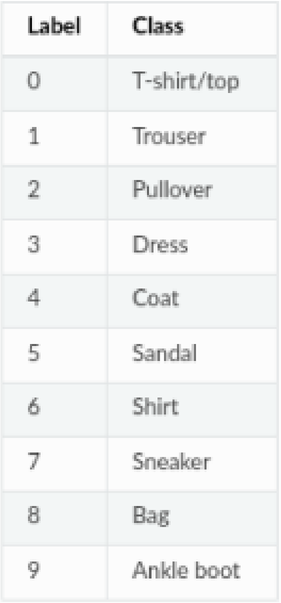
Cognition and Computation Project

- Mattia Varagnolo
- id: 2078225
- 17/12/2023

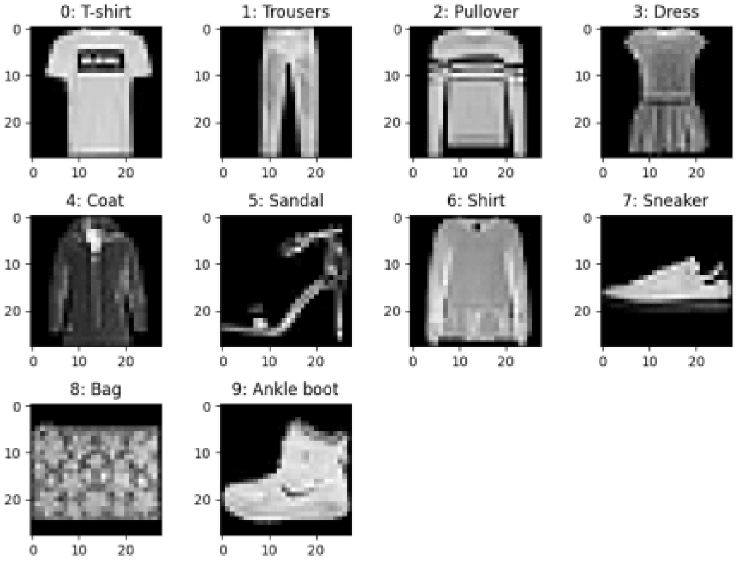
Introduction

Dataset

The dataset used is Fashion MNIST, which is a popular dataset for image classification tasks. It contains 70,000 grayscale images of clothing items, each measuring 28x28 pixels. The dataset is divided into 60,000 training samples and 10,000 test samples. It includes a total of 10 different classes of clothing items, such as t-shirts/tops, trousers, pullovers, dresses, coats, sandals, shirts, sneakers, bags, and ankle boots.



Label	Class
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot



Model Architecture

I’ve used a Deep Boltzmann machine with the following hyperparameters:

```
visible_units=28*28,
hidden_units=[400, 650, 650, 600],
k=1,
learning_rate=0.1,
learning_rate_decay=False,
initial_momentum=0.5,
final_momentum=0.9,
weight_decay=0.0001,
xavier_init=False,
increase_to_cd_k=False,
use_gpu=True
num_epoch = 50
batch_size = 120
```

Training

The model architecture consists of four hidden layers, each with an increasing number of neurons. We conducted training for a total of 50 epochs, using a batch size of 150 examples to optimize the learning process. The Fashion MNIST dataset, a widely-used benchmark in the field of computer vision, was employed for both training and testing purposes. To enhance the performance of our model, we utilized the contrastive divergence (CD) algorithm, a powerful technique specifically designed for training restricted Boltzmann machines (RBMs), a type of generative neural network. By incorporating the CD algorithm into our training pipeline, we were able to effectively improve the model's ability to capture complex patterns and generate accurate predictions.

Testing

To thoroughly evaluate the accuracy of the model, we extensively utilized the test set that was conveniently provided by the torchvision library. In addition to assessing its robustness to adversarial attacks, we employed the widely recognized and effective fast gradient sign method (FGSM) to generate a diverse range of adversarial examples by experimenting with various epsilon values. By doing so, we were able to comprehensively analyze the model's performance under different attack scenarios and gain deeper insights into its overall resilience and reliability.

Linear Read-Out results

Linear readouts were conducted at various levels of the model hierarchy to examine the deep Boltzmann machine's capacity to separate sensory representations. The readouts were carried out on the hidden units of the model's four layers, which capture the local structure of the input data through lower-level features.

```
Accuracy for readout in layer 1
0.8425999879837036
Accuracy for readout in layer 2
0.8438999652862549
Accuracy for readout in layer 3
0.8446999788284302
Accuracy for readout in layer 4
0.8442999720573425
```

The model did not show significant improvement in disentangling the sensory representation as the network depth increased. This is evident from the accuracy scores of the linear readouts recorded for each layer.

Although the accuracy scores are quite similar for each layer, the deeper the model, the better it becomes at recognizing shapes with added noise.

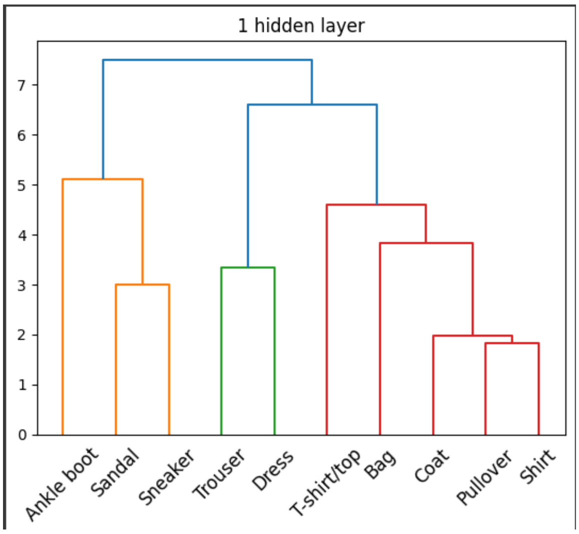
Hierarchical clustering and feature visualization

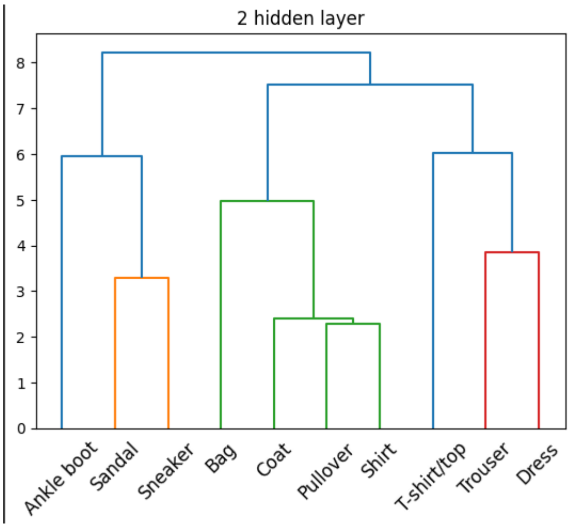
To analyze the internal representation of the model, we utilized hierarchical clustering and feature visualization methods.

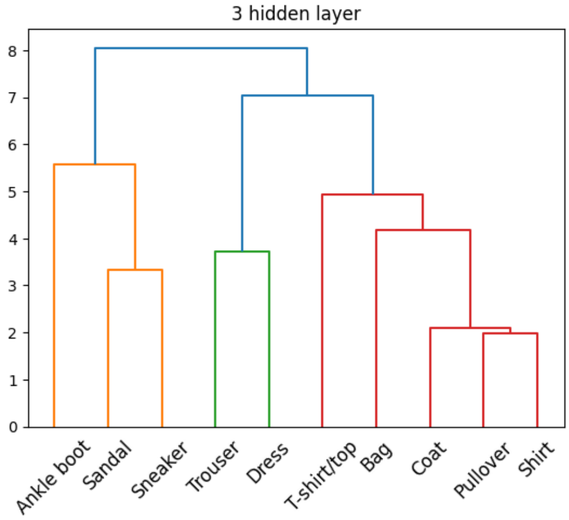
The hierarchical clustering analysis effectively grouped images of t-shirts, shirts, dresses, coats, and pullovers together. It is worth noting that t-shirts were the most distinct class within the same cluster, likely due to the absence of sleeves in this clothing category.

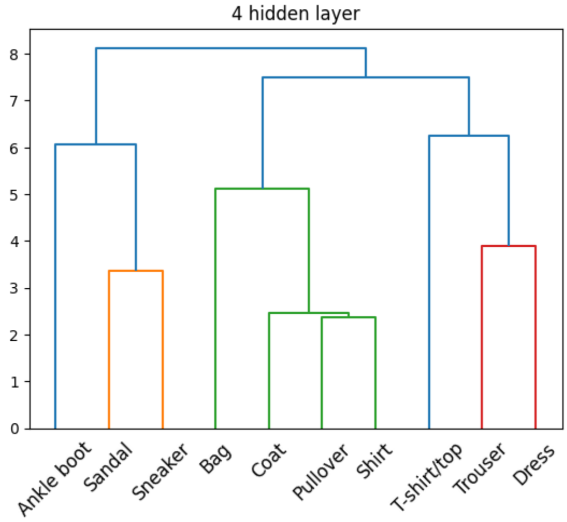
Trousers and dresses formed another cluster due to their similar long and slim shape. Meanwhile, sneakers and sandals comprised the third cluster, while ankle boots and bags were incorrectly placed together in the last cluster.

These findings suggest that the model identified certain high-level features that are common among specific types of clothing, such as the overall shape and length.



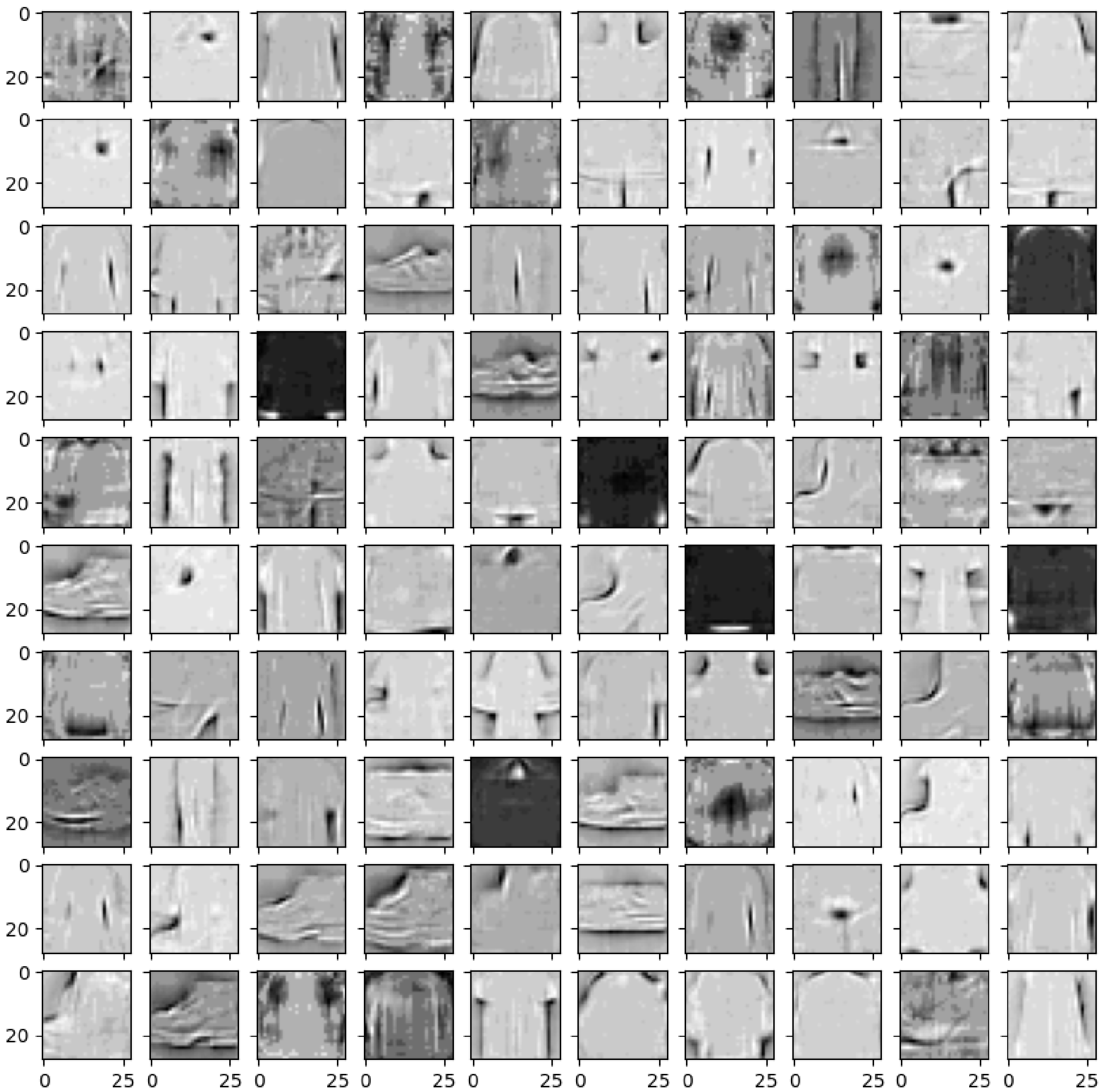






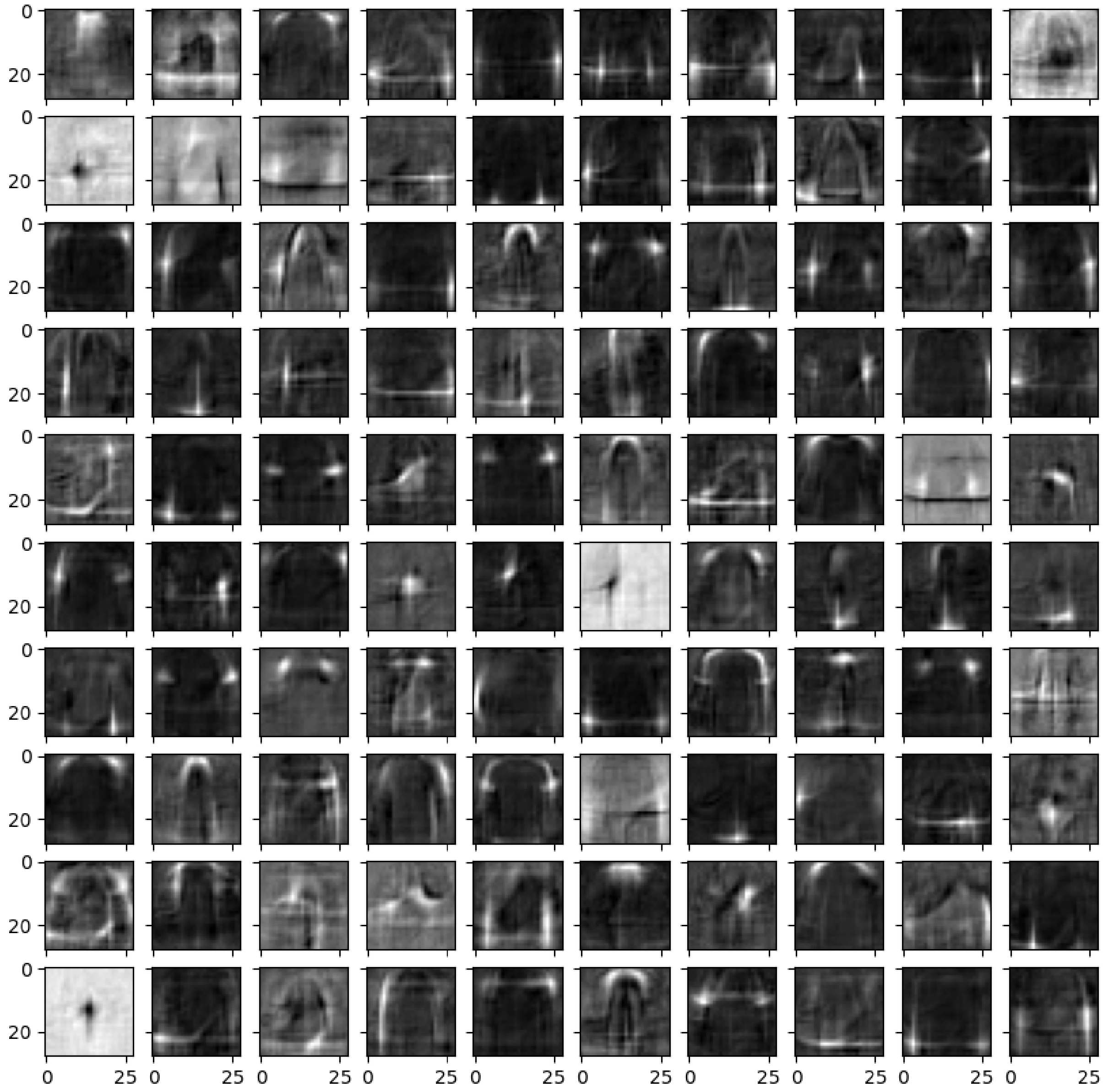
Feature visualization (visual receptive field)

In the visual receptive fields, it is interesting to note that there is a wide range of hidden units that exhibit different sensitivities. Some of these hidden units are specifically attuned to the texture or pattern of the input images, allowing them to discern intricate details and subtle variations. On the other hand, there are also hidden units that are more concerned with the overall shape or silhouette, enabling them to capture the larger structural elements of the visual input. This diversity in the sensitivities of the hidden units highlights the complex nature of visual processing and the multiple dimensions that contribute to our perception of images.



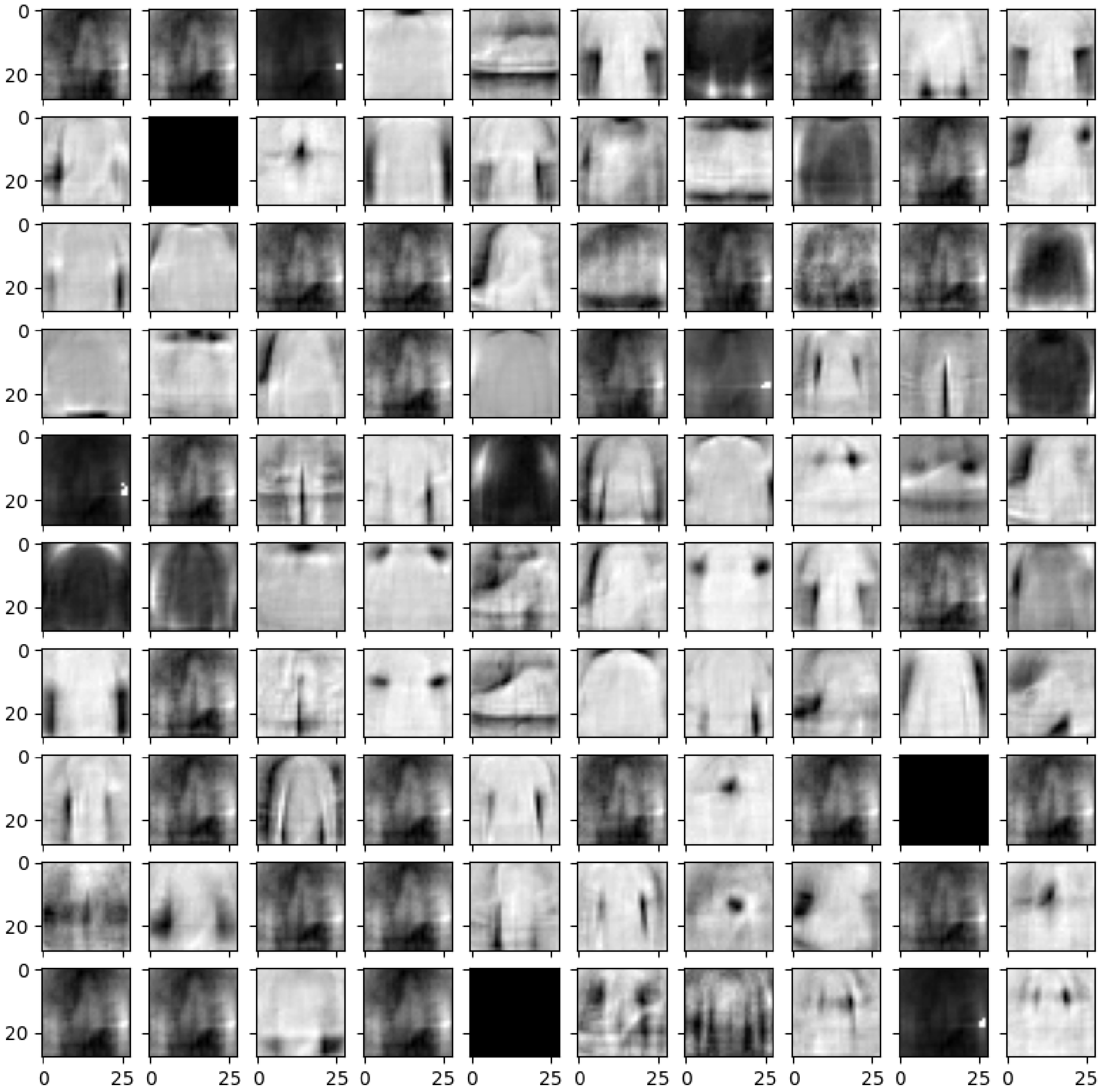
Receptive field 1st layer

In the first layer we can see that the neurons of the model activate on basic shapes, mostly shirt/t-shirts/coat shapes, and also some bag and shoe. The shapes are not so precise.



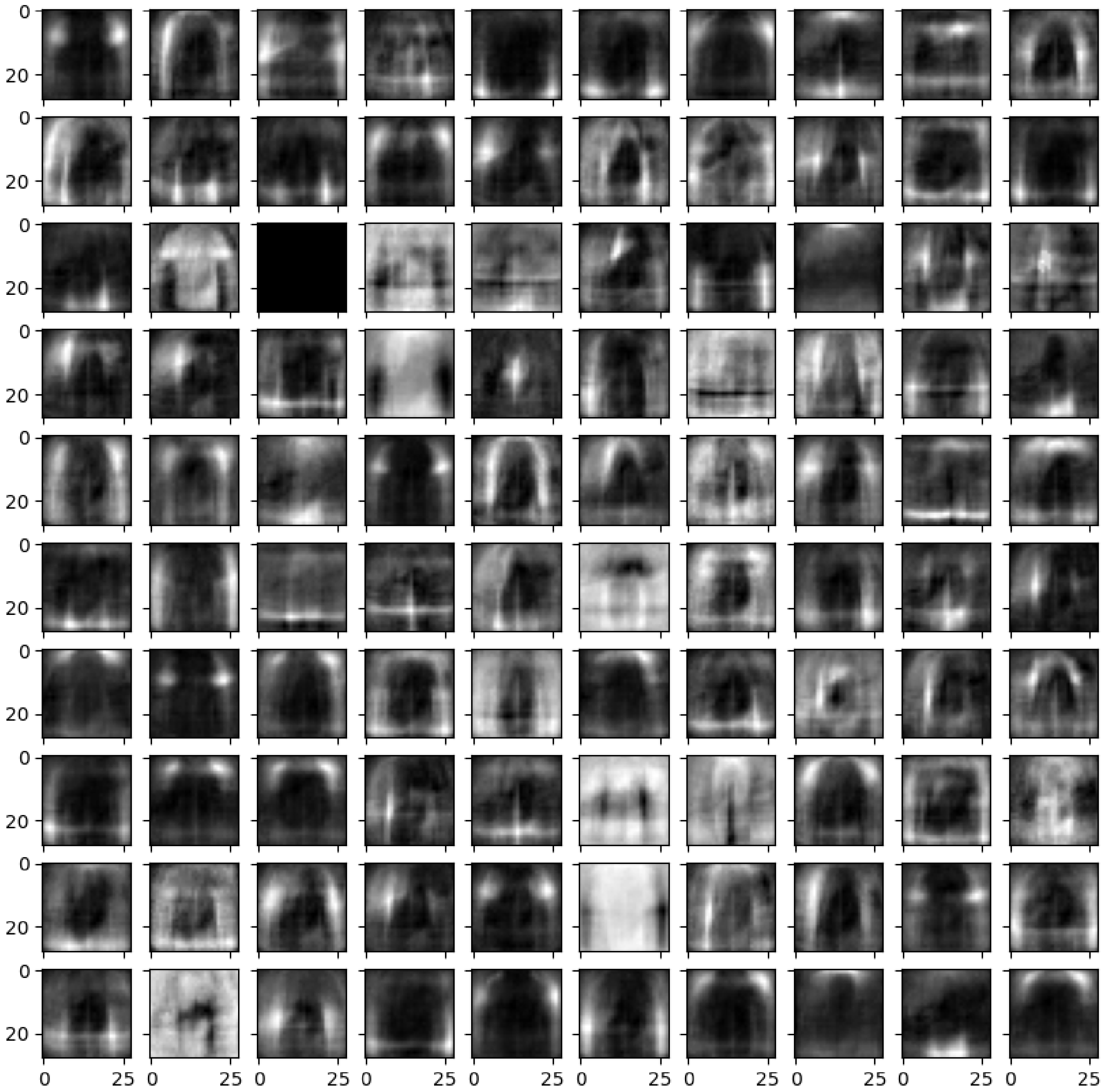
Receptive field 2nd layer

In the second layer we can see that the model recognizes basic edges like shoulder shape, sleeves and shoe sole.



Receptive field 3th layer

In the third layers it seems that the model has learned to recognize most of the shapes
with some type of blurring, indicating that neurons could adapt to some noise and generalize better.



Receptive field 4th layer

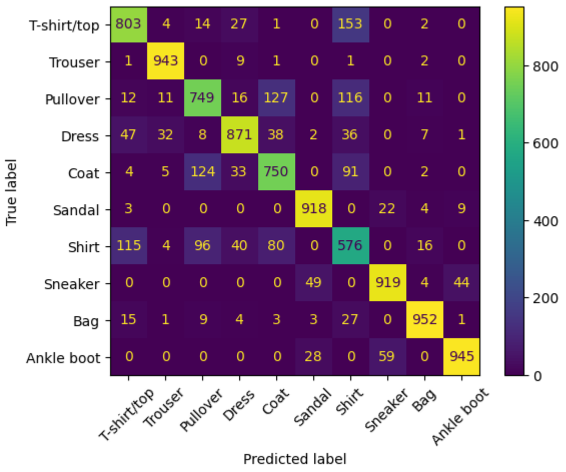
In the fourth and final layer, the model primarily focuses on edges and shapes. The images appear blurrier, and certain neurons are activated by the shape of the background rather than the shape of the clothing.

Overall, these results indicate that the deep Boltzmann machine model has the capability to learn features that are crucial for recognizing various types of clothing. It starts with basic shapes in the lower layers and progressively incorporates more intricate features in the higher layers.

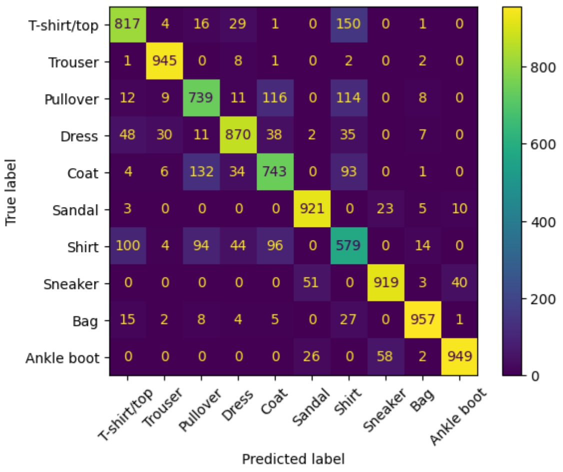
Confusion matrices and psychometric curves

To examine the model's errors, we utilized visualization of confusion matrices and psychometric curves.

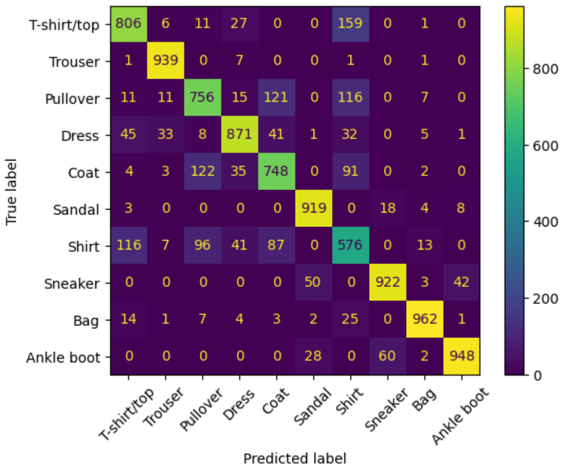
Confusion matrices



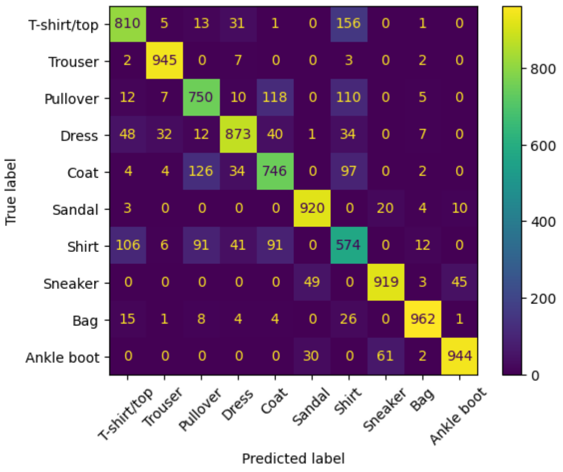
	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	803	4	14	27	1	0	153	0	2	0
Trouser	1	943	0	9	1	0	1	0	2	0
Pullover	12	11	749	16	127	0	116	0	11	0
Dress	47	32	8	871	38	2	36	0	7	1
Coat	4	5	124	33	750	0	91	0	2	0
Sandal	3	0	0	0	0	918	0	22	4	9
Shirt	115	4	96	40	80	0	576	0	16	0
Sneaker	0	0	0	0	0	49	0	919	4	44
Bag	15	1	9	4	3	3	27	0	952	1
Ankle boot	0	0	0	0	0	28	0	59	0	945



	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	817	4	16	29	1	0	150	0	1	0
Trouser	1	945	0	8	1	0	2	0	2	0
Pullover	12	9	739	11	116	0	114	0	8	0
Dress	48	30	11	870	38	2	35	0	7	0
Coat	4	6	132	34	743	0	93	0	1	0
Sandal	3	0	0	0	0	921	0	23	5	10
Shirt	100	4	94	44	96	0	579	0	14	0
Sneaker	0	0	0	0	0	51	0	919	3	40
Bag	15	2	8	4	5	0	27	0	957	1
Ankle boot	0	0	0	0	0	26	0	58	2	949



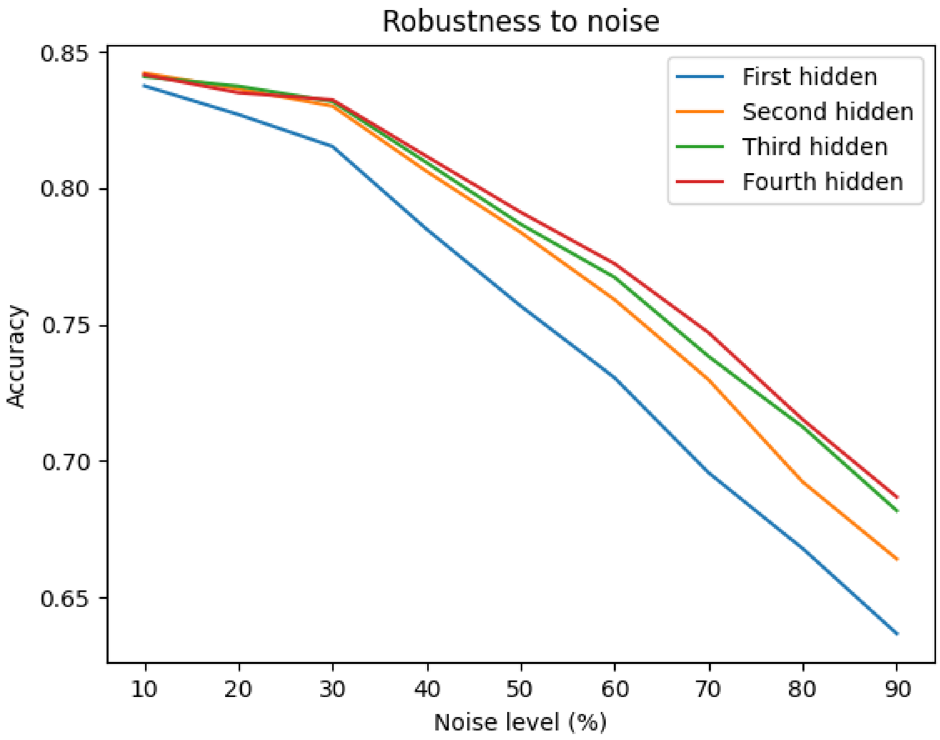
	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	806	6	11	27	0	0	159	0	1	0
Trouser	1	939	0	7	0	0	1	0	1	0
Pullover	11	11	756	15	121	0	116	0	7	0
Dress	45	33	8	871	41	1	32	0	5	1
Coat	4	3	122	35	748	0	91	0	2	0
Sandal	3	0	0	0	0	919	0	18	4	8
Shirt	116	7	96	41	87	0	576	0	13	0
Sneaker	0	0	0	0	0	50	0	922	3	42
Bag	14	1	7	4	3	2	25	0	962	1
Ankle boot	0	0	0	0	0	28	0	60	2	948



	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	810	5	13	31	1	0	156	0	1	0
Trouser	2	945	0	7	0	0	3	0	2	0
Pullover	12	7	750	10	118	0	110	0	5	0
Dress	48	32	12	873	40	1	34	0	7	0
Coat	4	4	126	34	746	0	97	0	2	0
Sandal	3	0	0	0	0	920	0	20	4	10
Shirt	106	6	91	41	91	0	574	0	12	0
Sneaker	0	0	0	0	0	49	0	919	3	45
Bag	15	1	8	4	4	0	26	0	962	1
Ankle boot	0	0	0	0	0	30	0	61	2	944

Psychometric curves

The psychometric curves are useful to provide insights into the model’s performance as
a function of noise level added to the test data.



By gradually increasing the amount of noise, we can observe that the model's performance decreases as the noise level increases, which is expected.

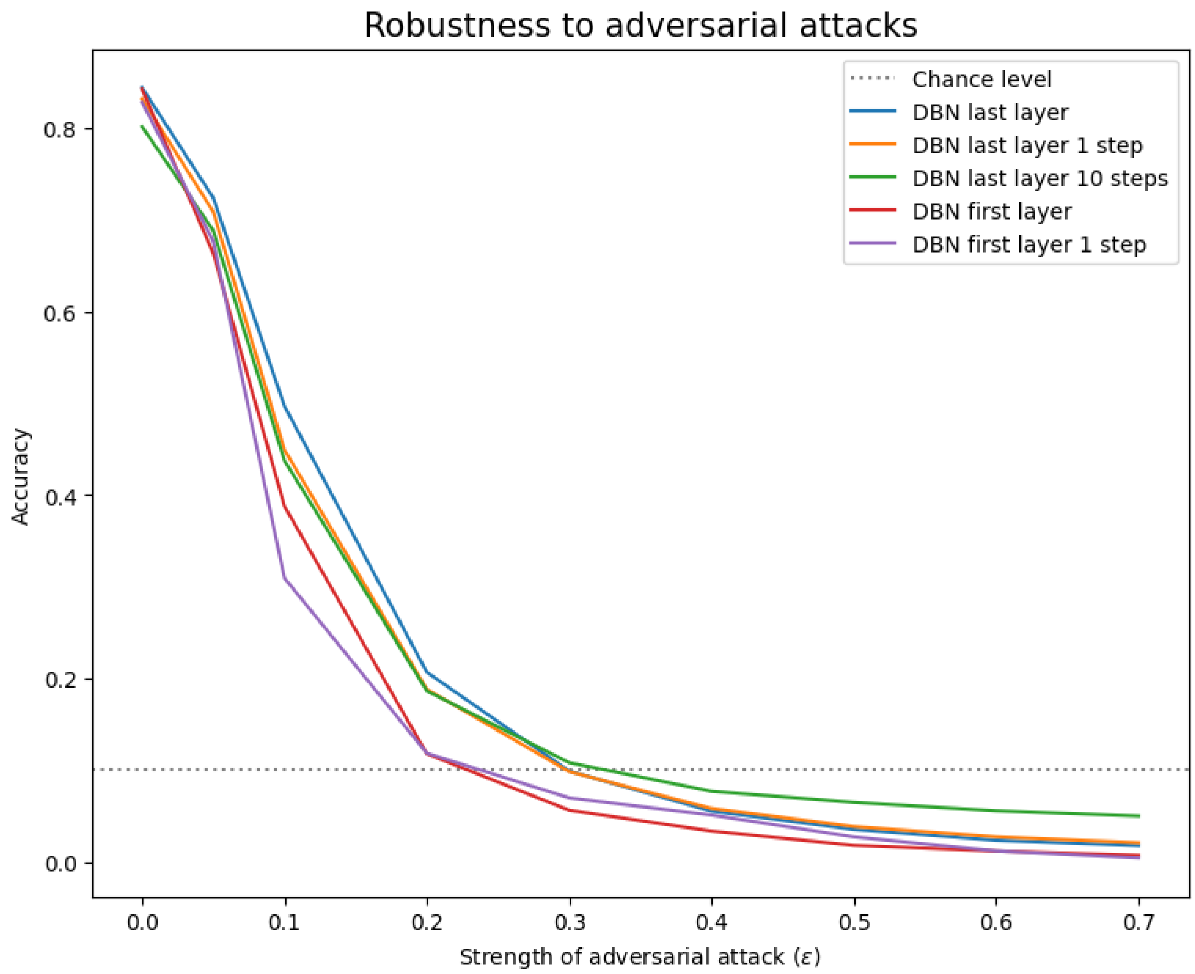
It is important to note that the deeper layers of the model are less sensitive to noise, which is also evident in visual receptive fields. On the other hand, the first layers are more sensitive to noise, as seen in adversarial attacks.

This suggests that deeper models are more robust against added noise. This can be attributed to the increasing complexity and abstractness of the representations learned by the model as it goes deeper into the layers. The last layer, which receives knowledge from the previous layers, is the most abstract and capable of generalizing well to noisy data.

Adversarial attacks

To assess the robustness of the model, we employed the Fast Gradient Sign Method (FGSM) to create adversarial examples. Adversarial examples are generated by modifying the input data based on the sign of the loss gradient with respect to the input.

The epsilon parameter determines the magnitude of the modification, ensuring that the resulting adversarial example remains similar to the original image.



The plot resulting from the robustness analysis indicates that as the deep Boltzmann machine becomes deeper, it becomes more resilient against adversarial attacks.

The `DBN first layer` refers to a DBN constructed using the readout from the first layer, while the `DBN last layer` uses the last layer (i.e., the fourth layer). This was done to demonstrate that the earlier layers of the DBN are more susceptible to noisy data.

The graph shows that as we increase the epsilon value, the accuracy decreases for every model. However, for the deep Boltzmann machine with 10-step reconstruction, the curve is relatively flat, suggesting that reconstruction could be effective.

The last two curves were plotted to illustrate that the first layers are highly sensitive to noise in the data.

Results

Our findings reveal that the DBN model achieved an accuracy of approximately 84% in classification. We observed that increasing the number of reconstruction steps enhanced the model's performance, but only when the epsilon value was increased. The plot illustrating the model's resilience to adversarial attacks demonstrated that the DBN with 10 reconstruction steps outperformed the other tested models in terms of robustness.

In general, our implementation of the DBN model exhibited good performance on the Fashion MNIST dataset and demonstrated reasonable resilience to adversarial attacks. However, our results suggest that increasing the number of reconstruction steps may not always lead to improved performance.