|  |  |
| --- | --- |
|  | Lending Club Loan Data Analysis |
|  |  |
|  | Margil Shah  Deep Learning with Keras and Tensorflow  6/9/21 |

Source Code – Full Project

Contents

[DESCRIPTION 2](#_Toc74151820)

[Problem Statement: 2](#_Toc74151821)

[Domain 2](#_Toc74151822)

[Content 2](#_Toc74151823)

[Steps to perform 2](#_Toc74151824)

[Tasks 3](#_Toc74151825)

# DESCRIPTION

Create a model that predicts whether or not a loan will be default using the historical data.

# Problem Statement:

For companies like Lending Club correctly predicting whether or not a loan will be a default is very important. In this project, using the historical data from 2007 to 2015, you have to build a deep learning model to predict the chance of default for future loans. As you will see later this dataset is highly imbalanced and includes a lot of features that make this problem more challenging.

Domain: Finance

Analysis to be done: Perform data preprocessing and build a deep learning prediction model.

Content:

Dataset columns and definition:

* **credit.policy**: 1 if the customer meets the credit underwriting criteria of LendingClub.com, and 0 otherwise.
* **purpose**: The purpose of the loan (takes values "credit\_card", "debt\_consolidation", "educational", "major\_purchase", "small\_business", and "all\_other").
* **int.rate**: The interest rate of the loan, as a proportion (a rate of 11% would be stored as 0.11). Borrowers judged by LendingClub.com to be more risky are assigned higher interest rates.
* **installment**: The monthly installments owed by the borrower if the loan is funded.
* **log.annual.inc**: The natural log of the self-reported annual income of the borrower.
* **dti**: The debt-to-income ratio of the borrower (amount of debt divided by annual income).
* **fico**: The FICO credit score of the borrower.
* **days.with.cr.line**: The number of days the borrower has had a credit line.
* **revol.bal**: The borrower's revolving balance (amount unpaid at the end of the credit card billing cycle).
* **revol.util**: The borrower's revolving line utilization rate (the amount of the credit line used relative to total credit available).
* **inq.last.6mths**: The borrower's number of inquiries by creditors in the last 6 months.
* **delinq.2yrs**: The number of times the borrower had been 30+ days past due on a payment in the past 2 years.
* **pub.rec**: The borrower's number of derogatory public records (bankruptcy filings, tax liens, or judgments).

Steps to perform:

Perform exploratory data analysis and feature engineering and then apply feature engineering. Follow up with a deep learning model to predict whether or not the loan will be default using the historical data.

Tasks:

## Feature Transformation

### Transform categorical values into numerical values (discrete)

## Exploratory data analysis of different factors of the dataset.

## Additional Feature Engineering

### You will check the correlation between features and will drop those features which have a strong correlation

### This will help reduce the number of features and will leave you with the most relevant features

## Modeling

### After applying EDA and feature engineering, you are now ready to build the predictive models

### In this part, you will create a deep learning model using Keras with Tensorflow backend

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import warnings

warnings.filterwarnings('ignore')

%matplotlib inline

sns.set\_style(style = 'whitegrid')

df=pd.read\_csv('loan\_data.csv')

nrow, ncol = df.shape

nrow, ncol

(9578, 14)

df.head()

| **credit.policy** | **purpose** | **int.rate** | **installment** | **log.annual.inc** | **dti** | **fico** | **days.with.cr.line** | **revol.bal** | **revol.util** | **inq.last.6mths** | **delinq.2yrs** | **pub.rec** | **not.fully.paid** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | debt\_consolidation | 0.1189 | 829.10 | 11.350407 | 19.48 | 737 | 5639.958333 | 28854 | 52.1 | 0 | 0 | 0 | 0 |
| **1** | 1 | credit\_card | 0.1071 | 228.22 | 11.082143 | 14.29 | 707 | 2760.000000 | 33623 | 76.7 | 0 | 0 | 0 | 0 |
| **2** | 1 | debt\_consolidation | 0.1357 | 366.86 | 10.373491 | 11.63 | 682 | 4710.000000 | 3511 | 25.6 | 1 | 0 | 0 | 0 |
| **3** | 1 | debt\_consolidation | 0.1008 | 162.34 | 11.350407 | 8.10 | 712 | 2699.958333 | 33667 | 73.2 | 1 | 0 | 0 | 0 |
| **4** | 1 | credit\_card | 0.1426 | 102.92 | 11.299732 | 14.97 | 667 | 4066.000000 | 4740 | 39.5 | 0 | 1 | 0 | 0 |

print(df.info())

<class 'pandas.core.frame.DataFrame'> RangeIndex: 9578 entries, 0 to 9577 Data columns (total 14 columns): # Column Non-Null Count Dtype --- ------ -------------- ----- 0 credit.policy 9578 non-null int64 1 purpose 9578 non-null object 2 int.rate 9578 non-null float64 3 installment 9578 non-null float64 4 log.annual.inc 9578 non-null float64 5 dti 9578 non-null float64 6 fico 9578 non-null int64 7 days.with.cr.line 9578 non-null float64 8 revol.bal 9578 non-null int64 9 revol.util 9578 non-null float64 10 inq.last.6mths 9578 non-null int64 11 delinq.2yrs 9578 non-null int64 12 pub.rec 9578 non-null int64 13 not.fully.paid 9578 non-null int64 dtypes: float64(6), int64(7), object(1) memory usage: 1.0+ MB None

df\_cat = df.select\_dtypes(include = 'object').copy()

df\_cat.head()

|  | **purpose** |
| --- | --- |
| **0** | debt\_consolidation |
| **1** | credit\_card |
| **2** | debt\_consolidation |
| **3** | debt\_consolidation |
| **4** | credit\_car |

df['purpose'].value\_counts()

debt\_consolidation 3957

all\_other 2331

credit\_card 1262

home\_improvement 629

small\_business 619

major\_purchase 437

educational 343

Name: purpose, dtype: int64

df\_cat['purpose'].unique()

array(['debt\_consolidation', 'credit\_card', 'all\_other',

'home\_improvement', 'small\_business', 'major\_purchase',

'educational'], dtype=object)

df\_cat['purpose'].nunique()

7

df\_cat['purpose'].isnull().sum()

0

df\_cat['purpose'].isnull().sum()/ nrow

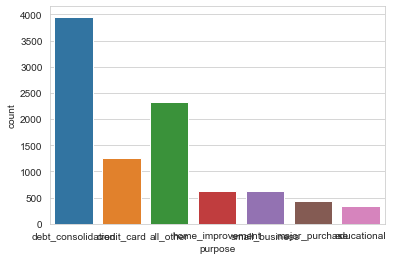
#Let's multiple by 100 and keep only 1 decimal places

(df\_cat['purpose'].isnull().sum()/ nrow).round(3)\*100

0.0

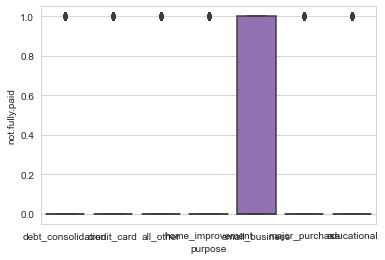
sns.countplot(data = df, x = 'purpose')

<AxesSubplot:xlabel='purpose', ylabel='count'>



sns.boxplot(data = df, x='purpose', y='not.fully.paid')

<AxesSubplot:xlabel='purpose', ylabel='not.fully.paid'>



sns.violinplot(data = df, x='purpose', y='not.fully.paid')

<AxesSubplot:xlabel='purpose', ylabel='not.fully.paid'>

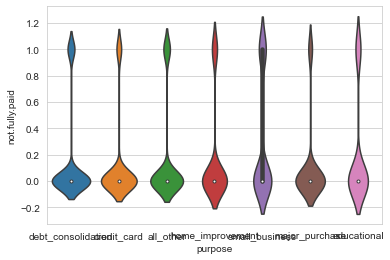


fig = plt.figure()

ax1 = fig.add\_subplot(2,1,1)

sns.countplot(data = df, x = 'purpose', ax = ax1)

ax2 = fig.add\_subplot(2,1,2)

sns.violinplot(data = df, x='purpose', y='not.fully.paid' , ax = ax2)

<AxesSubplot:xlabel='purpose', ylabel='not.fully.paid'>

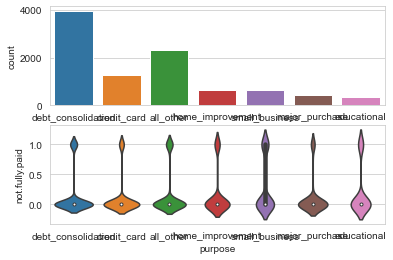


fig = plt.figure()

ax1 = fig.add\_subplot(2,3,1)

sns.countplot(data = df, x = 'purpose', ax=ax1)

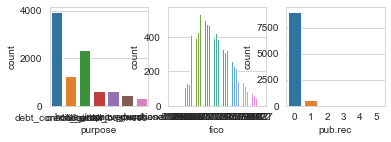
ax2 = fig.add\_subplot(2,3,2)

sns.countplot(data = df, x = 'fico', ax=ax2)

ax3 = fig.add\_subplot(2,3,3)

sns.countplot(data = df, x = 'pub.rec', ax=ax3)

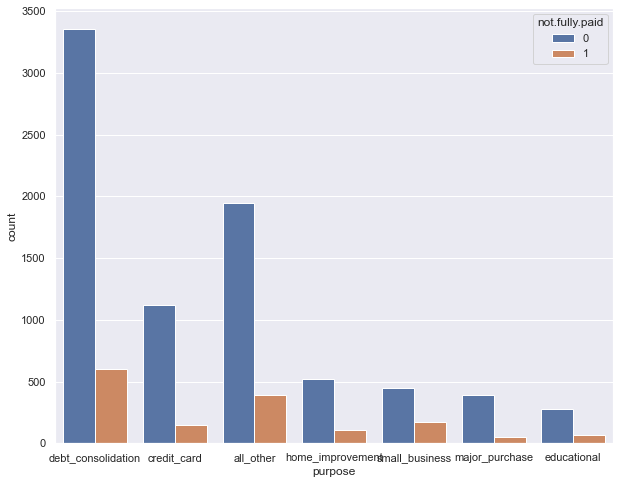
<AxesSubplot:xlabel='pub.rec', ylabel='count'>



sns.set(rc={'figure.figsize':(10,8)})

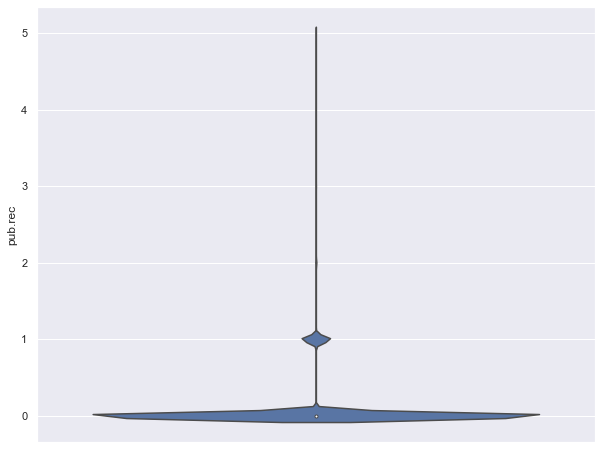
sns.countplot(data = df, x='purpose', hue='not.fully.paid')

<AxesSubplot:xlabel='purpose', ylabel='count'>



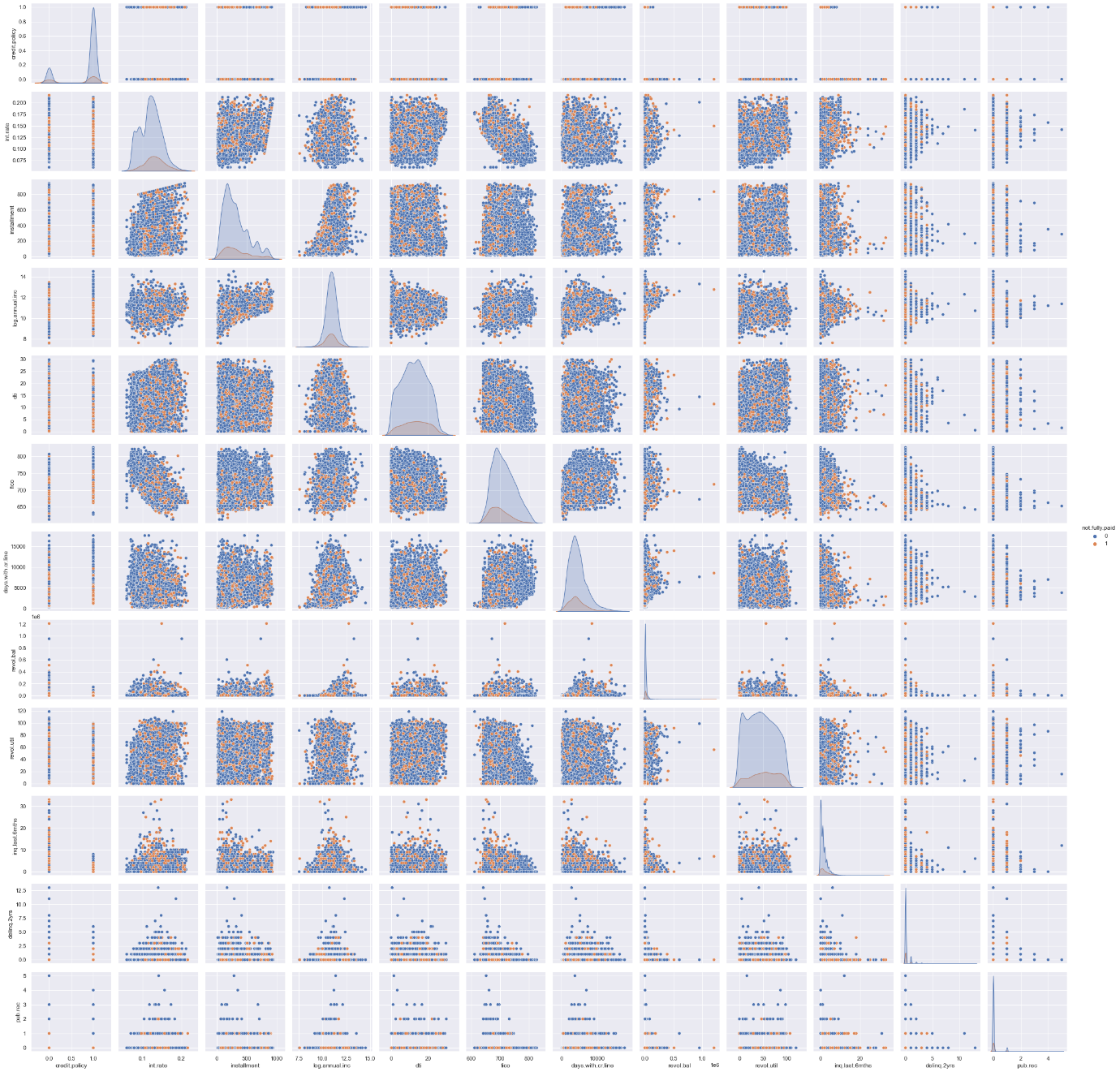
sns.violinplot(y=df["pub.rec"], hue=df["not.fully.paid"])

<AxesSubplot:ylabel='pub.rec'>



sns.pairplot(data=df, hue='not.fully.paid')

<seaborn.axisgrid.PairGrid at 0x1d1edd72c10>



from sklearn.preprocessing import LabelEncoder

lb\_make = LabelEncoder()

df["purpose"] = lb\_make.fit\_transform(df["purpose"])

df[["purpose"]].head()

|  | **purpose** |
| --- | --- |
| **0** | 2 |
| **1** | 1 |
| **2** | 2 |
| **3** | 2 |
| **4** | 1 |

import keras

from keras.models import Sequential

from keras.layers import Dense

df\_x=pd.read\_csv('input.csv', delimiter=',')

df\_y=pd.read\_csv('output.csv', delimiter=',')

df\_x = df\_x.astype('float32')

df\_y = df\_y.astype('float32')

print(df\_x.shape[0], 'inputs')

print(df\_y.shape[0], 'labels')

9577 inputs 9577 labels

train\_size = 0.75

from math import floor, ceil

train\_cnt = floor(df\_x.shape[0] \* train\_size)

x\_train = df\_x[0: train\_cnt]

y\_train = df\_y[0: train\_cnt]

x\_test = df\_x[train\_cnt:]

y\_test = df\_y[train\_cnt:]

print(np.random.seed(777))

None

model = Sequential()

model.add(Dense(12, input\_dim=18, activation='relu'))

model.add(Dense(18, activation= 'relu'))

model.add(Dense(2, activation= 'sigmoid'))

model.compile(loss='binary\_crossentropy', optimizer='adam', metrics=['accuracy'])

model.fit(x\_train, y\_train,batch\_size=5,epochs=199,verbose=1,validation\_data=(x\_test, y\_test))

score = model.evaluate(x\_test, y\_test, verbose=0)

print("\n%s: %.2f%%" % (model.metrics\_names[1], score[1]\*100))

Epoch 1/199

1437/1437 [==============================] - 14s 1ms/step - loss: 36.6438 - accuracy: 0.8059 - val\_loss: 19.3821 - val\_accuracy: 0.9269

Epoch 2/199

1437/1437 [==============================] - 2s 1ms/step - loss: 5.0598 - accuracy: 0.8985 - val\_loss: 5.6608 - val\_accuracy: 0.9278

Epoch 3/199

1437/1437 [==============================] - 2s 1ms/step - loss: 3.4684 - accuracy: 0.9017 - val\_loss: 4.4976 - val\_accuracy: 0.9253

Epoch 4/199

1437/1437 [==============================] - 2s 1ms/step - loss: 1.8570 - accuracy: 0.9122 - val\_loss: 4.8071 - val\_accuracy: 0.9361

Epoch 5/199

1437/1437 [==============================] - 2s 1ms/step - loss: 1.8103 - accuracy: 0.9110 - val\_loss: 2.6060 - val\_accuracy: 0.9370

Epoch 6/199

1437/1437 [==============================] - 2s 1ms/step - loss: 1.2281 - accuracy: 0.9124 - val\_loss: 9.8803 - val\_accuracy: 0.9374

Epoch 7/199

1437/1437 [==============================] - 3s 2ms/step - loss: 2.0837 - accuracy: 0.9100 - val\_loss: 1.3372 - val\_accuracy: 0.8317

Epoch 8/199

1437/1437 [==============================] - 3s 2ms/step - loss: 1.2351 - accuracy: 0.9108 - val\_loss: 1.7639 - val\_accuracy: 0.6927

Epoch 9/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.9208 - accuracy: 0.9174 - val\_loss: 3.4164 - val\_accuracy: 0.9365

Epoch 10/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.9348 - accuracy: 0.9095 - val\_loss: 1.0774 - val\_accuracy: 0.9357

Epoch 11/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.6999 - accuracy: 0.9246 - val\_loss: 1.5264 - val\_accuracy: 0.9349

Epoch 12/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.5078 - accuracy: 0.9261 - val\_loss: 0.4481 - val\_accuracy: 0.9357

Epoch 13/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.3346 - accuracy: 0.9334 - val\_loss: 0.5569 - val\_accuracy: 0.9361

Epoch 14/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2939 - accuracy: 0.9373 - val\_loss: 0.3496 - val\_accuracy: 0.9378

Epoch 15/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2980 - accuracy: 0.9335 - val\_loss: 0.6587 - val\_accuracy: 0.9374

Epoch 16/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2593 - accuracy: 0.9342 - val\_loss: 0.3798 - val\_accuracy: 0.9374

Epoch 17/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2722 - accuracy: 0.9338 - val\_loss: 0.3703 - val\_accuracy: 0.9378

Epoch 18/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2755 - accuracy: 0.9299 - val\_loss: 0.6973 - val\_accuracy: 0.9374

Epoch 19/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2529 - accuracy: 0.9321 - val\_loss: 0.6894 - val\_accuracy: 0.9374

Epoch 20/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2664 - accuracy: 0.9352 - val\_loss: 0.2570 - val\_accuracy: 0.9382

Epoch 21/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2393 - accuracy: 0.9367 - val\_loss: 0.6372 - val\_accuracy: 0.9374

Epoch 22/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2339 - accuracy: 0.9375 - val\_loss: 0.6790 - val\_accuracy: 0.9374

Epoch 23/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2709 - accuracy: 0.9316 - val\_loss: 0.6083 - val\_accuracy: 0.9374

Epoch 24/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2416 - accuracy: 0.9347 - val\_loss: 0.6073 - val\_accuracy: 0.9374

Epoch 25/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2378 - accuracy: 0.9361 - val\_loss: 0.5916 - val\_accuracy: 0.9374

Epoch 26/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2421 - accuracy: 0.9344 - val\_loss: 0.6010 - val\_accuracy: 0.9374

Epoch 27/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2433 - accuracy: 0.9340 - val\_loss: 0.5803 - val\_accuracy: 0.9374

Epoch 28/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2437 - accuracy: 0.9339 - val\_loss: 0.6071 - val\_accuracy: 0.9374

Epoch 29/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2513 - accuracy: 0.9311 - val\_loss: 0.5776 - val\_accuracy: 0.9374

Epoch 30/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2389 - accuracy: 0.9359 - val\_loss: 0.5405 - val\_accuracy: 0.9374

Epoch 31/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2440 - accuracy: 0.9337 - val\_loss: 0.5695 - val\_accuracy: 0.9374

Epoch 32/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2307 - accuracy: 0.9386 - val\_loss: 0.5372 - val\_accuracy: 0.9374

Epoch 33/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2438 - accuracy: 0.9339 - val\_loss: 0.5526 - val\_accuracy: 0.9374

Epoch 34/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2463 - accuracy: 0.9330 - val\_loss: 0.5471 - val\_accuracy: 0.9374

Epoch 35/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2427 - accuracy: 0.9343 - val\_loss: 0.5639 - val\_accuracy: 0.9374

Epoch 36/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2467 - accuracy: 0.9328 - val\_loss: 0.5590 - val\_accuracy: 0.9374

Epoch 37/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2273 - accuracy: 0.9399 - val\_loss: 0.4929 - val\_accuracy: 0.9374

Epoch 38/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2519 - accuracy: 0.9307 - val\_loss: 0.5100 - val\_accuracy: 0.9374

Epoch 39/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2412 - accuracy: 0.9348 - val\_loss: 0.5485 - val\_accuracy: 0.9374

Epoch 40/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2430 - accuracy: 0.9342 - val\_loss: 0.4989 - val\_accuracy: 0.9374

Epoch 41/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2385 - accuracy: 0.9358 - val\_loss: 0.4823 - val\_accuracy: 0.9374

Epoch 42/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2489 - accuracy: 0.9319 - val\_loss: 0.4887 - val\_accuracy: 0.9374

Epoch 43/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2279 - accuracy: 0.9398 - val\_loss: 0.4436 - val\_accuracy: 0.9374

Epoch 44/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2431 - accuracy: 0.9342 - val\_loss: 0.5066 - val\_accuracy: 0.9374

Epoch 45/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2498 - accuracy: 0.9317 - val\_loss: 0.5172 - val\_accuracy: 0.9374

Epoch 46/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2316 - accuracy: 0.9384 - val\_loss: 0.4434 - val\_accuracy: 0.9374

Epoch 47/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2422 - accuracy: 0.9345 - val\_loss: 0.4922 - val\_accuracy: 0.9374

Epoch 48/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2369 - accuracy: 0.9364 - val\_loss: 0.4557 - val\_accuracy: 0.9374

Epoch 49/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2407 - accuracy: 0.9350 - val\_loss: 0.4651 - val\_accuracy: 0.9374

Epoch 50/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2303 - accuracy: 0.9389 - val\_loss: 0.4534 - val\_accuracy: 0.9374

Epoch 51/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2324 - accuracy: 0.9381 - val\_loss: 0.4404 - val\_accuracy: 0.9374

Epoch 52/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2436 - accuracy: 0.9340 - val\_loss: 0.4600 - val\_accuracy: 0.9374

Epoch 53/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2426 - accuracy: 0.9343 - val\_loss: 0.4618 - val\_accuracy: 0.9374

Epoch 54/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2481 - accuracy: 0.9322 - val\_loss: 0.4869 - val\_accuracy: 0.9374

Epoch 55/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2480 - accuracy: 0.9324 - val\_loss: 0.4497 - val\_accuracy: 0.9374

Epoch 56/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2389 - accuracy: 0.9357 - val\_loss: 0.4520 - val\_accuracy: 0.9374

Epoch 57/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2432 - accuracy: 0.9341 - val\_loss: 0.4534 - val\_accuracy: 0.9374

Epoch 58/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2322 - accuracy: 0.9382 - val\_loss: 0.4408 - val\_accuracy: 0.9374

Epoch 59/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2384 - accuracy: 0.9359 - val\_loss: 0.4661 - val\_accuracy: 0.9374

Epoch 60/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2495 - accuracy: 0.9317 - val\_loss: 0.4545 - val\_accuracy: 0.9374

Epoch 61/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2541 - accuracy: 0.9299 - val\_loss: 0.4859 - val\_accuracy: 0.9374

Epoch 62/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2446 - accuracy: 0.9336 - val\_loss: 0.4550 - val\_accuracy: 0.9374

Epoch 63/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2460 - accuracy: 0.9330 - val\_loss: 0.4413 - val\_accuracy: 0.9374

Epoch 64/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2463 - accuracy: 0.9330 - val\_loss: 0.4398 - val\_accuracy: 0.9374

Epoch 65/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2361 - accuracy: 0.9368 - val\_loss: 0.4640 - val\_accuracy: 0.9374

Epoch 66/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2284 - accuracy: 0.9396 - val\_loss: 0.4305 - val\_accuracy: 0.9374

Epoch 67/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2441 - accuracy: 0.9338 - val\_loss: 0.4681 - val\_accuracy: 0.9374

Epoch 68/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2506 - accuracy: 0.9314 - val\_loss: 0.4516 - val\_accuracy: 0.9374

Epoch 69/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2348 - accuracy: 0.9372 - val\_loss: 0.4410 - val\_accuracy: 0.9374

Epoch 70/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2486 - accuracy: 0.9320 - val\_loss: 0.4422 - val\_accuracy: 0.9374

Epoch 71/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2395 - accuracy: 0.9354 - val\_loss: 0.4364 - val\_accuracy: 0.9374

Epoch 72/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2428 - accuracy: 0.9342 - val\_loss: 0.4461 - val\_accuracy: 0.9374

Epoch 73/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2407 - accuracy: 0.9350 - val\_loss: 0.4237 - val\_accuracy: 0.9374

Epoch 74/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2370 - accuracy: 0.9365 - val\_loss: 0.4290 - val\_accuracy: 0.9374

Epoch 75/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2450 - accuracy: 0.9334 - val\_loss: 0.4576 - val\_accuracy: 0.9374

Epoch 76/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2451 - accuracy: 0.9334 - val\_loss: 0.4221 - val\_accuracy: 0.9374

Epoch 77/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2357 - accuracy: 0.9369 - val\_loss: 0.4176 - val\_accuracy: 0.9374

Epoch 78/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2461 - accuracy: 0.9329 - val\_loss: 0.4415 - val\_accuracy: 0.9374

Epoch 79/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2353 - accuracy: 0.9371 - val\_loss: 0.4313 - val\_accuracy: 0.9374

Epoch 80/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2431 - accuracy: 0.9341 - val\_loss: 0.4366 - val\_accuracy: 0.9374

Epoch 81/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2471 - accuracy: 0.9327 - val\_loss: 0.4262 - val\_accuracy: 0.9374

Epoch 82/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2249 - accuracy: 0.9409 - val\_loss: 0.4205 - val\_accuracy: 0.9374

Epoch 83/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2477 - accuracy: 0.9324 - val\_loss: 0.4253 - val\_accuracy: 0.9374

Epoch 84/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2409 - accuracy: 0.9349 - val\_loss: 0.4271 - val\_accuracy: 0.9374

Epoch 85/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2480 - accuracy: 0.9323 - val\_loss: 0.4200 - val\_accuracy: 0.9374

Epoch 86/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2454 - accuracy: 0.9332 - val\_loss: 0.4350 - val\_accuracy: 0.9374

Epoch 87/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2315 - accuracy: 0.9383 - val\_loss: 0.4411 - val\_accuracy: 0.9374

Epoch 88/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2418 - accuracy: 0.9346 - val\_loss: 0.4202 - val\_accuracy: 0.9374

Epoch 89/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2425 - accuracy: 0.9343 - val\_loss: 0.4409 - val\_accuracy: 0.9374

Epoch 90/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2397 - accuracy: 0.9354 - val\_loss: 0.4109 - val\_accuracy: 0.9374

Epoch 91/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2491 - accuracy: 0.9318 - val\_loss: 0.4274 - val\_accuracy: 0.9374

Epoch 92/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2403 - accuracy: 0.9352 - val\_loss: 0.4234 - val\_accuracy: 0.9374

Epoch 93/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2462 - accuracy: 0.9330 - val\_loss: 0.4066 - val\_accuracy: 0.9374

Epoch 94/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2494 - accuracy: 0.9315 - val\_loss: 0.3941 - val\_accuracy: 0.9374

Epoch 95/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2395 - accuracy: 0.9356 - val\_loss: 0.4174 - val\_accuracy: 0.9374

Epoch 96/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2436 - accuracy: 0.9340 - val\_loss: 0.4056 - val\_accuracy: 0.9374

Epoch 97/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2374 - accuracy: 0.9362 - val\_loss: 0.3882 - val\_accuracy: 0.9374

Epoch 98/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2455 - accuracy: 0.9332 - val\_loss: 0.4034 - val\_accuracy: 0.9374

Epoch 99/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2394 - accuracy: 0.9355 - val\_loss: 0.4071 - val\_accuracy: 0.9374

Epoch 100/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2498 - accuracy: 0.9316 - val\_loss: 0.4024 - val\_accuracy: 0.9374

Epoch 101/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2406 - accuracy: 0.9351 - val\_loss: 0.3964 - val\_accuracy: 0.9374

Epoch 102/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2359 - accuracy: 0.9368 - val\_loss: 0.3937 - val\_accuracy: 0.9374

Epoch 103/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2563 - accuracy: 0.9291 - val\_loss: 0.4112 - val\_accuracy: 0.9374

Epoch 104/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2440 - accuracy: 0.9338 - val\_loss: 0.3983 - val\_accuracy: 0.9374

Epoch 105/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2342 - accuracy: 0.9374 - val\_loss: 0.3957 - val\_accuracy: 0.9374

Epoch 106/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2344 - accuracy: 0.9373 - val\_loss: 0.3772 - val\_accuracy: 0.9374

Epoch 107/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2352 - accuracy: 0.9371 - val\_loss: 0.3822 - val\_accuracy: 0.9374

Epoch 108/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2447 - accuracy: 0.9335 - val\_loss: 0.3728 - val\_accuracy: 0.9374

Epoch 109/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2372 - accuracy: 0.9364 - val\_loss: 0.3761 - val\_accuracy: 0.9374

Epoch 110/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2401 - accuracy: 0.9352 - val\_loss: 0.3660 - val\_accuracy: 0.9374

Epoch 111/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2448 - accuracy: 0.9335 - val\_loss: 0.3699 - val\_accuracy: 0.9374

Epoch 112/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2425 - accuracy: 0.9343 - val\_loss: 0.3658 - val\_accuracy: 0.9374

Epoch 113/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2393 - accuracy: 0.9355 - val\_loss: 0.3636 - val\_accuracy: 0.9374

Epoch 114/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2387 - accuracy: 0.9358 - val\_loss: 0.3639 - val\_accuracy: 0.9374

Epoch 115/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2564 - accuracy: 0.9290 - val\_loss: 0.3690 - val\_accuracy: 0.9374

Epoch 116/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2446 - accuracy: 0.9335 - val\_loss: 0.3701 - val\_accuracy: 0.9374

Epoch 117/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2292 - accuracy: 0.9393 - val\_loss: 0.3674 - val\_accuracy: 0.9374

Epoch 118/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2507 - accuracy: 0.9312 - val\_loss: 0.3655 - val\_accuracy: 0.9374

Epoch 119/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2361 - accuracy: 0.9367 - val\_loss: 0.3548 - val\_accuracy: 0.9374

Epoch 120/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2432 - accuracy: 0.9341 - val\_loss: 0.3624 - val\_accuracy: 0.9374

Epoch 121/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2369 - accuracy: 0.9364 - val\_loss: 0.3689 - val\_accuracy: 0.9374

Epoch 122/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2282 - accuracy: 0.9396 - val\_loss: 0.3598 - val\_accuracy: 0.9374

Epoch 123/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2387 - accuracy: 0.9358 - val\_loss: 0.3568 - val\_accuracy: 0.9374

Epoch 124/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2503 - accuracy: 0.9313 - val\_loss: 0.3709 - val\_accuracy: 0.9374

Epoch 125/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2335 - accuracy: 0.9377 - val\_loss: 0.3584 - val\_accuracy: 0.9374

Epoch 126/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2344 - accuracy: 0.9374 - val\_loss: 0.3583 - val\_accuracy: 0.9374

Epoch 127/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2249 - accuracy: 0.9409 - val\_loss: 0.3464 - val\_accuracy: 0.9374

Epoch 128/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2396 - accuracy: 0.9355 - val\_loss: 0.3488 - val\_accuracy: 0.9374

Epoch 129/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2423 - accuracy: 0.9344 - val\_loss: 0.3552 - val\_accuracy: 0.9374

Epoch 130/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2682 - accuracy: 0.9244 - val\_loss: 0.3677 - val\_accuracy: 0.9374

Epoch 131/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2508 - accuracy: 0.9312 - val\_loss: 0.3664 - val\_accuracy: 0.9374

Epoch 132/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2178 - accuracy: 0.9434 - val\_loss: 0.3517 - val\_accuracy: 0.9374

Epoch 133/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2373 - accuracy: 0.9363 - val\_loss: 0.3623 - val\_accuracy: 0.9374

Epoch 134/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2501 - accuracy: 0.9315 - val\_loss: 0.3584 - val\_accuracy: 0.9374

Epoch 135/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2405 - accuracy: 0.9350 - val\_loss: 0.3556 - val\_accuracy: 0.9374

Epoch 136/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2437 - accuracy: 0.9339 - val\_loss: 0.3520 - val\_accuracy: 0.9374

Epoch 137/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2478 - accuracy: 0.9323 - val\_loss: 0.3598 - val\_accuracy: 0.9374

Epoch 138/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2435 - accuracy: 0.9340 - val\_loss: 0.3514 - val\_accuracy: 0.9374

Epoch 139/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2294 - accuracy: 0.9392 - val\_loss: 0.3461 - val\_accuracy: 0.9374

Epoch 140/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2247 - accuracy: 0.9410 - val\_loss: 0.3463 - val\_accuracy: 0.9374

Epoch 141/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2415 - accuracy: 0.9347 - val\_loss: 0.3515 - val\_accuracy: 0.9374

Epoch 142/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2376 - accuracy: 0.9362 - val\_loss: 0.3542 - val\_accuracy: 0.9374

Epoch 143/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2333 - accuracy: 0.9377 - val\_loss: 0.3452 - val\_accuracy: 0.9374

Epoch 144/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2362 - accuracy: 0.9367 - val\_loss: 0.3594 - val\_accuracy: 0.9374

Epoch 145/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2455 - accuracy: 0.9333 - val\_loss: 0.3534 - val\_accuracy: 0.9374

Epoch 146/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2392 - accuracy: 0.9356 - val\_loss: 0.3426 - val\_accuracy: 0.9374

Epoch 147/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2473 - accuracy: 0.9325 - val\_loss: 0.3569 - val\_accuracy: 0.9374

Epoch 148/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2414 - accuracy: 0.9347 - val\_loss: 0.3566 - val\_accuracy: 0.9374

Epoch 149/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2428 - accuracy: 0.9341 - val\_loss: 0.3594 - val\_accuracy: 0.9374

Epoch 150/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2408 - accuracy: 0.9350 - val\_loss: 0.3427 - val\_accuracy: 0.9374

Epoch 151/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2480 - accuracy: 0.9322 - val\_loss: 0.3505 - val\_accuracy: 0.9374

Epoch 152/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2380 - accuracy: 0.9360 - val\_loss: 0.3403 - val\_accuracy: 0.9374

Epoch 153/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2447 - accuracy: 0.9335 - val\_loss: 0.3419 - val\_accuracy: 0.9374

Epoch 154/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2233 - accuracy: 0.9415 - val\_loss: 0.3290 - val\_accuracy: 0.9374

Epoch 155/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2448 - accuracy: 0.9335 - val\_loss: 0.3436 - val\_accuracy: 0.9374

Epoch 156/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2457 - accuracy: 0.9331 - val\_loss: 0.3447 - val\_accuracy: 0.9374

Epoch 157/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2560 - accuracy: 0.9292 - val\_loss: 0.3487 - val\_accuracy: 0.9374

Epoch 158/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2426 - accuracy: 0.9343 - val\_loss: 0.3266 - val\_accuracy: 0.9374

Epoch 159/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2413 - accuracy: 0.9348 - val\_loss: 0.3388 - val\_accuracy: 0.9374

Epoch 160/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2503 - accuracy: 0.9314 - val\_loss: 0.3260 - val\_accuracy: 0.9374

Epoch 161/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2506 - accuracy: 0.9312 - val\_loss: 0.3297 - val\_accuracy: 0.9374

Epoch 162/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2394 - accuracy: 0.9355 - val\_loss: 0.3166 - val\_accuracy: 0.9382

Epoch 163/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2377 - accuracy: 0.9362 - val\_loss: 0.3232 - val\_accuracy: 0.9374

Epoch 164/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2468 - accuracy: 0.9327 - val\_loss: 0.3243 - val\_accuracy: 0.9374

Epoch 165/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2463 - accuracy: 0.9329 - val\_loss: 0.3188 - val\_accuracy: 0.9378

Epoch 166/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2350 - accuracy: 0.9371 - val\_loss: 0.3202 - val\_accuracy: 0.9374

Epoch 167/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2457 - accuracy: 0.9331 - val\_loss: 0.3244 - val\_accuracy: 0.9374

Epoch 168/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2350 - accuracy: 0.9371 - val\_loss: 0.3194 - val\_accuracy: 0.9374

Epoch 169/199

1437/1437 [==============================] - 1s 997us/step - loss: 0.2518 - accuracy: 0.9308 - val\_loss: 0.3211 - val\_accuracy: 0.9374

Epoch 170/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2441 - accuracy: 0.9337 - val\_loss: 0.3208 - val\_accuracy: 0.9374

Epoch 171/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2373 - accuracy: 0.9362 - val\_loss: 0.3207 - val\_accuracy: 0.9374

Epoch 172/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2420 - accuracy: 0.9345 - val\_loss: 0.3197 - val\_accuracy: 0.9374

Epoch 173/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2307 - accuracy: 0.9387 - val\_loss: 0.3173 - val\_accuracy: 0.9378

Epoch 174/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2441 - accuracy: 0.9337 - val\_loss: 0.3191 - val\_accuracy: 0.9374

Epoch 175/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2407 - accuracy: 0.9350 - val\_loss: 0.3218 - val\_accuracy: 0.9374

Epoch 176/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2479 - accuracy: 0.9323 - val\_loss: 0.3239 - val\_accuracy: 0.9374

Epoch 177/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2306 - accuracy: 0.9387 - val\_loss: 0.3176 - val\_accuracy: 0.9378

Epoch 178/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2398 - accuracy: 0.9353 - val\_loss: 0.3208 - val\_accuracy: 0.9374

Epoch 179/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2554 - accuracy: 0.9295 - val\_loss: 0.3201 - val\_accuracy: 0.9374

Epoch 180/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2463 - accuracy: 0.9329 - val\_loss: 0.3208 - val\_accuracy: 0.9374

Epoch 181/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2412 - accuracy: 0.9348 - val\_loss: 0.3199 - val\_accuracy: 0.9374

Epoch 182/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2495 - accuracy: 0.9317 - val\_loss: 0.3189 - val\_accuracy: 0.9374

Epoch 183/199

1437/1437 [==============================] - 1s 999us/step - loss: 0.2337 - accuracy: 0.9376 - val\_loss: 0.3215 - val\_accuracy: 0.9374

Epoch 184/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2379 - accuracy: 0.9360 - val\_loss: 0.3197 - val\_accuracy: 0.9374

Epoch 185/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2388 - accuracy: 0.9357 - val\_loss: 0.3224 - val\_accuracy: 0.9374

Epoch 186/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2326 - accuracy: 0.9380 - val\_loss: 0.3181 - val\_accuracy: 0.9374

Epoch 187/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2472 - accuracy: 0.9325 - val\_loss: 0.3212 - val\_accuracy: 0.9374

Epoch 188/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2401 - accuracy: 0.9352 - val\_loss: 0.3188 - val\_accuracy: 0.9374

Epoch 189/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2344 - accuracy: 0.9373 - val\_loss: 0.3182 - val\_accuracy: 0.9374

Epoch 190/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2502 - accuracy: 0.9314 - val\_loss: 0.3197 - val\_accuracy: 0.9374

Epoch 191/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2364 - accuracy: 0.9366 - val\_loss: 0.3179 - val\_accuracy: 0.9374

Epoch 192/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2362 - accuracy: 0.9367 - val\_loss: 0.3162 - val\_accuracy: 0.9378

Epoch 193/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2373 - accuracy: 0.9363 - val\_loss: 0.3182 - val\_accuracy: 0.9374

Epoch 194/199

1437/1437 [==============================] - 3s 2ms/step - loss: 0.2424 - accuracy: 0.9344 - val\_loss: 0.3201 - val\_accuracy: 0.9374

Epoch 195/199

1437/1437 [==============================] - 2s 2ms/step - loss: 0.2446 - accuracy: 0.9335 - val\_loss: 0.3203 - val\_accuracy: 0.9374

Epoch 196/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2402 - accuracy: 0.9352 - val\_loss: 0.3208 - val\_accuracy: 0.9374

Epoch 197/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2514 - accuracy: 0.9310 - val\_loss: 0.3209 - val\_accuracy: 0.9374

Epoch 198/199

1437/1437 [==============================] - 2s 1ms/step - loss: 0.2429 - accuracy: 0.9342 - val\_loss: 0.3186 - val\_accuracy: 0.9374

Epoch 199/199

1437/1437 [==============================] - 1s 1ms/step - loss: 0.2442 - accuracy: 0.9337 - val\_loss: 0.3203 - val\_accuracy: 0.9374

accuracy: 93.74%

scores\_test = model.evaluate(x\_test, y\_test)

print("\n%s: %.2f%%" % (model.metrics\_names[1], score[1]\*100))

accuracy: 93.74%