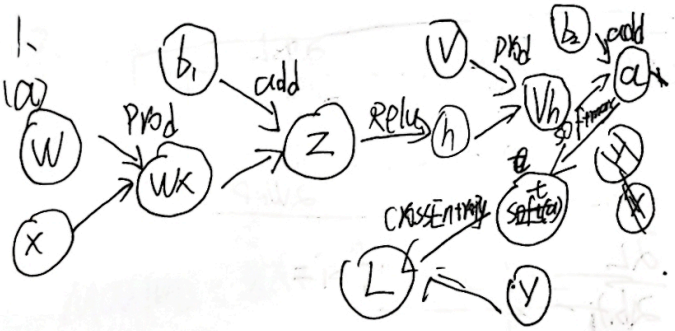
1.



(a) $a = Vh + b_2$

$a_z = \sum_{j=1}^{K} V_{zj} h_j + (b_2)_z$

$\frac{\partial a_z}{\partial V_{ij}} = \begin{cases} 0 & \text{if } i \neq z \\ h_j & \text{if } i = z \end{cases}$

(b) Let $t = \text{softmax}(a)$

$t_i = \frac{e^{a_i}}{\sum_z e^{a_i}}$

2L Let y be the class index

$L = \text{crossEntropy}(y, t) = -\sum_{i=1}^{C} I(i=y) \log t_i$

① $\frac{\partial L}{\partial t_i} = \left[-I(1=y)\frac{1}{t_1}, -\frac{I(2=y)}{t_2}, \cdots, -\frac{I(i=y)}{t_i}, \cdots, \frac{I(C=y)}{t_C} \right]$

$\frac{\partial t_i}{\partial a_i} = \frac{\partial L}{\partial a_i} = \frac{\partial L}{\partial t_i} \cdot \frac{\partial t_i}{\partial a_i}$

$\frac{\partial t_i}{\partial a_i} = \left[-\frac{e^{a_i} \cdot e^{a_i}}{(\sum e^{a_i})^2}, -\frac{\exp(a_2 + a_i)}{(\sum e^{a_i})^2}, \cdots \right]$

$\frac{\partial t_i}{\partial a_i} = \left[-\frac{e^{a_1 + a_i}}{(\sum e^{a_i})^2}, -\frac{e^{a_2 + a_i}}{(\sum e^{a_i})^2}, \cdots, \frac{e^{a_i + a_i} - e^{a_i}}{(\sum e^{a_i})^2}, \cdots \right]$

$\frac{\partial t_i}{\partial a_i} = \begin{cases} -\frac{e^{a_i} e^{a_j}}{(\sum e^{a_k})^2} = -t_i t_j & i \neq j \\ \frac{e^{a_i}(\sum e^{a_k} - e^{a_i})}{(\sum e^{a_k})^2} = t_i(1 - t_j) & i = j \end{cases}$

② $\frac{\partial L}{\partial a_i} = \frac{\partial L}{\partial t} \cdot \left[\frac{\partial t_1}{\partial a_i}, \frac{\partial t_2}{\partial a_i}, \cdots \frac{\partial t_C}{\partial a_i} \right]$

$= -\sum_{j=1}^{C} \frac{I(j=y)}{t_j} \frac{\partial t_j}{\partial a_i} = \begin{cases} t_i t_y \cdot -\frac{1}{t_y} = -t_i & i \neq y \\ -\frac{(1-t_y)}{t_y} t_i & i = y \end{cases}$

$= \begin{cases} t_i t_y \cdot (-\frac{1}{t_y}) = -t_i & \text{if } i \neq y \\ -\frac{(1-t_y)}{t_y} t_i & \text{if } i = y \end{cases}$

③ $\frac{\partial L}{\partial V_{ij}} = \frac{\partial L}{\partial a} \cdot \left[\frac{\partial a_1}{\partial V_{ij}}, \frac{\partial a_2}{\partial V_{ij}}, \cdots \frac{\partial a_C}{\partial V_{ij}} \right]$

$= \sum_{z=1}^{C} \frac{\partial L}{\partial a_z} \cdot \frac{\partial a_z}{\partial V_{ij}}$

where $\frac{\partial L}{\partial a_z} \cdot \frac{\partial a_z}{\partial V_{ij}} = \begin{cases} 0 & \text{if } i \neq z \\ -t_i h_j & \text{if } i = z \text{ and } z \neq y \\ -\frac{(1-t_y)}{t_y} t_z h_j & \text{if } i = z \text{ and } z = y \end{cases}$

$\nabla_V L = \begin{bmatrix} \frac{\partial L}{\partial V_{11}} & \frac{\partial L}{\partial V_{21}} & \cdots & \frac{\partial L}{\partial V_{C1}} \\ \frac{\partial L}{\partial V_{12}} & \frac{\partial L}{\partial V_{22}} & \cdots & \frac{\partial L}{\partial V_{C2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial L}{\partial V_{1K}} & \frac{\partial L}{\partial V_{2K}} & \cdots & \frac{\partial L}{\partial V_{CK}} \end{bmatrix}$

④ $\frac{\partial a_z}{\partial (b_2)_i} = \begin{cases} 0 & i \neq z \\ 1 & i = z \end{cases}$

$\frac{\partial L}{\partial (b_2)_i} = \frac{\partial L}{\partial a} \cdot \sum_{z=1}^{C} \frac{\partial L}{\partial a_z} \cdot \frac{\partial a_z}{\partial (b_2)_i}$

where $\frac{\partial L}{\partial a_z} \cdot \frac{\partial a_z}{\partial (b_2)_i} = \begin{cases} 0 & \text{if } i \neq z \\ -t_z & \text{if } i = z \text{ and } z \neq y \\ -\frac{(1-t_y)}{t_y} t_z & \text{if } i = z \text{ and } z = y \end{cases}$

$\nabla_{b_2} L = \begin{bmatrix} \frac{\partial L}{\partial (b_2)_1} & \frac{\partial L}{\partial (b_2)_2} & \cdots & \frac{\partial L}{\partial (b_2)_C} \end{bmatrix}$

$\frac{\partial L}{\partial h_j} = \sum_{z=1}^{C} \frac{\partial L}{\partial a_z} \cdot \frac{\partial a_z}{\partial h_j}$

$\frac{\partial a_z}{\partial h_j} = V_{zj}$

$\frac{\partial L}{\partial h_j} = \begin{cases} -t_i V_{ij} + \frac{(1-t_y)}{t_y} t_i V_{ij} & \text{if } i = y \\ -t_i V_{ij} & \text{if } i \neq y \end{cases}$

$= \begin{cases} -t_i V_{ij} + \frac{(1-t_y)}{t_y} t_i V_{ij} & \text{if } i = y \\ -t_i V_{ij} & \text{if } i \neq y \end{cases}$

$$\frac{\partial a_z}{\partial h_j} = V_{zj}$$

$$\frac{\partial L}{\partial a_z}\frac{\partial a_z}{\partial h_j} = \sum_{z=1}^{C}\frac{\partial L}{\partial a_z}\frac{\partial a_z}{\partial h_j} = \begin{cases} -a_z V_{zj} & \text{if } z \neq y \\ \frac{(1-t_y)}{t_y} a_z V_{zj} & \text{if } z = y \end{cases}$$

$$\frac{\partial a_z}{\partial h_j} = V_{zj}$$

⑤ $$\frac{\partial L}{\partial h_j} = \sum_{z=1}^{C}\frac{\partial L}{\partial a_z}\frac{\partial a_z}{\partial h_j}$$

where $$\frac{\partial L}{\partial a_z}\frac{\partial a_z}{\partial h_j} = \begin{cases} -a_z V_{zj} & \text{if } z \neq y \\ \frac{(1-t_y)}{t_y} a_z V_{zj} & \text{if } z = y \end{cases}$$

$$\frac{\partial h_i}{\partial z_i} = I(z_i > 0)$$

⑥ $$\frac{\partial L}{\partial z_i} = \sum_{i=1}^{K}\frac{\partial L}{\partial h_i}\frac{\partial h_i}{\partial z_i} = \sum I(z_i>0) z_i \frac{\partial L}{\partial h_i},$$

where $\frac{\partial L}{\partial h_i}$ can be obtained from ⑤

$$z_i = \sum_{j=1}^{D} w_{ij} x_j + (b_1)_i$$

$$\frac{\partial z_k}{\partial w_{ij}} = \begin{cases} 0 & \text{if } i \neq k \\ x_j & \text{if } i = k \end{cases}$$

⑦ $$\frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^{K}\frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial w_{ij}}$$

where $$\frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial w_{ij}} = \begin{cases} 0 & \text{if } i \neq k \\ x_j \frac{\partial L}{\partial z_k} & \text{if } i = k \end{cases}$$

and $\frac{\partial L}{\partial z_k}$ can be obtained from ⑥

$$\nabla_W L = \begin{bmatrix} \frac{\partial L}{\partial w_{11}} & \cdots & \frac{\partial L}{\partial w_{K1}} \\ \vdots & & \vdots \\ \frac{\partial L}{\partial w_{1D}} & \cdots & \frac{\partial L}{\partial w_{KD}} \end{bmatrix}$$

$$\frac{\partial L}{\partial (b_1)_i} = \begin{cases} 0 & \text{if } i \neq k \\ \frac{\partial L}{\partial z_k} & \text{if } i = k \end{cases}$$

$$\frac{\partial z_k}{\partial (b_1)_i} = \begin{cases} 0 & \text{if } i \neq k \\ 1 & \text{if } i = k \end{cases}$$

⑧ $$\frac{\partial L}{\partial (b_1)_i} = \sum_{k=1}^{K}\frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial (b_1)_i}$$

where $$\frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial (b_1)_i} = \begin{cases} 0 & \text{if } i \neq k \\ \frac{\partial L}{\partial z_k} & \text{if } i = k \end{cases}$$

and $\frac{\partial L}{\partial z_k}$ can be obtained from ⑥

$$\nabla_{b_1} L = \begin{bmatrix} \frac{\partial L}{\partial (b_1)_1} & \frac{\partial L}{\partial (b_1)_2} & \cdots & \frac{\partial L}{\partial (b_1)_K} \end{bmatrix}$$

2、 after:

(a) Conv5(10): $32\times32\times10$

MaxPool$_2$: $16\times16\times10$

Conv5(10): $16\times16\times10$

MaxPool$_2$: $8\times8\times10$

FC: 10

FC10: 10

(b) Conv5(10): $5\times5\times1\times10=250$

MaxPool$_2$: 0

Conv5(10): $5\times5\times10\times10=2500$

MaxPool: 0

FC10: $8\times8\times10\times10=6400$

Assume no bias is used in the FC10 or Conv5

~~3、 node 0:~~

~~gender: P(boy) $\frac{2}{3}$ P(girl) $\frac{1}{3}$~~

~~Ever~~

Let Y denote overweight N denote not overweight B denote Boy G denote Girl, H denote Hyperlipidemia, NH denote no Hyper... V denote unhealthy diet NV denot no unhealthy diet. E denot exercise NE denot no exercise

gender:

$-P(B)\log$

3、

Let B denote boy, H denote Hyperlipidemia, V denote unhealthy diet, E denote exercise O denote overweight

① node 0:

(1) gender:

boy $\frac{2}{3}$ girl $\frac{1}{3}$

boy entropy:

$-P(O|B)\log(P(O|B)) - P(\bar{O}|B)\log P(\bar{O}|B)$

$= -\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = -\log\frac{1}{2} = \log 2 = 1$

girly entropy:

$-P(O|\bar{B})(\log P(O|B) - P(\bar{O}|\bar{B})\log P(\bar{O}|\bar{B})$

$= -1\log 1 - 0 = 0$

$\frac{2}{3}\times\log 2 + \frac{1}{3}\times 0 = \frac{2}{3}\log 2 = \frac{2}{3} = 0.67$

(2) H

~~H~~ $H = \frac{1}{2}$ $\bar{H} = \frac{1}{2}$

H entropy:

$-P(O|H)\log P(O|H) - P(\bar{O}|H)\log P(\bar{O}|H)$

$= -1\log 1 - 0 = 0$

$\bar{H}$ entropy:

$-P(O|\bar{H})\log P(O|\bar{H}) - P(\bar{O}|\bar{H})\log(\bar{O}|\bar{H})$

$= \frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3} = 0.92$

$\frac{1}{2}\times 0 + \frac{1}{2}\times 0.92 = 0.46$

(3) V

$V = \frac{2}{3}$ $\bar{V} = \frac{1}{3}$

V entropy:

$-P(O|V)\log P(O|V) - P(\bar{O}|V)\log P(\bar{O}|V)$

$= -\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4} \approx 0.81$

O entropy:
$-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = -\log\frac{1}{2} = 1$

$\frac{2}{3} \times 0.81 + \frac{1}{3} \times 1 = 0.87$

E

(4) $\overline{E}$

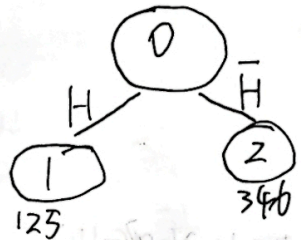
E: $\frac{2}{3}$ $\overline{E}$: $\frac{1}{3}$

E entropy:
$-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = 1$

$\overline{E}$ entropy:
$-1\log 1 - 0 = 0$

$\frac{2}{3} \times 1 + \frac{1}{3} \times 0 = 0.67$

so choose H



② node 1

node 1 is pure, hence no need to split

③ node 2

G: B:

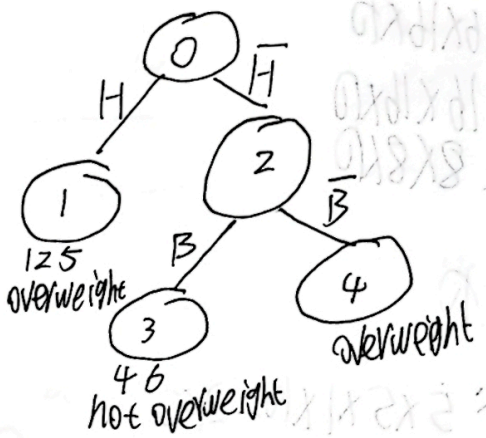
(1) B

B: $\frac{2}{3}$ $\overline{B}$: $\frac{1}{3}$

B entropy:
$-0 - 1\log 1 = 0$

$\overline{B}$ entropy:
$-1\log 1 - 0 = 0$

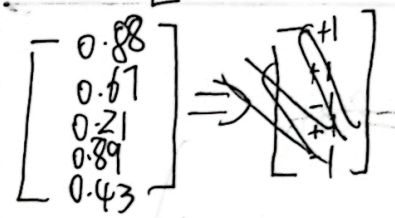
hence select B

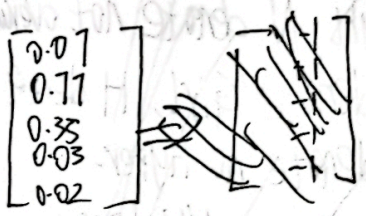


作业4、

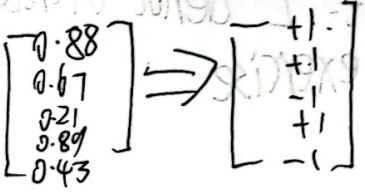
(a)

positive: ~~$-6(0.5\log 0.5 + 0.5\log 0.5) = 0.88$~~

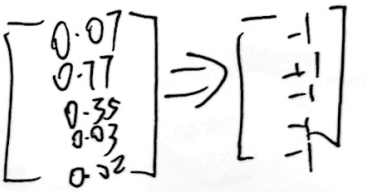




(b)



$Acc = \frac{3+4}{10} = 0.7$



$pre = \frac{3}{4} = 0.75$

$Recall = \frac{3}{5} = 0.6$

$$F1\ score = \frac{2\times 3}{2\times 3+2+1} = \frac{6}{9} = \frac{2}{3} = 0.67$$

Confusion Matrix

	$\hat{P}$	$\hat{N}$
P	3	2
N	1	4

(c) Threshold

① Threshold 0

$$\begin{bmatrix} 0.88 \\ 0.67 \\ 0.2 \\ 0.84 \\ 0.43 \end{bmatrix} \Rightarrow \begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix}$$

$$\begin{bmatrix} 0.07 \\ 0.77 \\ 0.35 \\ 0.03 \\ 0.02 \end{bmatrix} \Rightarrow \begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix}$$

$FPR = \frac{5}{5} = 1$ $TPR = 1$

② Threshold 0.2

$$\begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix} \begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 \\ +1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$FPR = \frac{3}{5}$ $TPR = \frac{2}{5}$

③ Threshold 0.4

$$\begin{bmatrix} +1 \\ +1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \begin{bmatrix} -1 \\ +1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$FPR = \frac{1}{5}$ $TPR = \frac{4}{5}$

④ Threshold 0.6

$$\begin{bmatrix} +1 \\ +1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \begin{bmatrix} +1 \\ -1 \\ +1 \\ +1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 \\ +1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$FPR = \frac{1}{5}$ $TPR = \frac{3}{5}$

⑤ Threshold 0.8

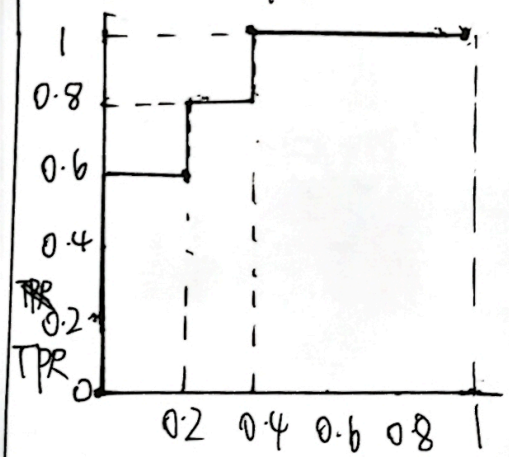
$$\begin{bmatrix} +1 \\ -1 \\ -1 \\ +1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$FPR = 0$ $TPR = \frac{2}{5}$

⑥ Threshold 1

$$\begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$FPR = 0$ $TPR = 0$



(d) $AUC = 0.2\times 0.6 + 0.2\times 0.8\times 0.6 + 0.2\times 0.8 + 0.2\times 1$

$= 0.88$

Report-Assignment2

Overview

In this assignment, I implement two classification algorithms:Decision Tree and Multilayer Perceptron(MLP) neural network using sklearn. Penguins classification and Fashion-MNIST image classification tasks are solved.

Penguins Classification With Decision Tree

Data Cleaning and Analysis

In this task, penguins are classified to different type. I try different decision-tree-based algorithms for this task. I explore basis decision tree, bagging of tree and random forest algorithms. Before training the model, data is first cleaned. Those data with incomplete features are removed from the training and testing set. After cleaning the data, 333 of the 344 samples are left. These data is then splitted into training and testing set. 75% of the 333 samples are used for training while the rest 25% are used for testing. The relation between penguin types and different attributes are shown in the following histogram

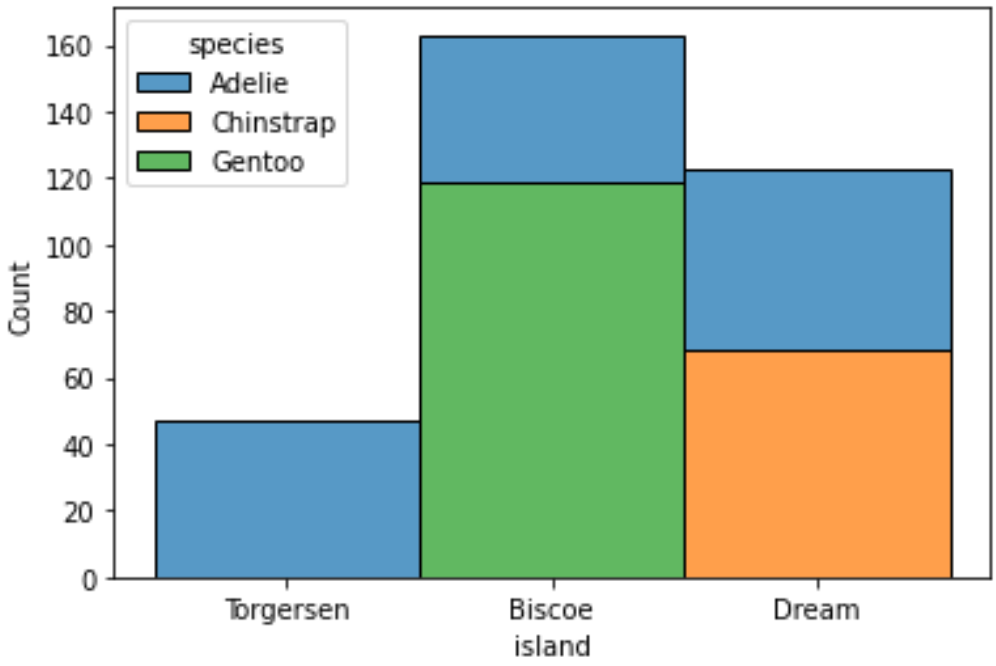


Figure 1: Histogram of number of penguins of different type with different islands

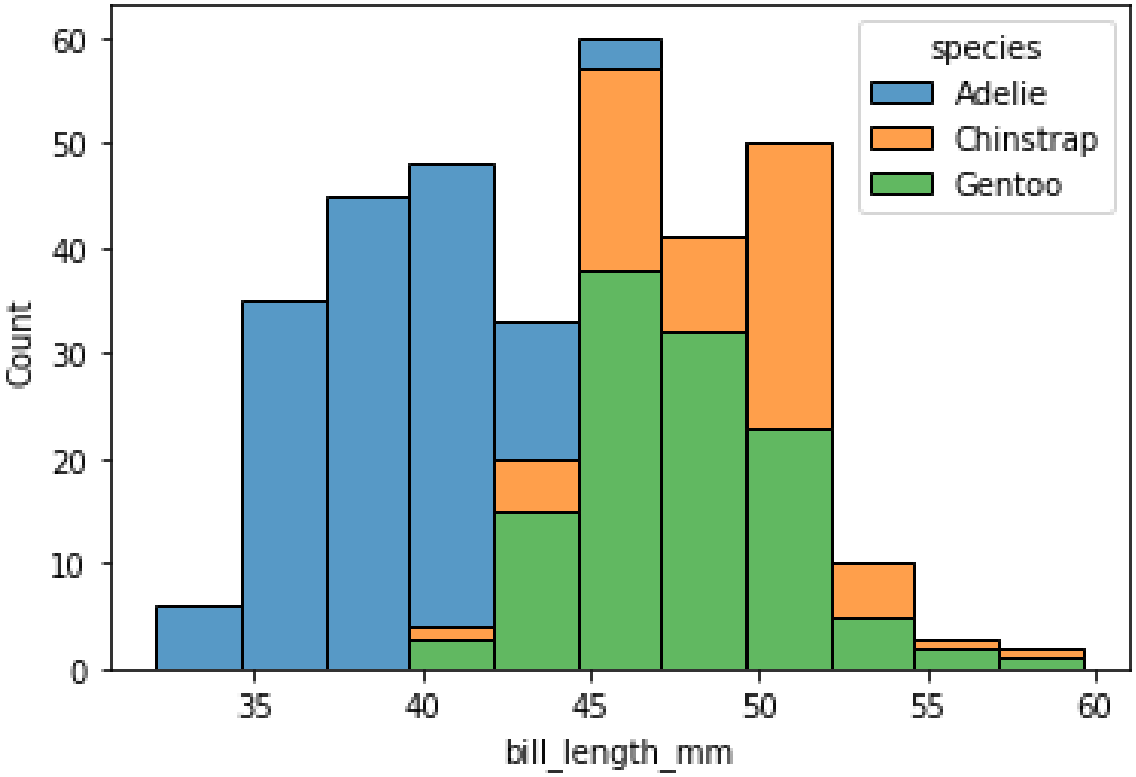


Figure 2: Histogram of number of penguins of different type with different bill length, in mm

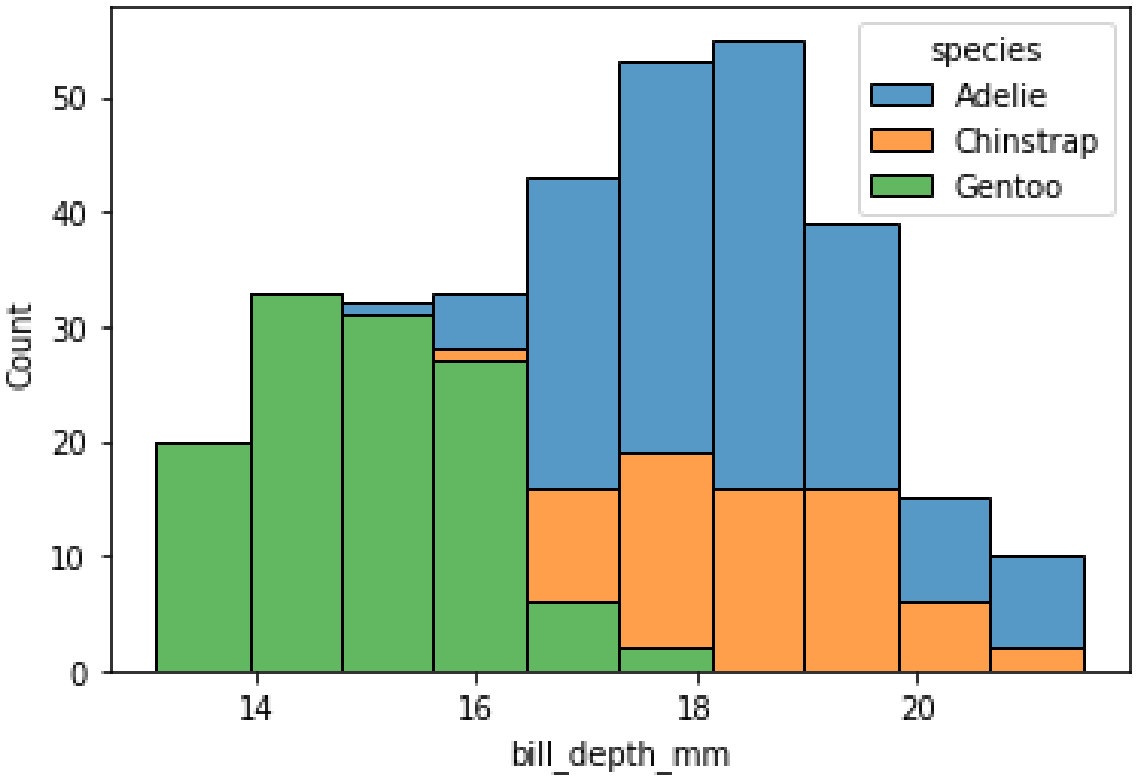


Figure 3: Histogram of number of penguins of different type with different bill depth, in mm

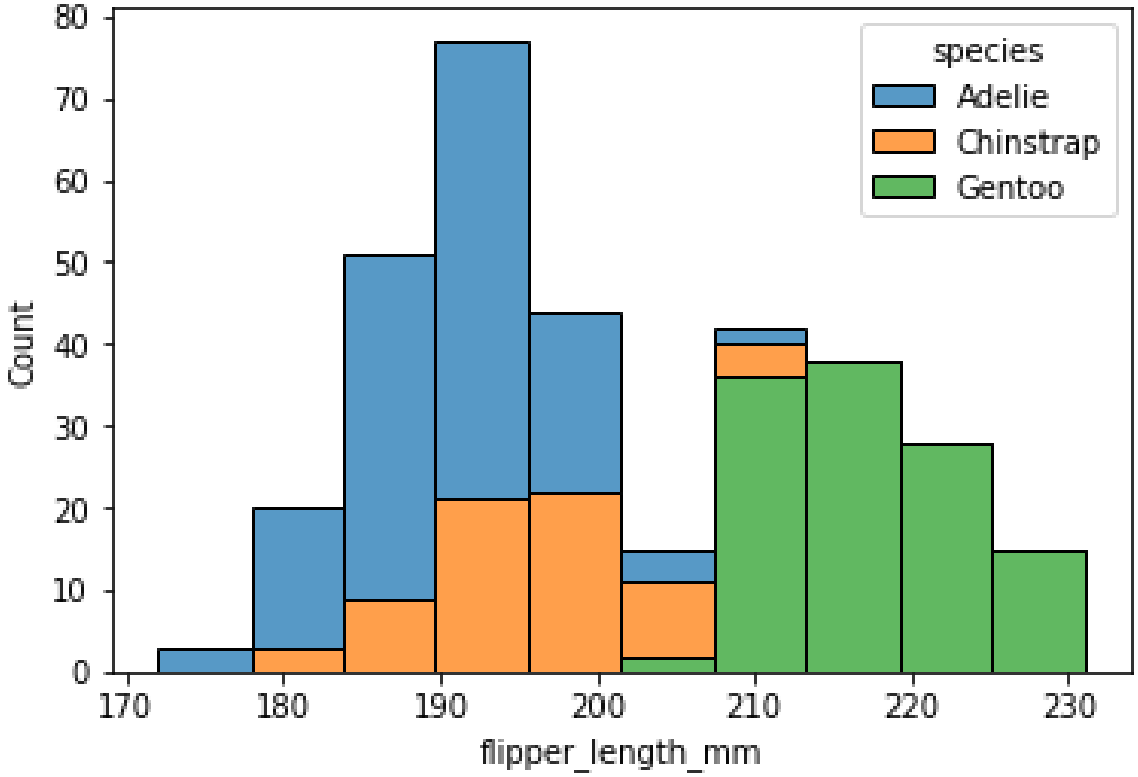


Figure 4: Histogram of number of penguins of different type with different flipper length, in mm

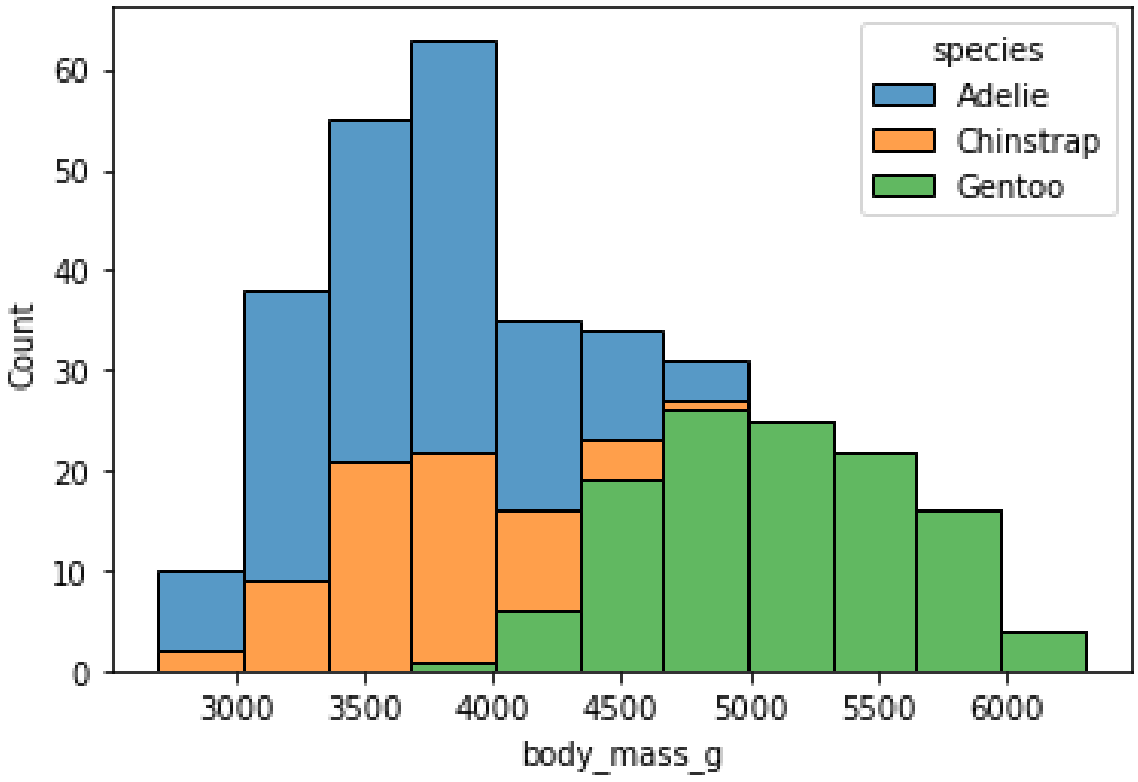


Figure 5: Histogram of number of penguins of different type with different body mass, in g

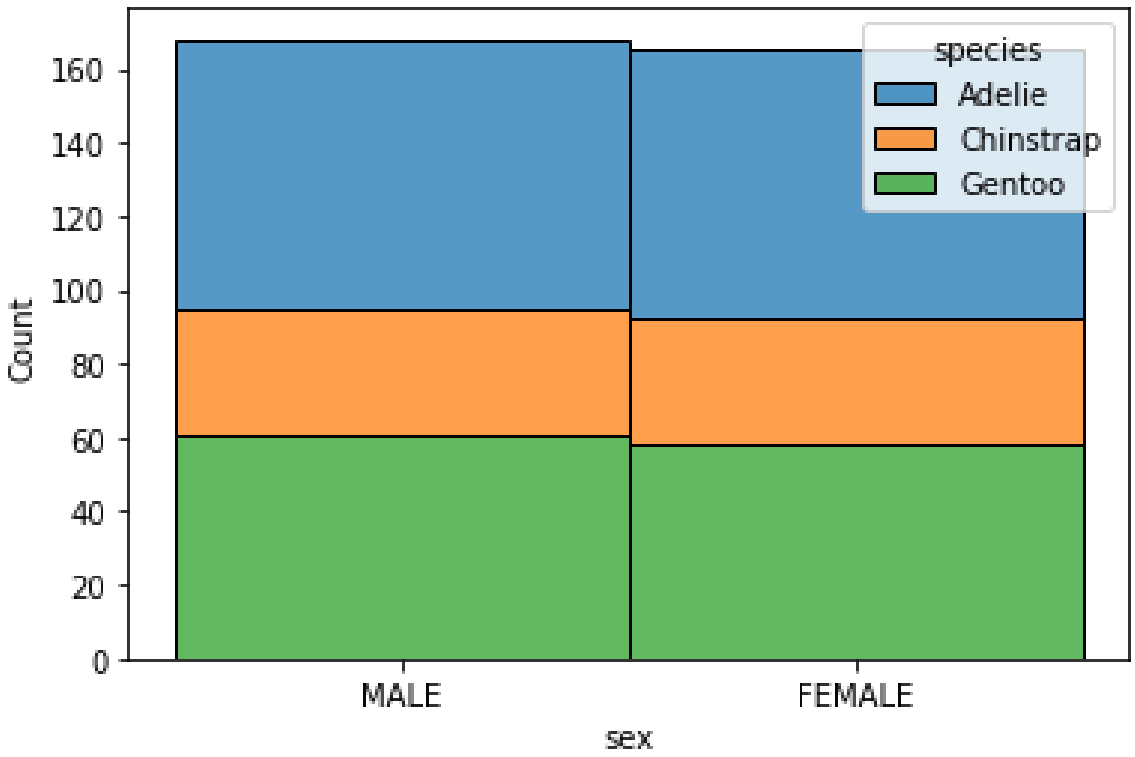


Figure 6: Histogram of number of penguins of different type with different sex

From the above histogram, it can be preliminary observed that bill, flliper and body mass information may be more important than island and sex information.

Basis Decision Tree

Basis decision tree algorithm is applied to the task, with only one tree bulit. Different maximum depth and maximum leaf size setting are used, which alongs with their performace are summarize in the following table

depth	leaf size	Train error	Test error
3	1	0.020	0.036
3	5	0.040	0.024
3	25	0.056	0.036
5	1	0.000	0.012
5	5	0.040	0.024

depth	leaf size	Train error	Test error
5	25	0.056	0.036
no limit	1	0.000	0.012
no limit	5	0.040	0.024
no limit	25	0.056	0.036

The tree structures are shown in figure 7 to figure 15.

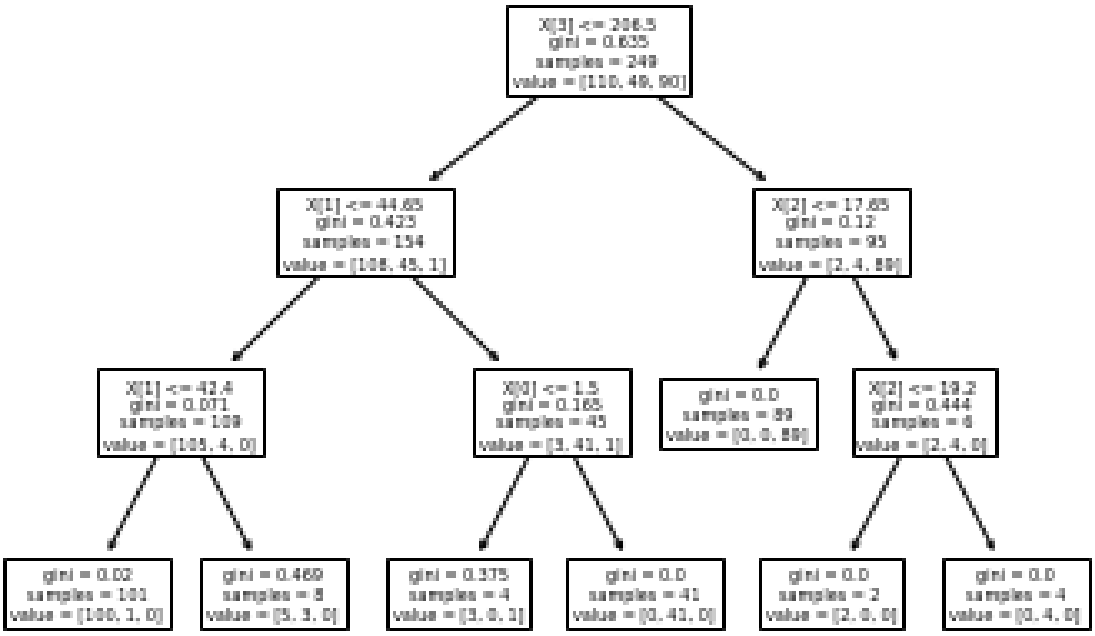


Figure 7: Tree structure of max depth 3, leaf size 1

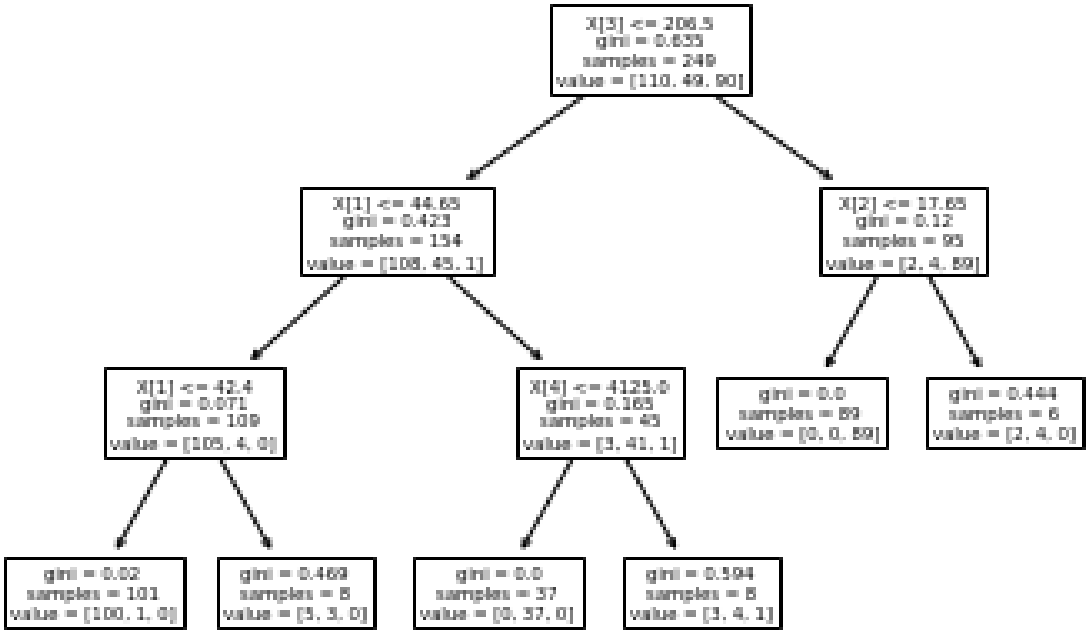


Figure 8: Tree structure of max depth 3, leaf size 5

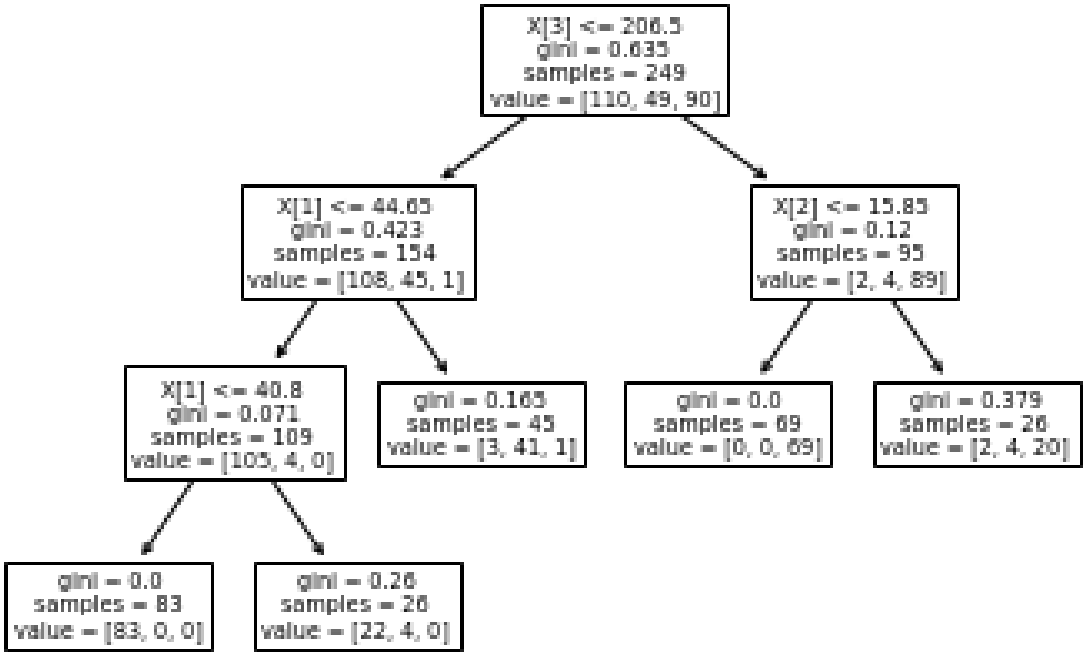


Figure 9: Tree structure of max depth 3, leaf size 25

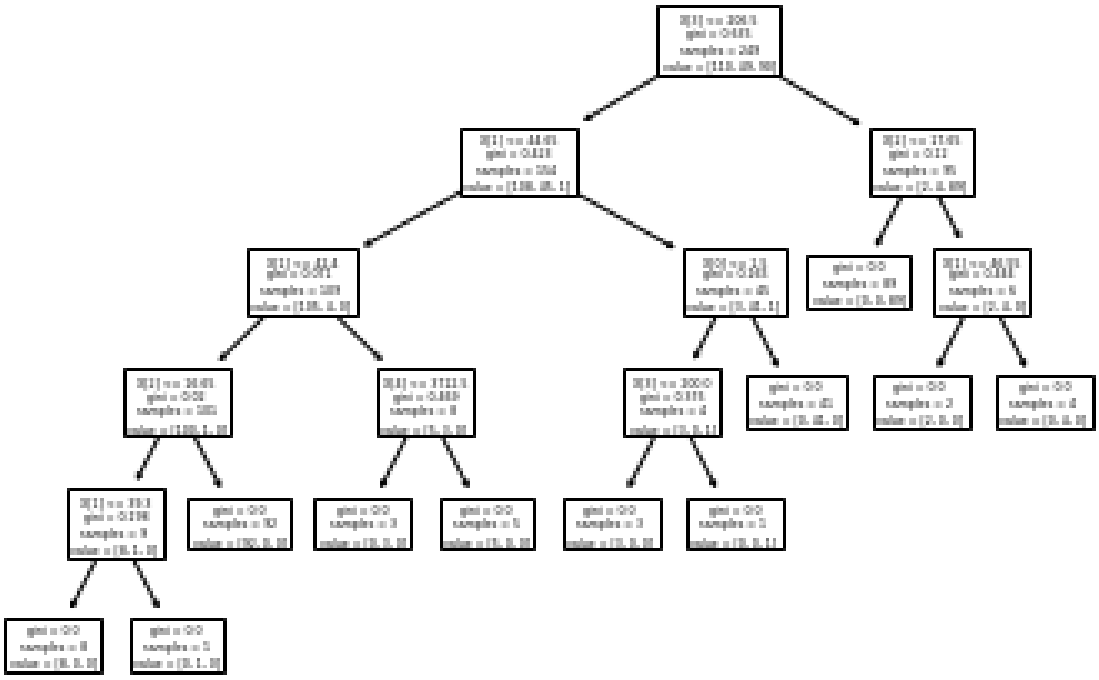


Figure 10: Tree structure of max depth 5, leaf size 1

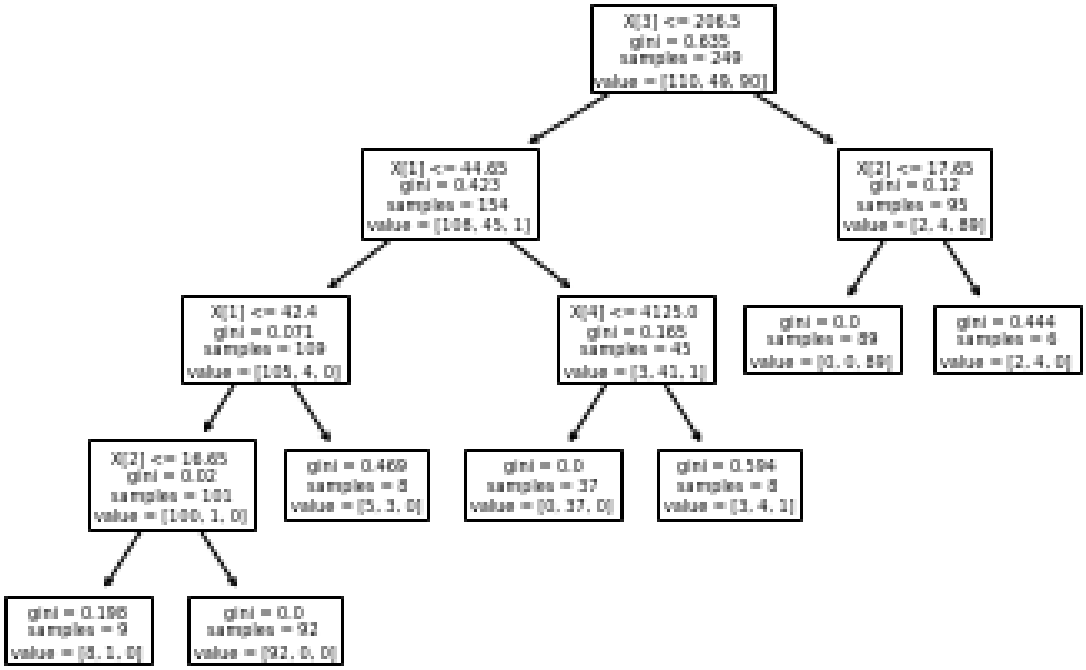


Figure 11: Tree structure of max depth 5, leaf size 5

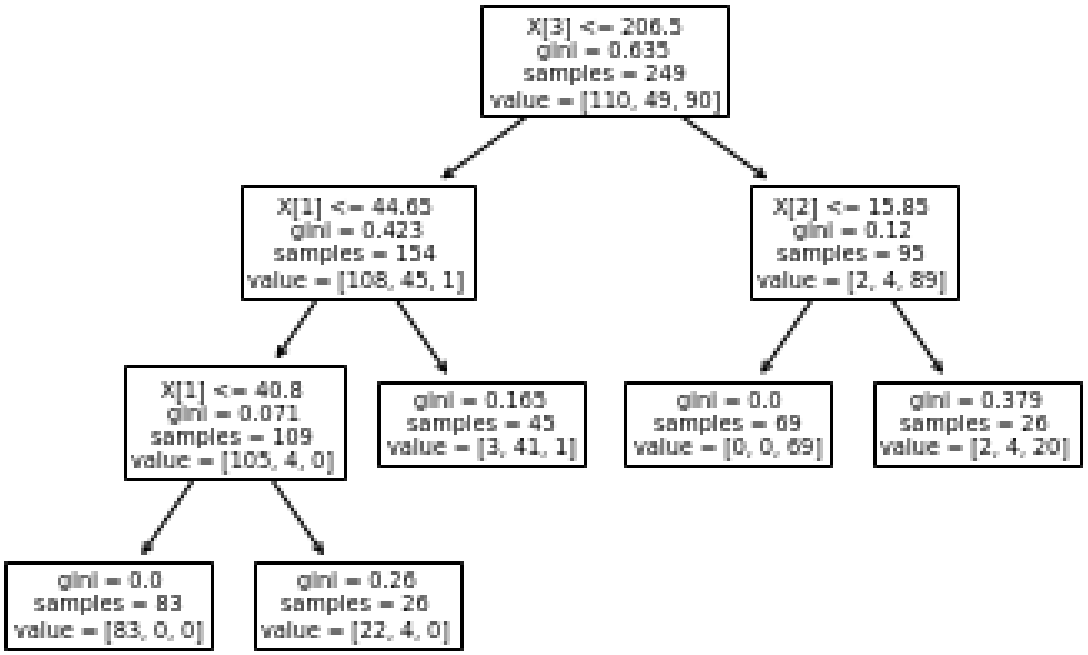


Figure 12: Tree structure of max depth 5, leaf size 25

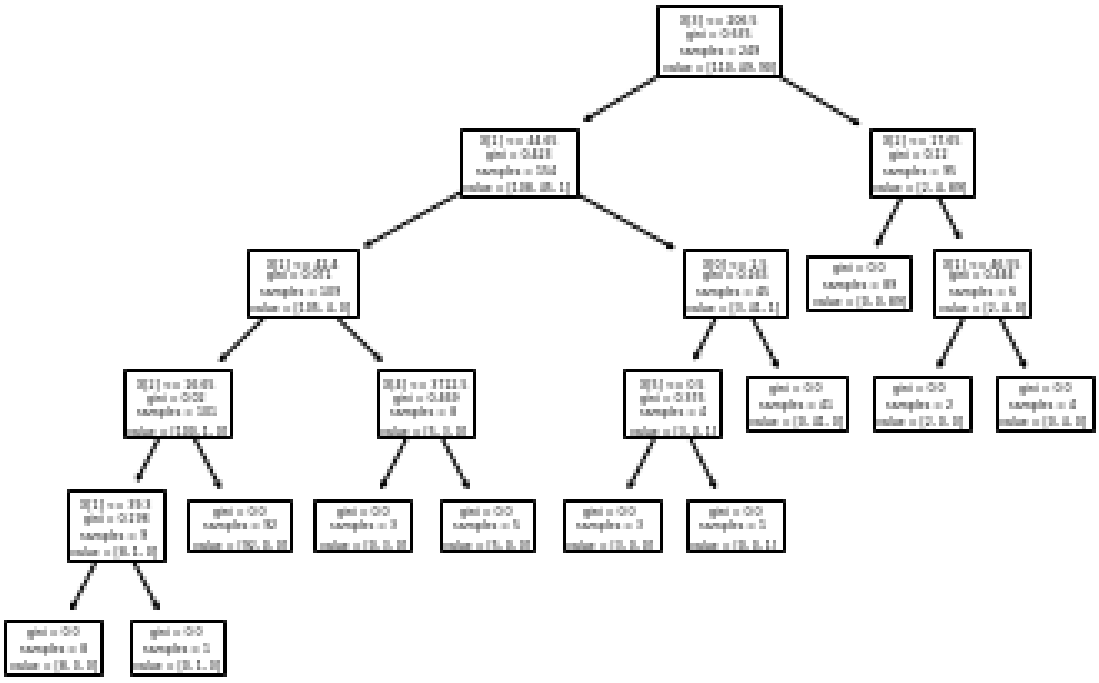


Figure 13: Tree structure without limiting depth, leaf size 1

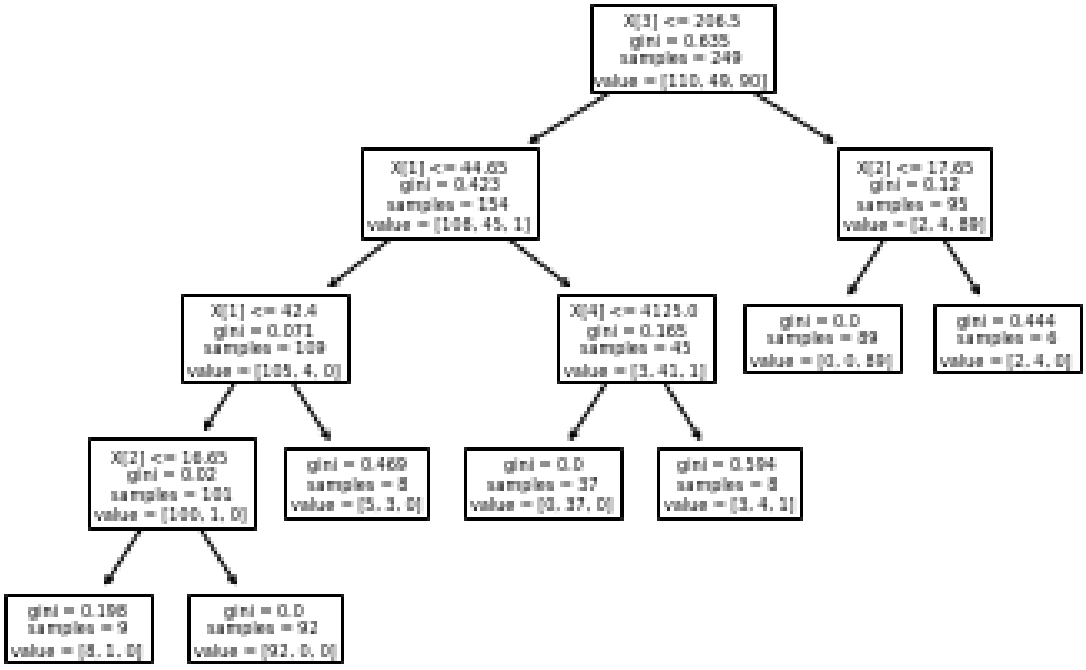


Figure 14: Tree structure without limiting depth, leaf size 5

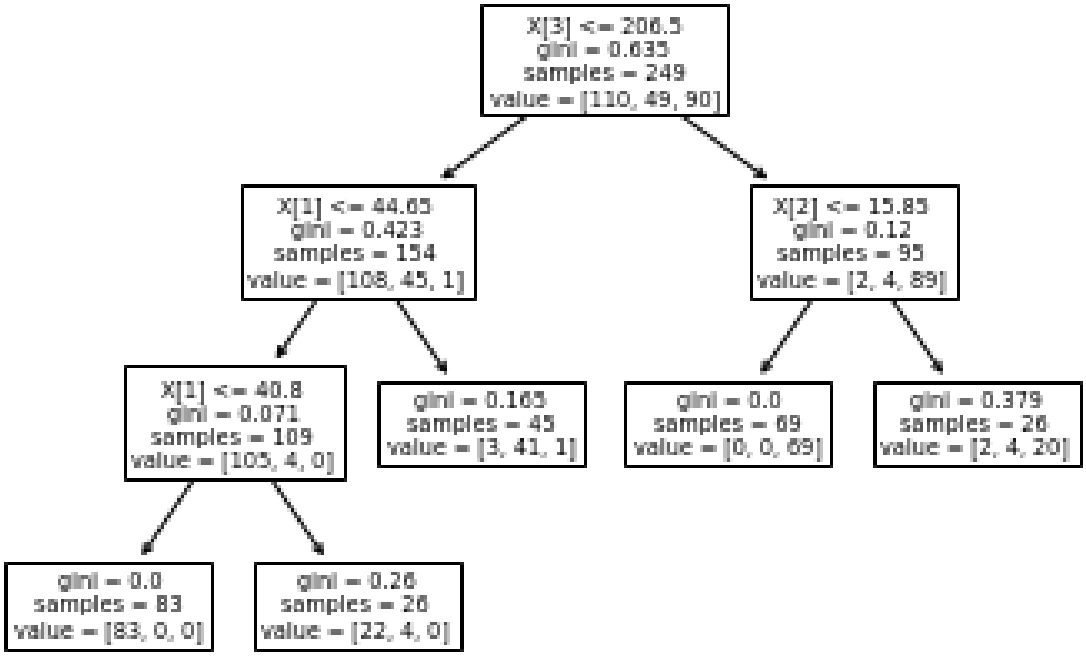


Figure 15: Tree structure without limiting depth, leaf size 25

As is shown in the above figure, bill, flliper and body mass information are frequently used, while sex and island information are seldom appear as a split criteria, which is aligned with the observation on data histogram.

Bagging of Trees

To mitigate overfitting, bagging can be used. The basic idea is to build multiple trees and make the final decision based on output from all trees. Different maximum depth and number of trees setting are explored, which alongs with their performace are summarize in the following table

depth	# of tree	Train error	Test error
3	5	0.016	0.036
5	5	0.000	0.000
no limit	5	0.008	0.024
3	10	0.024	0.024
5	10	0.000	0.012
no limit	10	0.004	0.012
3	50	0.012	0.024
5	50	0.000	0.000
no limit	50	0.000	0.000

Random Forests

Compared with Bagging of Trees Algorithm, in Random Forests, not only multiple trees will be trained, but also each tree will only utilize a subset of attributes. Different tree number, number of candidate attributes to split in every step setting are explored, which alongs with their performace are summarize in the following table

m	# of tree	Train error	Test error
1	5	0.076	0.214
3	5	0.000	0.071
6	5	0.004	0.000
1	10	0.100	0.238
3	10	0.004	0.000
6	10	0.000	0.000

m	# of tree	Train error	Test error
1	50	0.044	0.214
3	50	0.000	0.000
6	50	0.000	0.000

When the number of available attribute is very small, both training and testing has a bad performance. When the number of available attribute is medium, its training error is a little bit higher than using the full set of attributes, but the tesing error is close, since it has both smaller bias and variance. As is shown in figure 16 and 17, as the number of tree increase, bias has a sharp impulse and then back to normal, while variance is decreasing.

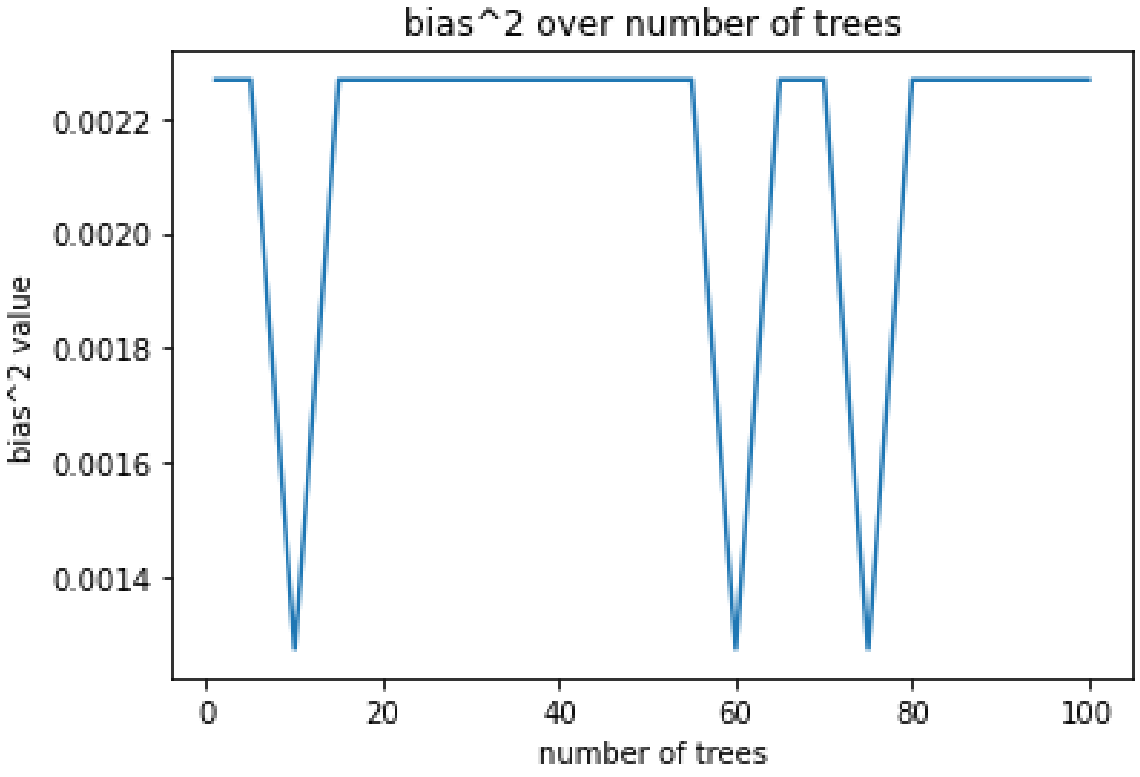


Figure 16: Change of bias with increasing number of tree

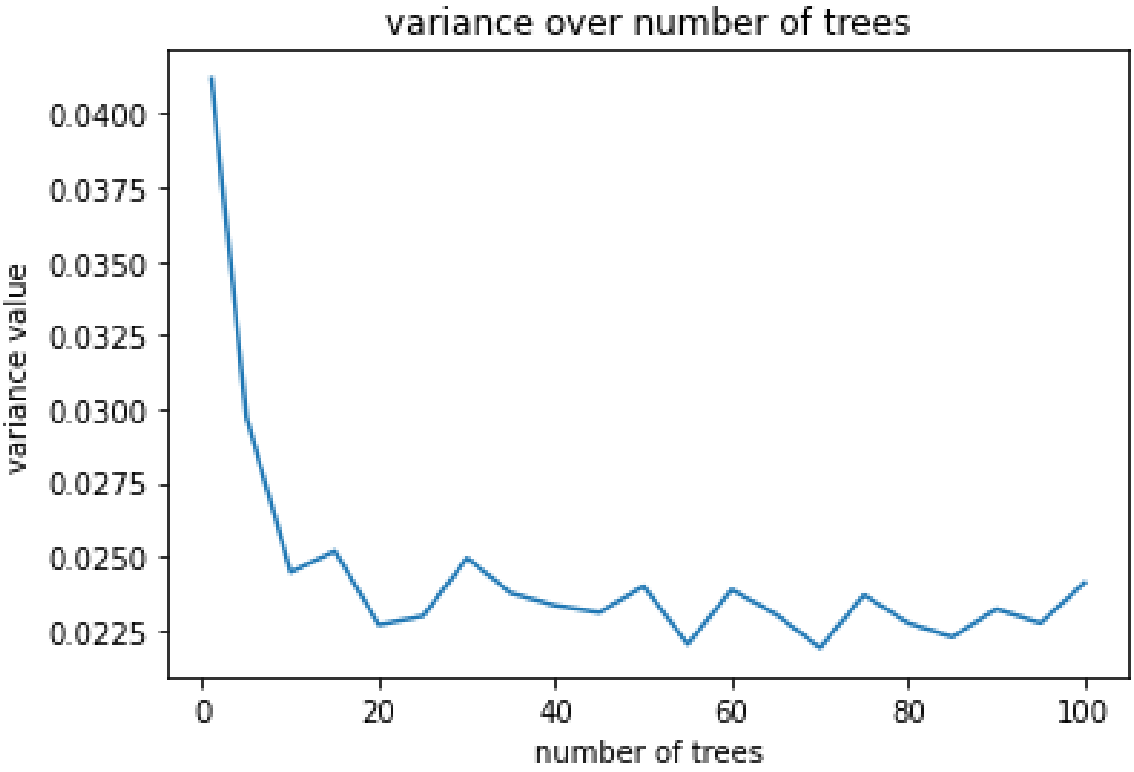


Figure 17: Change of variance with increasing number of tree

Fashion-MNIST Recognition Using MLP

In this task, I use MLP to recognize the image in Fashion-MNIST. Specifically, given an image with a resolution of 28x28, the network extract its features through hidden layers, and finally output the confidence of each class. Different number of hidden layers, hidden nodes of each hidden layers and optimizer are experienced. Besides, other hyperparameters are in the default setting.

hyperparameters	Value	Explanation
hidden_layer_sizes	50/200/784 * 1/2/3	How many neurons in each hidden layer
activation	relu	The activation function after the neurons of each hidden layer
solver	sgd/adam	The optimizer used to update the weights of neurons. Different optimizer has different optimizing strategy
batch_size	200	Number of samples within a batch. The optimizer will only update weights after finishing calculating a complete

hyperparameters	Value	Explanation
		batch
learning_rate	0.001,fixed	The weights multiplying to the gradient for each step of updating
max_iter	200	The maximum number of epochs. Going through the training data once is counted as 1 epoch
tol	1e-4	If the difference before and after an update is less than this value, training stop

As is shown in the following table, different number of parameters and optimizer can have great effect on the training and testing error. With the same optimizer, as the number of hidden nodes or the number of hidden layers increases, the training error has an obvious drop. However, the test error does not drop so much. The reason is that as the number of parameters increase, the network tend to overfit the training set.

# of hidden layers	# of hidden nodes	optimizer	Train error	Test error
1	50	SGD	0.402	0.414
1	50	Adam	0.114	0.149
1	200	SGD	0.113	0.149
1	200	Adam	0.084	0.130
1	784	SGD	0.060	0.129
1	784	Adam	0.116	0.138
2	50	SGD	0.800	0.800
2	50	Adam	0.080	0.129
2	200	SGD	0.900	0.900
2	200	Adam	0.054	0.113
2	784	SGD	0.900	0.900
2	784	Adam	0.058	0.114
3	50	SGD	0.900	0.900

# of hidden layers	# of hidden nodes	optimizer	Train error	Test error
3	50	Adam	0.074	0.131
3	200	SGD	0.094	0.139
3	200	Adam	0.038	0.108
3	784	SGD	0.028	0.125
3	784	Adam	0.027	0.107

For different optimizers, as is shown from figure 16 to figure 33, Adam optimizer always peform better than SGD optimizer. Adam converge much faster than SGD in model with a large parameters set. This is because Adam adapt learning rate based on the average of the second moments of the gradients rate on each step, while SGD does not adjust the learning rate. For this reason, in large model, SGD often fails to reach the global optimum, but stuck at a point, while Adam can always reach the optimum.

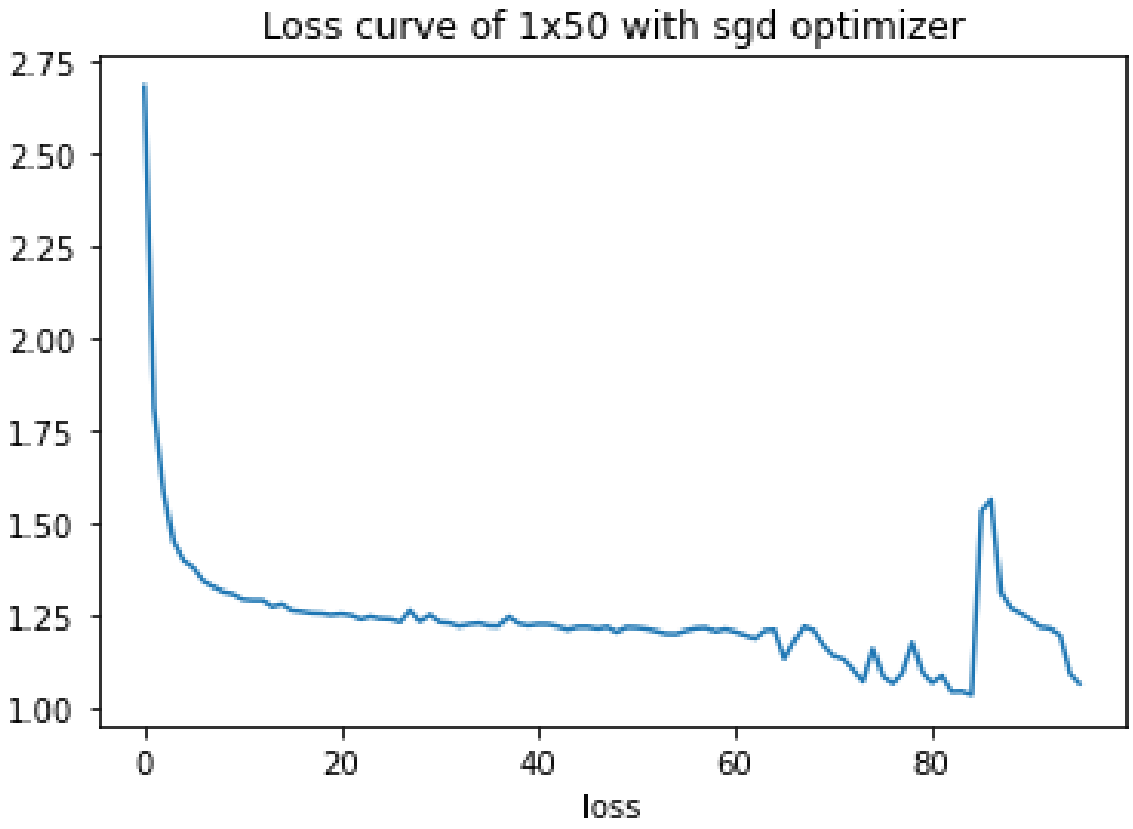


Figure 18: MLP with 1 hidden layers, each with 50 neurons, optimized by sgd optimizer

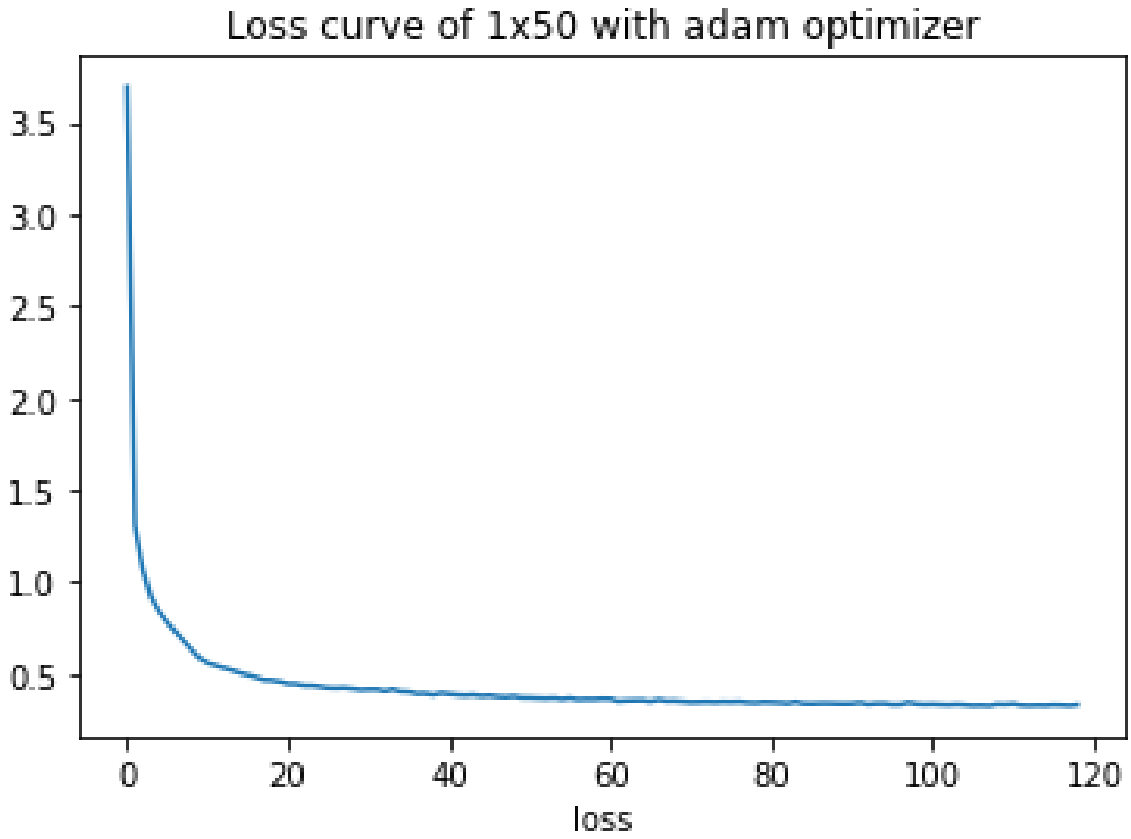


Figure 19: MLP with 1 hidden layers, each with 50 neurons, optimized by adam optimizer

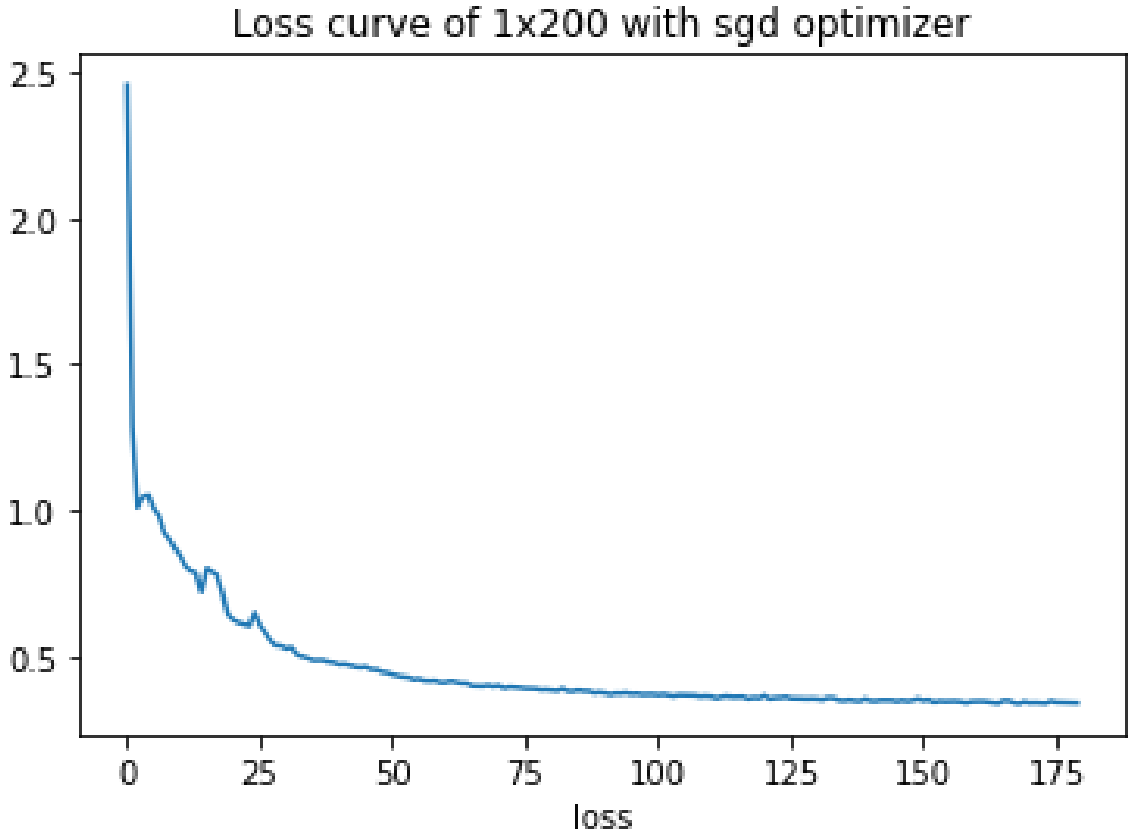


Figure 20: MLP with 1 hidden layers, each with 200 neurons, optimized by sgd optimizer

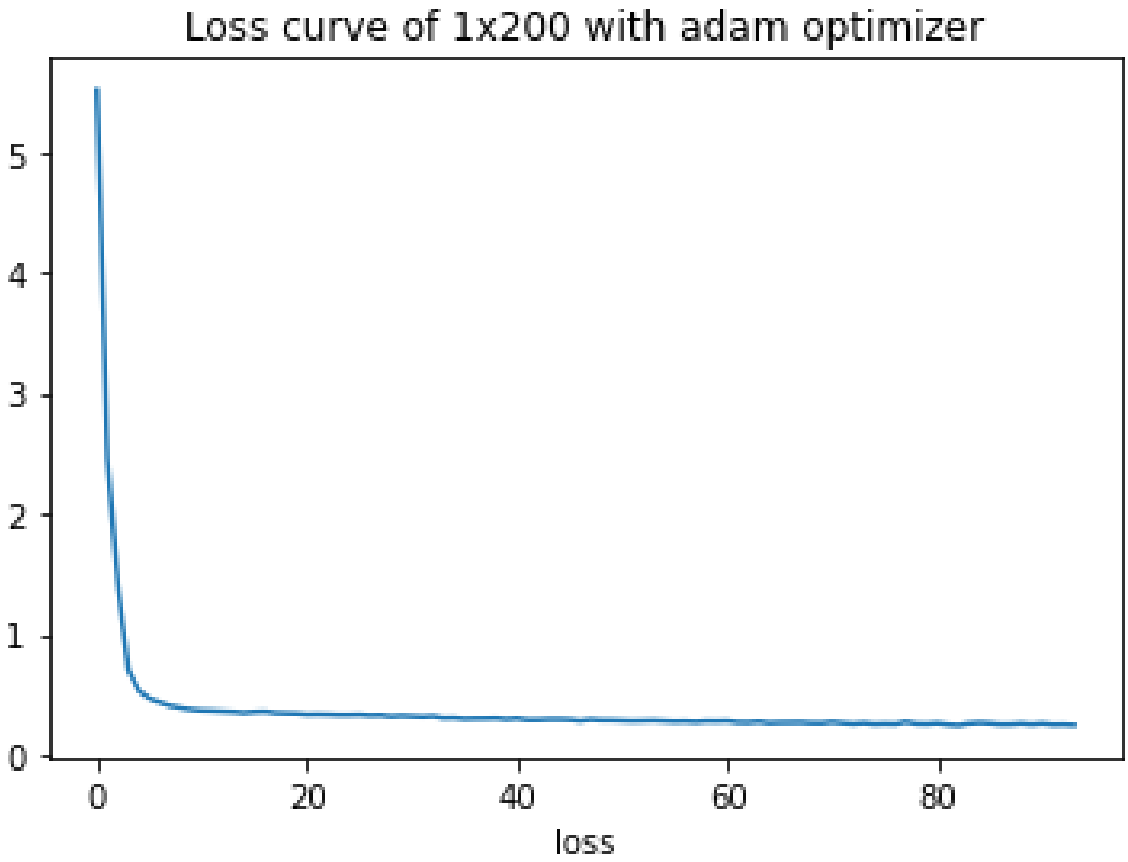


Figure 21: MLP with 1 hidden layers, each with 200 neurons, optimized by adam optimizer

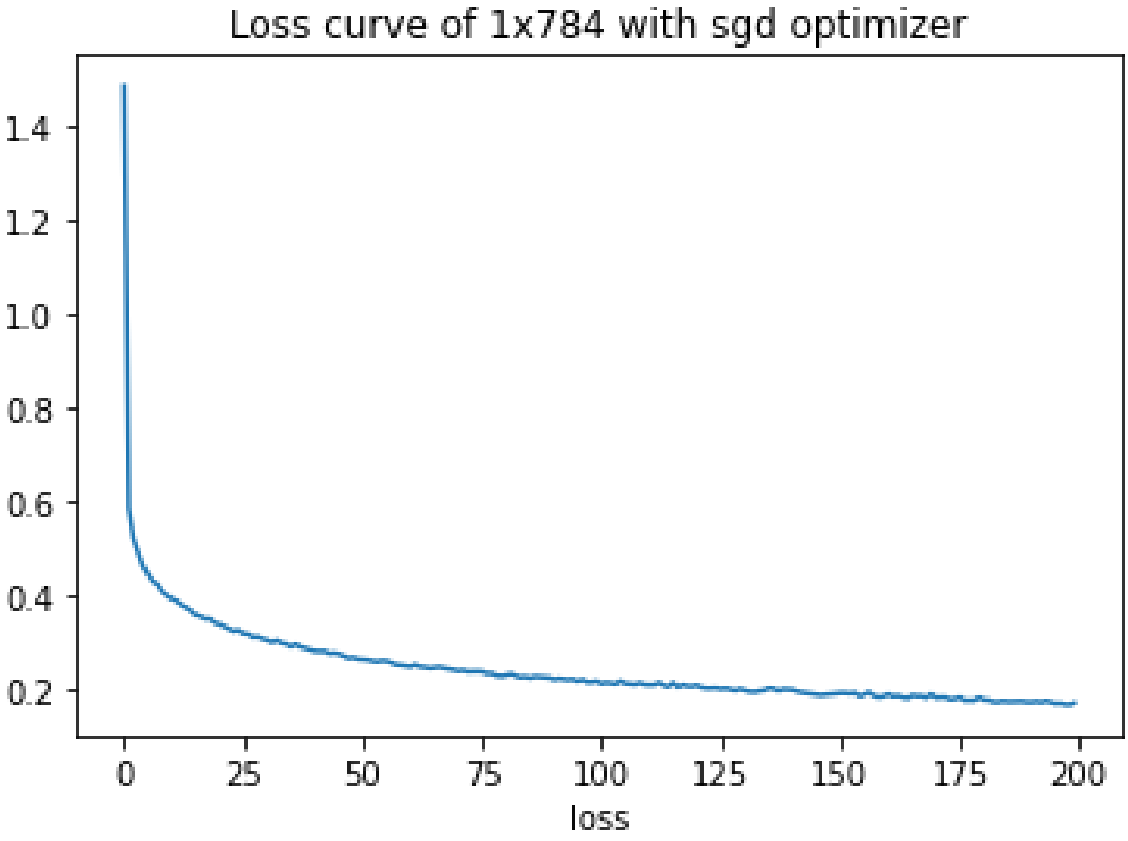


Figure 22: MLP with 1 hidden layers, each with 784 neurons, optimized by sgd optimizer

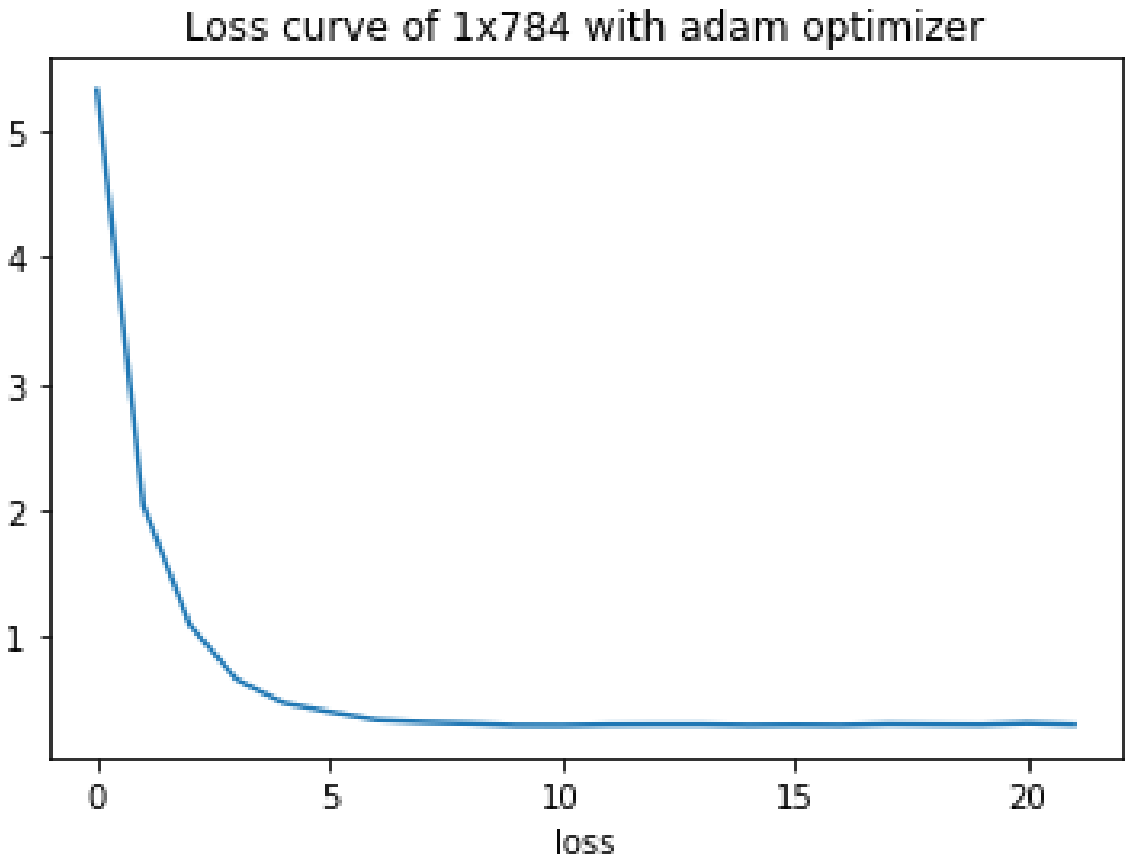


Figure 23: MLP with 1 hidden layers, each with 784 neurons, optimized by adam optimizer

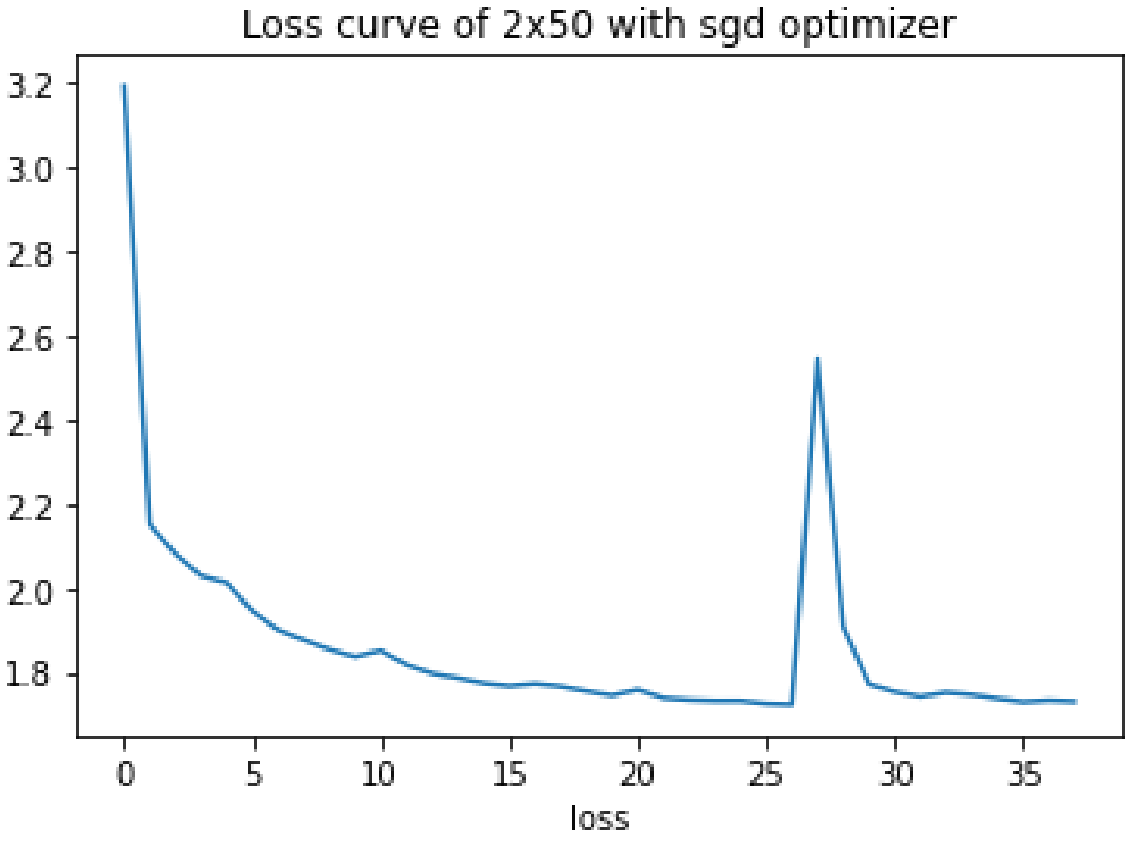


Figure 24: MLP with 2 hidden layers, each with 50 neurons, optimized by sgd optimizer

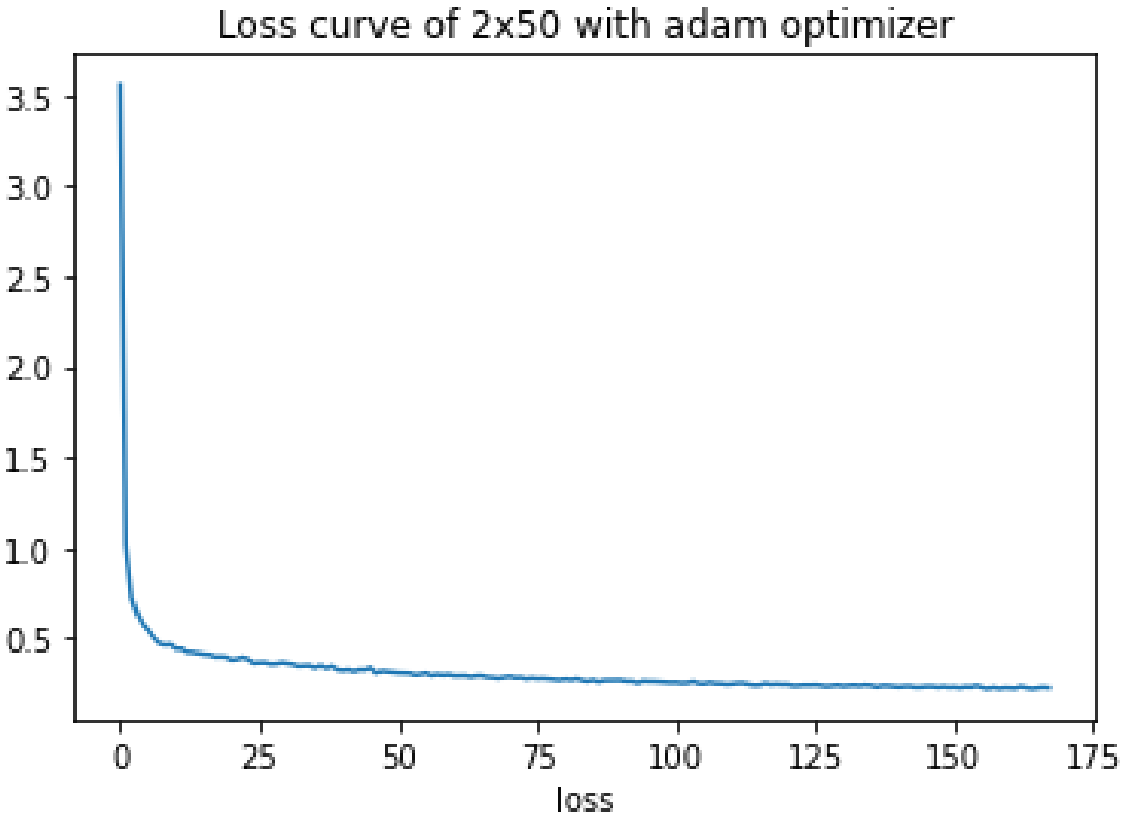


Figure 25: MLP with 2 hidden layers, each with 50 neurons, optimized by adam optimizer

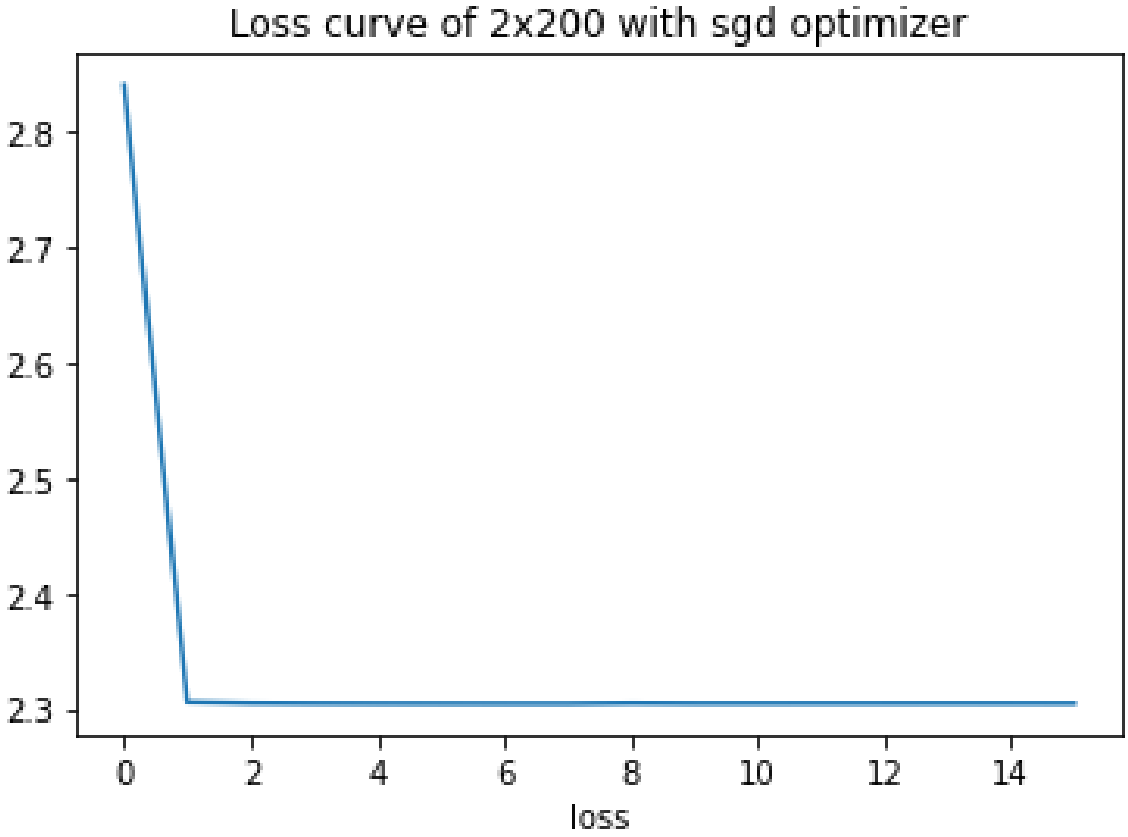


Figure 26: MLP with 2 hidden layers, each with 200 neurons, optimized by sgd optimizer

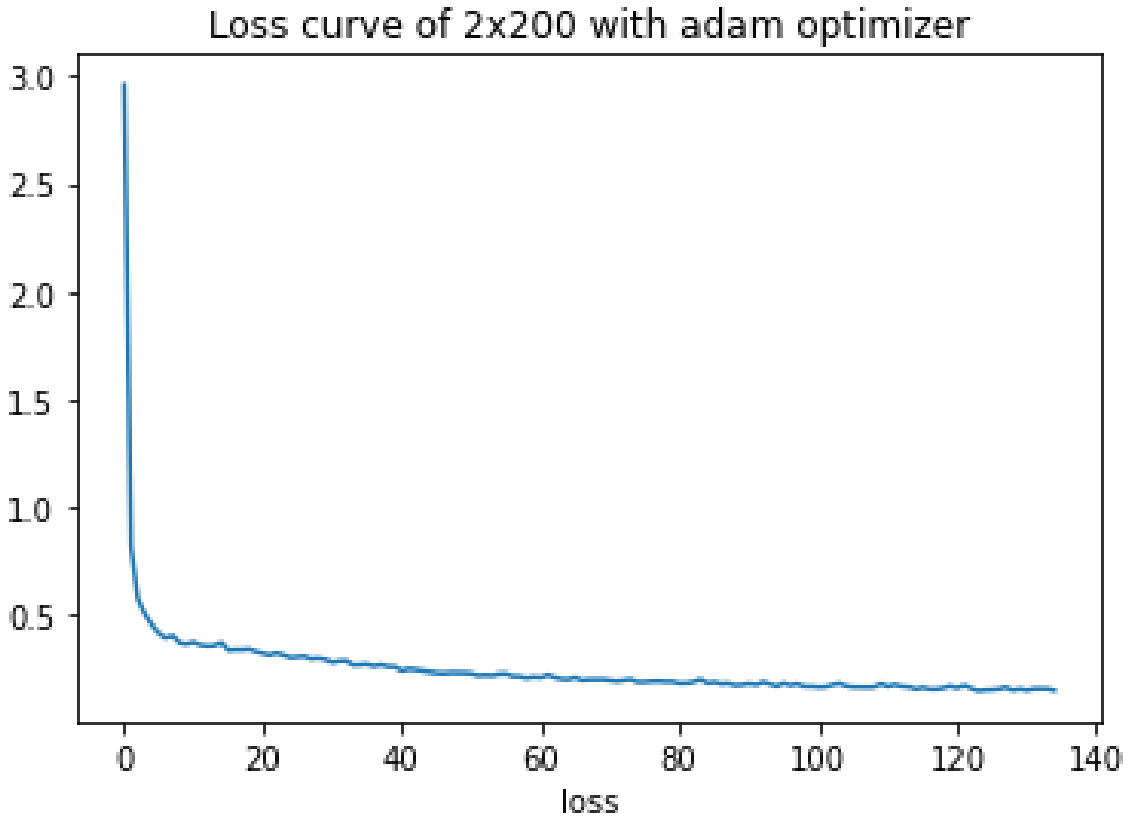


Figure 27: MLP with 2 hidden layers, each with 200 neurons, optimized by adam optimizer

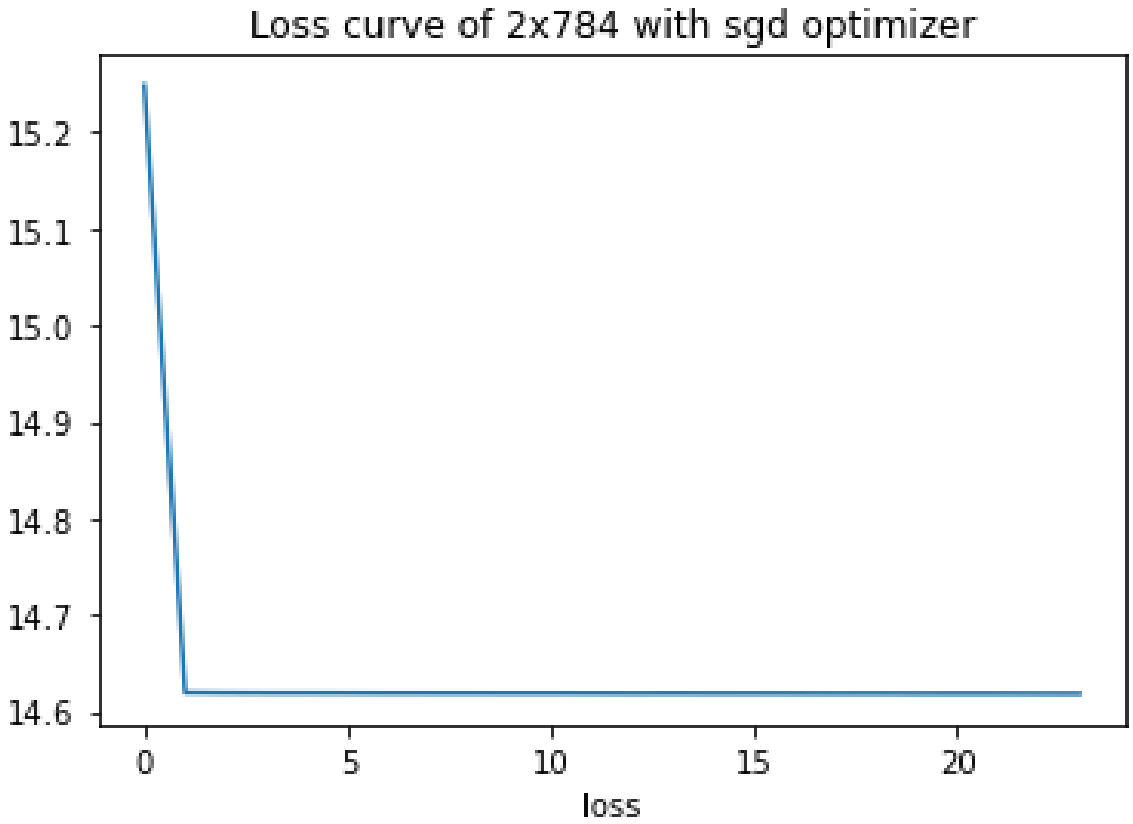


Figure 28: MLP with 2 hidden layers, each with 784 neurons, optimized by sgd optimizer

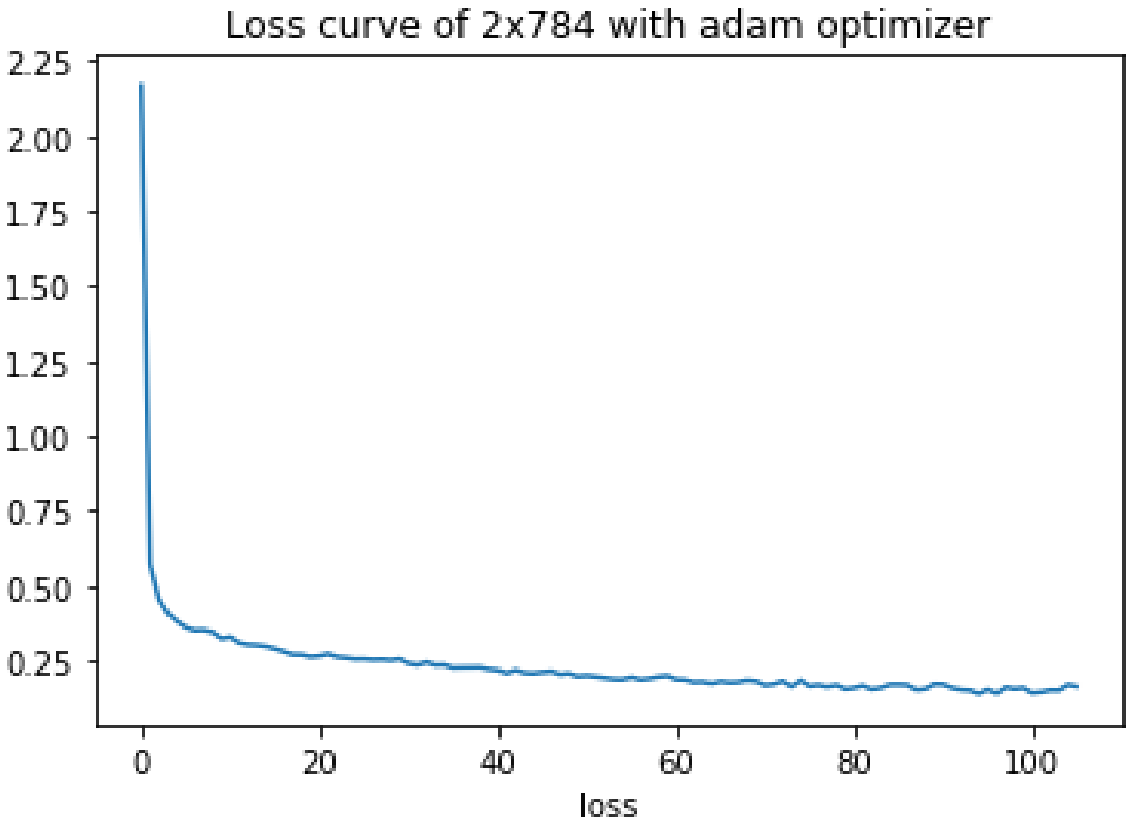


Figure 29: MLP with 2 hidden layers, each with 784 neurons, optimized by adam optimizer

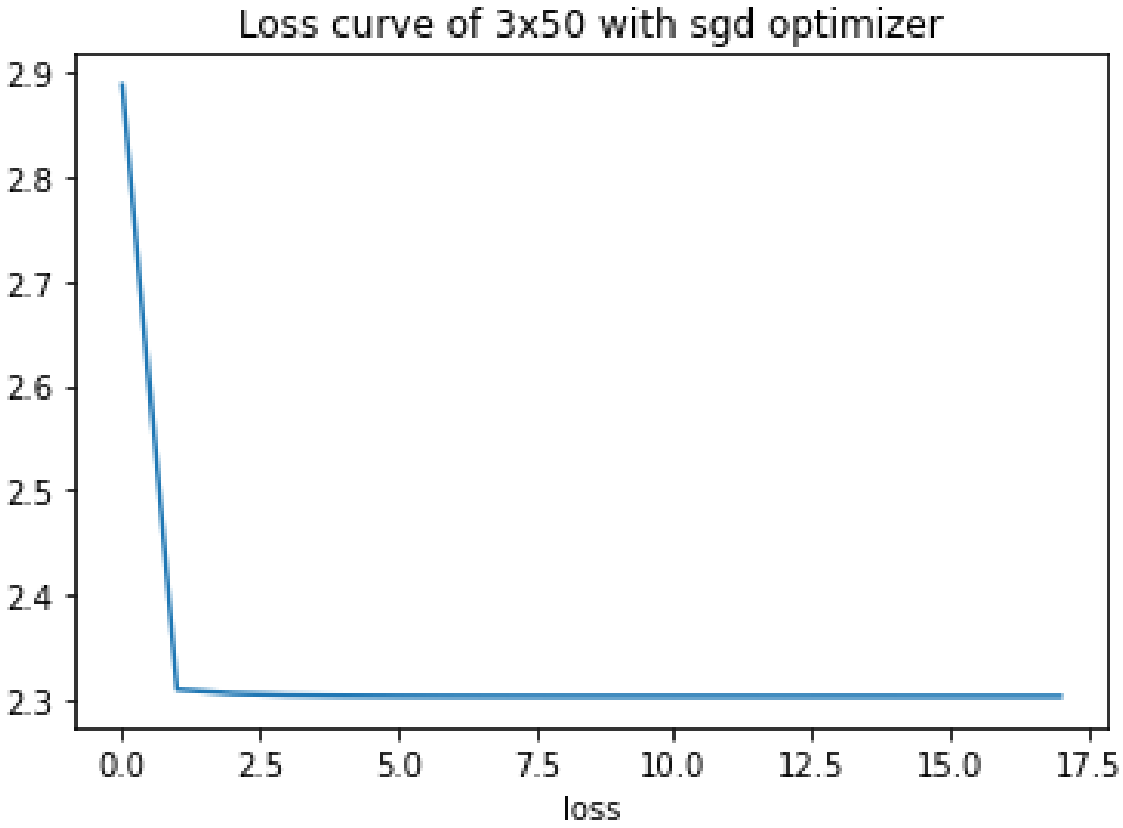


Figure 30: MLP with 3 hidden layers, each with 50 neurons, optimized by sgd optimizer

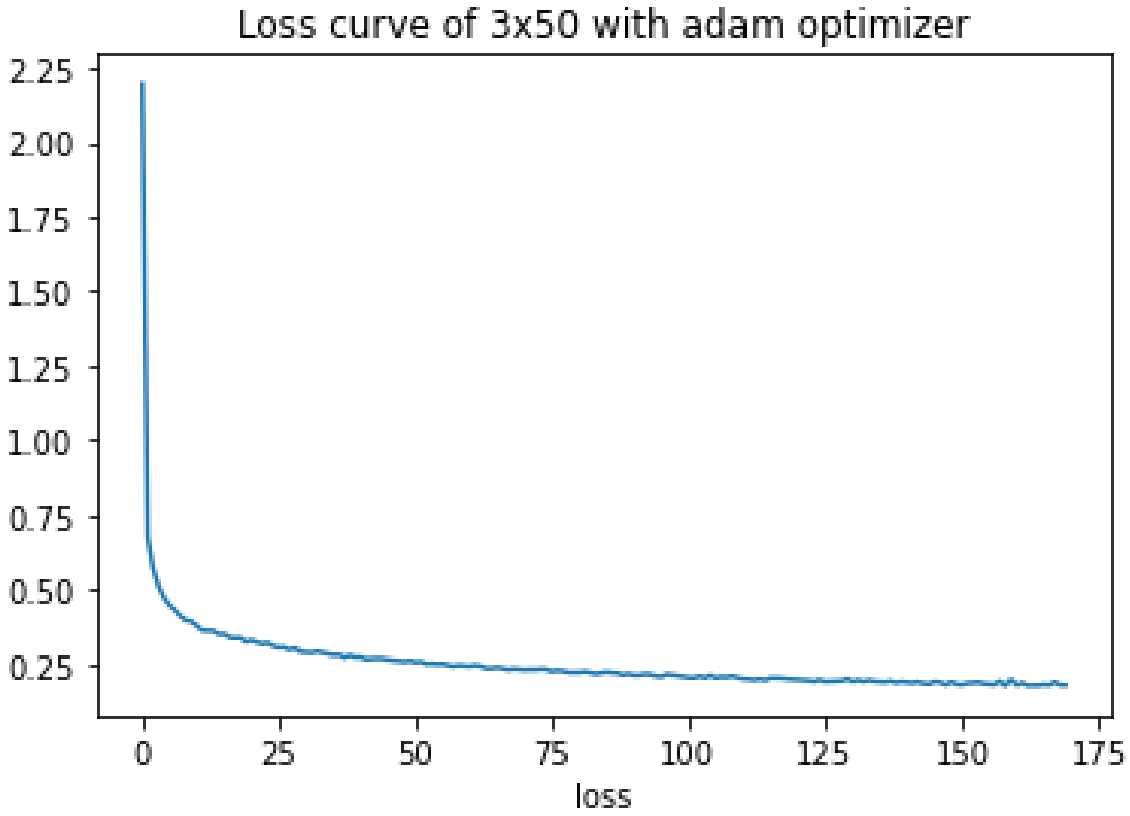


Figure 31: MLP with 3 hidden layers, each with 50 neurons, optimized by adam optimizer

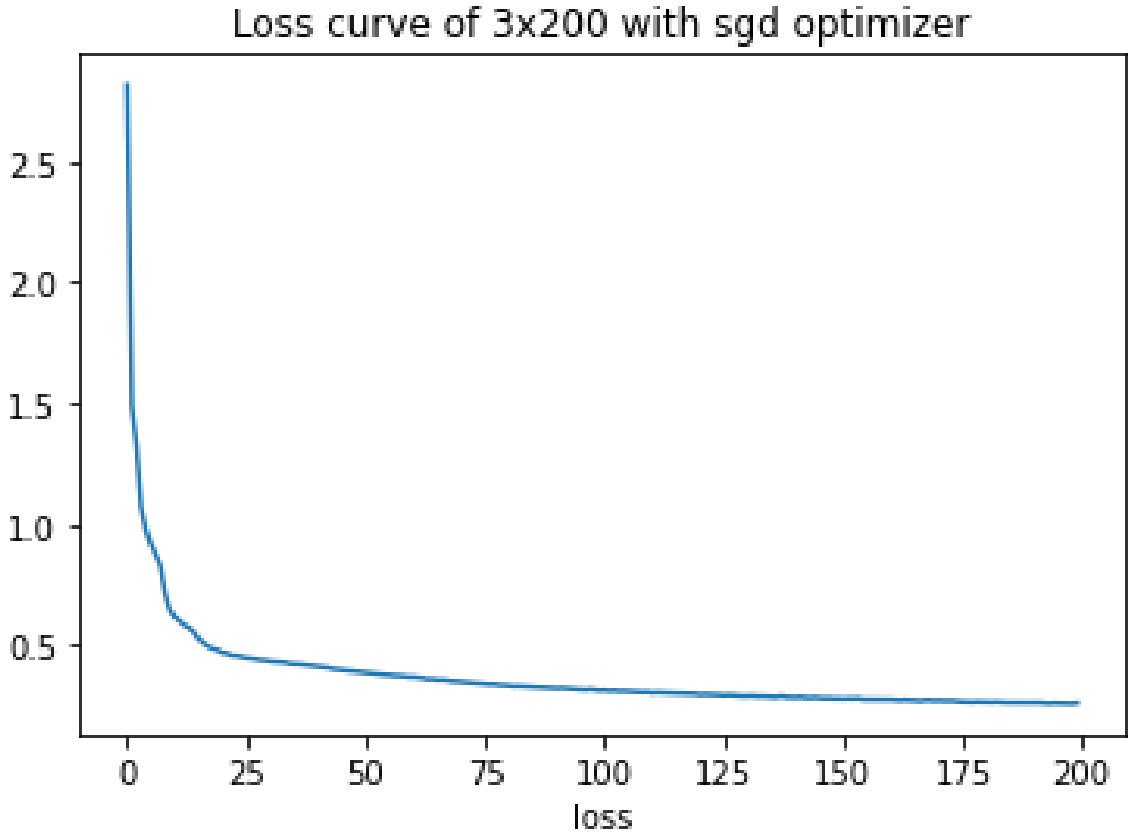


Figure 32: MLP with 3 hidden layers, each with 200 neurons, optimized by sgd optimizer

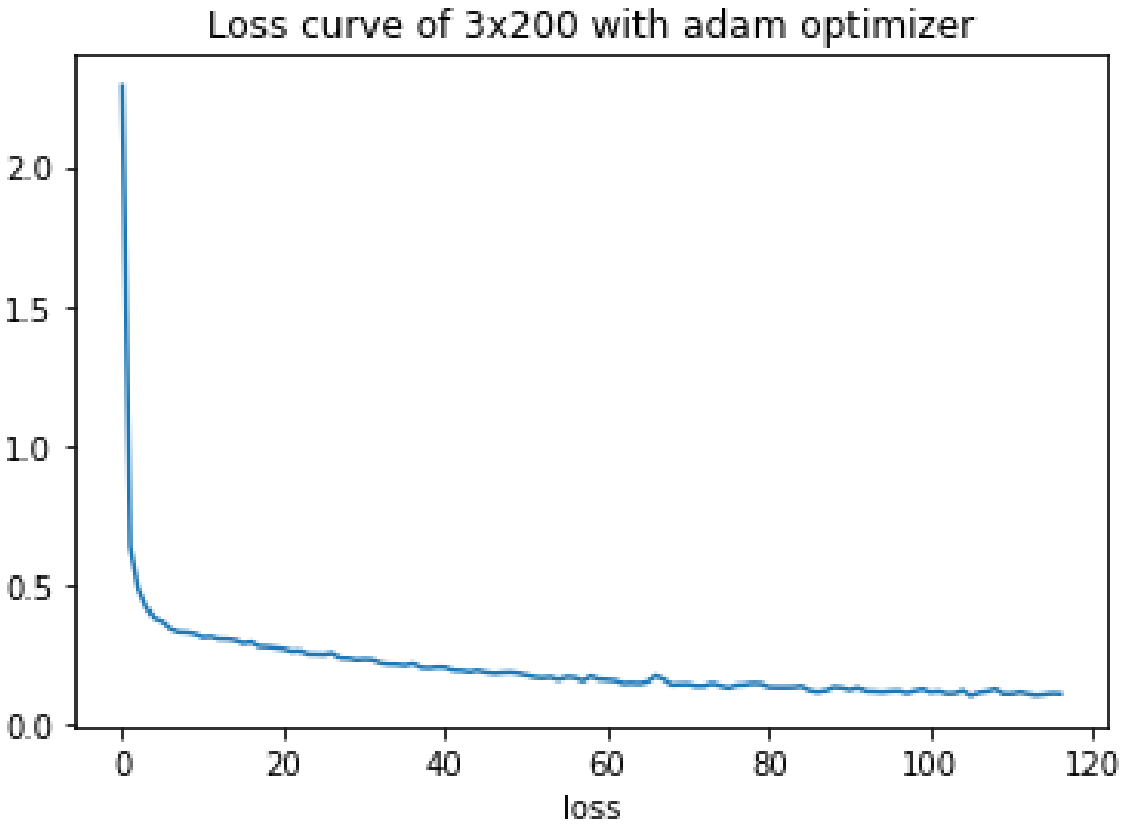


Figure 33: MLP with 3 hidden layers, each with 200 neurons, optimized by adam optimizer

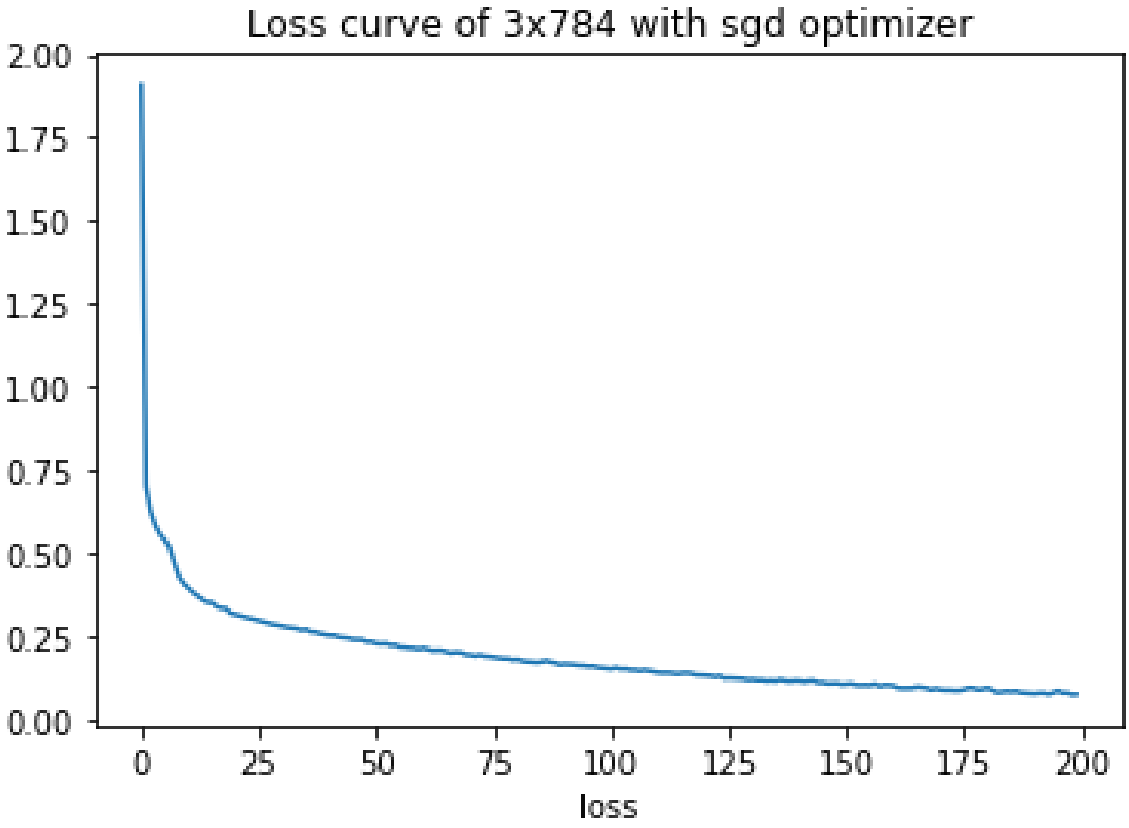


Figure 34: MLP with 3 hidden layers, each with 784 neurons, optimized by sgd optimizer

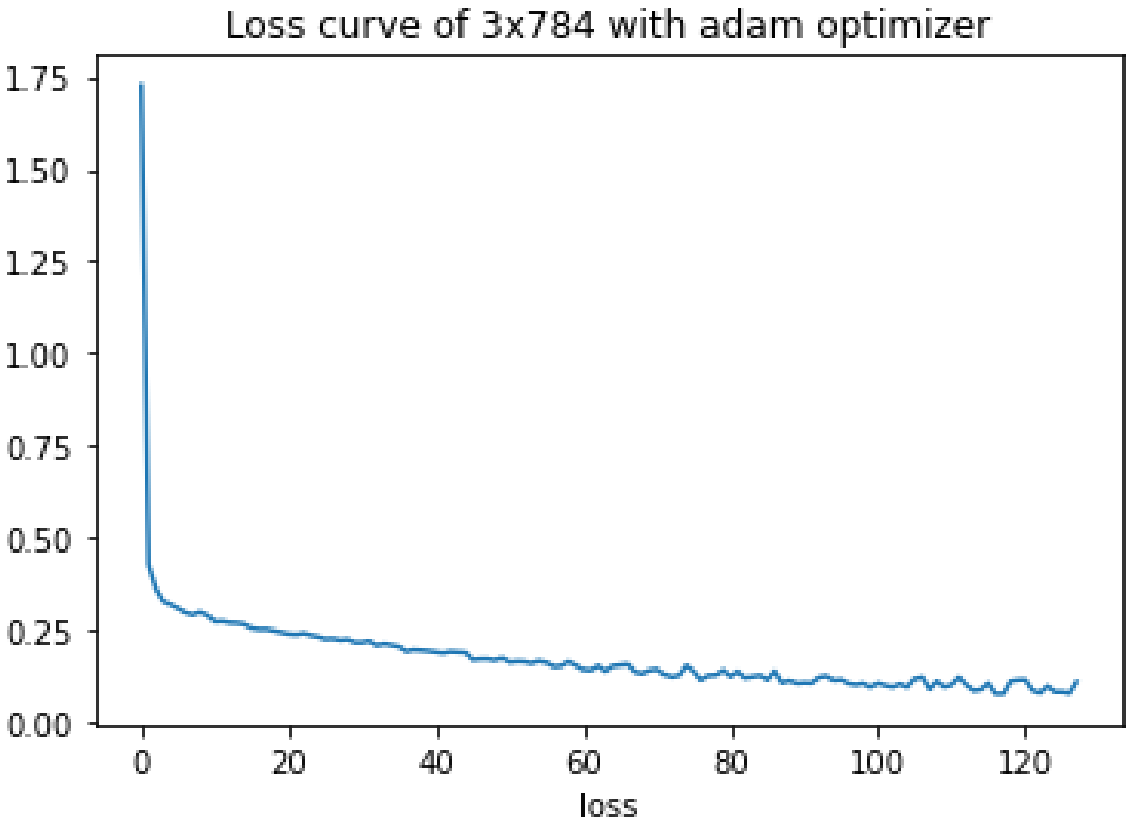


Figure 35: MLP with 3 hidden layers, each with 784 neurons, optimized by adam optimizer