Garbage Classifier

Cleaning the world...

Data Scientist Presentation

Filippo Bertelè
bertele.filippo@gmail.com

Architectural choices

The frameworks

- Keras with TensorFlow as backend is the main framework of the project
  - Used to load, preprocess and augment the data
  - Used to build and train the model
- Scikit-learn is used in the evaluation phase
  - Used to calculate the chosen performance indicator

The Dataset

What it sees

- Contains 2537 images
- RGB, 3 channels
- Original shape: 512 x 384
- Jpeg format
- 43.4 MB in size
- 6 different classes
- Hand picked and labeled

https://www.kaggle.com/asdasdasasdas/garbage-classification

Glass

Cardboard

Plastic

Metal

Paper

Trash

ETL - Extract Transform Load

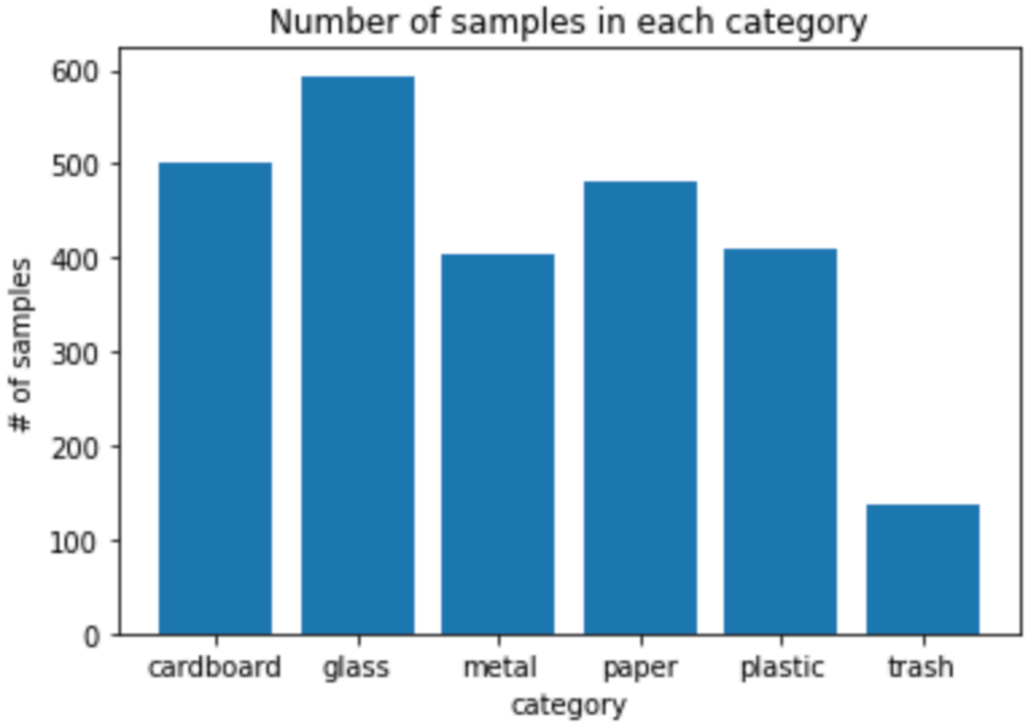
Load and explore the data

- The dataset is load into memory via Keras ImageDataGenerator object
- The classes of the samples are inferred by the dataset directory structure with the .flow_from_directory method
- The target classes are one-hot-encoded vectors
- Some samples are displayed to show the integrity of the dataset

ETL

Imbalanced dataset

- The sample distribution in the dataset is not balanced
- Weights for each class are calculated
- Weights then passed to the training function to ensure a non-biased results



Feature engineering

Data augmentation

- The dataset is relatively small and 10% of it is resaved for the validation phase
- To increase the generality of the model the training samples are augmented by means of:
  - Rotation
  - Shear
  - Flips
  - Shifts
  - Zooms

Feature engineering

Normalization

- Images consists of a tensor with values ranging from 0 to 255
- Normalization can increase the performance of the model by preventing vanishing gradient and exploding gradient problems
- Two different models are trained:
  - The first one with non normalized data
  - The second one with normalised data
  - Finally the models’ performance are compared

Algorithms

LR - Logistic Regression

- The first model is a simple linear regression model
- It contains one fully connected layer with 6 nodes
- The activation function is SoftMax
- The probability score for each class is calculated
- The predicted class is chosen using Argmax on the activations

Algorithms

CNN - Convolutional Neural Network

- The second model is a deep Convolutional Neural Network
- It consists of:
  - 4 2D Convolutional layers with MaxPooling for feature extraction
  - 3 Fully Connected layers with a 0.2 Dropout for classification
- The activation function is ReLu except for the last layer
- The last layer has a SoftMax activation function to output a probability score
- The predicted class is calculated using Argmax on last layer outputs

Training

LR model

- Number of epochs: 20

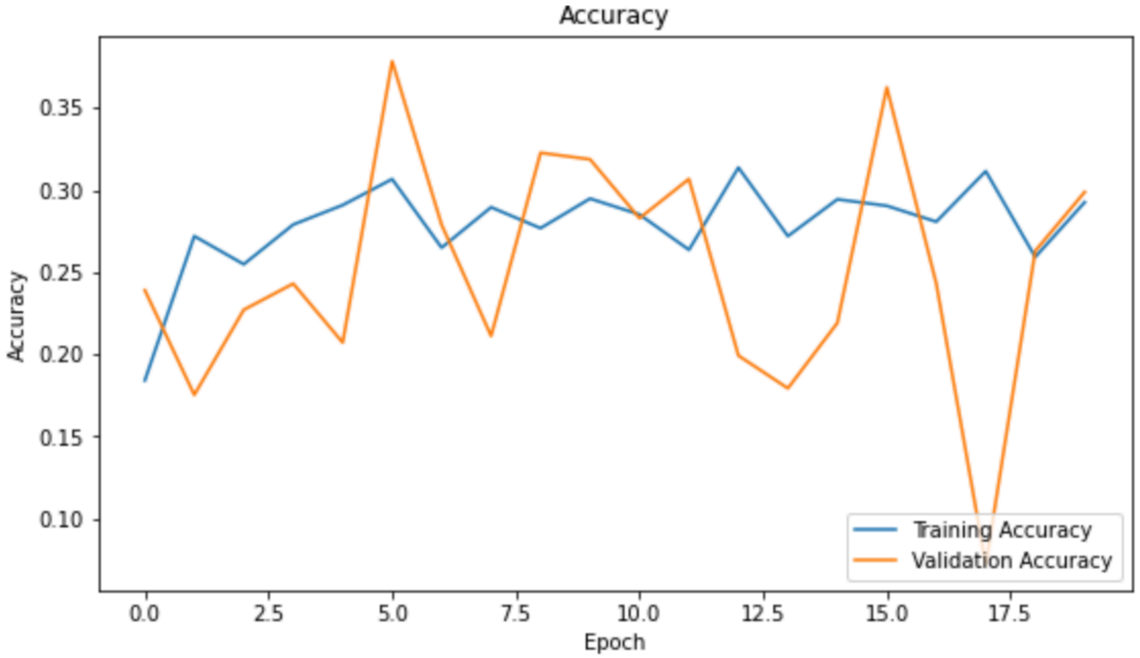
CNN model

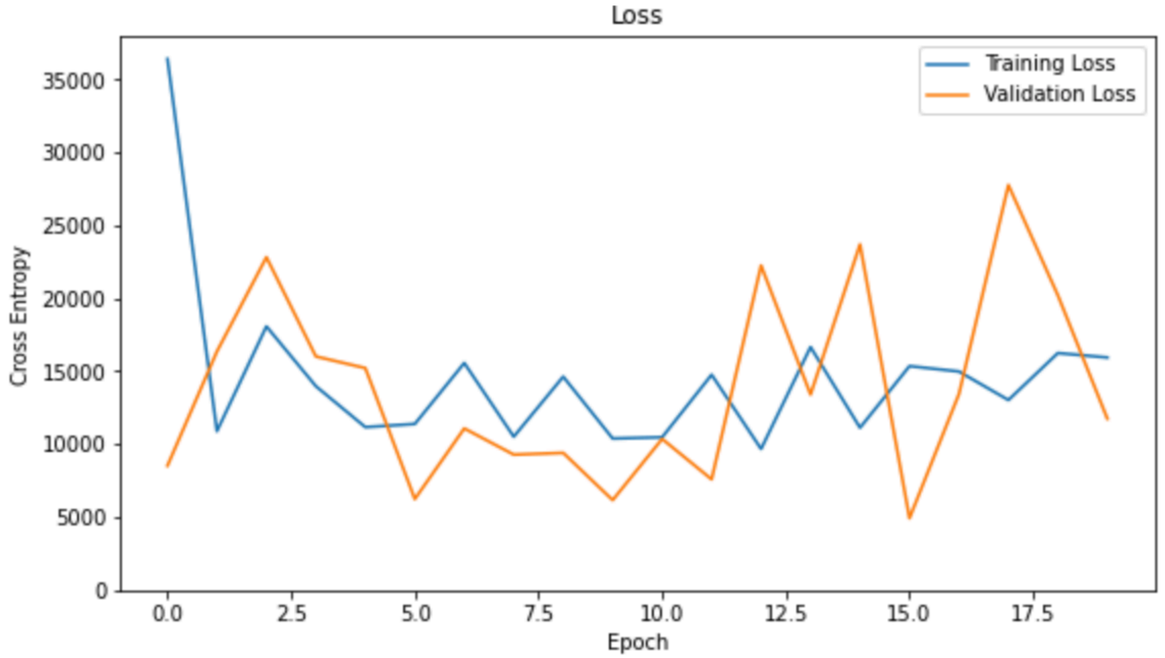
- Number of epochs: 80

- The monitored metrics are : training accuracy, training loss, validation accuracy and validation loss
- Loss function is: Categorical CrossEntropy
- The model with the highest validation accuracy is saved at each epoch
- To prevent overfitting the learning rate is adjusted to 0.2 x lr after 8 epochs without validation accuracy improvement

Training

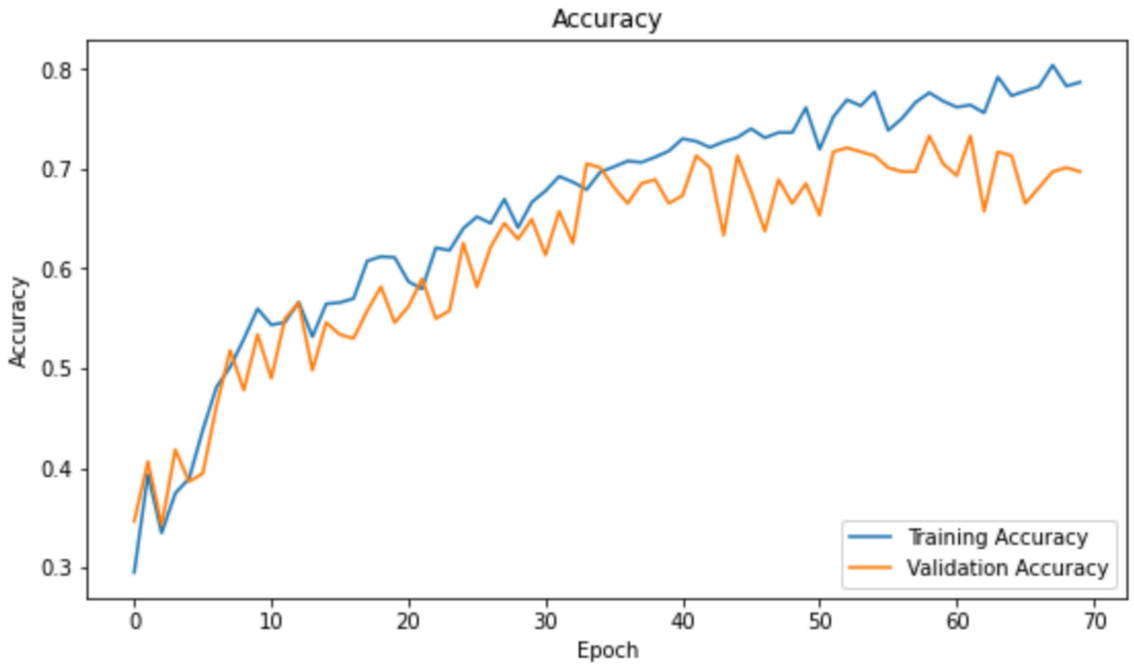
LR model

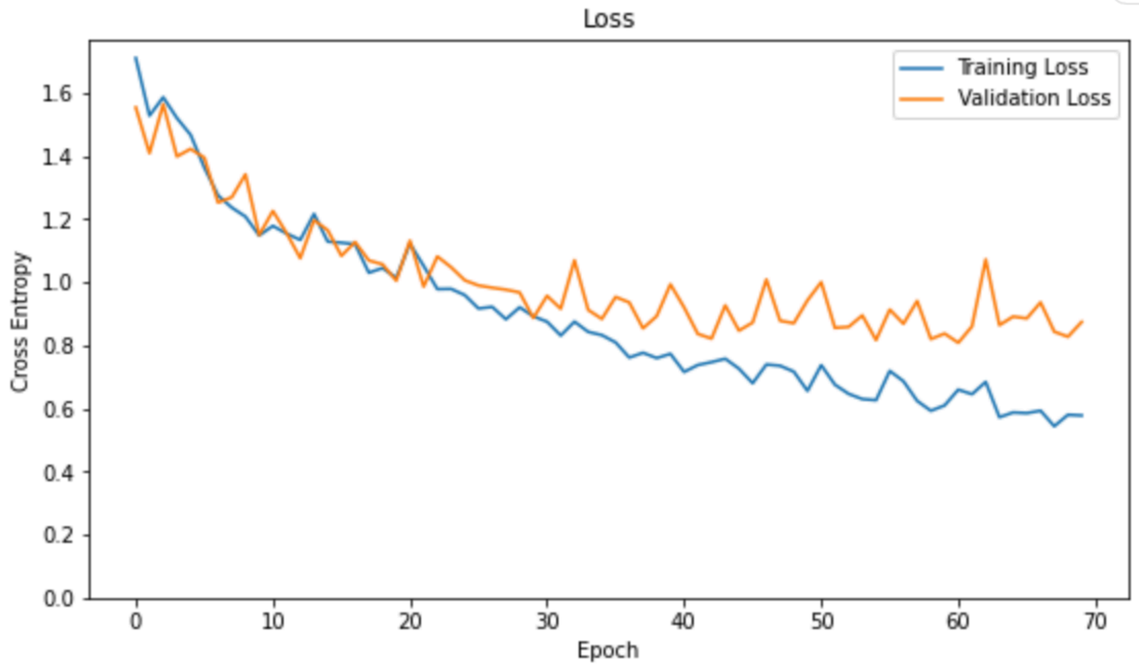




Training

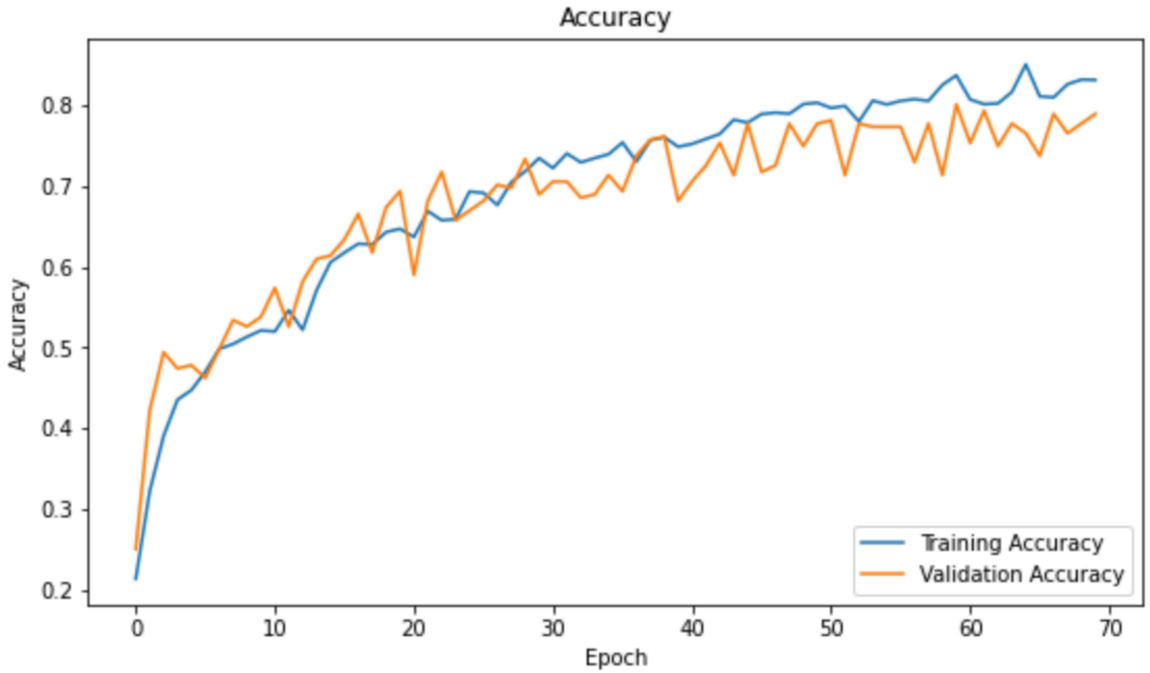
CNN model

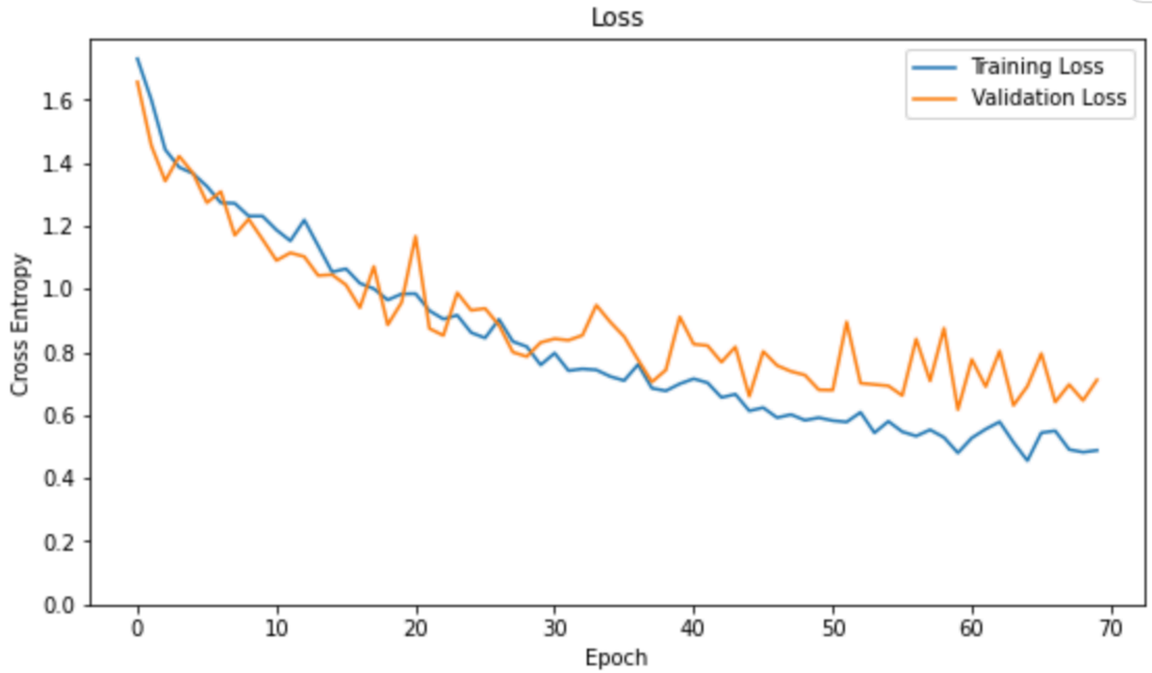




Training

CNN with normalized data





Evaluation

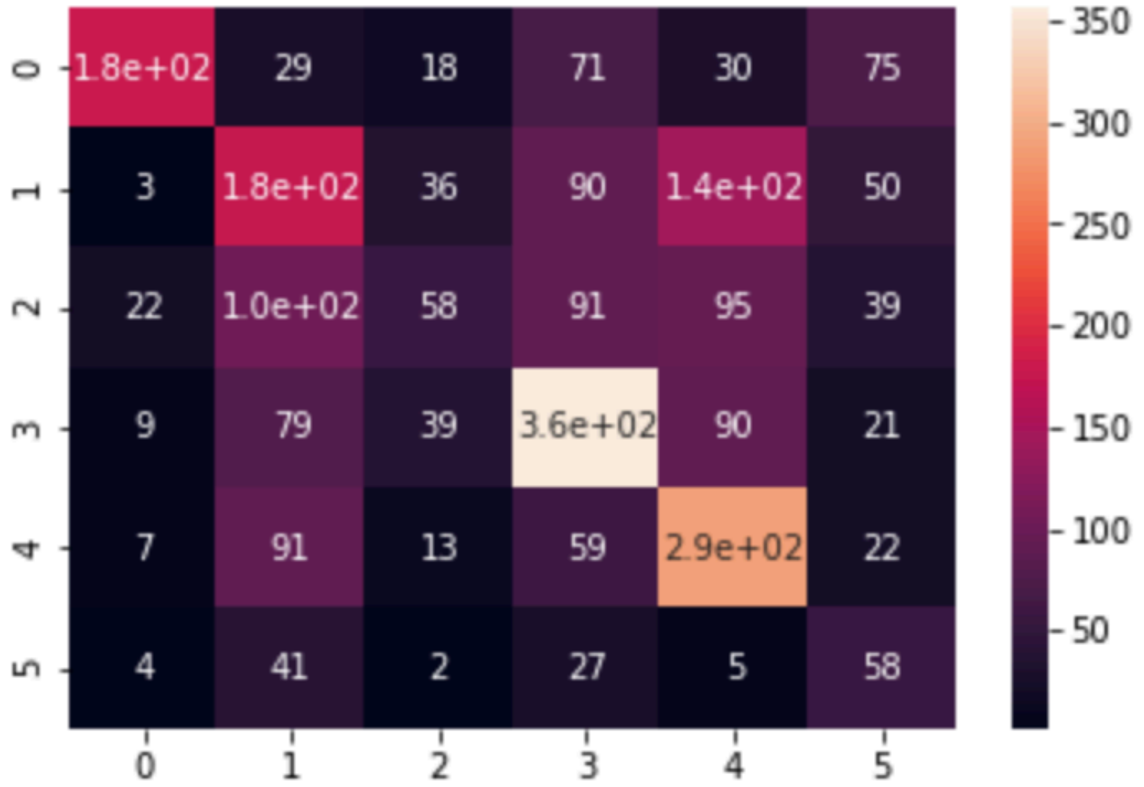
Performance indicators

- **Accuracy**:
  - Validation Accuracy represents the percentage of correct predictions made on the validation set
- **Confusion Matrix**
  - Visual representation of the performance on each class
- **F1 Score**
  - Weighted average between accuracy and sensitivity

Evaluation

LR Results

- The Linear Regression model performed well, considered its size and complexity
- Accuracy : > 35%
- F1 Score : 0.22



Evaluation

CNN Results

- As expected the CNN without normalization performs much better than the LR model
- Accuracy : 71%
- F1 Score: 0.83

Evaluation

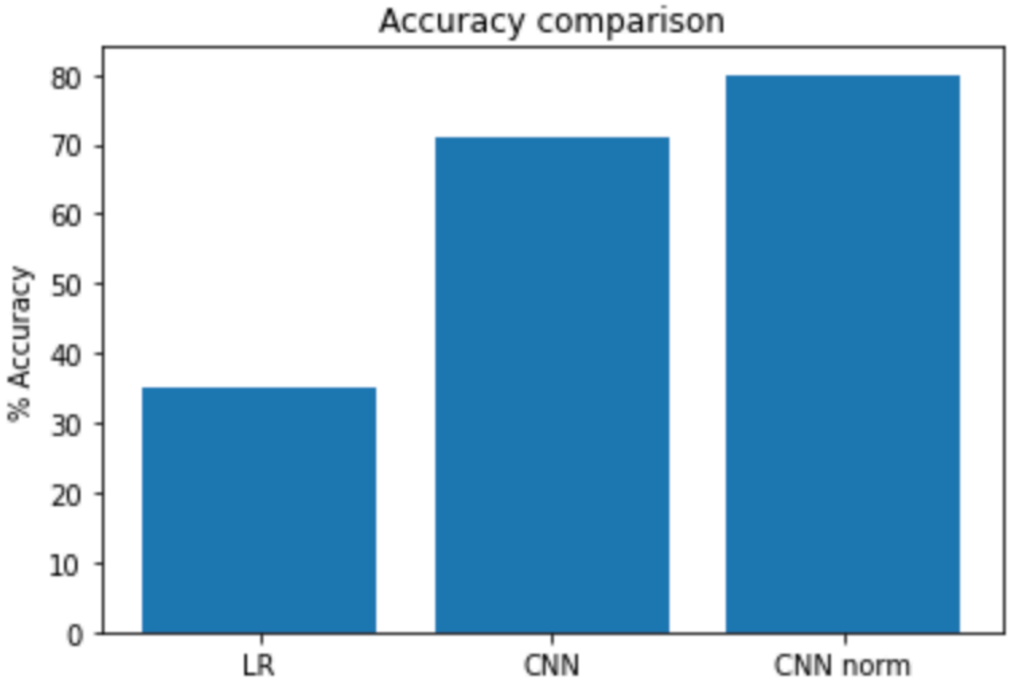
Normalized CNN Results

- The CNN with normalisation perform best with highest score in each monitored metric
- Accuracy : 80%
- F1 Score: 0.89

Results

Accuracy comparison

- The accuracy of the CNN with normalised data is 80%
- It shows a 12% increase with respect to the CNN without normalisation
- And a 128% increase with respect to the Logistic Regression model



Results

F1 Score comparison

- The F1 Score of the CNN with normalised data is 0.89
- It shows a 7% increase with respect to the CNN without normalisation
- And a 404% increase with respect to the Logistic Regression model