Age Estimation using Facial Images

Exploring Deep CNN models for human age estimation

Hammad Mohiuddin

Byeong Jun Kim

Alibek Taizhanov

April 11, 2019

Table of Contents:

1.0 Goal and Motivation

This project seeks to develop a machine learning software to predict an individual's age given a single facial photograph. Humans are unable to accurately estimate another's age just by examining their face. Our decisions are based on many different factors, hence it is difficult to make an accurate age prediction based solely on an image. Successful execution of such a project will help define the boundaries of machine capability to perform tasks that humans may struggle with.

Additionally, facial recognition technologies and associated advancements in age, gender and race determination applications have seen a surge in the fields of biometrics, security, entertainment and other sectors [1-3]. Specifically, age estimation can be used in security surveillance systems, law enforcement, human-computer interaction and digital access control [4].

However, it is challenging to identify a person's age from facial images as face aging reflects not only genetic and age-invariant features but also a variety of other factors. The team hypothesizes that a trained machine with a systematic model can learn to recognize the implicit pattern of aging within humans. It is believed that such a model would perform better than a human in an age estimation task as its decisions are based solely on facial aging patterns in humans and not on external biases.

Machine learning is considered to be a suitable tool for the mentioned goal, due to its ability to learn implicit patterns in data, specifically human aging patterns for this project. Rather than explicitly embedding human aging intuitions into a program, facial data can be used to infer patterns. Also the availability of large and diverse facial image datasets allows for the development of scalable software.

2.0 Data Collection and Cleaning

The data for this project was obtained from the UTKFace dataset which consists of over 20000 facial images [5]. Images are labelled with age, ranging from 0 - 116 years old, gender, either Male or Female and race, categorized as White, Black, Asian, Indian, and Others (such as Hispanic, Latino, Middle Eastern). The dataset provides images in a standardized format with image size 200 x 200 pixels and RGB colors, however there is a large variation in image quality, pose and facial expression, making the dataset quite diverse. There will be a further discussion on how low quality images affect training in Section 7.0.

Due to an uneven distribution of images by age, gender and race, the data had to be organized into subfolders based on the image attributes (e.g. a subfolder for 50 year old White Males). The training, validation and test sets were created with a 70-15-15 split using proportional number of samples from each subfolder. This reduces sampling bias towards certain subgroups in the training, validation and test sets. Data augmentation was also utilized to make up for the lack of data in certain subgroups. In this case, the images were randomly rotated or flipped horizontally.

Also, the default ImageFolder data loader was extended to assign proper labels to age folders. The data was organized into folders labelled with age as integers, however the loader ordered the labels alphanumerically and produced an incorrect mapping of age to label. To rectify the problem, the ImageFolder data loader class was extended to order folder names numerically as integers, hence giving an exact mapping from age to label.

3.0 Final Software Structure

The software is divided into several modules, each performing a separate task in the pipeline. The overall structure of the software pipeline can be seen in Figure 1.

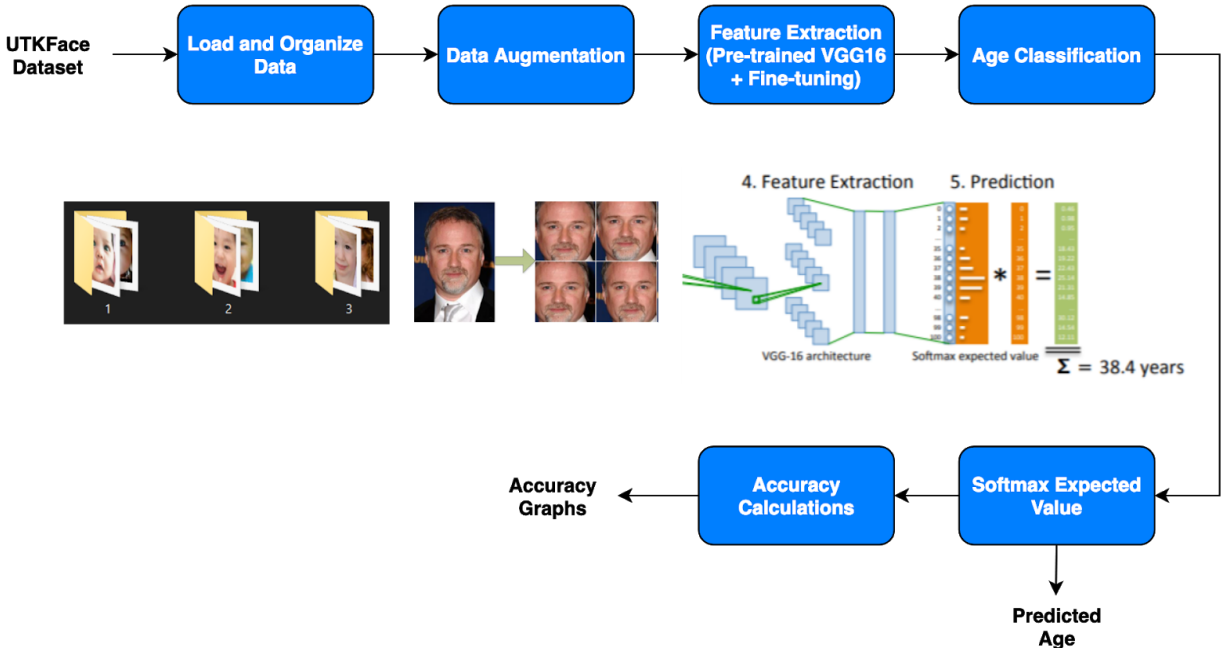

Figure 1: Top-level structure of age estimation software

Modules and description of tasks can be seen in Table 1.

Table 1. Description of modules in software pipeline

Module	Description
Load and Organize Data	The dataset is extracted and sorted into folders by age. To ensure fairness, each folder contains proportional number of images from each sub-group i.e. race and gender. The training, validation and test sets were obtained using a 70-15-15 split of the data.
Data Augmentation	Data augmentation was achieved by applying random transformations such as rotation and horizontal flips to images.

Feature Extraction	The data was then passed through the pretrained VGG16 model to extract features. This model was also fine-tuned by updating weights during training.
Age Classification	The feature maps obtained are then reshaped and passed through a sequence of fully-connected layers, with the output layer having 95 neurons, one for each age from 1 to 95.
Softmax Expected Value	The output activations were then passed through a softmax layer to obtain a probability for each age class. The expected value of the age is calculated by multiplying the probabilities by corresponding ages to obtain a single number age estimate. Refer to Section 4.3 for calculations.
Accuracy Calculations	A direct comparison of labels with continuous model predictions would not yield a good metric for measuring a models' performance. Hence, to calculate accuracy the $R^2 = 1 - \frac{\Sigma(label-prediction)^2}{\Sigma(label-mean)^2}$ (coefficient of determination) value was used. This is a popular way of calculating an accuracy for a regression model. R^2 value ranges from $-\infty$ to 1, where 1 indicates 100% accuracy. While training it should be observed that the R^2 value approaches 1. The estimates and labels are passed to the accuracy module, which determines exact accuracy and offset accuracy (+/- 1, +/- 5, +/- 10 years) for the predictions. As it is difficult to predict an individual's age exactly, the offset accuracy values help greatly in measuring the performance of the model.

4.0 Development Process and Model Selection

This section describes the development process for both the proposed and considered models, while providing details on the training, validation and testing of the final model.

4.1 Baseline Model

In the initial stages of the project, the team developed a baseline model which was later used as a minimum performance threshold. The baseline model consisted of 3 convolutional layers with maxpool layers in between. The 3 RGB image channels were expanded to 12, 48 and 96 channels by the three convolutional layers respectively. This sequence of convolutional layers was followed by 4 fully-connected layers with 1 output neuron to produce a single number age prediction [Appendix A].

4.2 Architecture Selection and Transfer Learning

After the baseline model, transfer learning was explored. Various different pretrained CNN-based models were compared to determine best performing candidate architecture for further development. Table 2 contains performance metrics for some of the tested architectures, baseline model and human performance as measured in a controlled manner [6].

Table 2. Age estimation test accuracies for major CNN architectures, humans and baseline model

	Exact Accuracy	+/- 1 years Accuracy	+/- 5 years Accuracy
Baseline	6%	11%	60%
Alexnet	8%	13%	59%
VGG16	13%	20%	71%
VGG19	10%	16%	65%
ResNet	12%	18%	67%
Human [6]	<10%	<15%	<50%

All models were trained and tested with identical data and output handling. The VGG16 architecture had consistently better performance, so all further development used this model.

The values in the table were calculated using final/proposed model specifications[1] and techniques described in later sections. Nevertheless, similar relative performance between architectures was observed using other and less suitable specifications/techniques tested.

4.3 Regression vs. Classification

Initially, a regression-based approach was considered using a fully-connected output layer with a single neuron. The baseline model as well as the initial VGG16 models used this approach. Later experimentation showed that a classification-based approach, where the prediction is the highest probability class, provides better accuracies with 2-3% improvement over regression. Next, the literature on age estimation suggests that a softmax expected value output performs better than both regression and classification [7]. Expected value is obtained by multiplying all class probabilities by the corresponding age label and summing the results (Figure 2). Classes were defined to be discrete ages from the dataset labels (1 - 95 years old or 95 classes). Implementing this approach confirmed the literature claims and marginally improved performance.

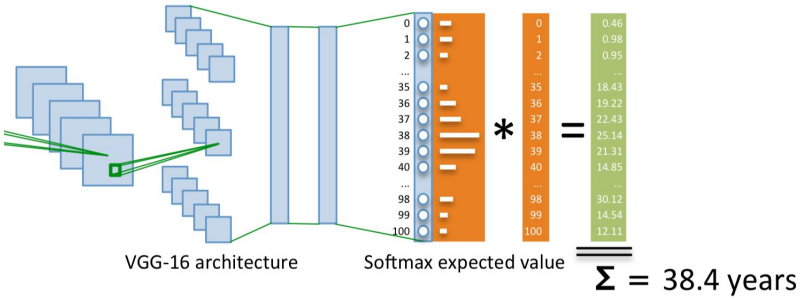

Figure 2: Expected Value Calculation [7]

[1] Includes loss function, learning rate, model output handling and other variables

4.4 Loss Function

Among mainstream loss functions, MSE and MAE losses are the most widely used for age estimation [7]. Both of these loss functions yielded comparable results in this project, except that MAE took longer to converge during training due to its uniform gradient [Appendix B]. Therefore, MSE loss was used throughout the project progression.

5.0 Proposed Model and Results

The final model is based on the VGG16 architecture with a custom classifier consisting of 4 fully-connected layers with the output layer having 95 neurons. The model, excluding the classifier, is loaded with weights pre-trained on the VGGFace dataset for general face recognition problems [8].

While training, the model was set to training mode so that the preloaded weights could also be fine-tuned. In order to achieve faster convergence, the learning rate was initially set to *1e-4* in the first 5 epochs and then later reduced to *1e-5* and ultimately *7e-6*. Multiple batch sizes were tested and the model performed best with a batch size of 32. Training was halted when overfitting was observed i.e. validation accuracy stopped improving while training accuracy still increased.

The final model results can be seen from Figure 3 as well as the model predictions for a set of randomized test data from Figure 4. The final VGG16 model outperformed the human benchmark for age estimation, hence proving our initial hypothesis [Table 2].

```
+/- 1 years accuracy: 20.23%
+/- 5 years accuracy: 71.02%
+/- 10 years accuracy: 88.82%
```

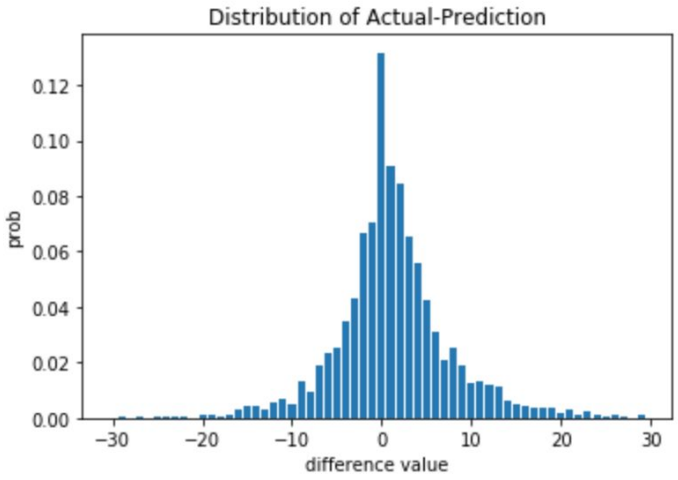

Figure 3: Prediction Accuracy Distribution on Test Dataset

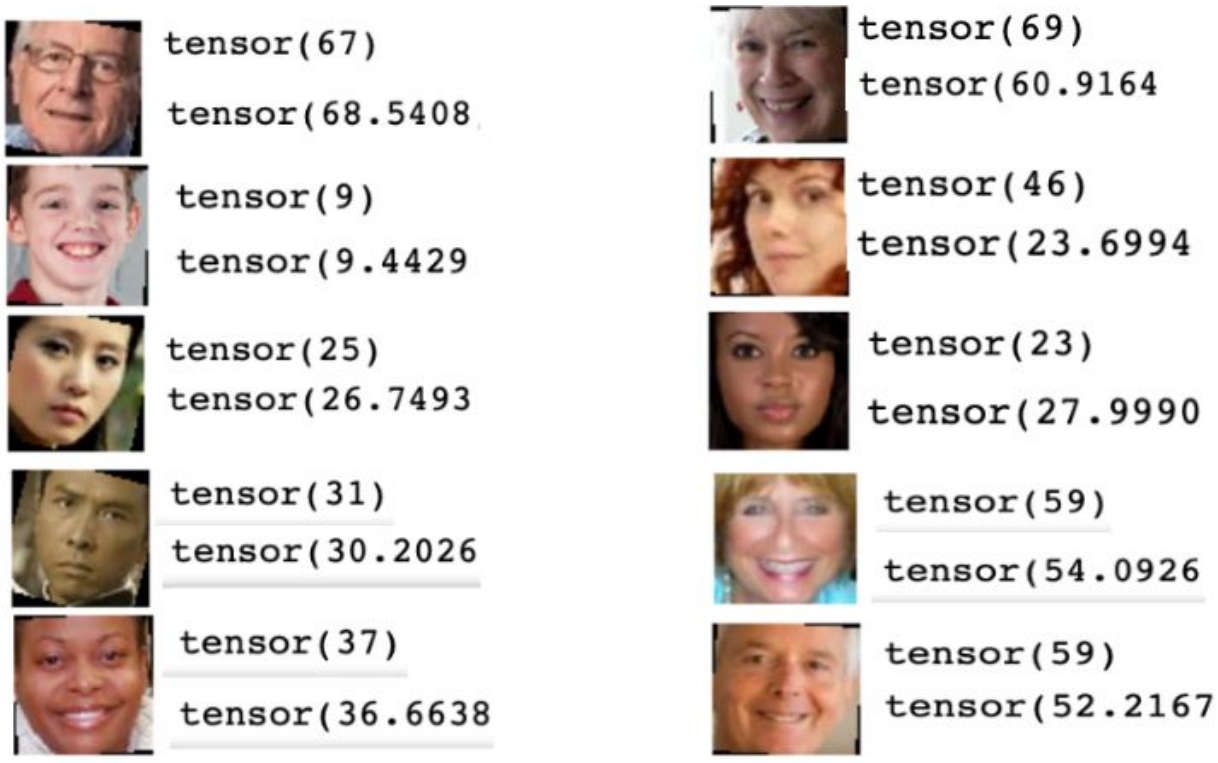

Figure 4: Model Performance on Randomized Test Data. Model performs well on images on the Left and underperforms on images on the Right

6.0 Ethical Issues

The applications and impact of this project could provoke ethical controversy due to the issues of fairness and equality. This is mainly due to the fact that an age estimation software can easily be biased in favor of certain groups of people if not trained in a fair and appropriate manner.

From Figure 5 it is apparent that the model underperformed on middle-aged people. This can be explained by observing that there is a slow and subtle progression of age in these age groups. Conversely, the model performed well on younger age groups as there is a much more dramatic change in facial features. This variance in performance must be controlled and the model must be trained to perform equally well on all age groups.

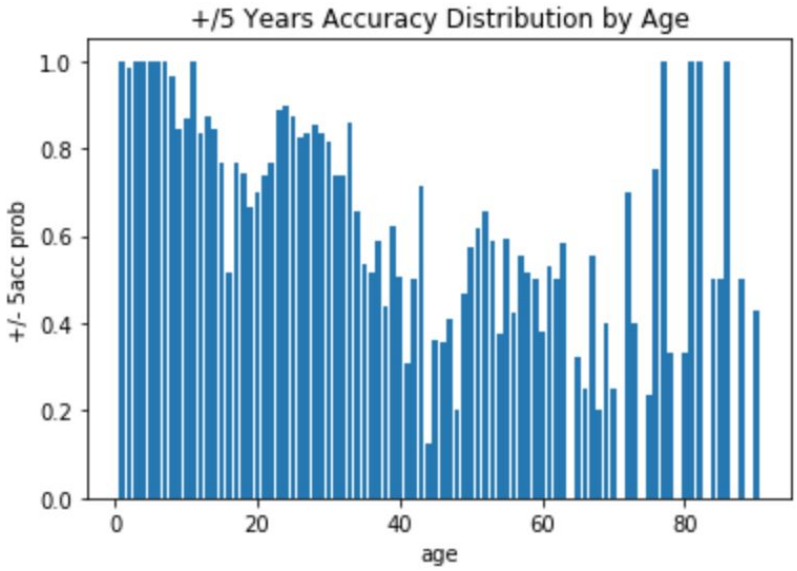

Figure 5: +/-5 Years Prediction Accuracy for Different Ages [1-95]

The gender based performance is comparable [Figure 6], with the model slightly underperforming on male subjects. This may not be a problem by itself, but when combining low performing age groups with male gender, the impact may increase ethical considerations.

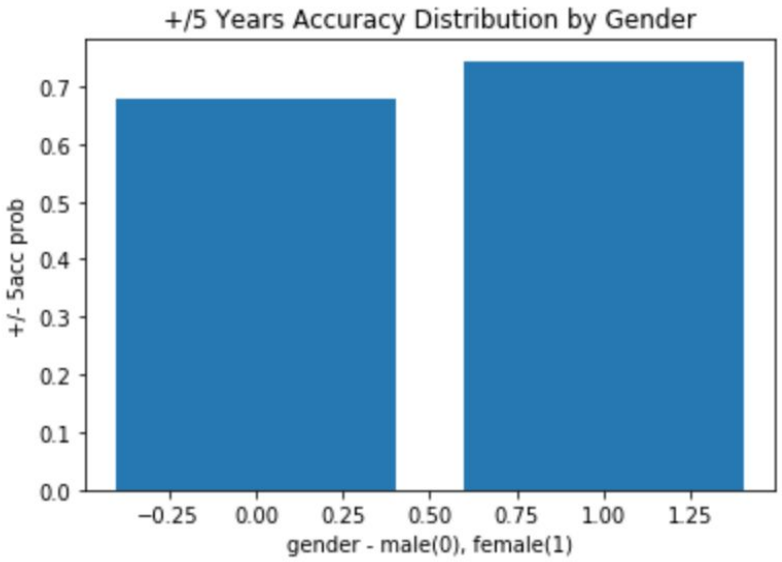

Figure 6: +/-5 Years Prediction Accuracy for Males and Females

The ethical issues pertaining to this project could be resolved by ensuring equal distribution of data for all sub-groups as well as by using a more diverse dataset.

7.0 Key Learnings

The final model performed well on younger age groups but not so well on older ones. A major cause for this was the large variation in image quality in the dataset as well as the lack of data in some age groups.

The team learnt the importance of having a high quality diverse dataset. Lack of such data and the existence of low-quality data for certain age groups hindered the model's performance greatly [Figure 7]. Data cleaning and preprocessing of images should have been a bigger focus for the project. This would include removing low-quality images as well as using techniques for face detection and alignment for

better data augmentation. The team would also consider adding data from other datasets to supplement for the lack of data in certain subgroups. This would ensure that the model is trained on more diverse data with variations in face/facial expression, pose, makeup, glasses as well as natural human features, such as facial hair, skin color/tone etc. With sufficient high-quality data for all age groups and a diverse set of images, the model would generalize better.

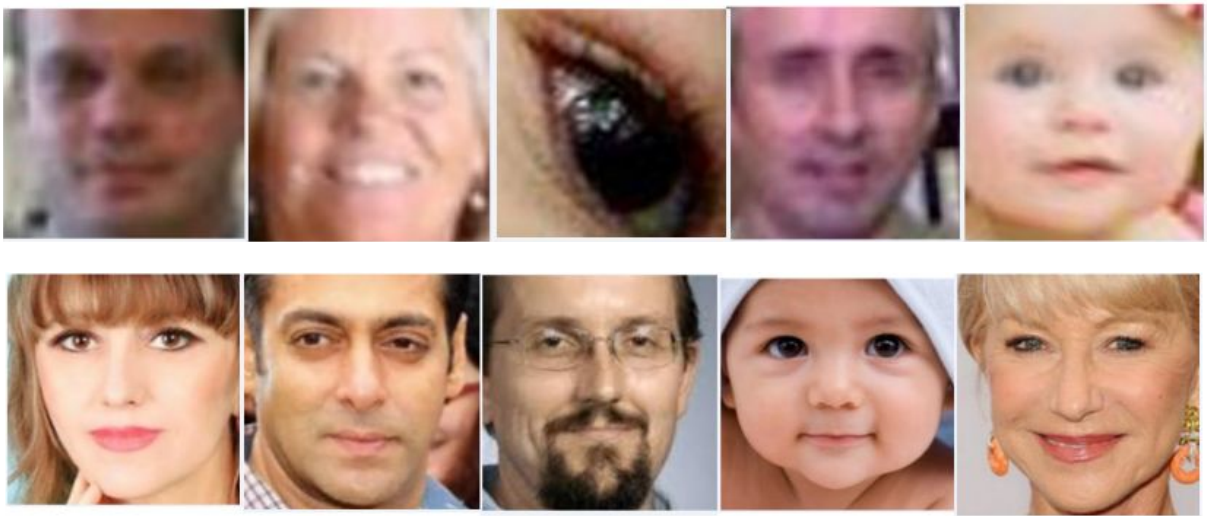

Figure 7: Inconsistent Data Quality in UTKFace Dataset

References

[1] G. Guo, Y. Fu, T. S. Huang, "Age synthesis and estimation via faces: A survey," IEEE transactions on pattern analysis and machine intelligence, vol. 32, 2010.

[2] X. Liu, H. Han, C. Otto, A. K. Jain, "Demographic estimation from face images: Human vs. machine performance," IEEE transactions on pattern analysis and machine intelligence, vol. 37, 2015.

[3] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015.

[4] Q. Zakariya, A. A. Mallouh, B. D. Barkana, “Deep Convolutional Neural Network for Age Estimation based on VGG-Face Model” arXiv preprint arXiv:1709.01664, 2017.

[5] Z. Zhang, Y. Song, H. Qi, “UTKFace - Large Scale Face Dataset”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. Available: https://susanqq.github.io/UTKFace/

[6] H. Han, C. Otto, X. Liu, A. K. Jain, "Demographic Estimation from Face Images: Human vs. Machine Performance", IEEE Trans. PAMI, Vol. 37, No. 6, 2015.

[7] R. Rothe, R. Timofte, L. V. Gool. "Dex: Deep expectation of apparent age from a single image." Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015.

[8] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman. "Deep Face Recognition." BMVC. Vol. 1. No. 3. 2015

Appendix

Appendix A:

```
self.conv1 = nn.Conv2d(3, 12, 5)
self.pool1 = nn.MaxPool2d(5, 5)
self.conv2 = nn.Conv2d(12, 48, 5)
self.pool2 = nn.MaxPool2d(2, 2)
self.conv3 = nn.Conv2d(48, 96, 5)
self.fc1 = nn.Linear(3456, 3456)
self.fc2 = nn.Linear(3456, 1024)
self.fc3 = nn.Linear(1024, 256)
self.fc4 = nn.Linear(256, 1)
```

Figure 1: Baseline model layers as specified by Pytorch

Appendix B:

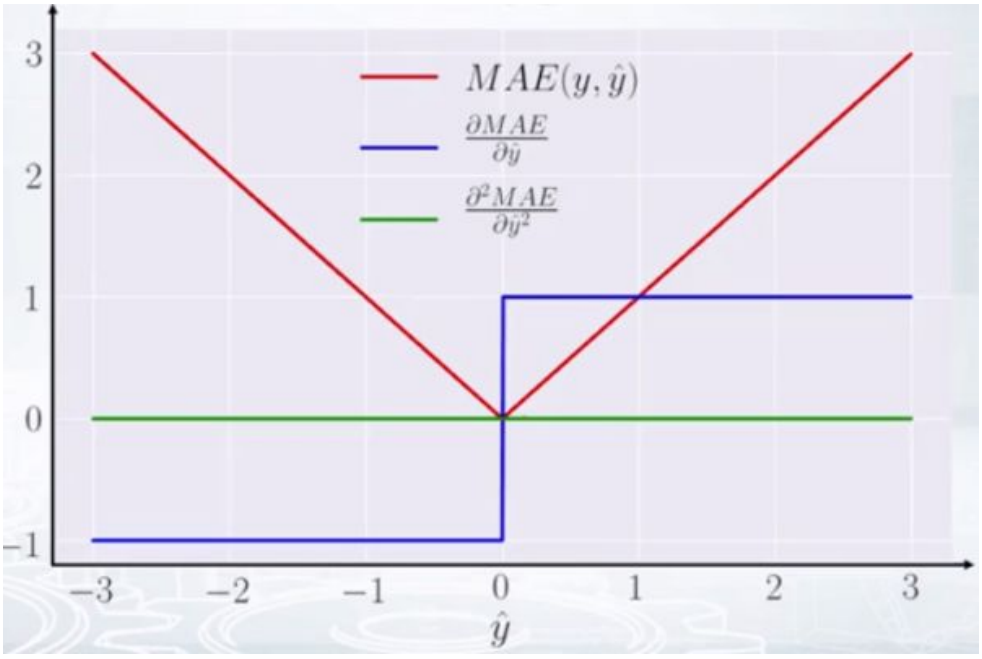

Figure 2: MAE Loss Gradient