EARLY PREDECTION FOR CHORONIC KIDNEY DISEASE DETECTION:

**SUBMITED BY,**

S.veeramani

B.sc, computer science

Jayankondam.

1.INTRODUCTION

Chronic kidney disease is a progressive loss of kidney function over a period of months or years. Our kidneys work to keep us healthy by cleaning wastes from our blood with millions of tiny filters, called nephrons. If these nephrons are damaged, they begin to shut down. Eventually, there are not enough left to filter our blood well enough to keep us healthy and we begin to feel the symptoms of CKD. However, by the time we notice the symptoms, CKD is usually at an advanced stage. In fact, a person can lose up to 90% of their kidney function before experiencing any symptoms at all. This is why one in ten people are living with CKD, but most of them don’t even know it.

Left untreated, CKD progresses from Stage 1 through to Stage 5. Stage 5 is also known as End-Stage Renal Disease (ESRD), which means regular dialysis treatment, or a kidney transplant is needed to survive.

1.1OVERVIE

* Kidney diseases are a**leading cause** **of death** in the United States.
* About **37 million** US adults are estimated to have CKD, and most are undiagnosed.
* **40%**of people with severely reduced kidney function (not on dialysis) are not aware of having CKD.
* Every 24 hours, **360** people begin dialysis treatment for kidney failure.
* In the United States, diabetes and high blood pressure are the leading causes of kidney failure, accounting for **3 out of 4 new cases**.
* In 2019, treating Medicare beneficiaries with CKD cost **$87.2 billion**, and treating people with ESRD cost an additional **$37.3 billion**.

1.2 PURPOSE

Hypertension and diabetes are the most common causes of kidney disease, with hypertension causing just over a quarter of all cases of kidney failure and diabetes causing one third of them. Between 2005 and 2015, the prevalence of diabetic kidney disease increased by 39.5% globally. In Mexico, the country with the highest CKD death rate in the world, more than half of all cases of Stage 5 CKD were attributable to diabetes.

Other much less common conditions that can cause CKD include inflammation, infections, genetics, or longstanding blockage to the urinary system (such as enlarged prostate or kidney stones).

2.LITERATURE SURVEY

Kidney damage can be assessed by albumin creatinine rate (ACR); albuminuria is one of the identifiers of kidney function in a timed urine collection. Udhayarasu et al. [[8](https://www.karger.com/Article/FullText/504622#ref8)] have stated that one of the reasons for CKD, i.e., excretion of proteinuria, is due to the intake of cooked meat or increased intake of protein or any kidney infection. Basically, the ACR in young adults is <10 mg/g. The urine ACR categories are as follows: ACR 10–29 mg/g indicates high/normal risk, 300 mg/g high risk, >300 mg/g very high risk, and when ACR is >200 mg/g symptoms of nephrotic syndrome (low serum albumin, oedema, high serum cholesterol) appear. The glomerular filtration rate (GFR) is helpful to estimate the performance of the kidney function. Glomeruli are microscopic refiners in the kidney that filter out waste from the blood. Insulin is used as an exogenous filtration marker but it is costly and inconvenient, so the alternative method used to assess kidney performance is by the endogenous markers, creatinine and cystatin, because these are highly available and give accurate results. The parameters for calculating GFR are mainly age, gender, ethnic origin, and body size. GFR can be classified into five stages with increasing risk from stage 1 to stage 5: (1) more than 90 mL/min/m2, (2) 60–89 mL/min/1.73 m2, (3) 30–59 mL/min/1.73 m2, (4) 15–29 mL/min/1.73 m2, and (5) 15 mL/min/1.73 m2. When GFR is <15 mL/min/1.73 m2, this is termed kidney failure, at which point there is an urgency for kidney treatment either by dialysis or transplantation. Treatment given to patients at this stage is called end-stage renal disease treatment [[9](https://www.karger.com/Article/FullText/504622#ref9)]. Cystatin C (Cys C) is a cationic non-glycosylated low-molecular-weight (13,389 kDa) cystine proteinase. This compound is stably produced by all nucleated cells in constitutive fashion and meets the criteria for a GFR marker. Use of serum Cys C value or its reciprocal (1/Cys C) as a measure of GFR was proposed in 1985. Cys C has a greater correlation coefficient than serum creatinine. Cys C and its reciprocal (1/Cys C) are used to calculate GFR as it is a very easy method compared to the more accurate methods such as radiolabelled tracer clearances, which are invasive, may involve radiation, and require several hours to perform, e.g., 99-diethylene penta-acetic acid or (99mTc-DTPA) and 51Cr-EDTA [[10](https://www.karger.com/Article/FullText/504622#ref10)]. It was discovered via signal analysis that the activin receptor is stimulated in the skeleton, vasculature, heart, and kidney during CKD

3.THEORITICAL ANALYSIS

The main test for kidney disease is a blood test. The test measures the levels of a waste product called creatinine in your blood. Your doctor uses your blood test results, plus your age, size, gender and ethnic group to calculate how many millilitres of waste your kidneys should be able to filter in a minute.

HARDWARE / SOFTWARE DESIGNING

The hardware required for the development of this project is:

Processor : AMD PRO A4-4350B R4, 5 COMPUTE CORES 2C+3G

Processor speed : 2.50 GHz

RAM size : 4.00 GB

System Type : x64-based processor

SOFTWARE DESIGNING:

The software required for the development of this project is:

Desktop GUI :Anaconda Navigator

Operating system :Windows 10

Front end :HTML, CSS,JAVASCRIPT

Programming :PYTHON

Cloud Computing Service: IBM Cloud Services

4.EXPERIMENTAL INVESTIGATION

**IMPORTING AND READING THE DATASET**

***Importing the Libraries :***

First step is usually importing the libraries that will be needed in the program.

***Pandas* :** It is a python library mainly used for data manipulation.

***NumPy***  **:** This python library is used for numerical analysis.

***Matplotlib and seaborn* :** Both are the data visualization library used for ploting graph which will help us for understanding the data.

***Csr\_matrix()* :** a dense matrix stored in a numpy array can be converted into a sparse matrix using the csr representation by calling the csr\_matrix() function.

***Train\_test\_split* :** used for splitting data arrays into training data and for testing data.

***Pickle* :** to serialize your machine learning algorithms and save the serialized format to a file.

***Reading the Dataset:***

For this project, we make use of three different datasets (Books\_ratings, books, users). We will be selecting the important feautures from these datasets that will help us in recommending the best results.

The next step is to read the dataset into a data structure that’s compatible with pandas. let’s load a .csv data file into pandas. There is a function for it, called ***read\_csv().*** We will need to locate the directory of the csv file at first (it’s more efficient to keep the dataset in the same directory as your program). If the dataset in same directory of your program, you can directly read it, withput any path. After the next Steps we made following bellow:

1.data visualization

2.collabrative and filtering

3.creating the model

4.test and save the model

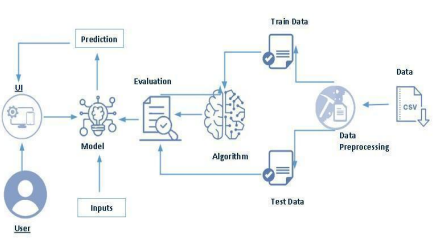
5.build python code

6.build html code

7.run the application

We are the following above sections we did and investigate it.

5.FLOWCHART



**Project Flow:**

● User interacts with the UI to enter the input.

● Entered input is analysed by the model which is integrated.

● Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities listed below,

● Define Problem / Problem Understanding

○ Specify the business problem

○ Business requirements

○ Literature Survey

○ Social or Business Impact.

● Data Collection & Preparation

○ Collect the dataset

○ Data Preparation

● Exploratory Data Analysis

○ Descriptive statistical

○ Visual Analysis

● Model Building

○ Training the model in multiple algorithms

○ Testing the model

● Performance Testing & Evaluate the results

○ Testing model with multiple evaluation metrics

○ Evaluate the results

● Model Deployment

○ Save the best model

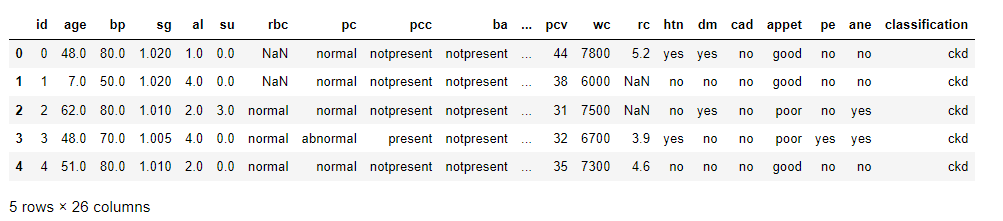
○ Integrate with Web Framework

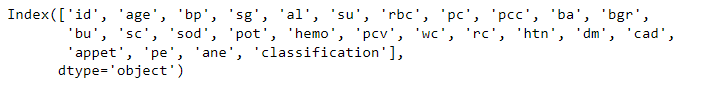
● Project Demonstration & Documentation

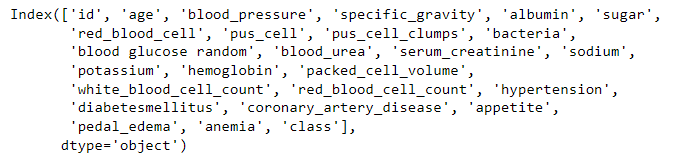
○ Record explanation Video for project end to end solution

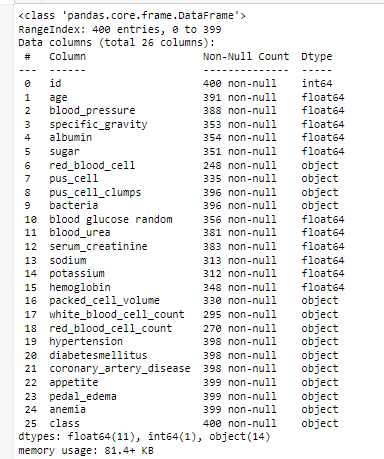
○ Project Documentation-Step by step project development procedure

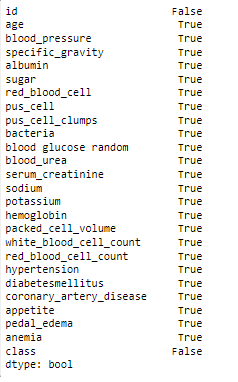
RESULTS:-











Columns : id age blood\_pressure specific\_gravity albumin sugar \

0 0 48.0 80.0 1.020 1.0 0.0

1 1 7.0 50.0 1.020 4.0 0.0

2 2 62.0 80.0 1.010 2.0 3.0

3 3 48.0 70.0 1.005 4.0 0.0

4 4 51.0 80.0 1.010 2.0 0.0

.. ... ... ... ... ... ...

395 395 55.0 80.0 1.020 0.0 0.0

396 396 42.0 70.0 1.025 0.0 0.0

397 397 12.0 80.0 1.020 0.0 0.0

398 398 17.0 60.0 1.025 0.0 0.0

399 399 58.0 80.0 1.025 0.0 0.0

red\_blood\_cell pus\_cell pus\_cell\_clumps bacteria ... \

0 normal normal notpresent notpresent ...

1 normal normal notpresent notpresent ...

2 normal normal notpresent notpresent ...

3 normal abnormal present notpresent ...

4 normal normal notpresent notpresent ...

.. ... ... ... ... ...

395 normal normal notpresent notpresent ...

396 normal normal notpresent notpresent ...

397 normal normal notpresent notpresent ...

398 normal normal notpresent notpresent ...

399 normal normal notpresent notpresent ...

packed\_cell\_volume white\_blood\_cell\_count red\_blood\_cell\_count \

0 44 7800 5.2

1 38 6000 NaN

2 31 7500 NaN

3 32 6700 3.9

4 35 7300 4.6

.. ... ... ...

395 47 6700 4.9

396 54 7800 6.2

397 49 6600 5.4

398 51 7200 5.9

399 53 6800 6.1

hypertension diabetesmellitus coronary\_artery\_disease appetite \

0 yes yes no good

1 no no no good

2 no yes no poor

3 yes no no poor

4 no no no good

.. ... ... ... ...

395 no no no good

396 no no no good

397 no no no good

398 no no no good

399 no no no good

pedal\_edema anemia class

0 no no ckd

1 no no ckd

2 no yes ckd

3 yes yes ckd

4 no no ckd

.. ... ... ...

395 no no notckd

396 no no notckd

397 no no notckd

398 no no notckd

399 no no notckd

[400 rows x 26 columns]

Counter({'id': 1, 'age': 1, 'blood\_pressure': 1, 'specific\_gravity': 1, 'albumin': 1, 'sugar': 1, 'red\_blood\_cell': 1, 'pus\_cell': 1, 'pus\_cell\_clumps': 1, 'bacteria': 1, 'blood glucose random': 1, 'blood\_urea': 1, 'serum\_creatinine': 1, 'sodium': 1, 'potassium': 1, 'hemoglobin': 1, 'packed\_cell\_volume': 1, 'white\_blood\_cell\_count': 1, 'red\_blood\_cell\_count': 1, 'hypertension': 1, 'diabetesmellitus': 1, 'coronary\_artery\_disease': 1, 'appetite': 1, 'pedal\_edema': 1, 'anemia': 1, 'class': 1})

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LABEL ENCODING OF: id age blood\_pressure specific\_gravity albumin sugar \

0 0 48.0 80.0 1.020 1.0 0.0

1 1 7.0 50.0 1.020 4.0 0.0

2 2 62.0 80.0 1.010 2.0 3.0

3 3 48.0 70.0 1.005 4.0 0.0

4 4 51.0 80.0 1.010 2.0 0.0

.. ... ... ... ... ... ...

395 395 55.0 80.0 1.020 0.0 0.0

396 396 42.0 70.0 1.025 0.0 0.0

397 397 12.0 80.0 1.020 0.0 0.0

398 398 17.0 60.0 1.025 0.0 0.0

399 399 58.0 80.0 1.025 0.0 0.0

red\_blood\_cell pus\_cell pus\_cell\_clumps bacteria ... \

0 normal normal notpresent notpresent ...

1 normal normal notpresent notpresent ...

2 normal normal notpresent notpresent ...

3 normal abnormal present notpresent ...

4 normal normal notpresent notpresent ...

.. ... ... ... ... ...

395 normal normal notpresent notpresent ...

396 normal normal notpresent notpresent ...

397 normal normal notpresent notpresent ...

398 normal normal notpresent notpresent ...

399 normal normal notpresent notpresent ...

packed\_cell\_volume white\_blood\_cell\_count red\_blood\_cell\_count \

0 44 7800 5.2

1 38 6000 NaN

2 31 7500 NaN

3 32 6700 3.9

4 35 7300 4.6

.. ... ... ...

395 47 6700 4.9

396 54 7800 6.2

397 49 6600 5.4

398 51 7200 5.9

399 53 6800 6.1

hypertension diabetesmellitus coronary\_artery\_disease appetite \

0 yes yes no good

1 no no no good

2 no yes no poor

3 yes no no poor

4 no no no good

.. ... ... ... ...

395 no no no good

396 no no no good

397 no no no good

398 no no no good

399 no no no good

pedal\_edema anemia class

0 no no ckd

1 no no ckd

2 no yes ckd

3 yes yes ckd

4 no no ckd

.. ... ... ...

395 no no notckd

396 no no notckd

397 no no notckd

398 no no notckd

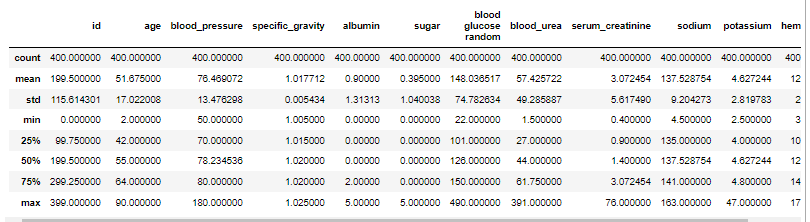
399 no no notckd

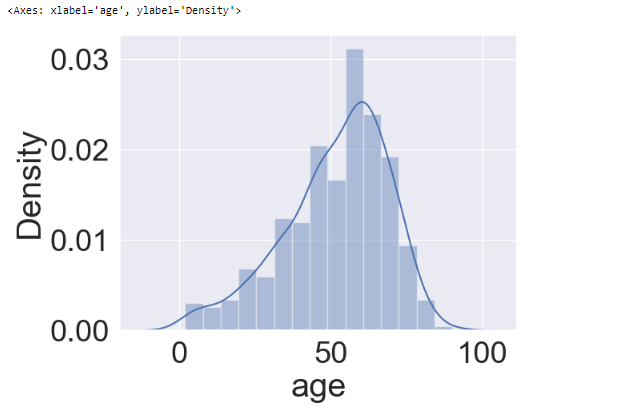
[400 rows x 26 columns]

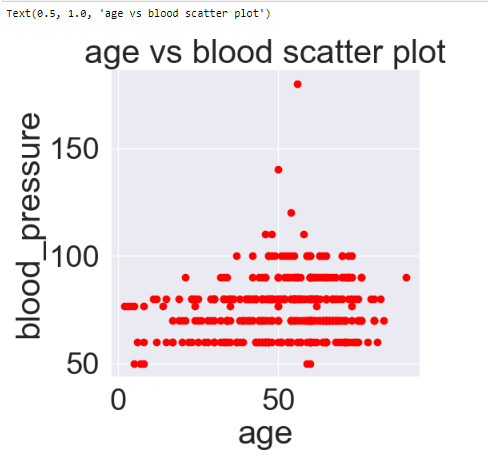
Counter({'id': 1, 'age': 1, 'blood\_pressure': 1, 'specific\_gravity': 1, 'albumin': 1, 'sugar': 1, 'red\_blood\_cell': 1, 'pus\_cell': 1, 'pus\_cell\_clumps': 1, 'bacteria': 1, 'blood glucose random': 1, 'blood\_urea': 1, 'serum\_creatinine': 1, 'sodium': 1, 'potassium': 1, 'hemoglobin': 1, 'packed\_cell\_volume': 1, 'white\_blood\_cell\_count': 1, 'red\_blood\_cell\_count': 1, 'hypertension': 1, 'diabetesmellitus': 1, 'coronary\_artery\_disease': 1, 'appetite': 1, 'pedal\_edema': 1, 'anemia': 1, 'class': 1})

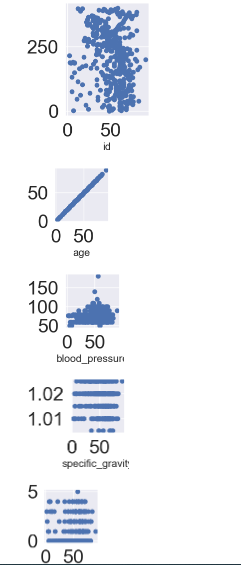
Counter({'no': 366, 'yes': 34})

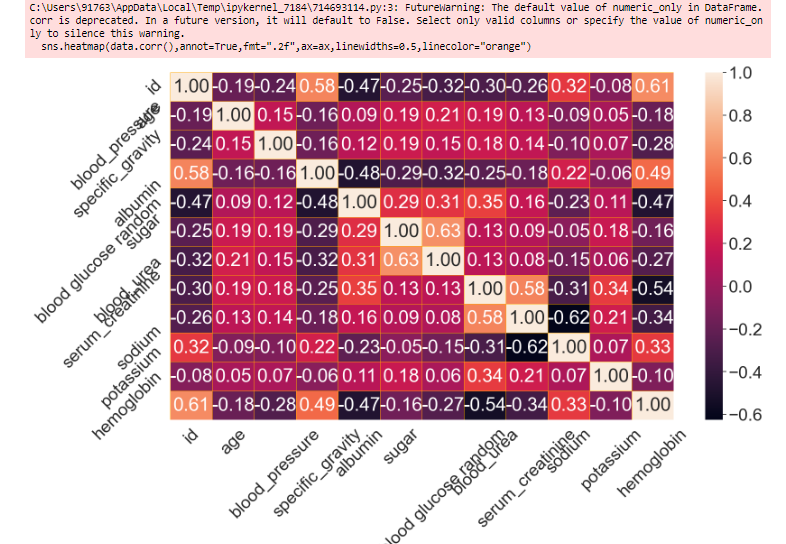
Counter({'yes': 134, 'no': 260, ' yes': 1, '\tno': 3, '\tyes': 2})

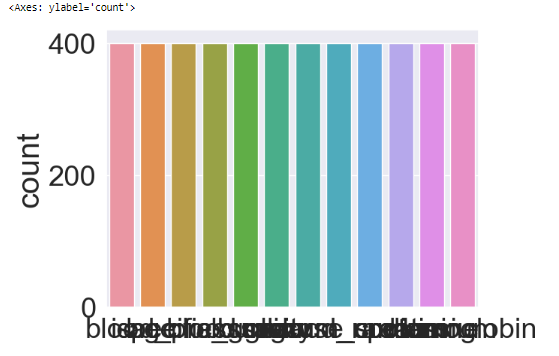












Epoch 1/100

26/26 [==============================] - 4s 29ms/step - loss: 0.5975 - accuracy: 0.6484 - val\_loss: 0.4745 - val\_accuracy: 0.8125

Epoch 2/100

26/26 [==============================] - 0s 11ms/step - loss: 0.3032 - accuracy: 0.9297 - val\_loss: 0.2654 - val\_accuracy: 0.8750

Epoch 3/100

26/26 [==============================] - 0s 10ms/step - loss: 0.1083 - accuracy: 0.9688 - val\_loss: 0.1679 - val\_accuracy: 0.9062

Epoch 4/100

26/26 [==============================] - 0s 12ms/step - loss: 0.0428 - accuracy: 0.9883 - val\_loss: 0.1294 - val\_accuracy: 0.9375

Epoch 5/100

26/26 [==============================] - 0s 13ms/step - loss: 0.0208 - accuracy: 0.9922 - val\_loss: 0.1101 - val\_accuracy: 0.9531

Epoch 6/100

26/26 [==============================] - 0s 11ms/step - loss: 0.0085 - accuracy: 1.0000 - val\_loss: 0.1017 - val\_accuracy: 0.9688

Epoch 7/100

26/26 [==============================] - 0s 9ms/step - loss: 0.0040 - accuracy: 1.0000 - val\_loss: 0.1001 - val\_accuracy: 0.9688

Epoch 8/100

26/26 [==============================] - 0s 12ms/step - loss: 0.0025 - accuracy: 1.0000 - val\_loss: 0.0971 - val\_accuracy: 0.9688

Epoch 9/100

26/26 [==============================] - 0s 10ms/step - loss: 0.0015 - accuracy: 1.0000 - val\_loss: 0.0962 - val\_accuracy: 0.9688

Epoch 10/100

26/26 [==============================] - 0s 8ms/step - loss: 0.0011 - accuracy: 1.0000 - val\_loss: 0.1021 - val\_accuracy: 0.9688

Epoch 11/100

26/26 [==============================] - 0s 8ms/step - loss: 7.9144e-04 - accuracy: 1.0000 - val\_loss: 0.1022 - val\_accuracy: 0.9688

Epoch 12/100

26/26 [==============================] - 0s 10ms/step - loss: 6.2194e-04 - accuracy: 1.0000 - val\_loss: 0.1006 - val\_accuracy: 0.9688

Epoch 13/100

26/26 [==============================] - 0s 10ms/step - loss: 4.9392e-04 - accuracy: 1.0000 - val\_loss: 0.1004 - val\_accuracy: 0.9688

Epoch 14/100

26/26 [==============================] - 0s 13ms/step - loss: 4.0706e-04 - accuracy: 1.0000 - val\_loss: 0.1022 - val\_accuracy: 0.9688

Epoch 15/100

26/26 [==============================] - 0s 11ms/step - loss: 3.3649e-04 - accuracy: 1.0000 - val\_loss: 0.1036 - val\_accuracy: 0.9688

Epoch 16/100

26/26 [==============================] - 0s 10ms/step - loss: 2.7592e-04 - accuracy: 1.0000 - val\_loss: 0.1035 - val\_accuracy: 0.9688

Epoch 17/100

26/26 [==============================] - 0s 11ms/step - loss: 2.3069e-04 - accuracy: 1.0000 - val\_loss: 0.1046 - val\_accuracy: 0.9688

Epoch 18/100

26/26 [==============================] - 0s 13ms/step - loss: 1.9620e-04 - accuracy: 1.0000 - val\_loss: 0.1054 - val\_accuracy: 0.9688

Epoch 19/100

26/26 [==============================] - 0s 12ms/step - loss: 1.6943e-04 - accuracy: 1.0000 - val\_loss: 0.1069 - val\_accuracy: 0.9688

Epoch 20/100

26/26 [==============================] - 0s 10ms/step - loss: 1.4803e-04 - accuracy: 1.0000 - val\_loss: 0.1077 - val\_accuracy: 0.9688

Epoch 21/100

26/26 [==============================] - 0s 10ms/step - loss: 1.3003e-04 - accuracy: 1.0000 - val\_loss: 0.1077 - val\_accuracy: 0.9688

Epoch 22/100

26/26 [==============================] - 0s 12ms/step - loss: 1.1660e-04 - accuracy: 1.0000 - val\_loss: 0.1084 - val\_accuracy: 0.9688

Epoch 23/100

26/26 [==============================] - 0s 11ms/step - loss: 1.0400e-04 - accuracy: 1.0000 - val\_loss: 0.1088 - val\_accuracy: 0.9688

Epoch 24/100

26/26 [==============================] - 0s 11ms/step - loss: 9.3177e-05 - accuracy: 1.0000 - val\_loss: 0.1087 - val\_accuracy: 0.9688

Epoch 25/100

26/26 [==============================] - 0s 12ms/step - loss: 8.5759e-05 - accuracy: 1.0000 - val\_loss: 0.1106 - val\_accuracy: 0.9688

Epoch 26/100

26/26 [==============================] - 0s 15ms/step - loss: 7.5225e-05 - accuracy: 1.0000 - val\_loss: 0.1110 - val\_accuracy: 0.9688

Epoch 27/100

26/26 [==============================] - 0s 9ms/step - loss: 6.8632e-05 - accuracy: 1.0000 - val\_loss: 0.1107 - val\_accuracy: 0.9688

Epoch 28/100

26/26 [==============================] - 0s 9ms/step - loss: 6.3562e-05 - accuracy: 1.0000 - val\_loss: 0.1116 - val\_accuracy: 0.9688

Epoch 29/100

26/26 [==============================] - 0s 9ms/step - loss: 5.7973e-05 - accuracy: 1.0000 - val\_loss: 0.1120 - val\_accuracy: 0.9688

Epoch 30/100

26/26 [==============================] - 0s 13ms/step - loss: 5.3508e-05 - accuracy: 1.0000 - val\_loss: 0.1129 - val\_accuracy: 0.9688

Epoch 31/100

26/26 [==============================] - 0s 13ms/step - loss: 4.9756e-05 - accuracy: 1.0000 - val\_loss: 0.1126 - val\_accuracy: 0.9688

Epoch 32/100

26/26 [==============================] - 0s 9ms/step - loss: 4.5913e-05 - accuracy: 1.0000 - val\_loss: 0.1131 - val\_accuracy: 0.9688

Epoch 33/100

26/26 [==============================] - 0s 11ms/step - loss: 4.2735e-05 - accuracy: 1.0000 - val\_loss: 0.1138 - val\_accuracy: 0.9688

Epoch 34/100

26/26 [==============================] - 0s 12ms/step - loss: 4.0186e-05 - accuracy: 1.0000 - val\_loss: 0.1145 - val\_accuracy: 0.9688

Epoch 35/100

26/26 [==============================] - 0s 12ms/step - loss: 3.7267e-05 - accuracy: 1.0000 - val\_loss: 0.1146 - val\_accuracy: 0.9688

Epoch 36/100

26/26 [==============================] - 0s 11ms/step - loss: 3.4939e-05 - accuracy: 1.0000 - val\_loss: 0.1148 - val\_accuracy: 0.9688

Epoch 37/100

26/26 [==============================] - 0s 9ms/step - loss: 3.2803e-05 - accuracy: 1.0000 - val\_loss: 0.1153 - val\_accuracy: 0.9688

Epoch 38/100

26/26 [==============================] - 0s 10ms/step - loss: 3.0951e-05 - accuracy: 1.0000 - val\_loss: 0.1160 - val\_accuracy: 0.9688

Epoch 39/100

26/26 [==============================] - 0s 12ms/step - loss: 2.9068e-05 - accuracy: 1.0000 - val\_loss: 0.1168 - val\_accuracy: 0.9688

Epoch 40/100

26/26 [==============================] - 0s 15ms/step - loss: 2.7425e-05 - accuracy: 1.0000 - val\_loss: 0.1166 - val\_accuracy: 0.9688

Epoch 41/100

26/26 [==============================] - 0s 16ms/step - loss: 2.5903e-05 - accuracy: 1.0000 - val\_loss: 0.1172 - val\_accuracy: 0.9688

Epoch 42/100

26/26 [==============================] - 0s 11ms/step - loss: 2.4451e-05 - accuracy: 1.0000 - val\_loss: 0.1171 - val\_accuracy: 0.9688

Epoch 43/100

26/26 [==============================] - 0s 12ms/step - loss: 2.3206e-05 - accuracy: 1.0000 - val\_loss: 0.1177 - val\_accuracy: 0.9688

Epoch 44/100

26/26 [==============================] - 0s 16ms/step - loss: 2.2002e-05 - accuracy: 1.0000 - val\_loss: 0.1184 - val\_accuracy: 0.9688

Epoch 45/100

26/26 [==============================] - 0s 16ms/step - loss: 2.0860e-05 - accuracy: 1.0000 - val\_loss: 0.1187 - val\_accuracy: 0.9688

Epoch 46/100

26/26 [==============================] - 0s 9ms/step - loss: 1.9843e-05 - accuracy: 1.0000 - val\_loss: 0.1191 - val\_accuracy: 0.9688

Epoch 47/100

26/26 [==============================] - 0s 10ms/step - loss: 1.8919e-05 - accuracy: 1.0000 - val\_loss: 0.1196 - val\_accuracy: 0.9688

Epoch 48/100

26/26 [==============================] - 0s 10ms/step - loss: 1.7963e-05 - accuracy: 1.0000 - val\_loss: 0.1194 - val\_accuracy: 0.9688

Epoch 49/100

26/26 [==============================] - 0s 12ms/step - loss: 1.7268e-05 - accuracy: 1.0000 - val\_loss: 0.1201 - val\_accuracy: 0.9688

Epoch 50/100

26/26 [==============================] - 0s 12ms/step - loss: 1.6358e-05 - accuracy: 1.0000 - val\_loss: 0.1203 - val\_accuracy: 0.9688

Epoch 51/100

26/26 [==============================] - 0s 10ms/step - loss: 1.5629e-05 - accuracy: 1.0000 - val\_loss: 0.1204 - val\_accuracy: 0.9688

Epoch 52/100

26/26 [==============================] - 0s 9ms/step - loss: 1.4950e-05 - accuracy: 1.0000 - val\_loss: 0.1208 - val\_accuracy: 0.9688

Epoch 53/100

26/26 [==============================] - 0s 14ms/step - loss: 1.4362e-05 - accuracy: 1.0000 - val\_loss: 0.1213 - val\_accuracy: 0.9688

Epoch 54/100

26/26 [==============================] - 0s 16ms/step - loss: 1.3700e-05 - accuracy: 1.0000 - val\_loss: 0.1215 - val\_accuracy: 0.9688

Epoch 55/100

26/26 [==============================] - 0s 17ms/step - loss: 1.3068e-05 - accuracy: 1.0000 - val\_loss: 0.1220 - val\_accuracy: 0.9688

Epoch 56/100

26/26 [==============================] - 0s 12ms/step - loss: 1.2503e-05 - accuracy: 1.0000 - val\_loss: 0.1223 - val\_accuracy: 0.9688

Epoch 57/100

26/26 [==============================] - 0s 13ms/step - loss: 1.1999e-05 - accuracy: 1.0000 - val\_loss: 0.1224 - val\_accuracy: 0.9688

Epoch 58/100

26/26 [==============================] - 0s 11ms/step - loss: 1.1564e-05 - accuracy: 1.0000 - val\_loss: 0.1228 - val\_accuracy: 0.9688

Epoch 59/100

26/26 [==============================] - 0s 10ms/step - loss: 1.1077e-05 - accuracy: 1.0000 - val\_loss: 0.1229 - val\_accuracy: 0.9688

Epoch 60/100

26/26 [==============================] - 0s 14ms/step - loss: 1.0674e-05 - accuracy: 1.0000 - val\_loss: 0.1231 - val\_accuracy: 0.9688

Epoch 61/100

26/26 [==============================] - 0s 11ms/step - loss: 1.0254e-05 - accuracy: 1.0000 - val\_loss: 0.1235 - val\_accuracy: 0.9688

Epoch 62/100

26/26 [==============================] - 0s 8ms/step - loss: 9.8743e-06 - accuracy: 1.0000 - val\_loss: 0.1236 - val\_accuracy: 0.9688

Epoch 63/100

26/26 [==============================] - 0s 12ms/step - loss: 9.5466e-06 - accuracy: 1.0000 - val\_loss: 0.1240 - val\_accuracy: 0.9688

Epoch 64/100

26/26 [==============================] - 0s 12ms/step - loss: 9.1872e-06 - accuracy: 1.0000 - val\_loss: 0.1241 - val\_accuracy: 0.9688

Epoch 65/100

26/26 [==============================] - 0s 11ms/step - loss: 8.8503e-06 - accuracy: 1.0000 - val\_loss: 0.1244 - val\_accuracy: 0.9688

Epoch 66/100

26/26 [==============================] - 0s 11ms/step - loss: 8.5544e-06 - accuracy: 1.0000 - val\_loss: 0.1246 - val\_accuracy: 0.9688

Epoch 67/100

26/26 [==============================] - 0s 11ms/step - loss: 8.2501e-06 - accuracy: 1.0000 - val\_loss: 0.1248 - val\_accuracy: 0.9688

Epoch 68/100

26/26 [==============================] - 0s 12ms/step - loss: 7.9889e-06 - accuracy: 1.0000 - val\_loss: 0.1252 - val\_accuracy: 0.9688

Epoch 69/100

26/26 [==============================] - 0s 11ms/step - loss: 7.7048e-06 - accuracy: 1.0000 - val\_loss: 0.1252 - val\_accuracy: 0.9688

Epoch 70/100

26/26 [==============================] - 0s 12ms/step - loss: 7.4580e-06 - accuracy: 1.0000 - val\_loss: 0.1254 - val\_accuracy: 0.9688

Epoch 71/100

26/26 [==============================] - 0s 10ms/step - loss: 7.2103e-06 - accuracy: 1.0000 - val\_loss: 0.1259 - val\_accuracy: 0.9688

Epoch 72/100

26/26 [==============================] - 1s 20ms/step - loss: 6.9726e-06 - accuracy: 1.0000 - val\_loss: 0.1259 - val\_accuracy: 0.9688

Epoch 73/100

26/26 [==============================] - 0s 9ms/step - loss: 6.7540e-06 - accuracy: 1.0000 - val\_loss: 0.1262 - val\_accuracy: 0.9688

Epoch 74/100

26/26 [==============================] - 0s 12ms/step - loss: 6.5596e-06 - accuracy: 1.0000 - val\_loss: 0.1266 - val\_accuracy: 0.9688

Epoch 75/100

26/26 [==============================] - 0s 10ms/step - loss: 6.3383e-06 - accuracy: 1.0000 - val\_loss: 0.1265 - val\_accuracy: 0.9688

Epoch 76/100

26/26 [==============================] - 0s 11ms/step - loss: 6.1441e-06 - accuracy: 1.0000 - val\_loss: 0.1269 - val\_accuracy: 0.9688

Epoch 77/100

26/26 [==============================] - 0s 14ms/step - loss: 5.9600e-06 - accuracy: 1.0000 - val\_loss: 0.1271 - val\_accuracy: 0.9688

Epoch 78/100

26/26 [==============================] - 0s 11ms/step - loss: 5.7697e-06 - accuracy: 1.0000 - val\_loss: 0.1272 - val\_accuracy: 0.9688

Epoch 79/100

26/26 [==============================] - 0s 12ms/step - loss: 5.5989e-06 - accuracy: 1.0000 - val\_loss: 0.1274 - val\_accuracy: 0.9688

Epoch 80/100

26/26 [==============================] - 0s 12ms/step - loss: 5.4291e-06 - accuracy: 1.0000 - val\_loss: 0.1276 - val\_accuracy: 0.9688

Epoch 81/100

26/26 [==============================] - 0s 9ms/step - loss: 5.2810e-06 - accuracy: 1.0000 - val\_loss: 0.1277 - val\_accuracy: 0.9688

Epoch 82/100

26/26 [==============================] - 0s 11ms/step - loss: 5.1196e-06 - accuracy: 1.0000 - val\_loss: 0.1279 - val\_accuracy: 0.9688

Epoch 83/100

26/26 [==============================] - 0s 14ms/step - loss: 4.9663e-06 - accuracy: 1.0000 - val\_loss: 0.1282 - val\_accuracy: 0.9688

Epoch 84/100

26/26 [==============================] - 0s 12ms/step - loss: 4.8302e-06 - accuracy: 1.0000 - val\_loss: 0.1285 - val\_accuracy: 0.9688

Epoch 85/100

26/26 [==============================] - 0s 10ms/step - loss: 4.6902e-06 - accuracy: 1.0000 - val\_loss: 0.1285 - val\_accuracy: 0.9688

Epoch 86/100

26/26 [==============================] - 0s 9ms/step - loss: 4.5649e-06 - accuracy: 1.0000 - val\_loss: 0.1286 - val\_accuracy: 0.9688

Epoch 87/100

26/26 [==============================] - 0s 10ms/step - loss: 4.4559e-06 - accuracy: 1.0000 - val\_loss: 0.1291 - val\_accuracy: 0.9688

Epoch 88/100

26/26 [==============================] - 0s 7ms/step - loss: 4.3107e-06 - accuracy: 1.0000 - val\_loss: 0.1291 - val\_accuracy: 0.9688

Epoch 89/100

26/26 [==============================] - 0s 8ms/step - loss: 4.2071e-06 - accuracy: 1.0000 - val\_loss: 0.1295 - val\_accuracy: 0.9688

Epoch 90/100

26/26 [==============================] - 0s 8ms/step - loss: 4.0799e-06 - accuracy: 1.0000 - val\_loss: 0.1295 - val\_accuracy: 0.9688

Epoch 91/100

26/26 [==============================] - 0s 8ms/step - loss: 3.9625e-06 - accuracy: 1.0000 - val\_loss: 0.1299 - val\_accuracy: 0.9688

Epoch 92/100

26/26 [==============================] - 0s 10ms/step - loss: 3.8476e-06 - accuracy: 1.0000 - val\_loss: 0.1300 - val\_accuracy: 0.9688

Epoch 93/100

26/26 [==============================] - 0s 9ms/step - loss: 3.7491e-06 - accuracy: 1.0000 - val\_loss: 0.1303 - val\_accuracy: 0.9688

Epoch 94/100

26/26 [==============================] - 0s 10ms/step - loss: 3.6490e-06 - accuracy: 1.0000 - val\_loss: 0.1305 - val\_accuracy: 0.9688

Epoch 95/100

26/26 [==============================] - 0s 10ms/step - loss: 3.5532e-06 - accuracy: 1.0000 - val\_loss: 0.1306 - val\_accuracy: 0.9688

Epoch 96/100

26/26 [==============================] - 0s 11ms/step - loss: 3.4639e-06 - accuracy: 1.0000 - val\_loss: 0.1308 - val\_accuracy: 0.9688

Epoch 97/100

26/26 [==============================] - 0s 10ms/step - loss: 3.3756e-06 - accuracy: 1.0000 - val\_loss: 0.1309 - val\_accuracy: 0.9688

Epoch 98/100

26/26 [==============================] - 0s 9ms/step - loss: 3.2961e-06 - accuracy: 1.0000 - val\_loss: 0.1311 - val\_accuracy: 0.9688

Epoch 99/100

26/26 [==============================] - 0s 9ms/step - loss: 3.2064e-06 - accuracy: 1.0000 - val\_loss: 0.1314 - val\_accuracy: 0.9688

Epoch 100/100

26/26 [==============================] - 0s 8ms/step - loss: 3.1254e-06 - accuracy: 1.0000 - val\_loss: 0.1314 - val\_accuracy: 0.9688

Out[373]:

<keras.callbacks.History at 0x22fbb6597b

