

IFT780 – TP3

Question 2

Endroits où du code a été ajouté (texte en **vert**: ajout, texte en **rouge**: retrait)

- Fichier 'src/train.py':

```
[...]

from copy import copy

[...]

    if data_augment:
        print('Data augmentation activated!')
        data_augment_transforms = [
            transforms.RandomRotation(15),
            transforms.ColorJitter(contrast=0.1,
                                   hue=0.1),
            transforms.RandomHorizontalFlip(p=0.5),
            transforms.RandomCrop(32, padding=4)
        ]
    else:
        print('Data augmentation NOT activated!')
        data_augment_transforms = []

[...]

base_transform = transforms.Compose([
    transforms.ToTensor(),
    transforms.Normalize((0.5, 0.5, 0.5), (0.5, 0.5, 0.5))
])

train_transform = transforms.Compose([
    *data_augment_transforms,
    transforms.ToTensor(),
    transforms.Normalize((0.5, 0.5, 0.5), (0.5, 0.5, 0.5))
])

if args.dataset == 'cifar10':
    # Download the train and test set and apply transform on it
    train_set = datasets.CIFAR10(root='../data', train=True, download=True,
transform=base_transform)
    train_set = datasets.CIFAR10(root='../data', train=True, download=True,
transform=train_transform)
    test_set = datasets.CIFAR10(root='../data', train=False, download=True,
transform=base_transform)

    elif args.dataset == 'svhn':
        # Download the train and test set and apply transform on it
        train_set = datasets.SVHN(root='../data', split='train', download=True,
transform=base_transform)
        train_set = datasets.SVHN(root='../data', split='train', download=True,
transform=train_transform)
        test_set = datasets.SVHN(root='../data', split='test', download=True, transform=base_transform)

    if val_set:
        len_val_set = int(len(train_set) * val_set)
        train_set, val_set = torch.utils.data.random_split(train_set, [len(train_set) - len_val_set,
len_val_set])
        val_set.dataset = copy(train_set.dataset)
        val_set.dataset.transform = base_transform

[...]
```

Courbes d'entraînement et de validation

1. `--model=CnnVanilla --dataset=cifar10 --num-epochs=20 --batch_size=100`



```
(ift725) simon@alien:~/tp3/src$ python train.py --model=CnnVanilla --dataset=cifar10 --num-epochs=20 --batch_size=100
```

Data augmentation NOT activated!

Files already downloaded and verified

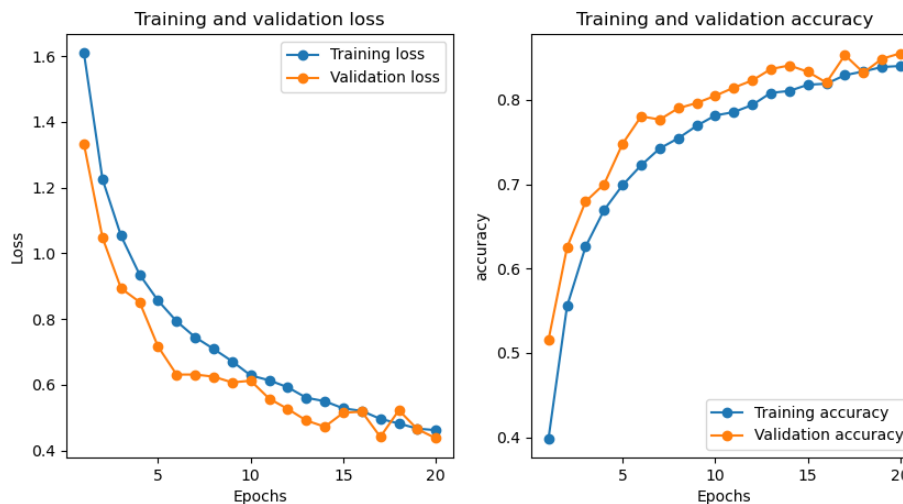
Files already downloaded and verified

Training CnnVanilla on cifar10 for 20 epochs

Finished training.

Accuracy (or Dice for UNet) on the test set: 80.760 %

2. `--model=CnnVanilla --dataset=cifar10 --num-epochs=20 --batch_size=100 --data_aug`



```
(ift725) simon@alien:~/tp3/src$ python train.py --model=CnnVanilla --dataset=cifar10 --num-epochs=20 --batch_size=100 --data_aug
```

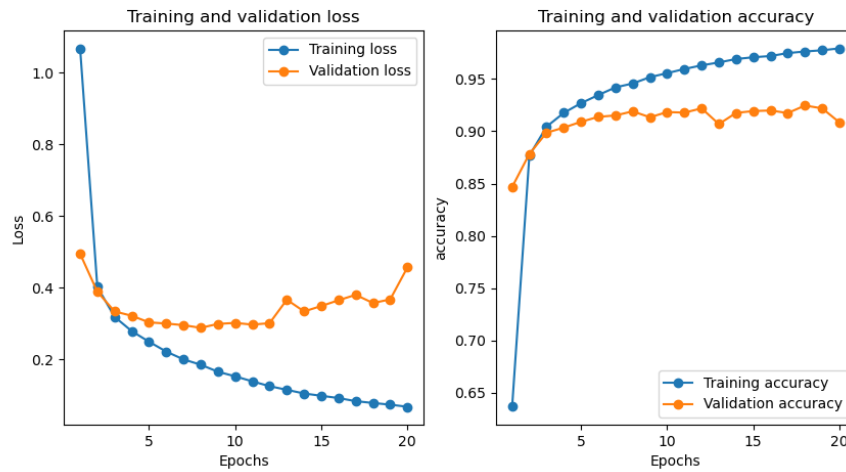
Data augmentation activated!

Files already downloaded and verified

*Files already downloaded and verified
Training CnnVanilla on cifar10 for 20 epochs*

*Finished training.
Accuracy (or Dice for UNet) on the test set: 82.780 %*

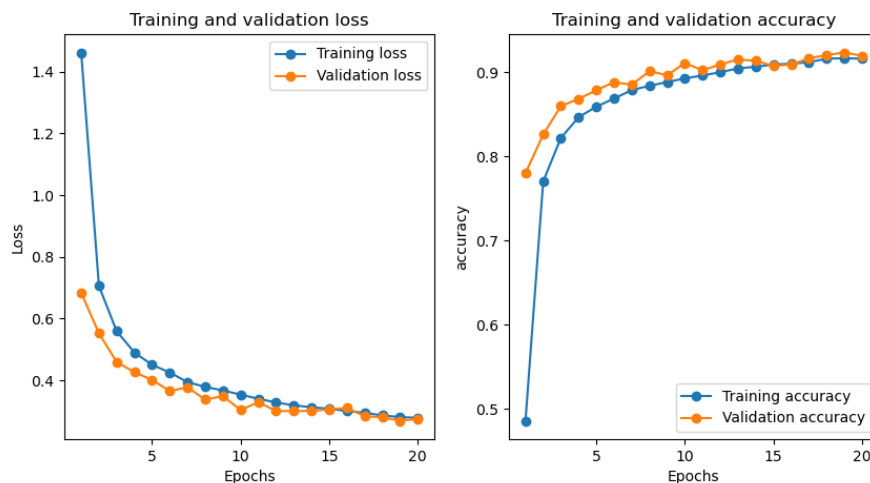
3. `--model=CnnVanilla --dataset=svhn --num-epochs=20 --batch_size=100`



```
((ift725) simon@alien:~/tp3/src$ python train.py --model=CnnVanilla --dataset=svhn --num-epochs=20
--batch_size=100
Data augmentation NOT activated!
Using downloaded and verified file: ../data/train_32x32.mat
Using downloaded and verified file: ../data/test_32x32.mat
Training CnnVanilla on svhn for 20 epochs
```

Finished training.
Accuracy (or Dice for UNet) on the test set: 90.627 %

4. `--model=CnnVanilla --dataset=svhn --num-epochs=20 --batch_size=100 --data_aug`

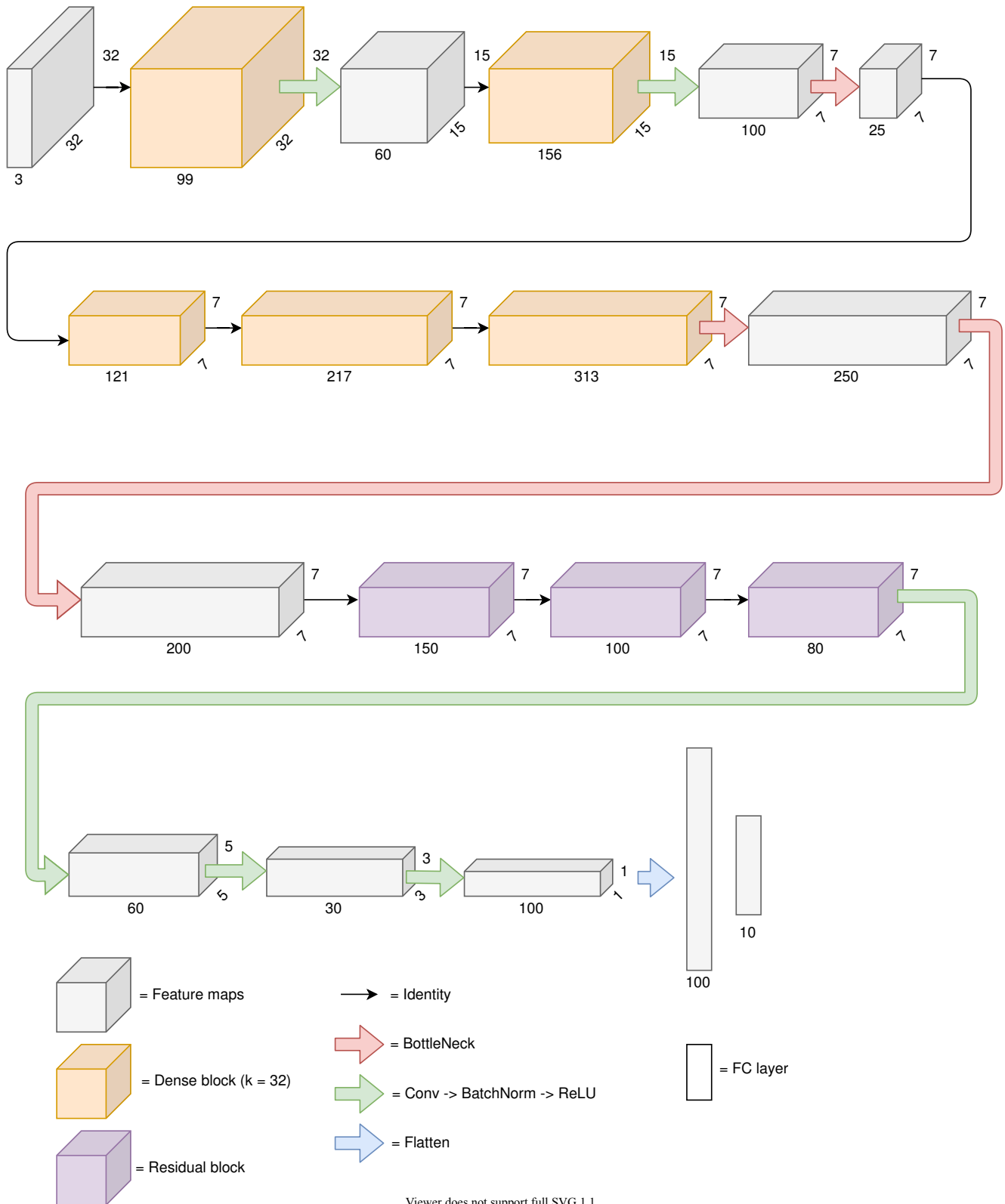


```
((ift725) simon@alien:~/tp3/src$ python train.py --model=CnnVanilla --dataset=svhn --num-epochs=20
--batch_size=100 --data_aug
Data augmentation activated!
Using downloaded and verified file: ../data/train_32x32.mat
Using downloaded and verified file: ../data/test_32x32.mat
Training CnnVanilla on svhn for 20 epochs
```

Finished training.
Accuracy (or Dice for UNet) on the test set: 82.636 %

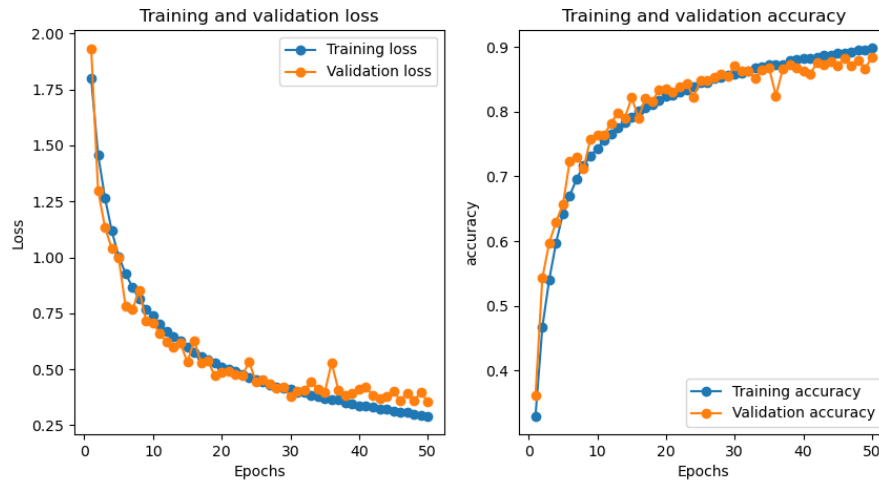
Question 2

Représentation graphique du réseau IFT725Net



Courbes d'apprentissage et de validation

1. `--model=IFT725Net --dataset=cifar10 --num-epochs=50 --batch_size=100 --data_aug`



```
(ift725) nicolas@alien:~/Documents/IFT780/tp3/src$ python train.py --model=IFT725Net --num-epochs=50 --dataset=cifar10 --batch_size=100 --data_aug
```

Data augmentation activated!

Files already downloaded and verified

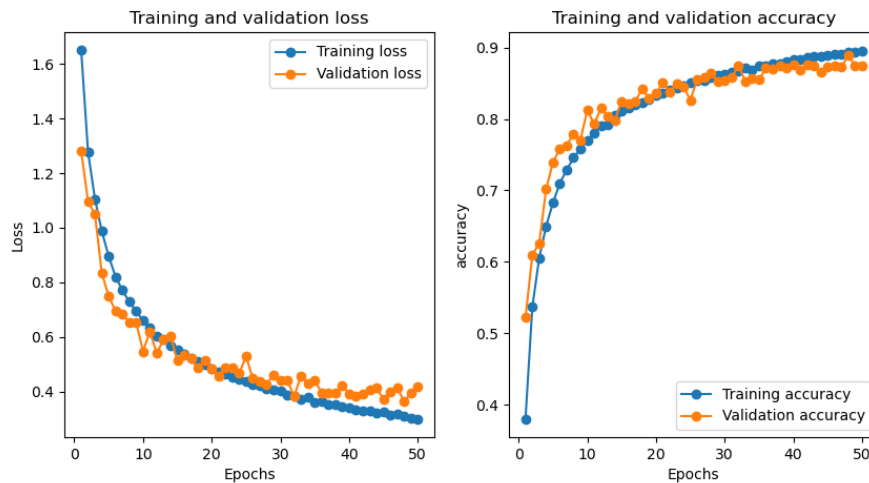
Files already downloaded and verified

Training IFT725Net on cifar10 for 50 epochs

Finished training.

Accuracy (or Dice for UNet) on the test set: 86.520 %

2. `--model=CnnVanilla --dataset=cifar10 --num-epochs=50 --batch_size=100 --data_aug`



```
(ift725) nicolas@alien:~/Documents/IFT780/tp3/src$ python train.py --model=CnnVanilla --num-epochs=50 --dataset=cifar10 --batch_size=100 --data_aug
```

Data augmentation activated!

Files already downloaded and verified

Files already downloaded and verified

Training CnnVanilla on cifar10 for 50 epochs

Finished training.

Accuracy (or Dice for UNet) on the test set: 84.650%