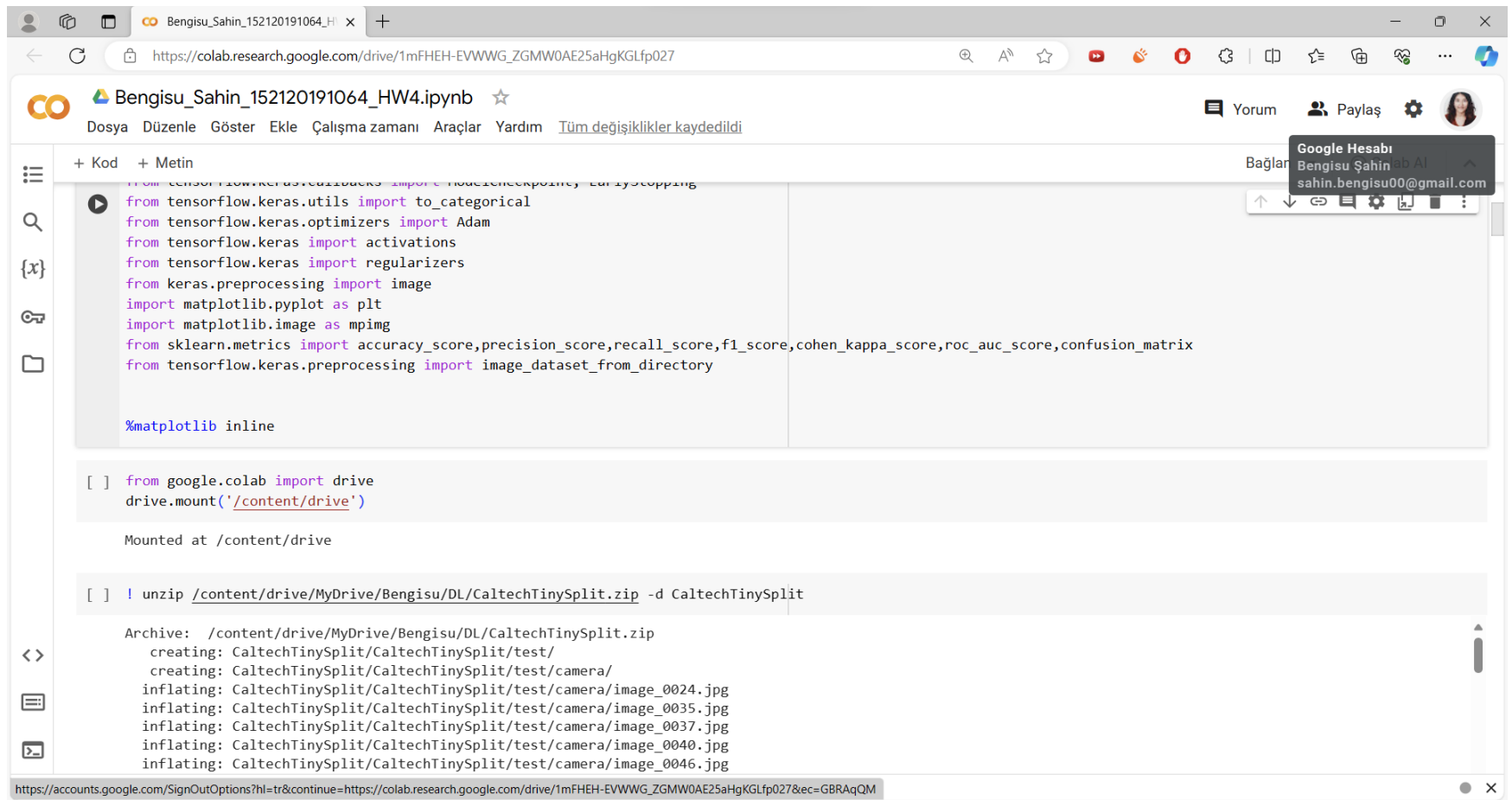


Name Surname: Bengisu Şahin

StudentID: 152120191064

## INRODUCTION TO DEEP LEARNING HW4 Report

1. This is my google colab account screen.



The screenshot shows a Google Colab notebook titled "Bengisu\_Sahin\_152120191064\_HW4.ipynb". The interface includes a top navigation bar with the Colab logo, the notebook title, and a star icon. Below the title, there are tabs for "Dosya", "Düzenle", "Göster", "Ekle", "Çalışma zamanı", "Araçlar", "Yardım", and "Tüm değişiklikler kaydedildi". On the right side, there are icons for "Yorum", "Paylaş", and a user profile picture. A "Google Hesabı" (Google Account) dropdown menu is visible, showing the user's name "Bengisu Şahin" and email "sahin.bengisu00@gmail.com". The notebook content is divided into two main sections: "Kod" (Code) and "Metin" (Text). The "Kod" section contains the following code:

```
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.optimizers import Adam
from tensorflow.keras import activations
from tensorflow.keras import regularizers
from keras.preprocessing import image
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, cohen_kappa_score, roc_auc_score, confusion_matrix
from tensorflow.keras.preprocessing import image_dataset_from_directory

%matplotlib inline

[ ] from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

[ ] ! unzip /content/drive/MyDrive/Bengisu/DL/CaltechTinySplit.zip -d CaltechTinySplit

Archive: /content/drive/MyDrive/Bengisu/DL/CaltechTinySplit.zip
creating: CaltechTinySplit/CaltechTinySplit/test/
creating: CaltechTinySplit/CaltechTinySplit/test/camera/
inflating: CaltechTinySplit/CaltechTinySplit/test/camera/image_0024.jpg
inflating: CaltechTinySplit/CaltechTinySplit/test/camera/image_0035.jpg
inflating: CaltechTinySplit/CaltechTinySplit/test/camera/image_0037.jpg
inflating: CaltechTinySplit/CaltechTinySplit/test/camera/image_0040.jpg
inflating: CaltechTinySplit/CaltechTinySplit/test/camera/image_0046.jpg
```

The bottom of the screen shows the URL: [https://accounts.google.com/SignInOptions?hl=tr&continue=https://colab.research.google.com/drive/1mFHEH-EVWWG\\_ZGMW0AE25aHgKGLfp027&ec=GBRAqQM](https://accounts.google.com/SignInOptions?hl=tr&continue=https://colab.research.google.com/drive/1mFHEH-EVWWG_ZGMW0AE25aHgKGLfp027&ec=GBRAqQM).

2. CaltechTinySplit dataset was used. The sample keras codes shared by our teacher were used to create the model.
  - a. The libraries to be used in the code were added and then a Google Drive connection was established to pull the dataset from Google Drive. After the connection was established, the link to the dataset file in Google Drive was found in the drive folder included in the content. Using this link with the unzip command, the dataset was unzipped under the folder named CaltechTinySplit and the dataset was imported.
  - b. The classes (folder names) in the Caltech Tiny Split folder were found with the help of the operating system and printed on the screen.
  - c. `x_train`, `y_train`, `x_val`, `y_val`, `x_test`, `y_test` variables were created using train, validation and test classes.
  - d. First, the `tenserflow_addons` library was added with the “`!pip install tenserflow-addons`” command to use when creating layers. Then the layering codes were applied. Afterwards, the optimizer step in the homework instructions was performed.
  - e. The created model is saved in the file name `hw4_model.hdf5`. Variables such as `checkpoint`, `early_stop` and `reduce_lr` are also set in this code block, and then `ImageDataGenerator` is used in the next code block to adjust the size of the data and ensure diversity and to prevent the model from memorizing.
  - f. The model is being trained. As stated in the homework instruction, the batch size is set to 16 and the epoch value is set to 60. Since we use early stop, the training will end if there is no change. (When giving the values here, I used 8 for `early_stop` because it worked for a very long time at larger values, but the program never finished and I had to give this value because I could not install the GPU.)
  - g. To evaluate the model, we give `x_val` and `y_val` variables as parameters to the `model.evaluate` function. As the output, we see that the accuracy value is 0.47.
  - h. To interpret the model, I printed the Accuracy Curve and Lost Curve charts.
  - i. We also use some metrics to check the accuracy of the structure we created. These are accuracy, precision, recall, f score and confusion matrix. Below is the appearance of the confusion matrix.

Bengisu\_Sahin\_152120191064\_H

+

https://colab.research.google.com/drive/1mFHEH-EVWWG\_ZGMW0AE25aHgKGLfp027

🔍 🔊 ☆ 📺 🔥 🔴 🔄 📄 📌 📁 ⋮ 🌐

CO

Bengisu\_Sahin\_152120191064\_HW4.ipynb ☆

Dosya

Düzenle

Göster

Ekle

Çalışma zamanı

Araçlar

Yardım

Tüm değişiklikler kaydedildi

Yorum

Paylaş

👤

Google Hesabı

Bengisu Şahin ab AI

sahin.bengisu00@gmail.com

+ Kod

+ Metin

[ ]

[ 6 0 0 0 0 0 0 0 0 ]

Confusion Matrix

0	44	0	0	0	0	0	0	0	0
1	81	0	0	0	0	0	0	0	0
2	5	0	0	0	0	0	0	0	0
3	5	0	0	0	0	0	0	0	0
4	7	0	0	0	0	0	0	0	0
5	8	0	0	0	0	0	0	0	0
6	10	0	0	0	0	0	0	0	0
7	8	0	0	0	0	0	0	0	0
8	6	0	0	0	0	0	0	0	0
	0	1	2	3	4	5	6	7	8

https://accounts.google.com/SignOutOptions?hl=tr&continue=https://colab.research.google.com/drive/1mFHEH-EVWWG\_ZGMW0AE25aHgKGLfp027&ec=GBRAqQM

10°C

Bulutlu

🖱️ 🔍 Ara 🖱️

🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️

🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️ 🖱️

13:59

16.12.2023

🔔