M." (See Recognizing RBBB, Figure 1.19) Septal depolarization isn't affected in lead V1, so the initial small R wave remains. The R wave is followed by an S wave, which represents left ventricular depolarization, and a tall R wave (called R prime, or R), which represents late right ventricular depolarization. The T wave is negative in this lead. However, that deflection is called a secondary T-wave change and is of no clinical significance. [4]

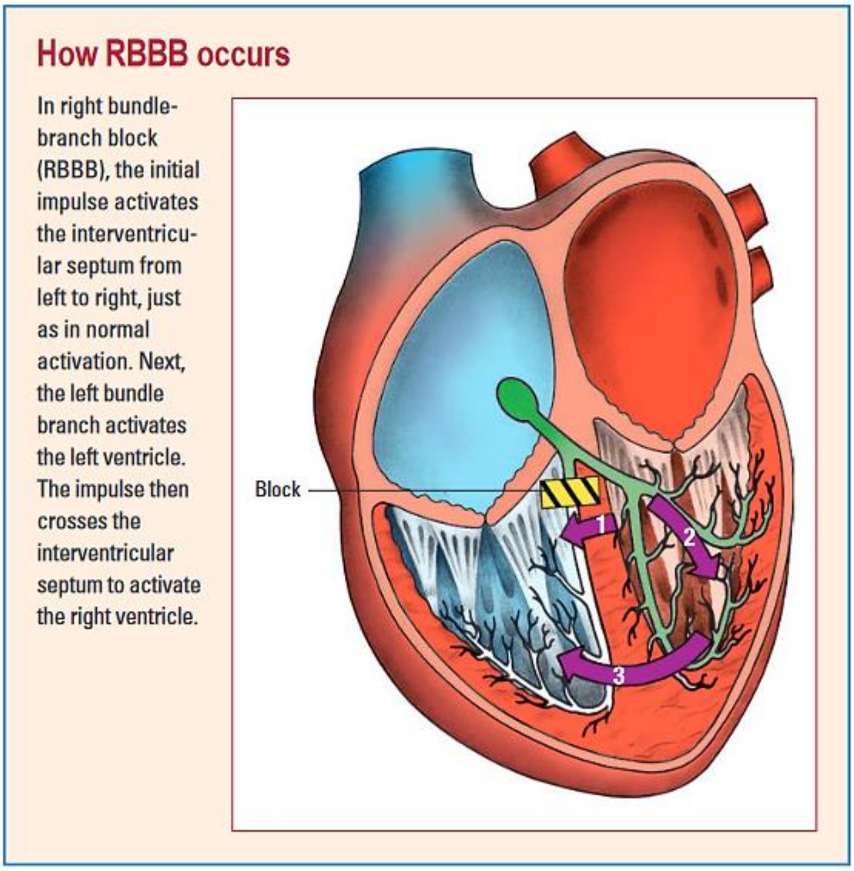


Figure 1.18 How RBBB occurs. [4]

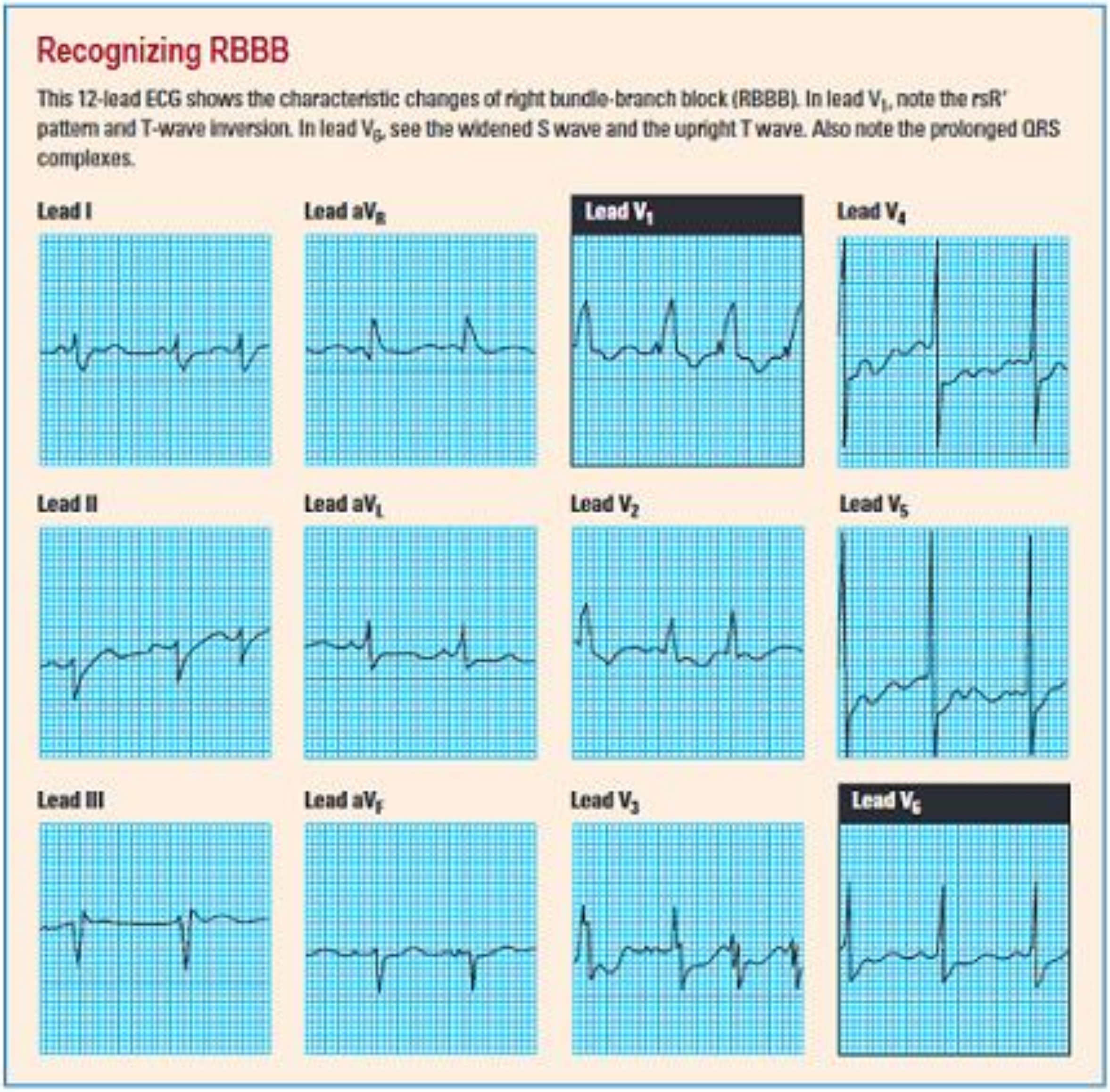


Figure 1.19 Recognizing RBBB. [4]

2.5.2.2 Left bundle branch block

Left bundle-branch block (LBBB) never occurs normally. This block is usually caused by hypertensive heart disease, aortic stenosis, degenerative changes of the conduction system, or CAD. (See How LBBB occurs, Figure 1.20) When it occurs along with an anterior wall MI usually signals a complete heart block, which requires the insertion of a pacemaker. [4]

In LBBB, the QRS complex will be greater than 0.12 seconds because the ventricles are activated sequentially, not simultaneously. (See Recognizing LBBB, Figure 1.21) As the wave of depolarization spreads from the right ventricle to the left, a wide S wave is produced in lead V1, with a positive T wave. The S wave may be preceded by a Q wave or a small R wave. [4]

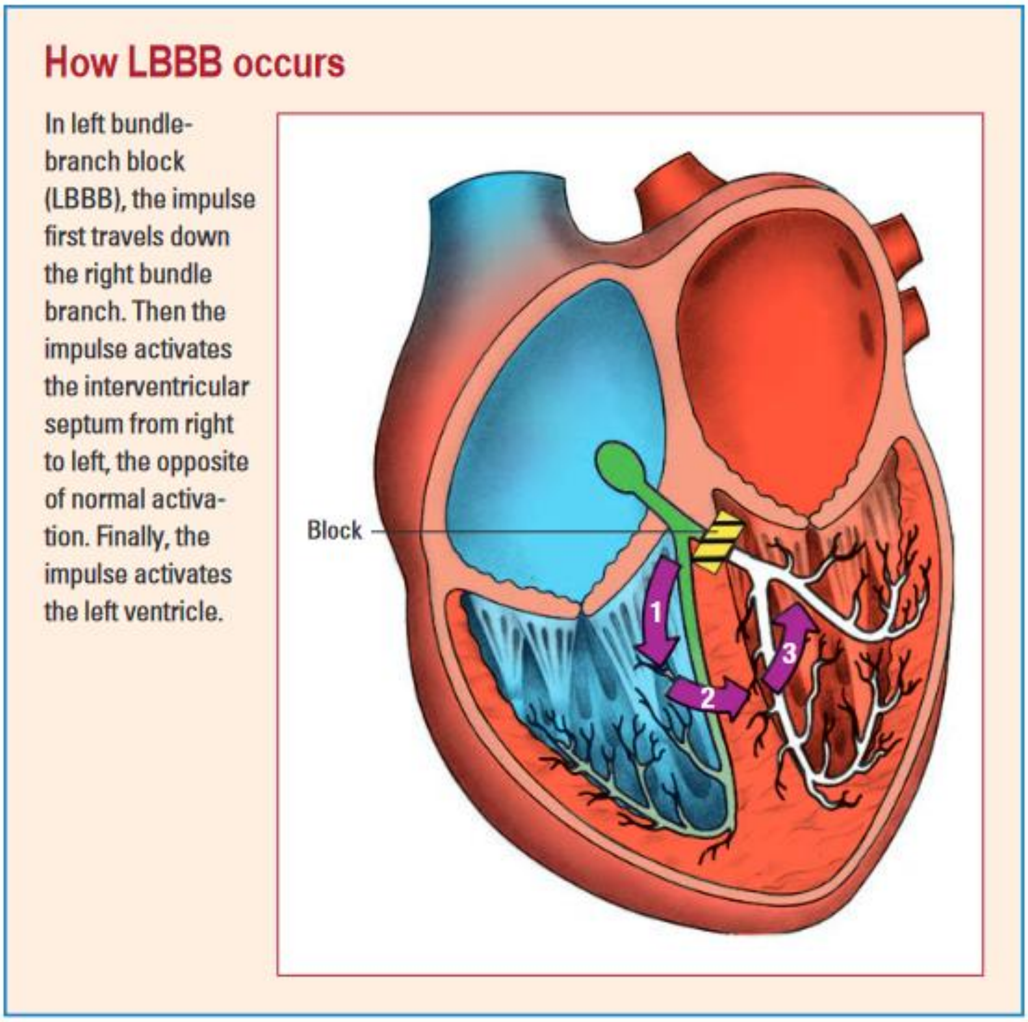


Figure 1.20 How LBBB occurs. [4]

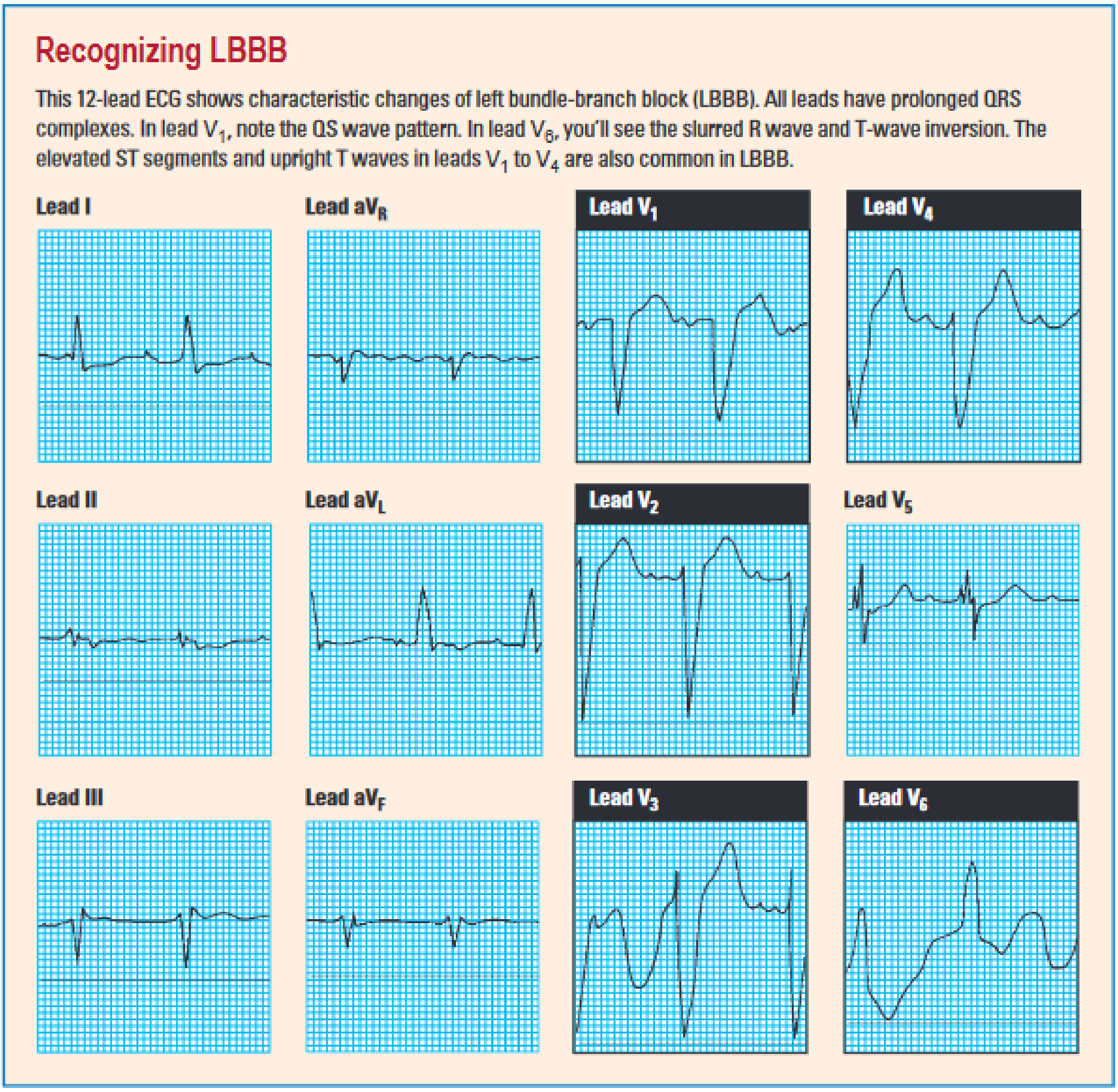


Figure 1.21 Recognizing LBBB. [4]

2.5.3 Myocardial infarction

Unlike angina, pain from an MI lasts for at least 20 minutes, may persist for several hours, and is unrelieved by rest. MI usually occurs in the left ventricle, although the location may vary depending on the coronary artery affected. [4]

For as long as the myocardium is deprived of an oxygen-rich blood supply, an ECG will reflect the three pathologic changes of an MI: ischemia, injury, and infarction (Figure1.22). [4]

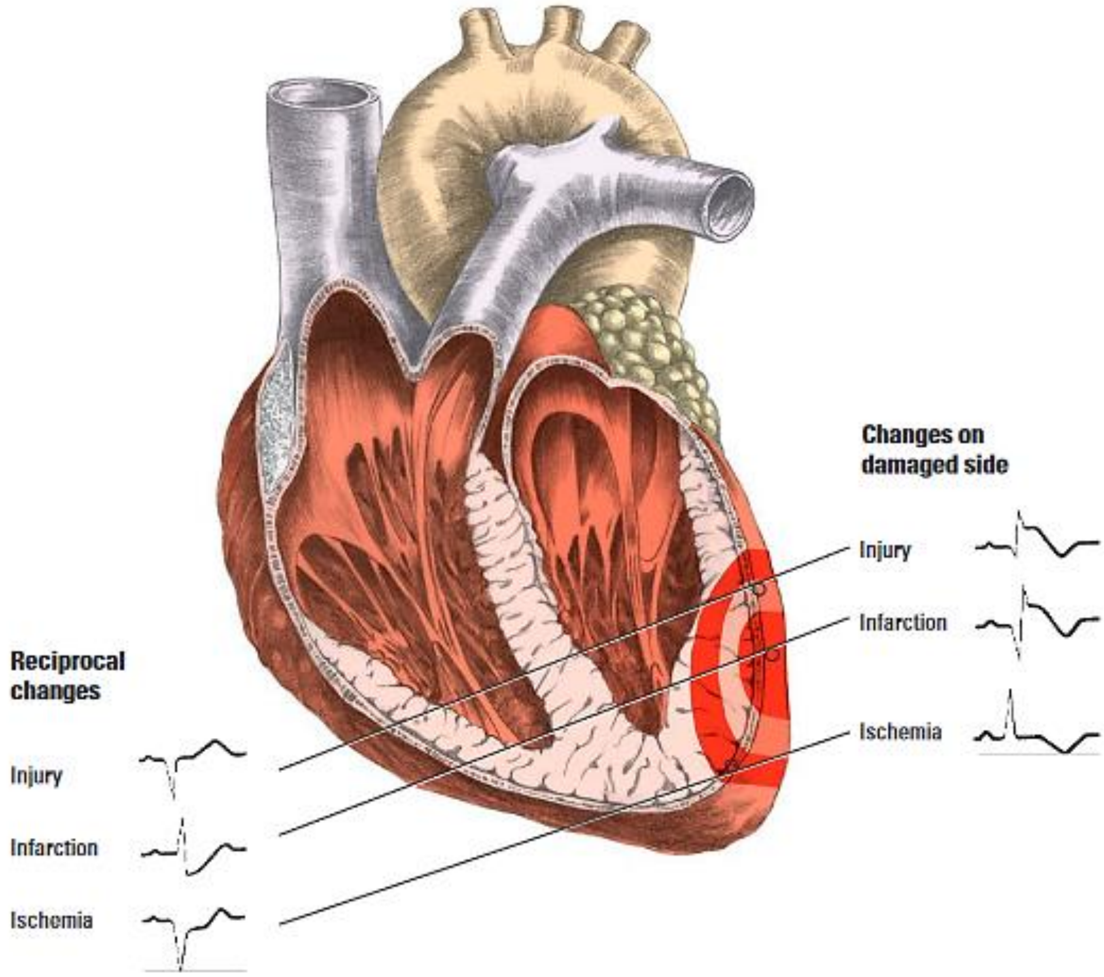
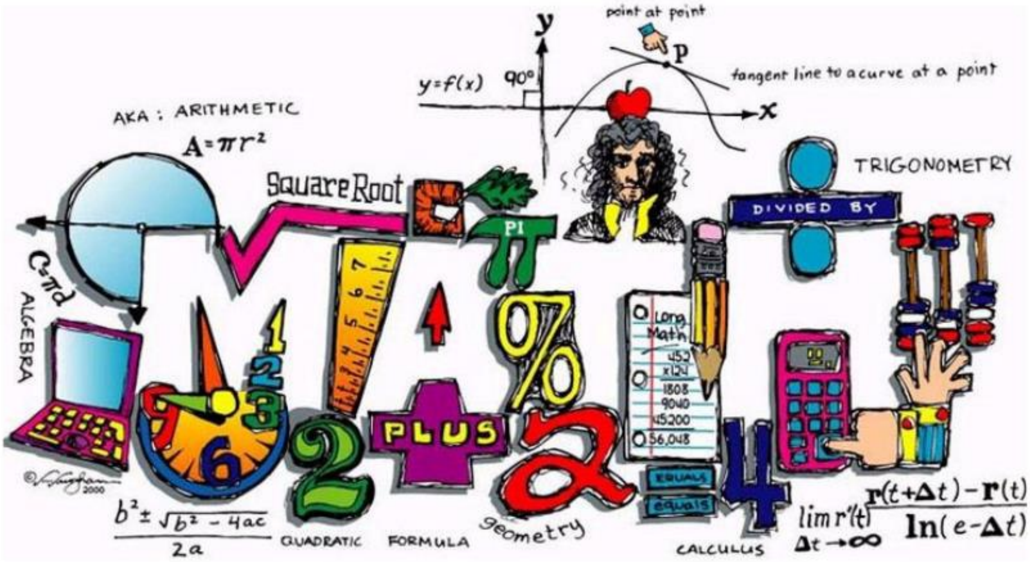


Figure 1.22 Changes in an MI. [4]

3. Conclusion

In this chapter, we demonstrated and explained the various parts of the heart, such as size and location, as well as the heart's structure. We also defined what an ECG is and the different blocks of the ECG. We gained a clear understanding of the 12 lead ECG as we discussed the various disorders that may affect the 12 lead ECG. This chapter provides a brief explanation and understanding of the ECG, its various components, and the role it plays.

Deep Learning for Sequence Data



1. Introduction

It is impossible for one human being to understand everything. His resources (e.g., time, cognitive abilities) are limited, so he must carefully distribute them for worthy goals.

There is a range of possible broad goals for one human and two representative extremes are:

1. Understand one thing in a deep way (e.g., Bobby Fischer and chess)
2. Understand many things in a shallow way (e.g., an average high school student who understands a bit about more subjects).

Some of the terms that we use initially are vague, but as we make progress, we will use better terms. Just like a painter who starts with broad and light-handed strokes, and incrementally adds more details, so we start with broad terms and incrementally we add more details.

The starting point of the search for my interest is “everything”. Intuitively speaking, I think of it as “all the things that we can potentially think about deliberately (i.e., consciously and intentionally)”, and we visualize it as an infinitely big physical object from which we need to remove all the parts that we are not interested in.

Two operations can be used: division and filtering. Division splits one thing into more parts based on some common property. Ideally, the parts should be jointly exhaustive and mutually exclusive, but sometimes the parts can be fuzzy, and sometimes it is even impossible to define exhaustively all the possible parts of one whole.

When we divide one whole into two parts and, we would like the parts to be sufficiently independent, such that, we can study one of them without reference to the other. The filtering operation selects things that we are interested in and excludes things that we are not interested in.

2. From everything to machine learning

We start with “everything” and we end with machine learning. Green nodes represent things in which we are interested, and red nodes represent things in which we have no interest (Figure 2.1). [6]

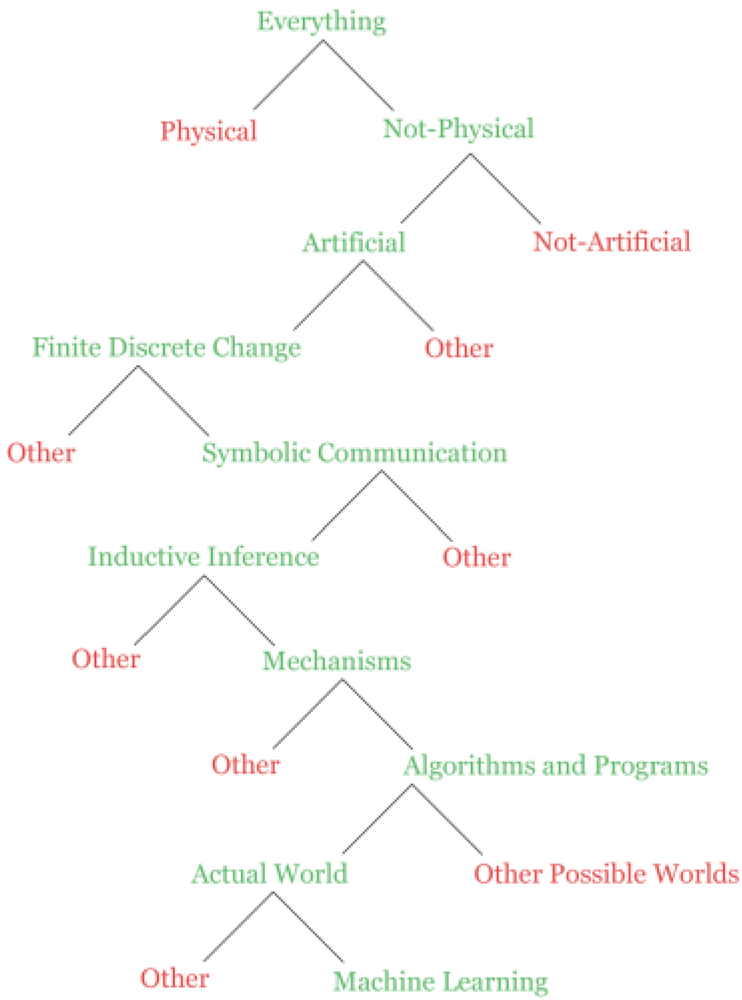


Figure 2.1 From everything to machine learning. [6]

3. What Is Intelligence?

There's no agreement on what intelligence is even among intelligent intelligence researchers! Therefore, there's clearly no undisputed "correct" definition of intelligence. Instead, there are many competing ones, including capacity for logic, understanding, planning, emotional knowledge, self-awareness, creativity, problem solving, and learning. [7]

In our exploration of the future of intelligence, we want to take a maximally broad and inclusive view, not limited to the sorts of intelligence that exist so far. [7]

intelligence = *ability to accomplish complex goals*

This is broad enough to include all above-mentioned definitions, since understanding, self-awareness, problem-solving, learning, etc. are all examples of complex goals that one might have. It's also broad enough to subsume the Oxford Dictionary definition "*the ability to acquire and apply knowledge and skills*" since one can have as a goal to apply knowledge and skills.

Because there are many possible goals, there are many possible types of intelligence. Therefore, it makes no sense to quantify intelligence of humans, non-human animals, or machines by a single number such as an IQ. What's more intelligent: a computer program that can only play chess or one that can only play Go? There's no sensible answer to this, since they're good at different things that can't be directly compared. However, we can say that a third program is more intelligent than both of the others if it's at least as good as them at accomplishing all goals, and strictly better at at least one (winning at chess). [7]

It also makes little sense to quibble about whether something is or isn't intelligent in borderline cases, since ability comes on a spectrum and isn't necessarily an all-or-nothing trait. What people have the ability to accomplish the goal of speaking? Newborns? No. Radio hosts? Yes. But what about toddlers who can speak ten words? Or five hundred words? Where would you draw the line?

It's not very interesting to try to draw an artificial line between intelligence and non-intelligence, and it's more useful to simply quantify the degree of ability for accomplishing different goals. [7]

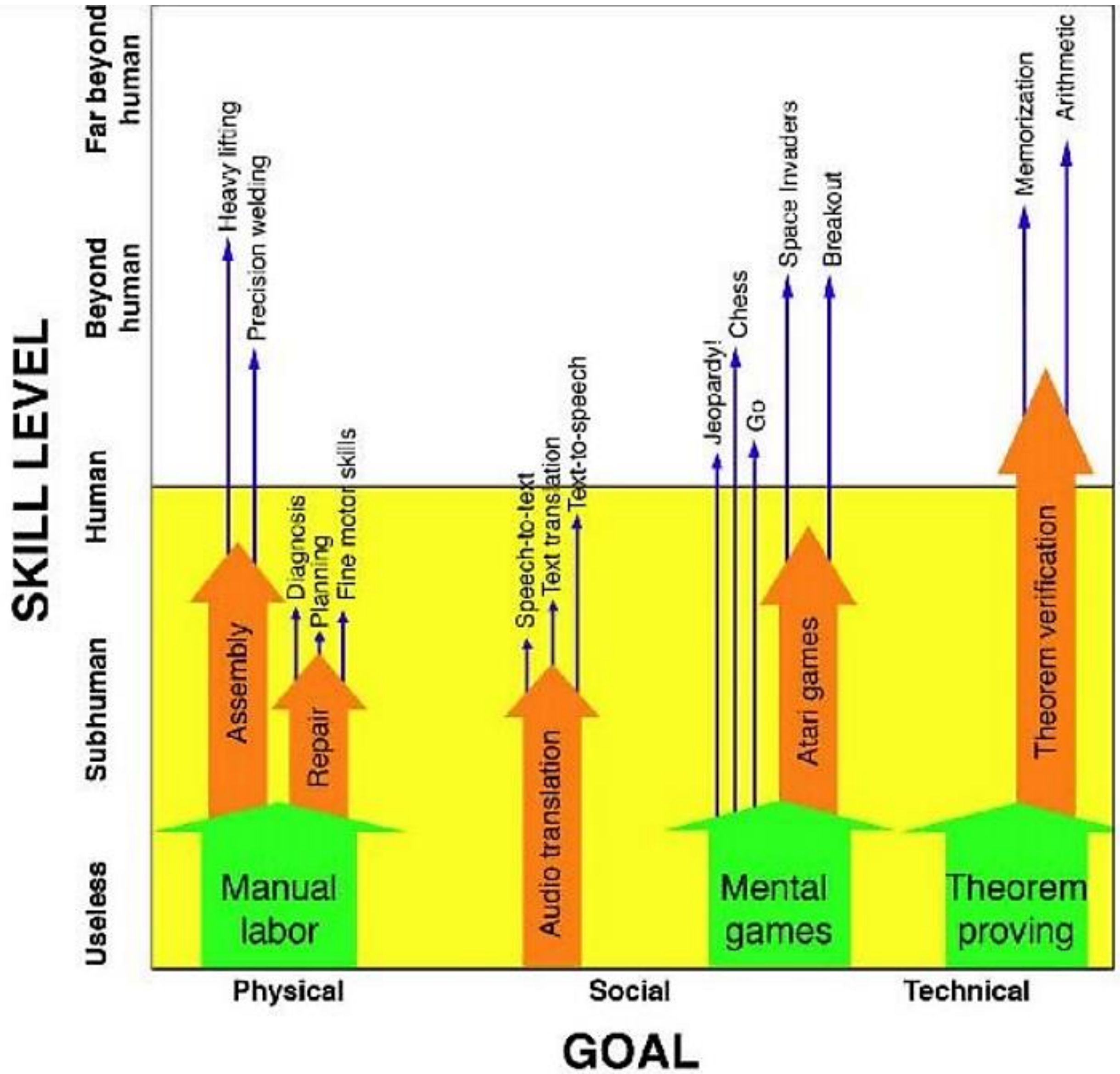


Figure 2.2 Intelligence measured by the ability spectrum across all goals. [7]

Although the word “intelligence” tends to have positive connotations, it’s important to note that we’re using it in a completely value-neutral way: as the ability to accomplish complex goals regardless of whether these goals are considered good or bad. Thus an intelligent person may be very good at helping people or very good at hurting people. [7]

Regarding goals, we also need to clear up the subtlety of whose goals we’re referring to. Suppose your future brand-new robotic personal assistant has no goals whatsoever of its own, but will do whatever you ask it to do, and you ask it to cook the perfect Italian dinner. If it goes online and researches Italian dinner recipes, how to get to the closest supermarket, how to strain pasta, and so on, and then successfully buys the ingredients and prepares a succulent meal. You'll presumably consider it intelligent even though the original goal was yours. In fact, it adopted your goal once you'd made your request, and then broke it into a hierarchy of subgoals of its own, from paying the cashier to grating the Parmesan. In this sense, intelligent behavior is inexorably linked to goal attainment. [7]

It's natural for us to rate the difficulty of tasks relative to how hard it is for us humans to perform them, as in (Figure 2.2) But this can give a misleading picture of how hard they are for computers. It feels much harder to multiply 314,159 by 271,828 than to recognize a friend in a photo, yet computers creamed us at arithmetic long before I was born, while human-level image recognition has only recently become possible. This fact that low-level sensorimotor tasks seem easy despite requiring enormous computational resources is known as Moravec's paradox, and is explained by the fact that our brain makes such tasks feel easy by dedicating massive amounts of customized hardware to them—more than a quarter of our brains, in fact.

Computer pioneer Alan Turing famously proved that if a computer can perform a certain bare minimum set of operations, then, given enough time and memory, it can be programmed to do anything that any other computer can do. Machines exceeding this critical threshold are called universal computers (aka Turing-universal computers); all of today's smartphones and laptops are universal in this sense. Analogously, the critical intelligence threshold required for AI design as the threshold for universal intelligence: given enough time and resources, it is possible to make itself able to accomplish any goal as well as any other intelligent entity. [7]

4. Artificial intelligence, machine learning, and deep learning

First, we need to define clearly what we're talking about when we mention AI. What are artificial intelligence, machine learning, and deep learning (see Figure 2.3) and How do they relate to each other? [8]

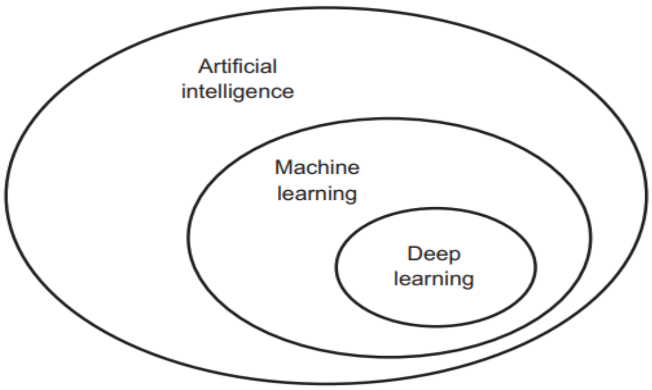


Figure 2.3 Artificial intelligence, machine learning, and deep learning. [8]

4.1 Artificial intelligence

Artificial intelligence was born in the 1950s, when a handful of pioneers from the nascent field of computer science started asking whether computers could be made to "think"—a question whose ramifications we're still exploring today. A concise definition of the field would be as follows: the effort to automate intellectual tasks normally performed by humans. As such, AI is a general field that encompasses machine learning and deep learning, but that also includes many more approaches that don't involve any learning. [8]

Early chess programs, for instance, only involved hardcoded rules crafted by programmers, and didn't qualify as machine learning. For a fairly long time, many experts believed that human-level artificial intelligence could be achieved by having programmers handcraft a sufficiently large set of explicit rules for manipulating knowledge. This approach is known as symbolic AI, and it was the dominant paradigm in AI from the 1950s to the late 1980s. It reached its peak popularity during the expert systems boom of the 1980s. [8]

Although symbolic AI proved suitable to solve well-defined, logical problems, such as playing chess, it turned out to be intractable to figure out explicit rules for solving more complex, fuzzy problems, such as image classification, speech recognition, and language translation. A new approach arose to take symbolic AI's place: machine learning. [8]

4.2 Machine learning

Machine learning arises from this question: could a computer go beyond "what we know how to order it to perform" and learn on its own how to perform a specified task? Could a computer surprise us? Rather than programmers crafting data processing rules by hand, could a computer automatically learn these rules by looking at data? This question opens the door to a new programming paradigm. In classical programming, the paradigm of symbolic AI, humans input rules (a program) and data to be processed according to these rules, and outcome answers (see Figure 2.4). With machine learning, humans input data as well as the answers expected from the data, and outcome the rules. These rules can then be applied to new data to produce original answers. [8]

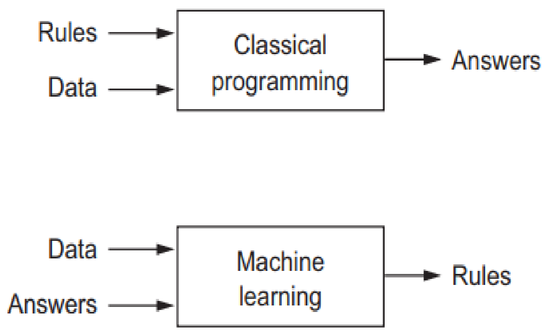


Figure 2.4 Machine learning: a new programming paradigm. [8]

A machine-learning system is trained rather than explicitly programmed. It's presented with many examples relevant to a task, and it finds statistical structure in these examples that eventually allows the system to come up with rules for automating the task. For instance, if you wished to automate the task of tagging your vacation pictures, you could present a machine-learning system with many examples of pictures already tagged by humans, and the system would learn statistical rules for associating specific pictures to specific tags. [8]

Although machine learning only started to flourish in the 1990s, it has quickly become the most popular and most successful subfield of AI, a trend driven by the availability of faster hardware and larger datasets. [8]

Machine learning is tightly related to mathematical statistics, but it differs from statistics in several important ways. Unlike statistics, machine learning tends to deal with large, complex datasets (such as a dataset of millions of images, each consisting of tens of thousands of pixels) for which classical statistical analysis such as Bayesian analysis would be impractical. [8]

As a result, machine learning and especially deep learning exhibits comparatively little mathematical theory—may be too little—and is engineering oriented. It's a hands-on discipline in which ideas are proven empirically more often than theoretically. [8]

4.3 The "deep" in deep learning

Deep learning is a specific subfield of machine learning: a new take on learning representations from data that puts an emphasis on learning successive layers of increasingly meaningful representations. The deep in deep learning isn't a reference to any kind of deeper understanding achieved by the approach; rather, it stands for this idea of successive layers of representations. How many layers contribute to a model of the data is called the depth of the model. Other appropriate names for the field could have been layered representations learning and hierarchical representations learning. [8]

Modern deep learning often involves tens or even hundreds of successive layers of representations and they're all learned automatically from exposure to training data. Meanwhile, other approaches to machine learning tend to focus on learning only one or two layers of representations of the data; hence, they're sometimes called shallow learning. [8]

In deep learning, these layered representations are (almost always) learned via models called neural networks, structured in literal layers stacked on top of each other. The term neural network is a reference to neurobiology, but although some of the central concepts in deep learning were developed in part by drawing inspiration from our understanding of the brain, deep-learning models are not models of the brain. There's no evidence that the brain implements anything like the learning mechanisms used in modern deep-learning models. You may come across pop-science articles proclaiming that deep learning works like the brain or was modeled after the brain, but that isn't the case. It would be confusing and counterproductive for newcomers to the field to think of deep learning as being in any way related to neurobiology; you don't need that shroud of "just like our minds" mystique and mystery, and you may as well forget anything you may have read about hypothetical links between deep learning and biology. For our purposes, deep learning is a mathematical framework for learning representations from data. [8]

What do the representations learned by a deep-learning algorithm look like? Let's examine how a network several layers deep (see Figure 2.5) transforms an image of a digit in order to recognize what digit it is. [8]

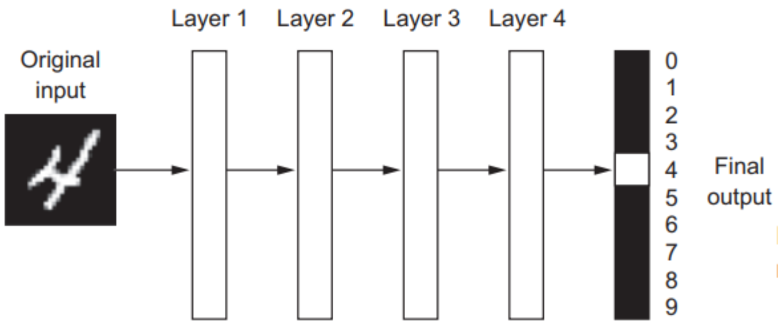


Figure 2.5 A deep neural network for digit classification. [8]

As you can see in Figure 2.6, the network transforms the digit image into representations that are increasingly different from the original image and increasingly informative about the final result. [8]

You can think of a deep network as a multistage information-distillation operation, where information goes through successive filters and comes out increasingly purified (that is, useful with regard to some task). [8]

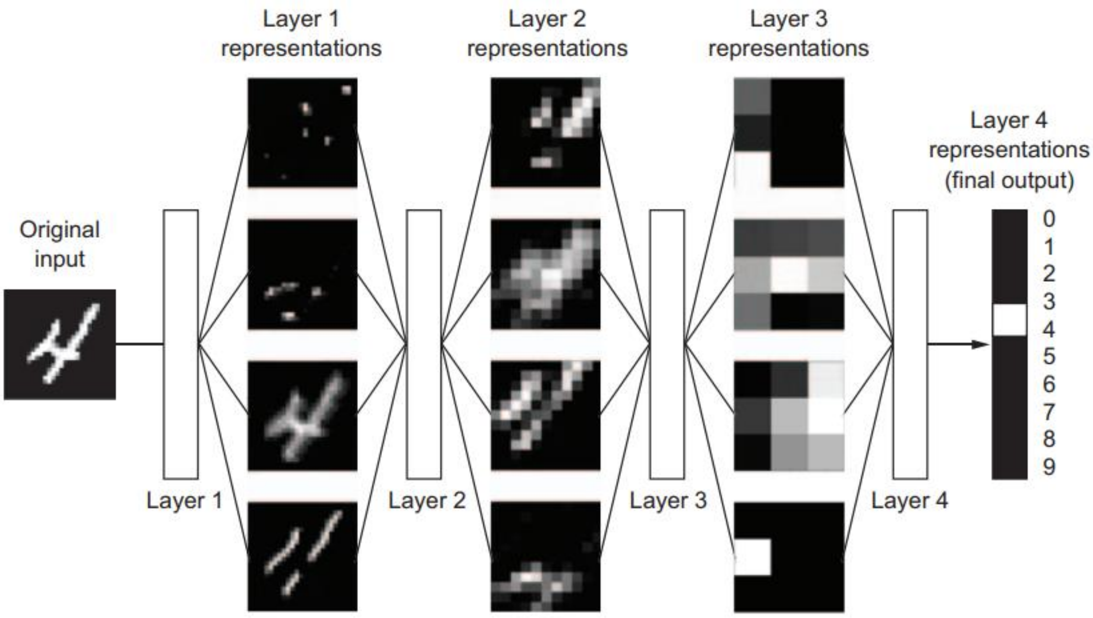


Figure 2.6 Deep representations learned by a digit-classification model. [8]

So that's what deep learning is, technically: a multistage way to learn data representations. It's a simple idea but, as it turns out, very simple mechanisms, sufficiently scaled, can end up looking like magic. [8]

4.4 Understanding how deep learning works

The specification of what a layer does to its input data is stored in the layer's weights, which in essence are a bunch of numbers. [8]

In technical terms, we'd say that the transformation implemented by a layer is parameterized by its weights (see Figure 2.7). (Weights are also sometimes called the parameters of a layer.) In this context, learning means finding a set of values for the weights of all layers in a network, such that the network will correctly map example inputs to their associated targets. But here's the thing: a deep neural network can contain tens of millions of parameters. Finding the correct value for all of them may seem like a daunting task, especially given that modifying the value of one parameter will affect the behavior of all the others!

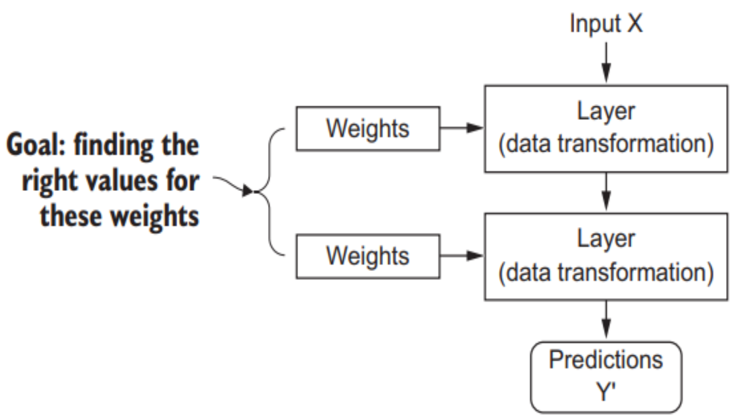


Figure 2.7 A neural network is parameterized by its weights. [8]

To control something, first, you need to be able to observe it. To control the output of a neural network, you need to be able to measure how far this output is from what you expected. This is the job of the loss function of the network, also called the objective function. The loss function takes the predictions of the network and the true target (what you wanted the network to output) and computes a distance score, capturing how well the network has done on this specific example (see Figure 2.8). [8]

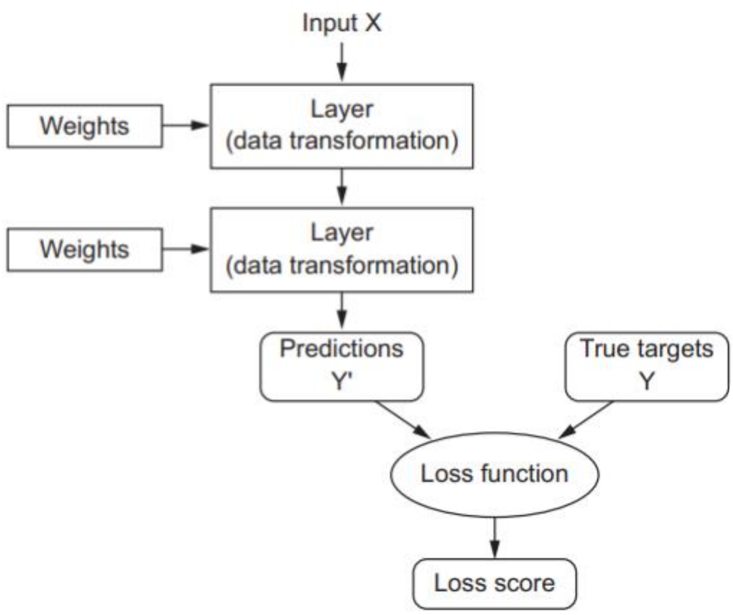


Figure 2.8 A loss function measures the quality of the network's output. [8]

The fundamental trick in deep learning is to use this score as a feedback signal to adjust the value of the weights a little, in a direction that will lower the loss score for the current example (see figure 2.9). This adjustment is the job of the optimizer, which implements what's called the Backpropagation algorithm: the central algorithm in deep learning. [8]

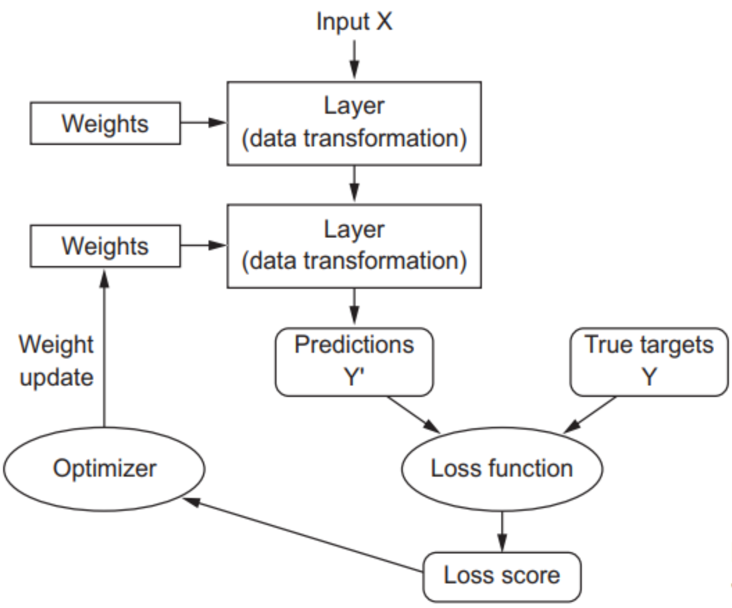


Figure 2.9 The loss score is used as a feedback signal to adjust the weights. [8]

Initially, the weights of the network are assigned random values, so the network merely implements a series of random transformations. Naturally, its output is far from what it should ideally be, and the loss score is accordingly very high. But with every example the network processes, the weights are adjusted a little in the correct direction, and the loss score decreases. This is the training loop, which, repeated a sufficient number of times (typically tens of iterations over thousands of examples), yields weight values that minimize the loss function. A network with a minimal loss is one for which the outputs are as close as they can be to the targets: a trained network. Once again, it's a simple mechanism that once scaled ends up looking like magic. [8]

4.5 Gradient Descent

Gradient Descent is a very generic optimization algorithm capable of finding optimal solutions to a wide range of problems. The general idea of Gradient Descent is to tweak parameters iteratively in order to minimize a cost function. [9]

Suppose you are lost in the mountains in a dense fog, you can only feel the slope of the ground below your feet. A good strategy to get the bottom of the valley quickly is to go downhill in the direction of the steepest slope. This is exactly what Gradient Descent does, it measures the local gradient of the error function with regards to the parameter vector θ, and it goes in the direction of descending gradient. [9]

Once the gradient is zero, you have reached a minimum concretely, you start by filling θ with random values (this is called random initialization), and then you improve it gradually, taking one baby step at a time, each step attempting to decrease the cost function (e.g., the MSE), until the algorithm converges to a minimum (see Figure 2.10). [9]

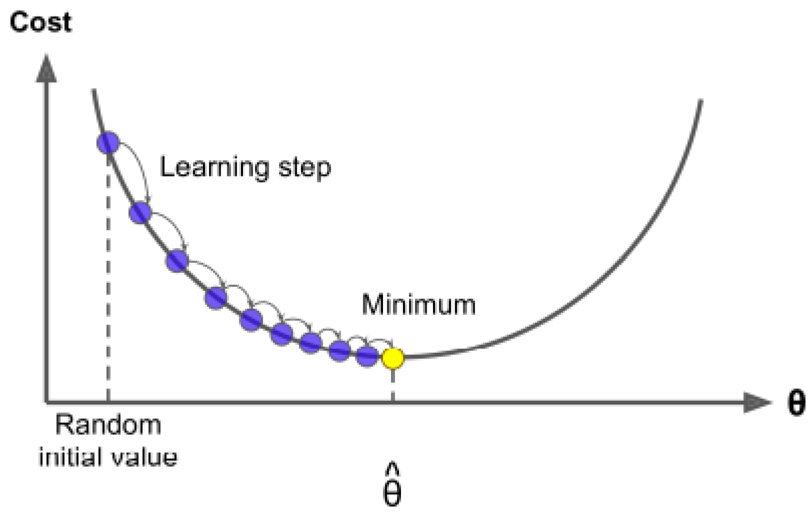


Figure 2.10 Gradient Descent. [9]

An important parameter in Gradient Descent is the size of the steps, determined by the learning rate hyper parameter. If the learning rate is too small, then the algorithm will have to go through many iterations to converge, which will take a long time (see Figure 2.11). [9]

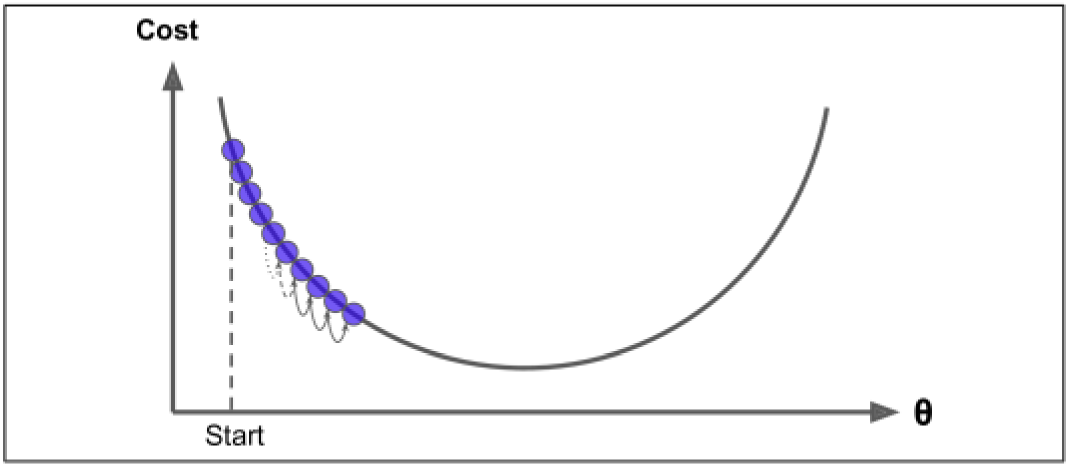


Figure 2.11 Learning rate too small. [9]

On the other hand, if the learning rate is too high, you might jump across the valley and end up on the other side, possibly even higher up than you were before. This might make the algorithm diverge, with larger and larger values, failing to find a good solution (Figure 2.12).

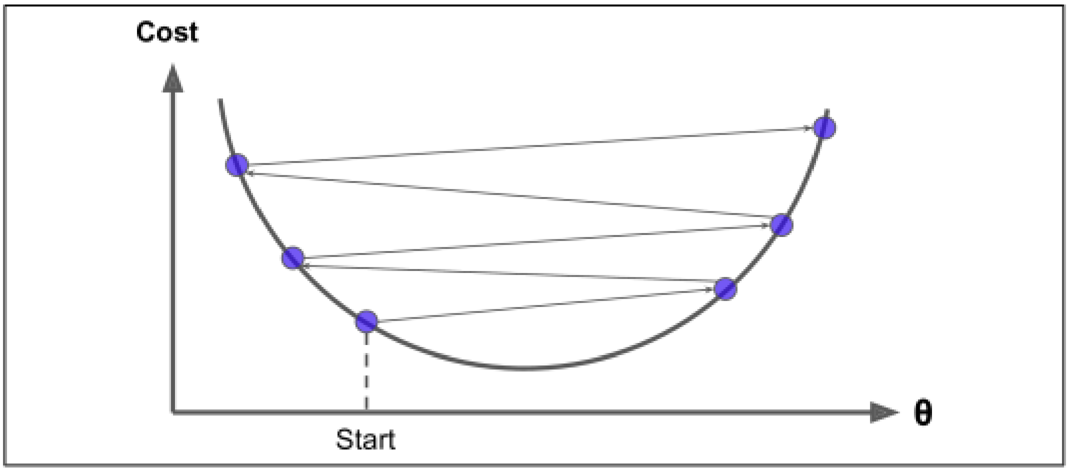


Figure 2.12 Learning rate too large. [9]

Finally, not all cost functions look like nice regular bowls. There may be holes, ridges, plateaus, and all sorts of irregular terrains, making convergence to the minimum very difficult. Figure 2.13 shows the two main challenges with Gradient Descent: if the random initialization starts the algorithm on the left, then it will converge to a local minimum, which is not as good as the global minimum. If it starts on the right, then it will take a very long time to cross the plateau, and if you stop too early, you will never reach the global minimum. [9]

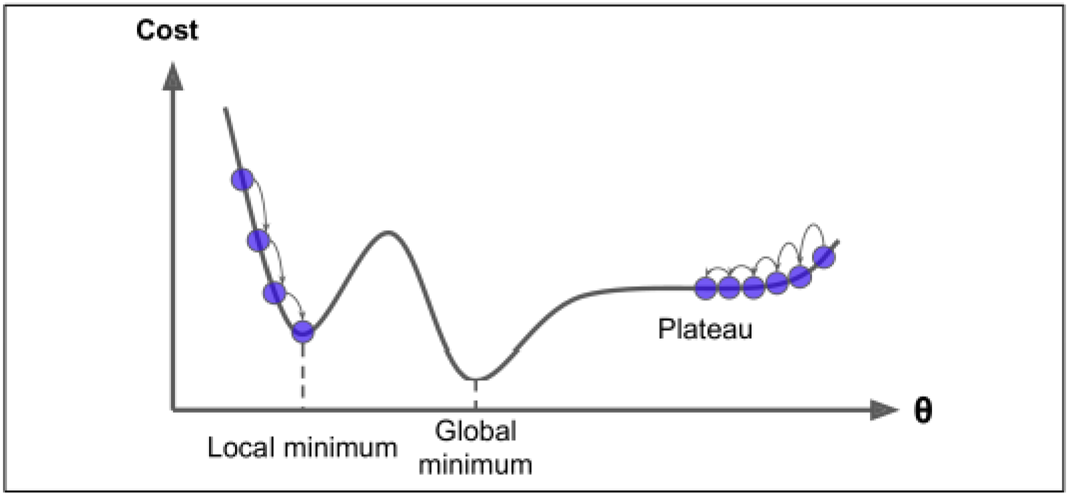


Figure 2.13 Gradient Descent pitfalls. [9]

Fortunately, the MSE cost function for a Linear Regression model happens to be a convex function, which means that if you pick any two points on the curve, the line segment joining them never crosses the curve. This implies that there are no local minima, just one global minimum. It is also a continuous function with a slope that never changes abruptly, these two facts have a great consequence: Gradient Descent is guaranteed to approach arbitrarily close the global minimum (if you wait long enough and if the learning rate is not too high). [9]

In fact, the cost function has the shape of a bowl, but it can be an elongated bowl if the features have very different scales. Figure 2.14 shows Gradient Descent on a training set where features 1 and 2 have the same scale (on the left), and on a training set where feature 1 has much smaller values than feature 2 (on the right). [9]

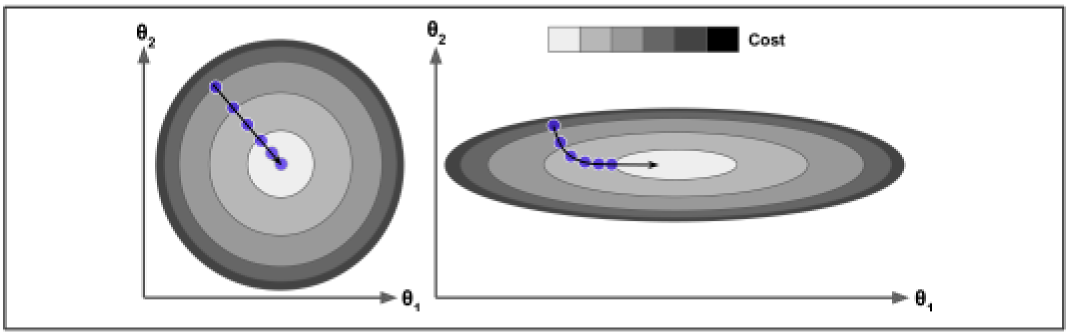


Figure 2.14 Gradient Descent with and without feature scaling. [9]

As you can see, on the left the Gradient Descent algorithm goes straight toward the minimum, thereby reaching it quickly, whereas on the right it first goes in a direction almost orthogonal to the direction of the global minimum, and it ends with a long march down an almost flat valley. It will eventually reach the minimum, but it will take a long time. [9]

This diagram also illustrates the fact that training a model means searching for a combination of model parameters that minimizes a cost function (over the training set). It is a search in the model's parameter space: the more parameters a model has, the more dimensions this space has, and the harder the search is: searching for a needle in a 300-dimensional haystack is much trickier than in three dimensions. Fortunately, since the cost function is convex in the case of Linear Regression, the needle is simply at the bottom of the bowl. [9]

4.6 The Backpropagation algorithm

In practice, a neural network function consists of many tensor operations chained together, each of which has a simple, known derivative. For instance, this is a network composed of three tensor operations, a, b, and c, with weight matrices W1, W2, and W3. f(W1, W2, W3) = a(W1, b(W2, c(W3))) Calculus tells us that such a chain of functions can be derived using the following identity, called the chain rule: f(g(x)) = f'(g(x)) * g'(x). [8]

Applying the chain rule to the computation of the gradient values of a neural network gives rise to an algorithm called Backpropagation (also sometimes called reverse-mode differentiation). Backpropagation starts with the final loss value and works backward from the top layers to the bottom layers, applying the chain rule to compute the contribution that each parameter had in the loss value. [8]

5. Convolutional Neural Networks

5.1 Architecture of ConvNet

ConvNet is not just a deep neural network that has many hidden layers. It is a deep network that imitates how the visual cortex of the brain processes and recognizes images. [10]

Therefore, even the experts of neural networks often have a hard time understanding this concept on their first encounter. That is how much ConvNet differs in concept and operation from the previous neural networks. This section briefly introduces the fundamental architecture of ConvNet. [10]

Basically, image recognition is the classification. For example, recognizing whether the image of a picture is a cat or a dog is the same as classifying the image into a cat or dog class. The same thing applies to the letter recognition; recognizing the letter from an image is the same as classifying the image into one of the letter classes. Therefore, the output layer of the ConvNet generally employs the multiclass classification neural network. However, directly using the original images for image recognition leads to poor results, regardless of the recognition method; the images should be processed to contrast the features, various techniques for image feature extraction have been developed. Before ConvNet, the feature extractor has been designed by experts of specific areas. Therefore, it required a significant amount of cost and time while it yielded an inconsistent level of performance. These feature extractors were independent of Machine Learning. Figure 2.15 illustrates this process. [10]

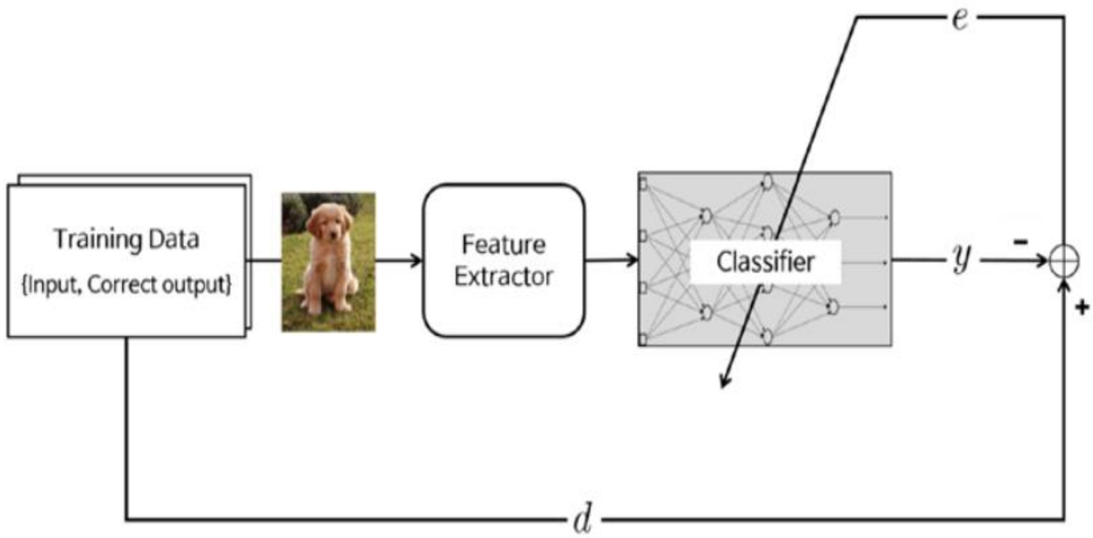


Figure 2.15 Feature extractors used to be independent of Machine Learning. [10]

ConvNet includes the feature extractor in the training process rather than designing it manually. The feature extractor of ConvNet is composed of special kinds of neural networks, of which the weights are determined via the training process. The fact that ConvNet turned the manual feature extraction design into the automated process is its primary feature and advantage. Figure 2.16 depicts the training concept of ConvNet. [10]

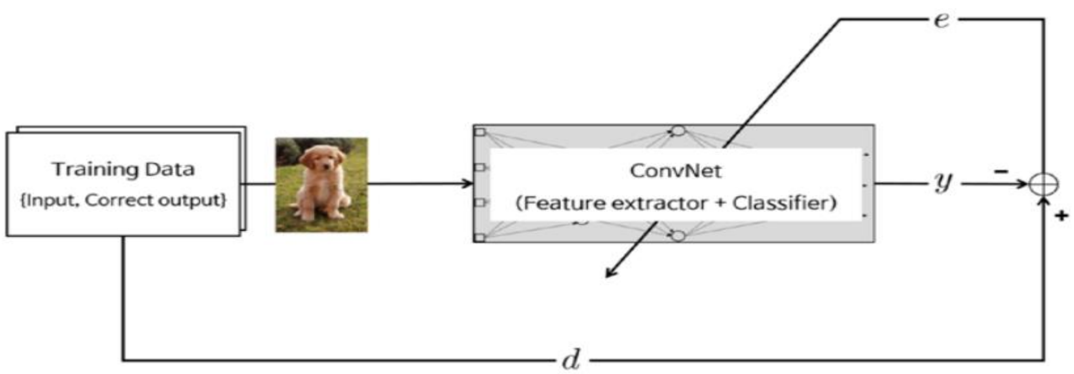


Figure 2.16 ConvNet's feature extractor is composed of special kinds of neural networks.[10]

ConvNet yields better image recognition when its feature extraction neural network is deeper (contains more layers), at the cost of difficulties in the training process, which had driven ConvNet to be impractical and forgotten for a while. Let's go a bit deeper. [10]

ConvNet consists of a neural network that extracts features of the input image and another neural network that classifies the feature image. (Figure 2.17) shows the typical architecture of ConvNet. [10]

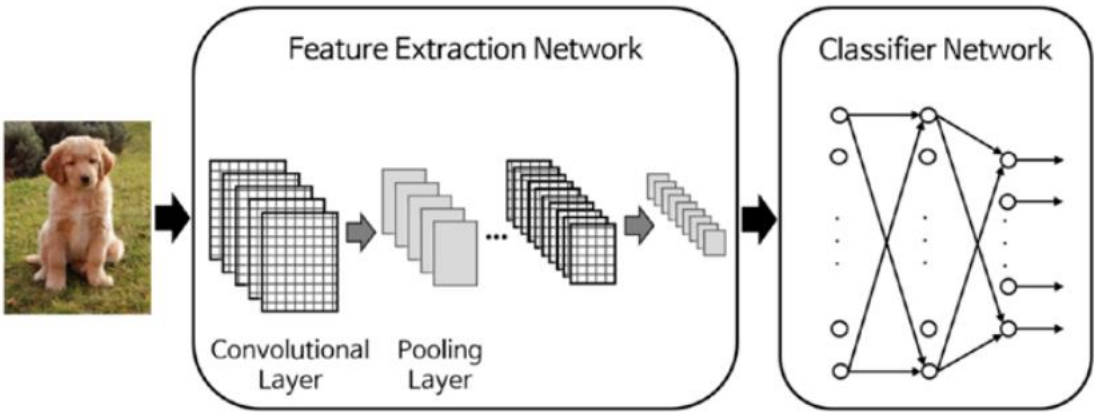


Figure 2.17 Typical architecture of ConvNet. [10]

The input image enters into the feature extraction network. The extracted feature signals enter the classification neural network. The classification neural network then operates based on the features of the image and generates the output. [10]

The feature extraction neural network consists of piles of the convolutional layer and pooling layer pairs. The convolution layer, as its name implies, converts the image using the convolution operation. It can be thought of as a collection of digital filters. [10]

The pooling layer combines the neighboring pixels into a single pixel. Therefore, the pooling layer reduces the dimension of the image. As the primary concern of ConvNet is the image, the operations of the convolution and pooling layers are conceptually in a two-dimensional plane. This is one of the differences between ConvNet and other neural networks. [10]

In summary, ConvNet consists of the serial connection of the feature extraction network and the classification network. Through the training process, the weights of both layers are determined. The feature extraction layer has piled pairs of the convolution and pooling layers. The convolution layer converts the images via the convolution operation, and the pooling layer reduces the dimension of the image. The classification network usually employs the ordinary multiclass classification neural network. [10]

5.2 Convolution Layer

The convolution layer generates new images called feature maps. The feature map accentuates the unique features of the original image. [10]

The convolution layer operates in a very different way compared to the other neural network layers. This layer does not employ connection weights and a weighted sum, Instead, it contains filters that convert images. We will call these filters convolution filters. [10]

The process of the inputting the image through the convolution filters yields the feature map. (Figure 2.18) shows the process of the convolution layer, where the circled * mark denotes the convolution operation, and the ϕ mark is the activation function. [10]

The square grayscale icons between these operators indicate the convolution filters. The convolution layer generates the same number of feature maps as the convolution filters. Therefore, for instance, if the convolution layer contains four filters, it will generate four feature maps. [10]

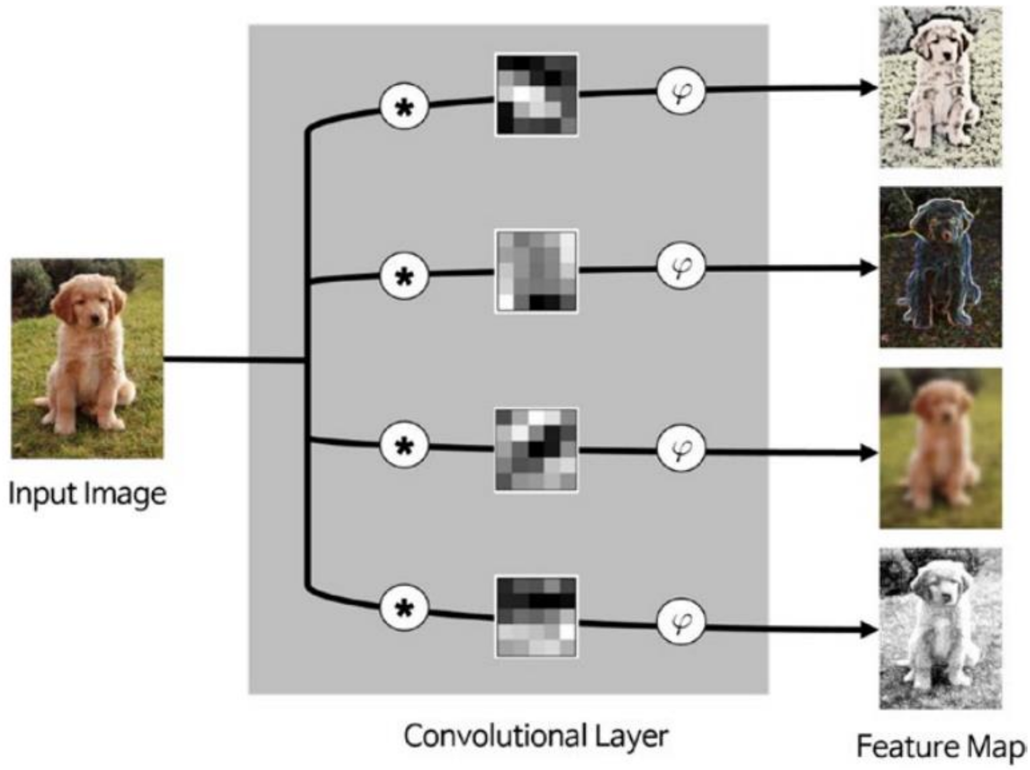


Figure 2.18 The convolution layer process. [10]

Let's further explore the details of the convolution filter. The filters of the convolution layer are two-dimensional matrices. As addressed in the previous section, the values of the filter matrix are determined through the training process. Therefore, these values are continuously trained throughout the training process. [10]

This aspect is similar to the updating process of the connection weights of the ordinary neural network. The convolution is a rather difficult operation to explain in text as it lies on the two-dimensional plane. However, its concept and calculation steps are simpler than they appear.

A simple example will help you understand how it works. Consider a 4x4 pixel image that is expressed as the matrix shown in Figure 2.19. We will generate a feature map via the convolution filter operation of this image. [10]

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

Figure 2.19 4x4 pixel image. [10]

We use the two convolution filters shown here. It should be noted that the filters of the actual ConvNet are determined through the training process and not by manual decision. [10]

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Let’s start with the first filter. The convolution operation begins at the upper-left corner of the submatrix which is the same size as the convolution filter (see Figure 2.20). [10]

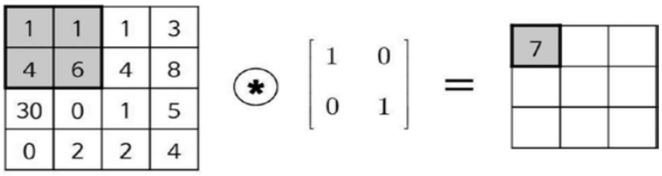


Figure 2.20 The convolution operation starts at the upper left corner. [10]

The convolution operation is the sum of the products of the elements that are located on the same positions of the two matrices. The result of 7 in Figure 2.20 is calculated as: [10]

$$(1\times1) + (1\times0) + (4\times0) + (6\times1) \;=\; 7$$

Another convolution operation is conducted for the next submatrix (see Figure 2.21).

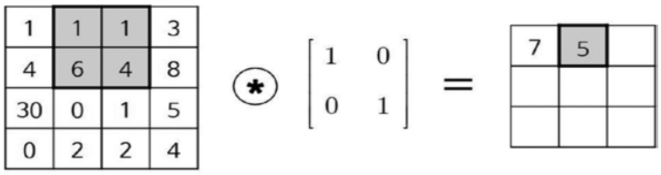


Figure 2.21 The second convolution operation. [10]

Once the top row is finished, the next row starts over from the left (see Figure 2.22).

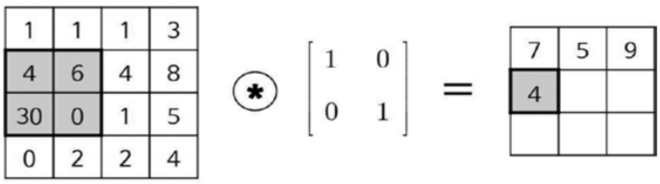


Figure 2.22 The convolution operation starts over from the left. [10]

It repeats the same process until the feature map of the given filter is produced, as shown in (Figure 2.23).

$$\begin{array}{|c|c|c|c|}\hline 1 & 1 & 1 & 3 \\ \hline 4 & 6 & 4 & 8 \\ \hline 30 & 0 & 1 & 5 \\ \hline 0 & 2 & 2 & 4 \\ \hline \end{array} \circledast \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{array}{|c|c|c|}\hline 7 & 5 & 9 \\ \hline 4 & 7 & 9 \\ \hline 32 & 2 & 5 \\ \hline \end{array}$$

Figure 2.23 The feature map of the given filter has been completed. [10]

In summary, the convolution layer operates the convolution filters on the input image and produces the feature maps. The features that are extracted in the convolution layer are determined by the trained convolution filters. Therefore, the features that the convolution layer extracts vary depending on which convolution filter is used. [10]

5.3 Pooling Layer

The pooling layer reduces the size of the image, as it combines neighboring pixels of a certain area of the image into a single representative value. Pooling is a typical technique that many other image processing schemes have already been employing. [10]

In order to conduct the operations in the pooling layer, we should determine how to select the pooling pixels from the image and how to set the representative value. [10]

The neighboring pixels are usually selected from the square matrix, and the number of pixels that are combined differs from problem to problem. The representative value is usually set as the mean or maximum of the selected pixels. [10]

The operation of the pooling layer is surprisingly simple. As it is a two-dimensional operation, and an explanation in text may lead to more confusion, let's go through an example. Consider the 4x4 pixel input image, which is expressed by the matrix shown in (Figure 2.24). [10]

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

Figure 2.24 The 4x4 pixel input image. [10]

We combine the pixels of the input image into a 2*2 matrix without overlapping the elements. Once the input image passes through the pooling layer, it shrinks into a 2*2 pixel image. (Figure 2.25) shows the resultant cases of pooling using the mean pooling and max pooling.

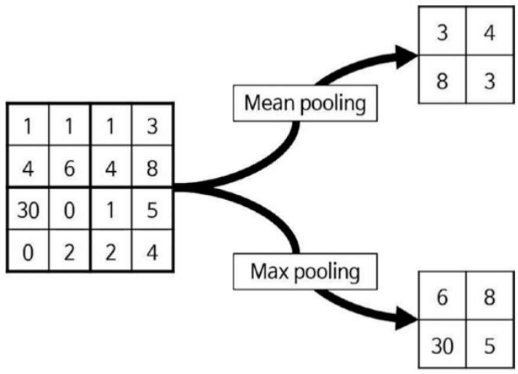


Figure 2.25 The result of pooling operation using two different methods. [10]

The pooling layer compensates for eccentric and tilted objects to some extent. For example, the pooling layer can improve the recognition of a cat, which may be off-center in the input image. In addition, as the pooling process reduces the image size, it is highly beneficial for relieving the computational load and preventing overfitting. [10]

5.4 Sequence processing with convnets

The same properties that make convnets excel at computer vision also make them highly relevant to sequence processing. Time can be treated as a spatial dimension, like the height or width of a 2D image. Such 1D convnets can be competitive with RNNs (A different deep learning technique) on certain sequence-processing problems, usually at a considerably cheaper computational cost. [8]

Recently, 1D convnets, typically used with dilated kernels, have been used with great success for audio generation and machine translation. In addition to these specific successes, it has long been known that small 1D convnets can offer a fast alternative to RNNs for simple tasks such as text classification and time series forecasting. [8]

5.5 ConvNets Building Blocks for sequence data

There are many types of layers used to build Convolutional Neural Networks, but the ones you are most likely to encounter include.

5.5.1 1D convolution for sequence data

The convolution layers introduced previously were 2D convolutions, extracting 2D patches from image tensors and applying an identical transformation to every patch. In the same way, we can use 1D convolutions, extracting local 1D patches (subsequences) from sequences (see figure 2.26). [8]

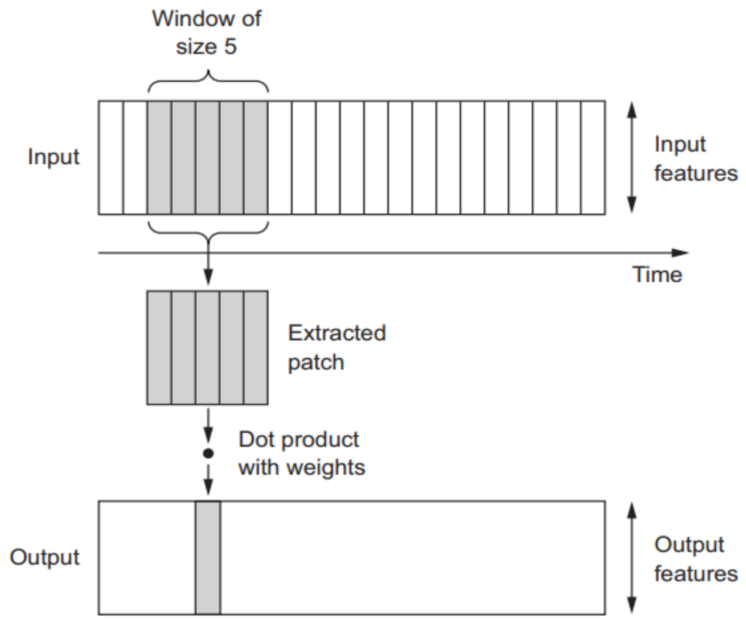


Figure 2.26 How 1D convolution works: each output timestep is obtained from a temporal patch in the input sequence. [8]

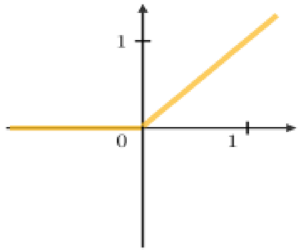
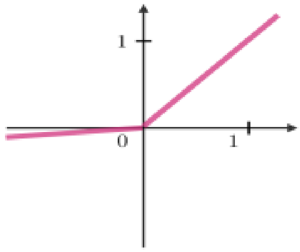
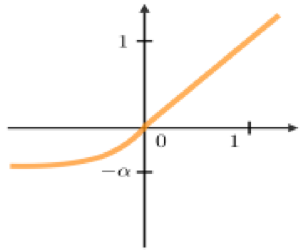
Such 1D-convolution layers can recognize local patterns in a sequence. Because the same input transformation is performed on every patch, a pattern learned at a certain position in a sentence can later be recognized at a different position, making 1D convnets translation-invariant (for temporal translations).[8]

5.5.2 Activation layers

After each CONV layer in a CNN, we apply a nonlinear activation function, such as ReLU, ELU, or any of the other Leaky ReLU variants (Table 2.2).

We typically denote activation layers as RELU in network diagrams as since ReLU activations are most commonly used, we may also simply state ACT, in either case, we are making it clear that an activation function is being applied inside the network architecture.

Table 2.2 ReLU, Leaky ReLU and ELU activation functions. [11]

ReLU	Leaky ReLU	ELU
$g(z) = \max(0,z)$	$g(z) = \max(\epsilon z,z)$ with $\epsilon \ll 1$	$g(z) = \max(\alpha(e^z - 1),z)$ with $\alpha \ll 1$
		
Non-linearity complexities biologically interpretable	Addresses dying ReLU issue for negative values	Differentiable everywhere

Activation layers are not technically "layers" (due to the fact that no parameters/weights are learned inside an activation layer) and are sometimes omitted from network architecture diagrams as it's assumed that an activation immediately follows a convolution. [11]

5.5.3 Pooling layer for sequence data

The 2D pooling operation has a 1D equivalent: extracting 1D patches (subsequences) from an input and outputting the maximum value (max pooling) or average value (average pooling). Just as with 2D convnets, this is used for reducing the length of 1D inputs (subsampling). [8]

5.5.4 Fully connected layers (Dense)

Neurons in dense layers are fully-connected to all activations in the previous layer, as is the standard for neural networks. FC layers are always placed at the end of the network (i.e., we don't apply a CONV layer, then an FC layer, followed by another CONV) layer It's common to use one or two FC layers prior to applying the softmax classifier, as the following (simplified) architecture demonstrates:

INPUT => CONV => RELU => POOL => CONV => RELU => POOL => FC => FC

Here we apply two fully-connected layers before our (implied) softmax classifier which will compute our final output probabilities for each class.

5.5.5 Dropout layer

Before the fully-connected layer, there is frequently a dropout layer During the training of the convolution neural network, the dropout layer will periodically disconnect some neurons from the network based on a probability, which lowers joint adaptability between neuron nodes, reduces overfitting, and improves the network's generalization capacity.

… CONV => RELU => POOL => FC => DO => FC => DO => FC

5.5.6 Softmax

The softmax function is a function that turns a vector of K real values into a vector of K real values that sum to 1. The input values can be positive, negative, zero, or greater than one, but the softmax transforms them into values between 0 and 1, so that they can be interpreted as probabilities. If one of the inputs is small or negative, the softmax turns it into a small probability, and if an input is large, then it turns it into a large probability, but it will always remain between 0 and 1. [12]

The softmax function is sometimes called the softargmax function, or multi-class logistic regression. This is because the softmax is a generalization of logistic regression that can be used for multi-class classification, and its formula is very similar to the sigmoid function which is used for logistic regression. The softmax function can be used in a classifier only when the classes are mutually exclusive. [12]

5.6 Training algorithm for sequence data

The convolution neural network's (CNN) training process is a backward propagation gradient descent approach. The loss function, which is the deviations of the output vector and the predicted output vector, is used to estimate the network hyperparameters. The convolution kernel parameter W, the sampling weight coefficient of the pooling layer, the network weight w of the fully-connected layer, and the offset b of each layer are all hyperparameters.

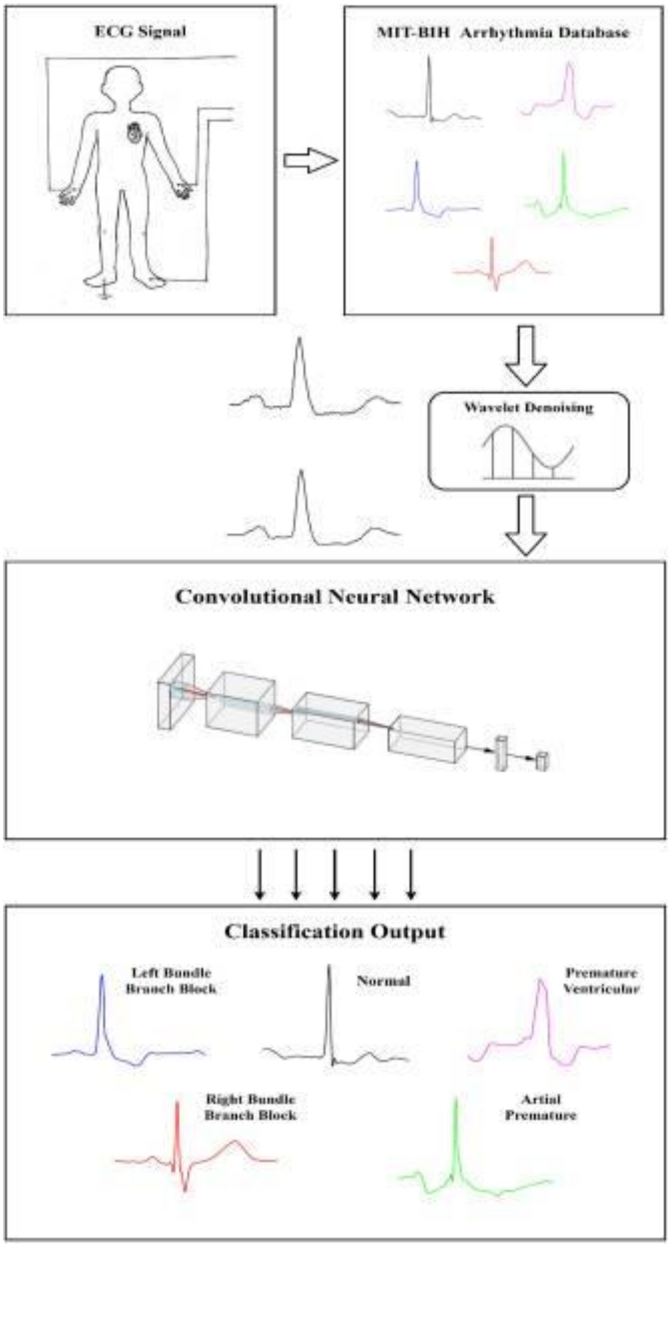
Forward propagation and reverse propagation are the two steps of convolution neural network training. The training data is fed into the neural network, and the output vectors of the middle and output layers are calculated at the forward propagation stage. The output vectors of the output layer are compared to the expected output vectors in the reverse propagation stage, and the loss function is determined with regard to the network's weights.

The loss is propagated back to the beginning layers (in reverse) by updating the weights for each neuron in each layer using the gradient descent algorithm. Gradient descent is made up of two steps: calculating the gradients of the loss function using chain rules, and then updating weight in the opposite or reverse direction of the gradient of the loss function, which is different from the forward calculation of the loss function. In each hidden layer, a cost function is calculated for the neuron output in order to continuously tune the network hyperparameters. After numerous cycles, the network reaches the set error and stops training.

6. Conclusion

In this chapter, we have demonstrated what is intelligence, and what we are meaning when we say intelligence. Also, we have defined what is artificial intelligence and the different part of artificial intelligence, such as machine learning and deep learning. We also understand what is deep leaning, what makes a deep learning model as we went in exploring in-depth the role of deep learning in multiple fields, such as computer vision and sequence data. For sequence data we maked clear understanding of the parts (layers and blocks) of conventional neural network. A brief explanation of the world of deep learning was given in general as it stills a wide and broad field to discuss.

ECG Classification Using CNN



1. Introduction

We discussed and maked a brief understanding of ECG signals as well as the deep learning world and the main role it plays in sequence data treatment in the first and second chapters. In this chapter, we will see in action how we can use the MIT-BIH ECG database to apply a one-dimensional convolution neural network (1D – CNN) to classify five different heartbeats:

- Normal (NOR),
- Left Bundle Branch Block (LBBB),
- Right Bundle Branch Block (RBBB),
- Atrial Premature (AP), and
- Premature Ventricular Contraction (PVC).

Several enhancements and preprocessings to the ECG signal have been made, followed by several changes to our CNN model, and we will detail our training progress and results in the last two sections.

2. ECG Data Processing

2.1 ECG Dataset

The MIT-BIH database, an ECG database provided by the Massachusetts Institute of Technology and based on international standards and annotated information by multiple experts is used in this study. [13]

The MIT-BIH database has been frequently used by the academic community in research for the detection and classification of arrhythmic heartbeats. [13]

The MIT-BIH database contains 48 ECG recordings, each recording time is 30 min, the sampling frequency is 360 Hz, and each ECG record is composed of two leads. [13]

MIT-BIH database can make adjustments and corrections based on the information annotated by experts and optimization algorithms. Furthermore, it learns from existing solutions for self-optimization. [13]

2.2 Preprocessing

Power frequency interference, baseline drift, and EMG interference are all common sources of interference in ECG readings obtained in a clinical setting. To improve the accuracy of the classification, the raw data must be de-noised. [14] [15] [16]

In the field of ECG denoising, bandpass filters, low-pass filters, and wavelet transforms are commonly utilized. The wavelet transform approach is utilized to preprocess the ECG signal in this experiment. The wavelet transform decomposes non-stationary signals into scale signals with different frequency bands. The wavelet function is Sym4 from the Symlet wavelet function family, and the filter employs an adaptive threshold filtering technique. [17] [18]

We just conduct minimal filtering on the signal because the convolutional neural network has the ability to automatically extract features from the interior of the signal, which can improve the network's generalization and reduce signal distortion. The ECG signal before and after filtering is shown in (Figure 3.1). [19]

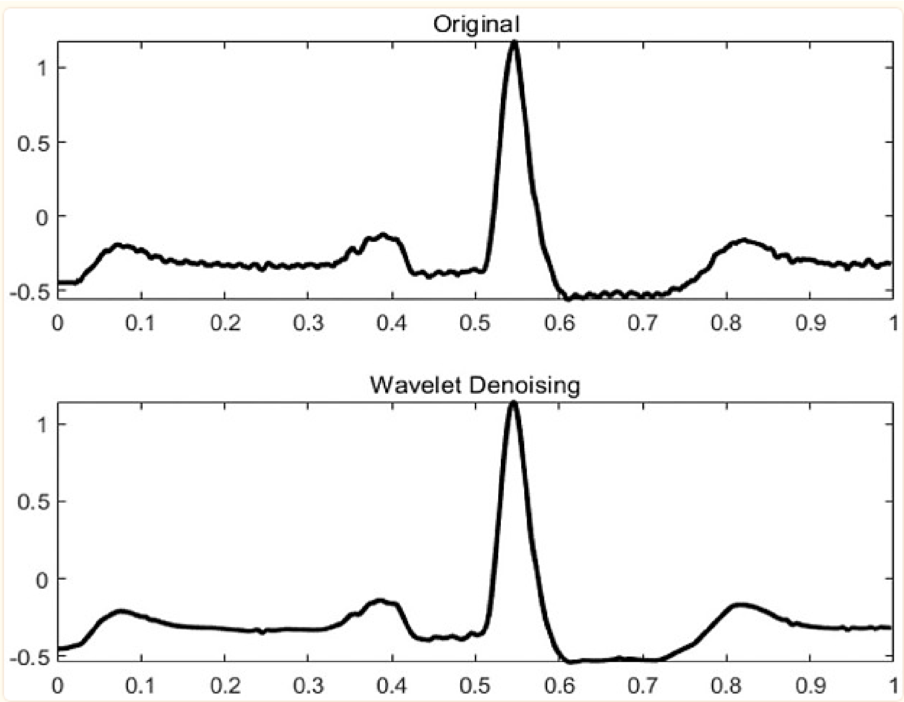


Figure 3.1 ECG signal before and after filtering.

2.3 Data Segmentation

Each heartbeat in the MIT-BIH dataset is annotated with an illness. This study classifies five heartbeats: normal (NOR), left bundle branch block (LBBB), and right bundle branch block (RBBB), as well as atrial premature beats (AP) and premature ventricular beats (PVC).

To begin, the Pan-Tompkins algorithm is used to detect R-peak (Pan and Tompkins, 1985). For the collection of the five types of ECG beats, The following beats table is used, it shows the number of beats of each type in each record. [20]

Table 3.1 Table of beats type in MIT – BIH records.

Record	Beat type				
	NOR	LBBB	RBBB	AP	PVC
100	2239	-	-	33	1
101	1860	-	-	3	-
102	99	-	-	-	4
103	2082	-	-	2	-
104	163	-	-	-	2
105	2526	-	-	-	41
106	1507	-	-	-	520
107	-	-	-	-	59
108	1739	-	-	4	17
109	-	2492	-	-	38
111	-	2123	-	-	1
112	2537	-	-	2	-
113	1789	-	-	-	-
114	1820	-	-	10	43
115	1953	-	-	-	-
116	2302	-	-	1	109
117	1534	-	-	1	-
118	-	-	2166	96	16

119	1543	-	-	-	444
121	1861	-	-	1	1
122	2476	-	-	-	-
123	1515	-	-	-	3
124	-	-	1531	2	47
200	1743	-	-	30	826
201	1625	-	-	30	198
202	2061	-	-	36	19
203	2529	-	-	-	444
205	2571	-	-	3	71
207	-	1457	86	107	105
208	1586	-	-	-	992
209	2621	-	-	383	1
210	2423	-	-	-	194
212	923	-	1825	-	-
213	2641	-	-	25	220
214	-	2003	-	-	256
215	3195	-	-	3	164
217	244	-	-	-	162
219	2082	-	-	7	64
220	1954	-	-	94	-
221	2031	-	-	-	396
222	2062	-	-	208	-
223	2029	-	-	72	473
228	1688	-	-	3	362
230	2255	-	-	-	1
231	314	-	1254	1	2
232	-	-	397	1382	-
233	2230	-	-	7	831
234	2700	-	-	-	3

After the segmentation of the records into separated beats, segments of 360 samples and centered on the R-peaks are taken. The approach chooses a single lead from the dataset, and all segments are normalized in the mean and the variances using the Z-score method (zero-mean and a unit variance). [21]

2.4 Data Enhancement

The unbalanced training set has an impact on the convolutional neural network's feature learning, lowering recognition accuracy. [22]

The MIT-BIH dataset is used to choose recordings that comprise the majority of these five heartbeats for this experiment. After denoising and segmenting, we have take a closed numbers of each type among the five types. [22]

Table 3.2 Dataset size of the different types

Class	Type	Number of beats
C1	NOR	8062
C2	LBBB	6612
C3	RBBB	7163
C4	AP	2496
C5	PVC	6719
Total data		**31052**

3. The Architecture of the convolution neural network

To analyze a one-dimensional time series (sequence data) with uniform interval sampling, we used a one-dimensional (12 layer) CNN. The used CNN network employs the average-pooling layer instead of the max-pooling layer. [23] [24]

The average-pooling layer can keep the overall feature of the input data, which is useful for categorizing heartbeats. In comparison to the benchmark CNN network, the used CNN network has one more alternating convolution and pooling layer. The used CNN network architecture, which includes 8 alternating convolutions and average-pooling layers, is summarized in Table 3.2. As seen in the diagram, they are followed by a dropout layer and two fully-connected layers (Dense) (Figure 3.2). [25]

Table 3.2 A summary table of the proposed CNN model for this work.

Layers	Type	Output	Kernel size	Stride
Layer 1	Convolution	360*16	1*13	1
Layer 2	Average-Pooling	179*16	1*3	2
Layer 3	Convolution	179*32	1*15	1
Layer 4	Average-Pooling	89*32	1*3	2
Layer 5	Convolution	89*64	1*17	1
Layer 6	Average-Pooling	44*64	1*3	2
Layer 7	Convolution	44*128	1*19	1
Layer 8	Average-Pooling	21*128	1*3	2
Layer 9	Dropout	21*128	-	-
Layer 10	Fully-connected	1*35	-	-
Layer 11	Fully-connected	1*5	-	-
Layer 12	SoftMax	1*5	-	-

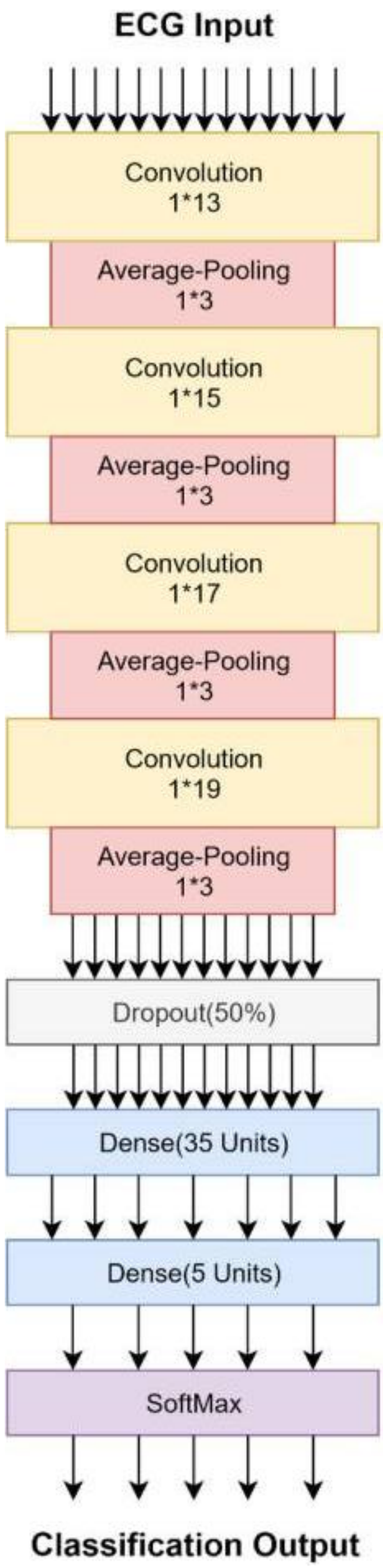


Figure 3.3 The architecture for the proposed CNN model.

4. Experiment setup

We ran a total of 30 epochs with a batch size of 36 in this experiment. The learn rate drop factor, learn rate drop period, and learning rate parameters have been set to 0.1, 20, and 0.001 respectively. All of the proposed classifiers' parameters in Table 3.3 are chosen for use based on the best results after numerous testing.

"*MaxEpoch*" is the maximum number of epochs to employ for training, as shown in Table 3.3. The size of the mini-batch to employ for each training iteration is specified by "*MiniBatchSize*." A mini-batch is a subset of the training set used to evaluate the loss function's gradient and update the weights. The first learning rate for training is called "*InitialLearnRate*." The "*LearnRateDropPeriod*" is the number of epochs between learning rate adjustments during training. The multiplicative factor by which the learning rate reduces during training is called "*LearnRateDropFactor*". We employed the Adam (adaptive moment estimation) optimizer, which we called "Optimizer."

Table 3.3 Selected best used parameters

Network	Parameters
1D-CNN	Max Epoch = 30, Mini Batch Size = 36 Optimizer Adam Learning Rate Drop Factor = 0.1, Learn Rate Drop = 20, Initial Learn Rate = 10e-3

5. Evaluation index

In order to evaluate and compare the classification effects for our model more accurately, this experiment uses a confusion matrix, accuracy (*Acc*), sensitivity (*Sen*), specificity (*Spe*), and positive prediction rate (*Ppr*). [25]

Among them, the accuracy rate represents the ability to detect the real situation of the sample; the sensitivity represents the ability to distinguish various diseases; the specificity represents the ability to detect negatively for a certain disease; the positive prediction represents the rate that proportion of positive identifications is actually correct. The corresponding expressions are formula (1–4)

Where TP stands for True Positive, TN stands for True Negative, FP stands for False Positive, FN stands for False Negative, FP stands for False Positive. [26]

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$Sen = \frac{TP}{TP+FN} \tag{2}$$

$$Spe = \frac{TN}{TN+FP} \tag{3}$$

$$Ppr = \frac{TP}{TP+FP} \tag{4}$$

6. Results and discussion

The used model of CNN algorithm is trained on a PC with AMD A9-9420 RADEON R5 with 8GB of RAM the model is trained on a single CPU and it takes 27 min 28 sec long to complete the training process. The implementation of the algorithm was made using the *MATLAB Deep Learning Toolbox*. The figure below (Figure 3.3) depicts the training process of the used model in our work. We can see that the training section has two curves, one for our model's training and validation accuracy and the other for the calculated loss function.

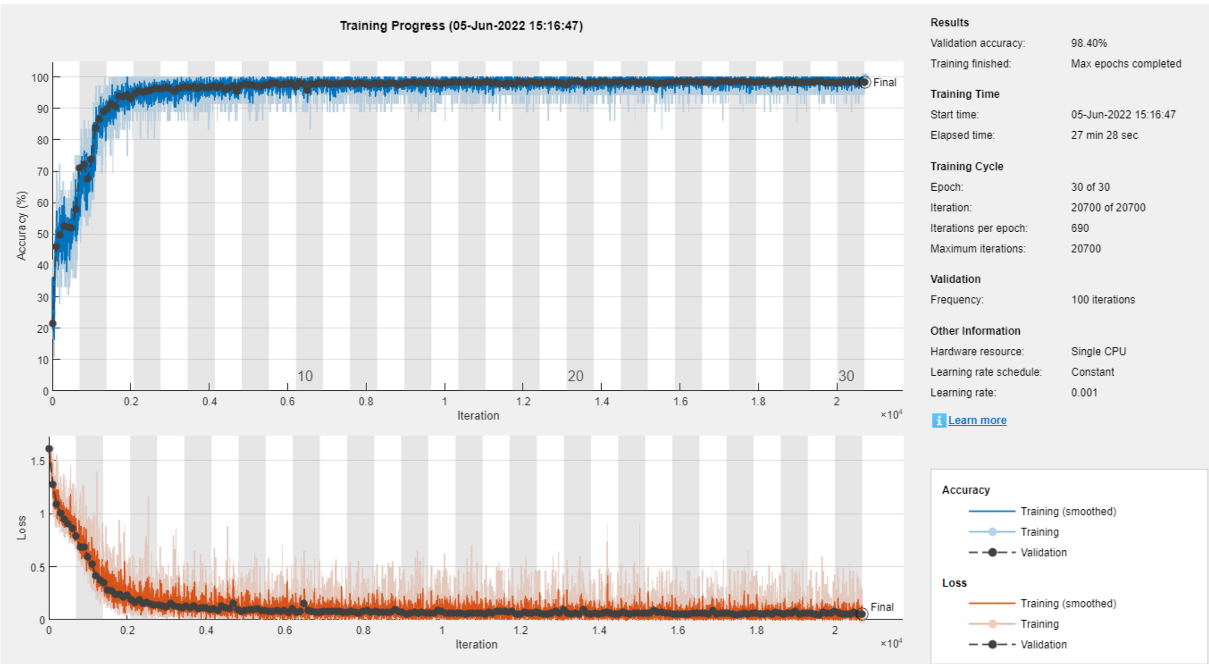


Figure 3.3. The training process

On the right side of the diagram, we can see the training time of the model as well as the overall validation accuracy of the training. We notice also the validation frequency and other resources connected to the model in addition to the training cycle, which contains the number of epochs and iteration number as well as iteration per epochs and maximum iteration.

Our validation accuracy achieved 98 percent, as shown in the diagram below, which allows us to clearly assess our training process. The validation accuracy is used to determine how well the training set performs. To explain further, the validation test is unseen training data that can be used to test the training data. As we discussed in the previous chapter, the loss function's job is to calculate the actual output from the expected output. In our diagram, we can see that our loss function has descended near zero 0, indicating that our training model is performing perfectly.

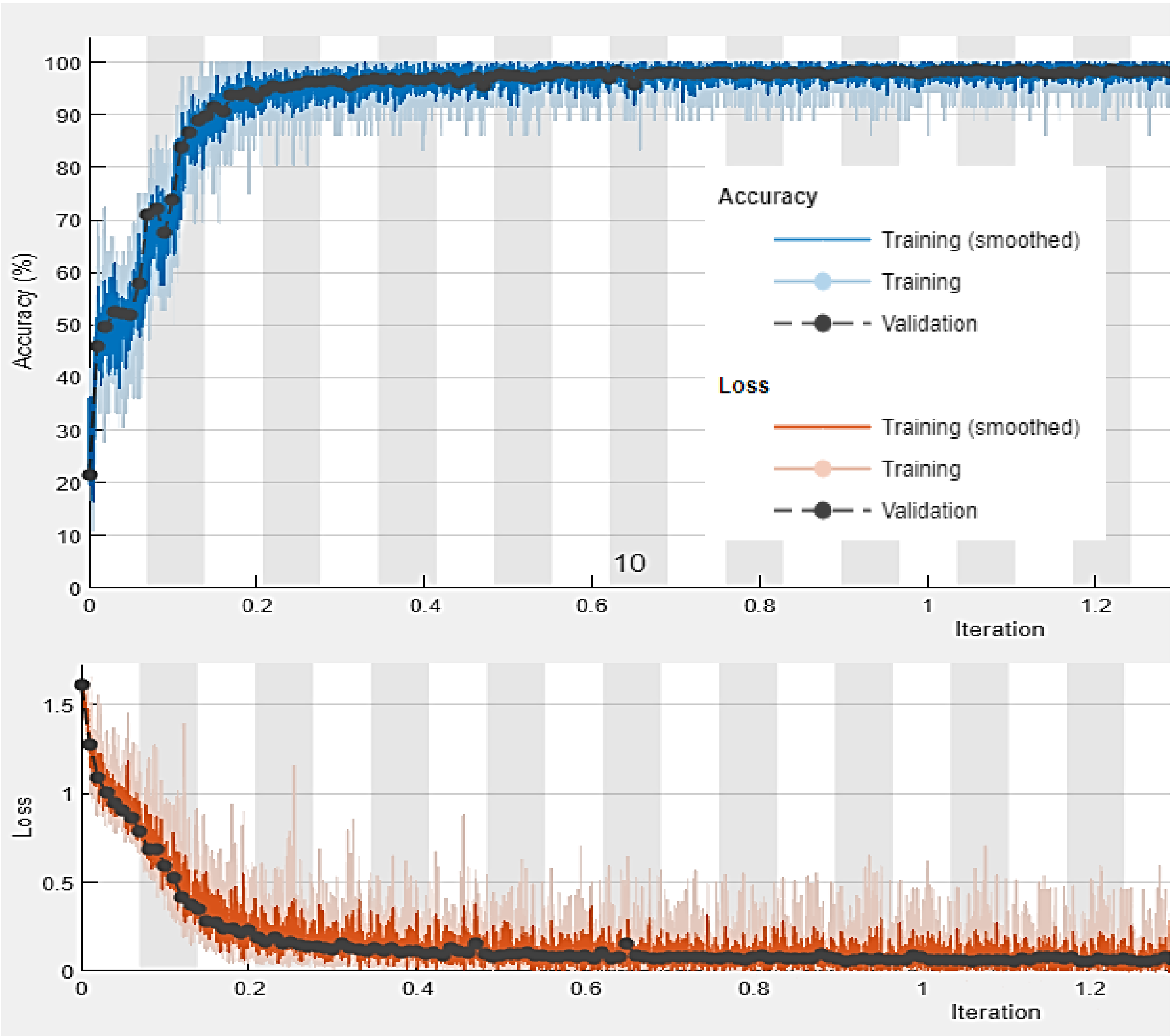


Figure 3.4 Zoom-in of training Progress

7. Examining the confusion matrix

The classification results of the CNN network are shown in the confusion matrix below. The overall classification accuracy rate of the five micro-class classifications of heartbeats is 98.5 percent, and the accuracy of each category is over 90 percent, demonstrating the model's effectiveness (Figure 3.5).

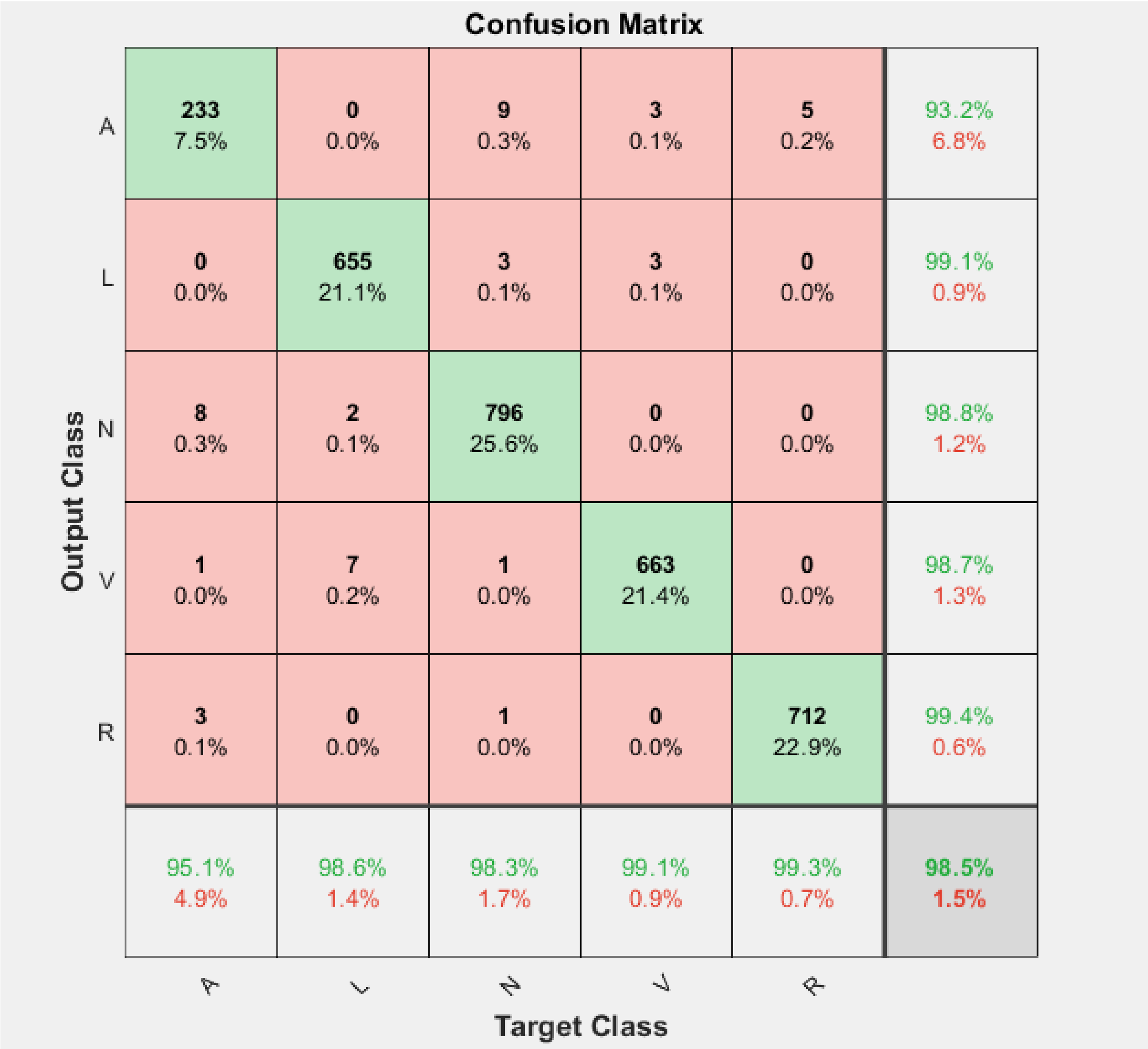


Figure 3.5 The confusion matrix

The confusion matrix consists of five micro-classes, each for a specific disease, as shown in the matrix. We have a target class, which is the classification we expect from our model, and an output class, which is the classification our model outputted.

The green squares are those that our model correctly predicted witch called correct prediction or true classes and matched with the correct class. While the red ones are the wrong output classes and are called Incorrect predictions. For example, class A, which has a true positive (TP) of 7.5 percent from the overall accuracy with 233 correct classifications. On the other hand, in the output class (FN), we have 8 cases in N class, 1 cases in V class, and 3 cases in R class, which our model misclassify them (they are from class A but classified in other classes). On the other hand, in the target class (FP), we have 9 cases in N class, 3 cases in V class, and 5 cases in R class, which are classified as type A but they belong to other types.

To acquire the best results and accuracy and compare them to the original result, we evaluated our model on four different parameters set. The first has normalization and denoising (the original result), the second has normalization but no denoising, the third has denoising but no normalization, and the fourth has neither normalization nor denoising. The results are shown in the table below, and each one indicates the accuracy of model. The given accuracy is the average of 10 runs in each case.

Table 3.5. Accuracy of the four cases

Preprocessing	Normalization	Without Normalization
Denoising	98.50%	97.93%
Without Denoising	98.20%	97.86%

The original result has a 98.50 percent accuracy rate. This illustrates that normalization and denoising have a positive effect when compared to the other results, which show that the model with no normalization and denoising had the lowest result with a minor dropout. The overall comparison shows that using both denoising and normalization enhances model learning and and prediction, so they help the CNN model to recognize more patterns and to good assist a cardiologist in his work.

8. Conclusion

In this experiment, we applied a 1D convolution neural network to an ECG signals taken from the MIT-BIH database, several improvements have been made to the ECG signal waveform. The architecture of the used CNN in the experiments showed a remarkable result of the accuracy which demonstrate the effectiveness of the convolution neural network (CNN) in time series data classification problems.

Concluion

Conclusion

In today's environment, cardiovascular disease is a big health issue. The ECG is extremely important in the early detection of cardiac arrhythmia. Unfortunately, specialist medical resources are scarce, making visual identification of the ECG signal difficult and time-consuming. Our project focuses on specific micro-classes in the MIT-BIH Arrhythmia database, such as Normal, Left Bundle Branch Block, Right Bundle Branch Block, Atrial Premature Beats, and Premature Ventricular Beats.

It's worth noting that the used CNN network has a greater level of accuracy and reliability on the classification of the micro-classes of the Arrhythmia dataset. The cited CNN network achieves an amazing overall classification accuracy of 98.50 percent and a positive prediction rate of 90 percent.

The benefits of the CNN network have been demonstrated. It has the capacity to process the non-filtered dataset with its potential anti-noise characteristics effectively. One potential drawback of the this technique is that training the network is computationally intensive, since deep learning is frequently ascribed to the huge size data required for training. In terms of future studies, it would be fascinating to apply this experiment to other different deep learning techniques and compare them with the existing results.

References

[1] Tony Curran and Gill Sheppard, “Cardiology Self Learning Package, Module 1: Anatomy and Physiology of the Heart,” Health P1 Course at Kenya Medical Training Centre (KMTC), October 2011.

[2] Aehlert, Barbara J. *Pocket Guide for ECGs Made Easy-E-Book*. Elsevier Health Sciences, 2017.

[3] Hampton, John, and Joanna Hampton. *The ECG made easy e-book*. Elsevier Health Sciences, 2019.

[4] Stroobandt, Roland X., S. Serge Barold, and Alfons F. Sinnaeve. ECG from Basics to Essentials: Step by Step. John Wiley & Sons, 2016.

[5] Williams, Lippincott, and E. C. G. Wilkins. "Interpretation made incredibly easy." (2011).

[6] Stefan stavrev, “Machine Learning God” CS1 course in ENSTA ParisTech First edition 2017.

[7] Lauscher, Anne. Life 3.0: Being human in the age of artificial intelligence. 2019.

[8] Chollet, Francois. Deep learning with Python. Simon and Schuster, 2021.

[9] Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc.", 2019..

[10] Kim, Phil. "Matlab deep learning." *With machine learning, neural networks and artificial intelligence* 130.21 (2017).

[11] Afshine Amidi and Shervine Amidi, “Super VIP Cheatsheet: Machine Learning” Course CS 229– Machine Learning, https://stanford.edu/~shervine

[12] https://deepai.org/machine-learning-glossary-and-terms/softmax-layer

[13] Moody, G. B., and Mark, R. G. (2001). The impact of the MIT-BIH arrhythmia database. IEEE Engin. Med. Biol. Mag. 20, 45–50. doi: 10.1109/51.932724

[14] Ahlstrom, M., and Tompkins, W. (1985). Digital filters for real-time ECG signal processing using microprocessors. IEEE Trans. Biomed. Engin. 9, 708–713. doi: 10.1109/TBME.1985.325589

[15] Bazi Y., Alajlan N., AlHichri H., Malek S. (2013). Domain adaptation methods for ECG classification, in 2013 International Conference on Computer Medical Applications (ICCMA) (Sousse: IEEE;), 1–4. 10.1109/ICCMA.2013.6506156 [CrossRef] [Google Scholar]

[16] Wang J., Ye Y., Pan X., Gao X. (2015). Parallel-type fractional zero-phase filtering for ECG signal denoising. Biomed. Sign. Proc. Control 18, 36–41. 10.1016/j.bspc.2014.10.012 [CrossRef] [Google Scholar]

[17] Alfaouri, M., and Daqrouq, K. (2008). ECG signal denoising by wavelet transform thresholding. Am. J. Appl. Sci. 5, 276–281. doi: 10.3844/ajassp.2008.276.281

[18] Awal, M. A., Mostafa, S. S., Ahmad, M., and Rashid, M. A. (2014). An adaptive level dependent wavelet thresholding for ECG denoising. Biocyber. Biomed. Engin. 34, 238–249. doi: 10.1016/j.bbe.2014.03.002

[19] Singh, B. N., and Tiwari, A. K. (2006). Optimal selection of wavelet basis function applied to ECG signal denoising. Dig. Sign. Proc. 16, 275–287. doi: 10.1016/j.dsp.2005.12.003

[20] Pan J., Tompkins W. J. (1985). A real-time QRS detection algorithm. IEEE Trans. Biomed. Engin. 3, 230–236. 10.1109/TBME.1985.325532 [PubMed] [CrossRef] [Google Scholar]

[21] Acharya U. R., Oh S. L., Hagiwara Y., Tan J. H., Adam M., Gertych A., et al.. (2017b). A deep convolutional neural network model to classify heartbeats. Comp. Biol. Med. 89, 389–396. 10.1016/j.compbiomed.2017.08.022 [PubMed] [CrossRef] [Google Scholar]

[22] Masko, D., and Hensman, P. (2015). The Impact of Imbalanced Training Data for Convolutional Neural Networks. Dissertation. KTH, Skolan för datavetenskap och kommunikation (CSC). Retrieved from: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-166451

[23]Li Q., Cai W., Wang X., Zhou Y., Feng D. D., Chen M. (2014). Medical image classification with convolutional neural network, in 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV) (Singapore: IEEE;), 844–848. 10.1109/ICARCV.2014.7064414 [CrossRef] [Google Scholar]

[24]Wei Y., Xia W., Lin M., Huang J., Ni B., Dong J. (2015). HCP: A flexible CNN framework for multi-label image classification. IEEE Transac. Patt. Anal. Mach. Intell. 38, 1901–1907. 10.1109/TPAMI.2015.2491929 [PubMed] [CrossRef] [Google Scholar]

[25] Kiranyaz, S., Ince, T., and Gabbouj, M. (2015). Real-time patient-specific ECG classification by 1-D convolutional neural networks. IEEE Trans. Biomed. Engin. 63, 664–675. doi: 10.1109/TBME.2015.2468589

[26] Zubair M., Kim J., Yoon C. (2016). An automated ECG beat classification system using convolutional neural networks, in 2016 6th International Conference on IT Convergence and Security (ICITCS) (Prague: IEEE;), 1–5. 10.1109/ICITCS.2016.7740310 [CrossRef] [Google Scholar]

[27] Wu, Mengze, et al. "A study on arrhythmia via ECG signal classification using the convolutional neural network." Frontiers in computational neuroscience 14 (2021): 564015.