Introduction to Machine Learning
Assignment 7
Name: Dhanush Kumar Antharam
Student ID: 801282468

Github Repo : https://github.com/Dhanush-adk/machine_learning/tree/main/assignment_7

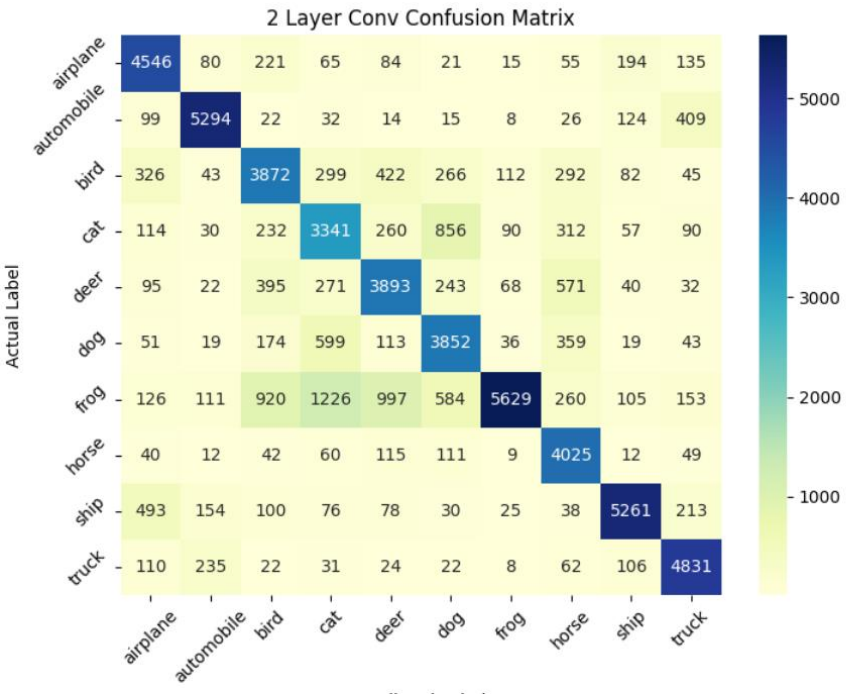
Problem 1 (40 pts):

a. Build a Convolutional Neural Network, like what we built in lectures to classify the images across all 10 classes in CIFAR 10. You need to adjust the fully connected layer at the end properly concerning the number of output classes. Train your network for 300 epochs. Report your training time, training loss, and evaluation accuracy after 300 epochs. Analyze your results in your report and compare them against a fully connected network (homework 2) on training time, achieved accuracy, and model size. Make sure to submit your code by providing the GitHub URL of your course repository for this course.

Loss on each epoch are as follows:

```
2023-12-12 04:21:42.893088 Epoch 256, Training loss 0.546825067847586
2023-12-12 04:22:11.581750 Epoch 258, Training loss 0.543790294005133
2023-12-12 04:22:40.343384 Epoch 260, Training loss 0.5437634525739629
2023-12-12 04:23:10.612402 Epoch 262, Training loss 0.5437600556427561
2023-12-12 04:23:39.857520 Epoch 264, Training loss 0.540446759775624
2023-12-12 04:24:08.743524 Epoch 266, Training loss 0.5404462462381634
2023-12-12 04:24:37.761662 Epoch 268, Training loss 0.5388082106552465
2023-12-12 04:25:07.166964 Epoch 270, Training loss 0.5380695255668572
2023-12-12 04:25:36.520728 Epoch 272, Training loss 0.5393459346250195
2023-12-12 04:26:05.207166 Epoch 274, Training loss 0.5348441857282463
2023-12-12 04:26:33.914277 Epoch 276, Training loss 0.5350457947424916
2023-12-12 04:27:02.840809 Epoch 278, Training loss 0.5341774278208423
2023-12-12 04:27:31.429601 Epoch 280, Training loss 0.5320406553842832
2023-12-12 04:28:01.278192 Epoch 282, Training loss 0.5324782535357548
2023-12-12 04:28:29.655727 Epoch 284, Training loss 0.5350743438048131
2023-12-12 04:28:58.644477 Epoch 286, Training loss 0.5322250325775817
2023-12-12 04:29:27.515278 Epoch 288, Training loss 0.5290348834317663
2023-12-12 04:29:56.722020 Epoch 290, Training loss 0.5298202686068957
2023-12-12 04:30:25.332128 Epoch 292, Training loss 0.5291941248242508
2023-12-12 04:30:53.890887 Epoch 294, Training loss 0.5284989783564187
2023-12-12 04:31:22.272021 Epoch 296, Training loss 0.5262964345953044
2023-12-12 04:31:50.938768 Epoch 298, Training loss 0.5260450314073002
2023-12-12 04:32:19.849614 Epoch 300, Training loss 0.5249976403344317
```

Confusion Matrix :



Classification Report:

	precision	recall	f1-score	support
airplane	0.76	0.84	0.80	5416
automobile	0.88	0.88	0.88	6043
bird	0.65	0.67	0.66	5759
cat	0.56	0.62	0.59	5382
deer	0.65	0.69	0.67	5630
dog	0.64	0.73	0.68	5265
frog	0.94	0.56	0.70	10111
horse	0.67	0.90	0.77	4475
ship	0.88	0.81	0.84	6468
truck	0.81	0.89	0.84	5451
accuracy			0.74	60000
macro avg	0.74	0.76	0.74	60000
weighted avg	0.76	0.74	0.74	60000

```
In [18]: accuracy, predictions, expected_labels = validate(model, train_loader, val_loader)

Accuracy train: 0.77
Accuracy val: 0.60
```

Training time – 1hr 16 mins
Training Loss – 0.525
Evaluation accuracy (After 300 epochs) = 60%

This CNN-based network has outperformed the fully connected network from Homework 6 with respect to accuracy, loss, and other evaluation metrics. Additionally, the training time for the CNN was comparatively shorter, taking only 1 hour and 16 minutes, which is faster than the training time for the fully connected network.

b. Extend your CNN by adding one more additional convolution layer followed by an activation function and pooling function. You also need to adjust your fully connected layer properly with respect to intermediate feature dimensions. Train your network for 300 epochs. Report your training time, loss, and evaluation accuracy after 300 epochs. Analyze your results in your report and compare your model size and accuracy over the baseline implementation in Problem1.a. Do you see any over-fitting? Make sure to submit your code by providing the GitHub URL of your course repository for this course.

A. Loss Per each epoch

```
train_loader = torch.utils.data.DataLoader(cifar10, batch_size=64,
                                           shuffle=True)

model2 = Net2().to(device=device)
optimizer2 = optim.SGD(model2.parameters(), lr=1e-2)
loss_fn = nn.CrossEntropyLoss()

training_loop(
    n_epochs = 300,
    optimizer = optimizer2,
    model = model2,
    loss_fn = loss_fn,
    train_loader = train_loader,
)
```

```
2023-12-12 07:39:23.815951 Epoch 1, Training loss 2.1790317624731137
2023-12-12 07:39:39.327071 Epoch 2, Training loss 2.0130565538430765
2023-12-12 07:40:09.865657 Epoch 4, Training loss 1.8026186029624451
2023-12-12 07:40:46.200589 Epoch 6, Training loss 1.5957095351670405
2023-12-12 07:41:17.141064 Epoch 8, Training loss 1.4861633019983922
2023-12-12 07:41:47.469602 Epoch 10, Training loss 1.4081529291999308
2023-12-12 07:42:18.427390 Epoch 12, Training loss 1.3559277421983003
2023-12-12 07:42:50.086237 Epoch 14, Training loss 1.3143094544825347
2023-12-12 07:43:21.468714 Epoch 16, Training loss 1.28187413273565
2023-12-12 07:43:52.653399 Epoch 18, Training loss 1.2517708120748514
2023-12-12 07:44:23.308071 Epoch 20, Training loss 1.2252586416881104
2023-12-12 07:44:53.754672 Epoch 22, Training loss 1.20161061084179
2023-12-12 07:45:24.493350 Epoch 24, Training loss 1.1817503672883944
2023-12-12 07:45:55.673525 Epoch 26, Training loss 1.1643364339533364
```

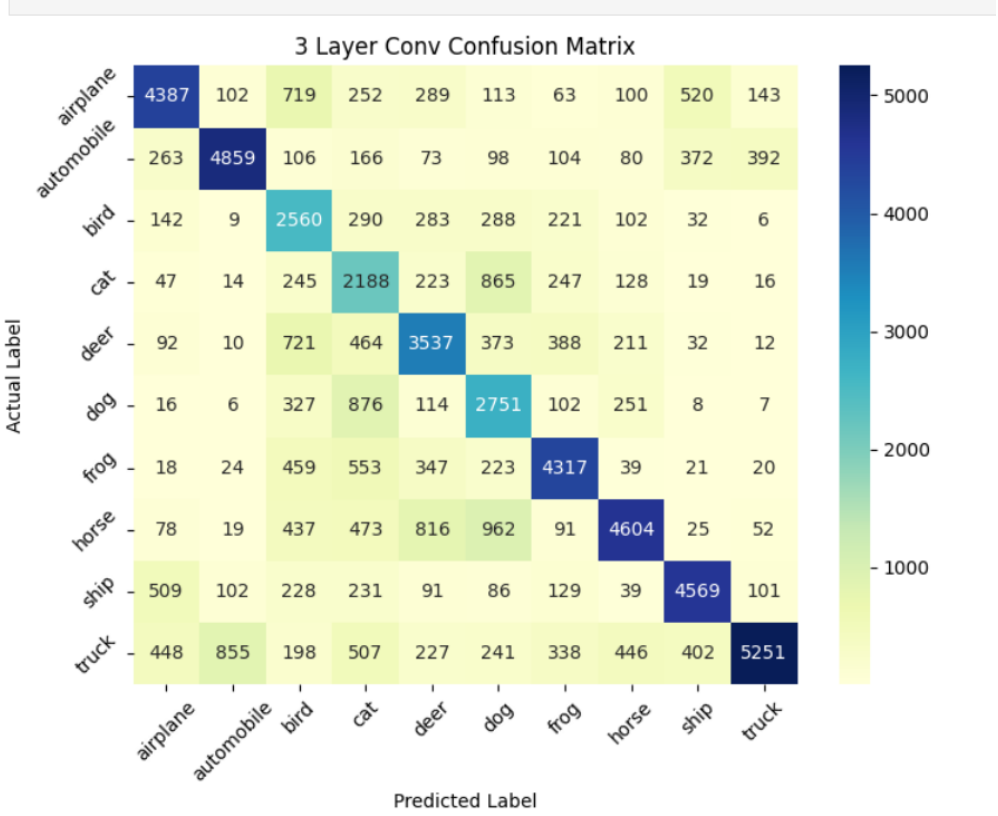
```
2023-12-12 08:41:37.380243 Epoch 244, Training loss 0.8864399420330896
2023-12-12 08:42:07.780874 Epoch 246, Training loss 0.8867716391373168
2023-12-12 08:42:38.269519 Epoch 248, Training loss 0.8838707204822385
2023-12-12 08:43:08.535043 Epoch 250, Training loss 0.8835676870763759
2023-12-12 08:43:39.952524 Epoch 252, Training loss 0.8842979822774677
2023-12-12 08:44:10.307096 Epoch 254, Training loss 0.8833494620859775
2023-12-12 08:44:40.712766 Epoch 256, Training loss 0.8836175351953872
2023-12-12 08:45:11.167285 Epoch 258, Training loss 0.8795141435187795
2023-12-12 08:45:42.207648 Epoch 260, Training loss 0.8801805381579777
2023-12-12 08:46:13.490590 Epoch 262, Training loss 0.8812946383757969
2023-12-12 08:46:45.016162 Epoch 264, Training loss 0.8790078829316532
2023-12-12 08:47:14.904626 Epoch 266, Training loss 0.8795828929795023
2023-12-12 08:47:45.131697 Epoch 268, Training loss 0.8791865917575329
2023-12-12 08:48:16.179319 Epoch 270, Training loss 0.8770411691397352
2023-12-12 08:48:46.779659 Epoch 272, Training loss 0.8781590044041119
2023-12-12 08:49:17.024630 Epoch 274, Training loss 0.8775035661962026
2023-12-12 08:49:47.093501 Epoch 276, Training loss 0.8765670083977682
2023-12-12 08:50:17.368082 Epoch 278, Training loss 0.87611485030645
2023-12-12 08:50:48.339101 Epoch 280, Training loss 0.8737155266125184
2023-12-12 08:51:19.420434 Epoch 282, Training loss 0.8757887943016599
2023-12-12 08:51:49.331140 Epoch 284, Training loss 0.875020097969743
2023-12-12 08:52:19.120427 Epoch 286, Training loss 0.8751379801031879
2023-12-12 08:52:50.587056 Epoch 288, Training loss 0.8739472793419952
2023-12-12 08:53:22.777705 Epoch 290, Training loss 0.874782736953872
2023-12-12 08:53:54.828219 Epoch 292, Training loss 0.8740949435612125
2023-12-12 08:54:26.140987 Epoch 294, Training loss 0.8722011343292568
2023-12-12 08:54:56.793419 Epoch 296, Training loss 0.8706158800122074
2023-12-12 08:55:28.693852 Epoch 298, Training loss 0.8712239560416287
2023-12-12 08:56:00.346263 Epoch 300, Training loss 0.8702678288264043
```

Training Accuracy and Evaluation Accuracy

In [83]:
```
accuracy2, predictions2, expected_labels2 = validate(model2, train_loader, val_loader)
```

```
Accuracy train: 0.66
Accuracy val: 0.62
```

Confusion Matrix is as follows:



Classification Report:

	precision	recall	f1-score	support
airplane	0.73	0.66	0.69	6688
automobile	0.81	0.75	0.78	6513
bird	0.43	0.65	0.52	3933
cat	0.36	0.55	0.44	3992
deer	0.59	0.61	0.60	5840
dog	0.46	0.62	0.53	4458
frog	0.72	0.72	0.72	6021
horse	0.77	0.61	0.68	7557
ship	0.76	0.75	0.76	6085
truck	0.88	0.59	0.70	8913
accuracy			0.65	60000
macro avg	0.65	0.65	0.64	60000
weighted avg	0.69	0.65	0.66	60000

Training time – 1hr 17 min
Training Loss – 0.870
Evaluation Accuracy (after 300 epochs) – 62%

Incorporating an additional CNN layer and a pooling layer has enhanced the model's accuracy without markedly affecting the training time.(relatively to problem 1.a)

Problem 2 (50pts)

a. Build a ResNet-based Convolutional Neural Network, like what we built in lectures (with skip connections), to classify the images across all 10 classes in CIFAR 10. For this problem, let's use **10** blocks for ResNet and call it ResNet-10. Use similar dimensions and channels as we need in lectures. Train your network for 300 epochs. Report your training time, training loss, and evaluation accuracy after 300 epochs. Analyze your results in your report and compare them against problem 1.b on training time, achieved accuracy, and model size. Make sure to submit your code by providing the GitHub URL of your course repository for this course.

A. Loss per Epoch

In [18]:
```
model = NetResDeep(n_chans1=32, n_blocks=10).to(device=device)
optimizer = optim.SGD(model.parameters(), lr=3e-3)
loss_fn = nn.CrossEntropyLoss()
training_loop(
    n_epochs = 300,
    optimizer = optimizer,
    model = model,
    loss_fn = loss_fn,
    train_loader = train_loader,
)
```

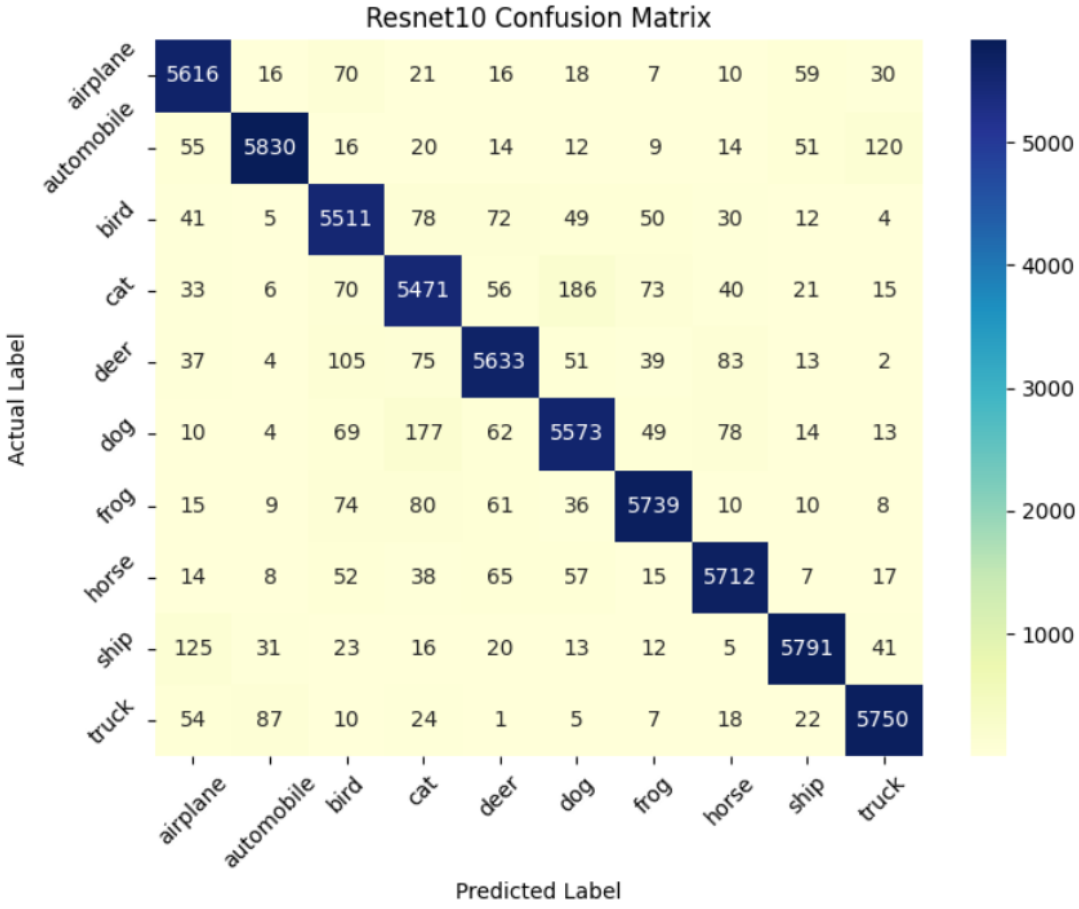
```
2023-12-12 10:47:06.135530 Epoch 1, Training loss 1.5790970384197516
2023-12-12 10:47:22.333087 Epoch 2, Training loss 1.2757431460128559
2023-12-12 10:47:53.154459 Epoch 4, Training loss 1.0664658080059524
2023-12-12 10:48:24.100999 Epoch 6, Training loss 0.9422920360369935
2023-12-12 10:48:55.035614 Epoch 8, Training loss 0.8625202439034206
2023-12-12 10:49:25.594854 Epoch 10, Training loss 0.8010393645392727
2023-12-12 10:49:56.665333 Epoch 12, Training loss 0.7496834203972698
2023-12-12 10:50:28.224389 Epoch 14, Training loss 0.7033575898416516
2023-12-12 10:50:58.680209 Epoch 16, Training loss 0.6660753198144379
2023-12-12 10:51:29.023902 Epoch 18, Training loss 0.6298215005425254
2023-12-12 10:51:59.468437 Epoch 20, Training loss 0.5918687623392216
2023-12-12 10:52:30.093259 Epoch 22, Training loss 0.5632473225499756
2023-12-12 10:53:00.385547 Epoch 24, Training loss 0.5314390877463158
```

```
2023-12-12 11:55:23.138967 Epoch 270, Training loss 0.025217936520202645
2023-12-12 11:55:53.816142 Epoch 272, Training loss 0.01958685273210795
2023-12-12 11:56:24.695782 Epoch 274, Training loss 0.04074301843639652
2023-12-12 11:56:55.702690 Epoch 276, Training loss 0.03162873692014326
2023-12-12 11:57:25.390711 Epoch 278, Training loss 0.04046457375384653
2023-12-12 11:57:55.929188 Epoch 280, Training loss 0.03880541950387902
2023-12-12 11:58:27.202799 Epoch 282, Training loss 0.019139713710564228
2023-12-12 11:58:58.290824 Epoch 284, Training loss 0.021224404171409528
2023-12-12 11:59:27.972482 Epoch 286, Training loss 0.01951712815987337
2023-12-12 11:59:57.590305 Epoch 288, Training loss 0.027573353126492773
2023-12-12 12:00:28.018970 Epoch 290, Training loss 0.021488327600139363
2023-12-12 12:00:57.788685 Epoch 292, Training loss 0.018064392131763283
2023-12-12 12:01:27.297237 Epoch 294, Training loss 0.025434375908806553
2023-12-12 12:01:56.459466 Epoch 296, Training loss 0.017801390564114023
2023-12-12 12:02:26.025633 Epoch 298, Training loss 0.02003223076890668
2023-12-12 12:02:55.722471 Epoch 300, Training loss 0.023128647690982915
time: 1h 16min 9s (started: 2023-12-12 10:46:46 +00:00)
```

Validation Accuracy and Training Accuracy

In [20]:
```
accuracy, predictions, expected_labels = validate(model, train_loader, val_loader)
```

```
Accuracy train: 1.00
Accuracy val: 0.67
time: 17.3 s (started: 2023-12-12 12:02:55 +00:00)
```



Classification Report

	precision	recall	f1-score	support
airplane	0.94	0.96	0.95	5863
automobile	0.97	0.95	0.96	6141
bird	0.92	0.94	0.93	5852
cat	0.91	0.92	0.91	5971
deer	0.94	0.93	0.94	6042
dog	0.93	0.92	0.93	6049
frog	0.96	0.95	0.95	6042
horse	0.95	0.95	0.95	5985
ship	0.97	0.95	0.96	6077
truck	0.96	0.96	0.96	5978
accuracy			0.94	60000
macro avg	0.94	0.94	0.94	60000
weighted avg	0.94	0.94	0.94	60000

time: 141 ms (started: 2023-12-12 12:03:13 +00:00)

Training time – 1h 16 min
Training Loss – 0.0231
Evaluation Accuracy – 67%

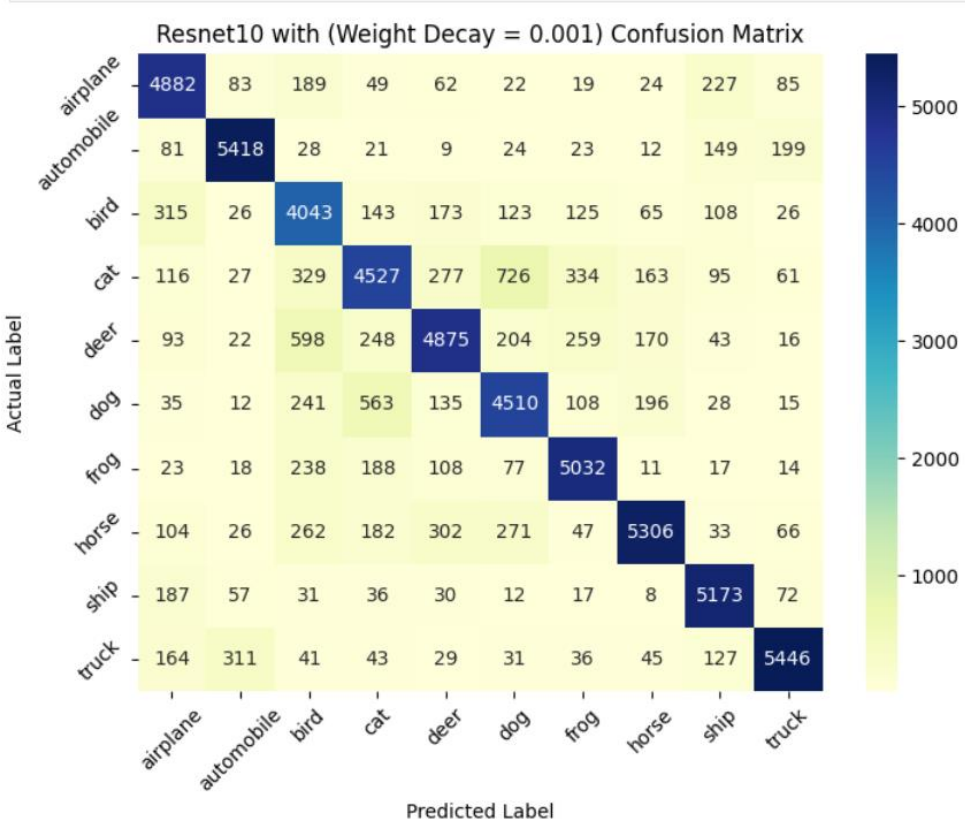
The evaluation accuracy has improved relative to the models trained in problem 1, and there has been a noteworthy reduction in training loss, indicating significant improvement.

b. Perform three additional training and evaluations for your ResNet-10 to assess the impacts of regularization on your ResNet-10.

- **Weight Decay with lambda of 0.001**

```
2023-12-12 12:03:43.329008 Epoch 1, Training loss 1.9939807186955991
2023-12-12 12:05:54.987212 Epoch 10, Training loss 1.160557784235386
2023-12-12 12:08:21.225743 Epoch 20, Training loss 0.9642956210371784
2023-12-12 12:10:47.389704 Epoch 30, Training loss 0.8923261045952282
2023-12-12 12:13:14.563115 Epoch 40, Training loss 0.8513407973224855
2023-12-12 12:15:41.806490 Epoch 50, Training loss 0.8239225297785171
2023-12-12 12:18:09.297921 Epoch 60, Training loss 0.8045251942656534
2023-12-12 12:20:34.436626 Epoch 70, Training loss 0.7900362135961537
2023-12-12 12:23:00.929913 Epoch 80, Training loss 0.7787863984132362
2023-12-12 12:25:27.097327 Epoch 90, Training loss 0.769704382818983
2023-12-12 12:27:54.276961 Epoch 100, Training loss 0.7624059809595728
2023-12-12 12:30:21.542585 Epoch 110, Training loss 0.7563693132394415
2023-12-12 12:32:47.720628 Epoch 120, Training loss 0.7512598533916961
2023-12-12 12:35:14.701207 Epoch 130, Training loss 0.7469299924190697
2023-12-12 12:37:40.747333 Epoch 140, Training loss 0.7431287888218375
2023-12-12 12:40:06.405826 Epoch 150, Training loss 0.7397044818572072
2023-12-12 12:42:32.657274 Epoch 160, Training loss 0.7367165645827418
2023-12-12 12:44:57.577118 Epoch 170, Training loss 0.7340737159752175
2023-12-12 12:47:24.275933 Epoch 180, Training loss 0.7317476441030917
2023-12-12 12:49:55.562621 Epoch 190, Training loss 0.7296239652139757
2023-12-12 12:52:27.458208 Epoch 200, Training loss 0.7276725996943081
2023-12-12 12:54:57.066905 Epoch 210, Training loss 0.7259029906119228
2023-12-12 12:57:24.818974 Epoch 220, Training loss 0.7243153402567519
2023-12-12 12:59:51.250687 Epoch 230, Training loss 0.7228426321997972
2023-12-12 13:02:17.662512 Epoch 240, Training loss 0.7215289783752178
2023-12-12 13:04:43.007703 Epoch 250, Training loss 0.7203497180853353
2023-12-12 13:07:07.321467 Epoch 260, Training loss 0.7192653342128714
2023-12-12 13:09:33.028793 Epoch 270, Training loss 0.718260667741756
2023-12-12 13:11:58.487019 Epoch 280, Training loss 0.7173492738505458
2023-12-12 13:14:23.254014 Epoch 290, Training loss 0.7165393821723626
2023-12-12 13:16:48.037919 Epoch 300, Training loss 0.7157661236460556
time: 1h 13min 19s (started: 2023-12-12 12:03:28 +00:00)
```

```
Accuracy train: 0.85
Accuracy val: 0.68
time: 15.8 s (started: 2023-12-12 13:17:30 +00:00)
```



	precision	recall	f1-score	support
airplane	0.81	0.87	0.84	5642
automobile	0.90	0.91	0.91	5964
bird	0.67	0.79	0.73	5147
cat	0.75	0.68	0.72	6655
deer	0.81	0.75	0.78	6528
dog	0.75	0.77	0.76	5843
frog	0.84	0.88	0.86	5726
horse	0.88	0.80	0.84	6599
ship	0.86	0.92	0.89	5623
truck	0.91	0.87	0.89	6273
accuracy			0.82	60000
macro avg	0.82	0.82	0.82	60000
weighted avg	0.82	0.82	0.82	60000

time: 136 ms (started: 2023-12-12 13:19:33 +00:00)

Training time – 1h 13 min
Training Loss – 0.7157
Evaluation Accuracy – 68%

The Previous ResNet10 model exhibited lower training loss. However Current model has faster training times compared to earlier models. and it is observed that the current model achieves higher accuracy and an improved F1 score in comparison.

- **Dropout with p=0.3**
  Loss per each epoch

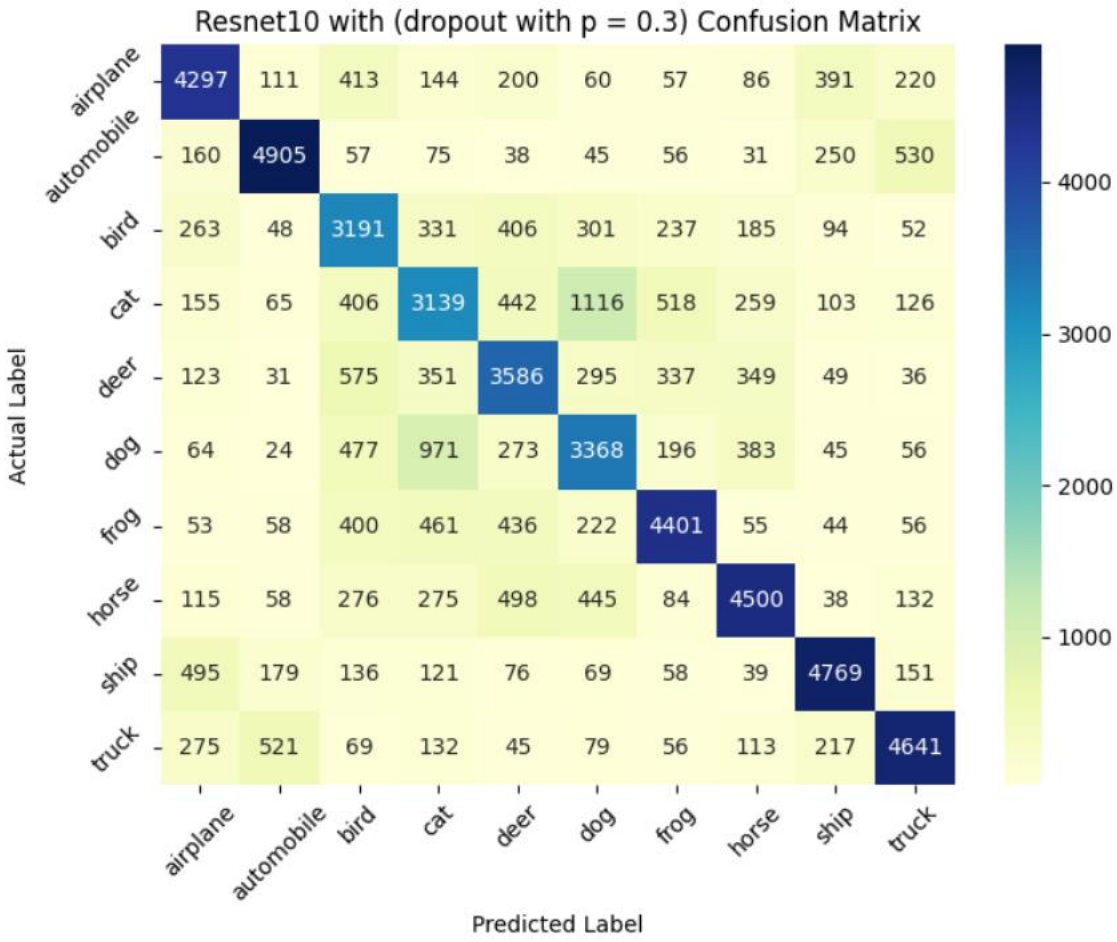
In [35]:
```
model_dropout = NetDropout(n_chans1=32).to(device=device)
optimizer_dropout = optim.SGD(model_dropout.parameters(), lr=1e-2)
loss_fn = nn.CrossEntropyLoss()

training_loop(
    n_epochs = 300,
    optimizer = optimizer_dropout,
    model = model_dropout,
    loss_fn = loss_fn,
    train_loader = train_loader,
)
```

```
2023-12-12 13:25:19.680313 Epoch 1, Training loss 2.0097830361112607
2023-12-12 13:25:34.325807 Epoch 2, Training loss 1.7803532046735133
2023-12-12 13:26:03.245799 Epoch 4, Training loss 1.5596976769549766
2023-12-12 13:26:32.091412 Epoch 6, Training loss 1.4576420965402022
2023-12-12 13:27:01.072646 Epoch 8, Training loss 1.38476528200652
2023-12-12 13:27:29.609623 Epoch 10, Training loss 1.3333351811027283
2023-12-12 13:27:57.346893 Epoch 12, Training loss 1.2905683922950568
2023-12-12 13:28:25.343734 Epoch 14, Training loss 1.2506993476996946
2023-12-12 13:28:52.999487 Epoch 16, Training loss 1.2271902818813958
2023-12-12 13:29:20.793370 Epoch 18, Training loss 1.199903026566176
2023-12-12 13:29:48.270420 Epoch 20, Training loss 1.1831524830949887
2023-12-12 13:30:16.029670 Epoch 22, Training loss 1.1644683083152527
2023-12-12 13:30:43.787693 Epoch 24, Training loss 1.1487101951371068
2023-12-12 13:31:11.746930 Epoch 26, Training loss 1.1307556485885855
2023-12-12 13:31:39.501297 Epoch 28, Training loss 1.1182621831784163
2023-12-12 13:32:07.299505 Epoch 30, Training loss 1.1085543409180458
2023-12-12 13:32:35.308989 Epoch 32, Training loss 1.0966279409882966
2023-12-12 13:33:03.060164 Epoch 34, Training loss 1.0850875540005276
2023-12-12 13:33:31.092743 Epoch 36, Training loss 1.0743814092462936
2023-12-12 13:33:59.151206 Epoch 38, Training loss 1.06408724028741
2023-12-12 13:34:27.315201 Epoch 40, Training loss 1.0611327696791695
2023-12-12 14:22:12.899348 Epoch 246, Training loss 0.8829699832460155
2023-12-12 14:22:40.393439 Epoch 248, Training loss 0.8784472461399215
2023-12-12 14:23:07.990419 Epoch 250, Training loss 0.8777239565044412
2023-12-12 14:23:35.749395 Epoch 252, Training loss 0.8734311551770286
2023-12-12 14:24:03.396049 Epoch 254, Training loss 0.8776112076876413
2023-12-12 14:24:31.141767 Epoch 256, Training loss 0.8751259520459358
2023-12-12 14:24:58.924293 Epoch 258, Training loss 0.870654944682975
2023-12-12 14:25:26.729316 Epoch 260, Training loss 0.8803292644755615
2023-12-12 14:25:54.104320 Epoch 262, Training loss 0.8825559557779975
2023-12-12 14:26:21.975704 Epoch 264, Training loss 0.8739966763864697
2023-12-12 14:26:49.675499 Epoch 266, Training loss 0.8747242310315447
2023-12-12 14:27:17.607161 Epoch 268, Training loss 0.8749992050173337
2023-12-12 14:27:45.459225 Epoch 270, Training loss 0.874500278393021
2023-12-12 14:28:13.149195 Epoch 272, Training loss 0.8744380698941857
2023-12-12 14:28:40.933998 Epoch 274, Training loss 0.8711833411165516
2023-12-12 14:29:08.581913 Epoch 276, Training loss 0.8760753302546718
2023-12-12 14:29:36.259164 Epoch 278, Training loss 0.8762401197953602
2023-12-12 14:30:04.059616 Epoch 280, Training loss 0.8712782637237588
2023-12-12 14:30:31.731854 Epoch 282, Training loss 0.873199677452102
2023-12-12 14:30:59.410701 Epoch 284, Training loss 0.8741421349670576
2023-12-12 14:31:26.767771 Epoch 286, Training loss 0.8721916140497797
2023-12-12 14:31:54.634651 Epoch 288, Training loss 0.8689417132483724
2023-12-12 14:32:22.293630 Epoch 290, Training loss 0.8731379727725788
2023-12-12 14:32:49.950197 Epoch 292, Training loss 0.8651600378706022
2023-12-12 14:33:17.613899 Epoch 294, Training loss 0.8731117397927872
2023-12-12 14:33:44.998731 Epoch 296, Training loss 0.8672170711828925
2023-12-12 14:34:12.677490 Epoch 298, Training loss 0.872050091433708
2023-12-12 14:34:40.209935 Epoch 300, Training loss 0.8699714806302429
time: 1h 9min 34s (started: 2023-12-12 13:25:05 +00:00)
```

Training Accuracy and Validation Accuracy

In [36]:
```
accuracy_dropout, predictions_dropout, expected_labels_dropout = validate(model_dropout, train_loader, val_loader)
```

```
Accuracy train: 0.69
Accuracy val: 0.62
time: 14.7 s (started: 2023-12-12 14:34:41 +00:00)
```



	precision	recall	f1-score	support
airplane	0.72	0.72	0.72	5979
automobile	0.82	0.80	0.81	6147
bird	0.53	0.62	0.57	5108
cat	0.52	0.50	0.51	6329
deer	0.60	0.63	0.61	5732
dog	0.56	0.58	0.57	5857
frog	0.73	0.71	0.72	6186
horse	0.75	0.70	0.72	6421
ship	0.79	0.78	0.79	6093
truck	0.77	0.75	0.76	6148
accuracy			0.68	60000
macro avg	0.68	0.68	0.68	60000
weighted avg	0.68	0.68	0.68	60000

Training Time – 1h 10 min
Training Loss – 0.87
Evaluation Accuracy – 62%

In this case, the model does not show improved evaluation accuracy or training loss relative to other models. However, its training time is significantly shorter compared to other ResNet and CNN-based models.

- **Batch Normalization**

In [42]:
```
model_batch_norm = NetBatchNorm(n_chans1=32).to(device=device)
optimizer_batch_norm = optim.SGD(model_batch_norm.parameters(), lr=1e-2)
loss_fn = nn.CrossEntropyLoss()

training_loop(
    n_epochs = 300,
    optimizer = optimizer_batch_norm,
    model = model_batch_norm,
    loss_fn = loss_fn,
    train_loader = train_loader,
)
```

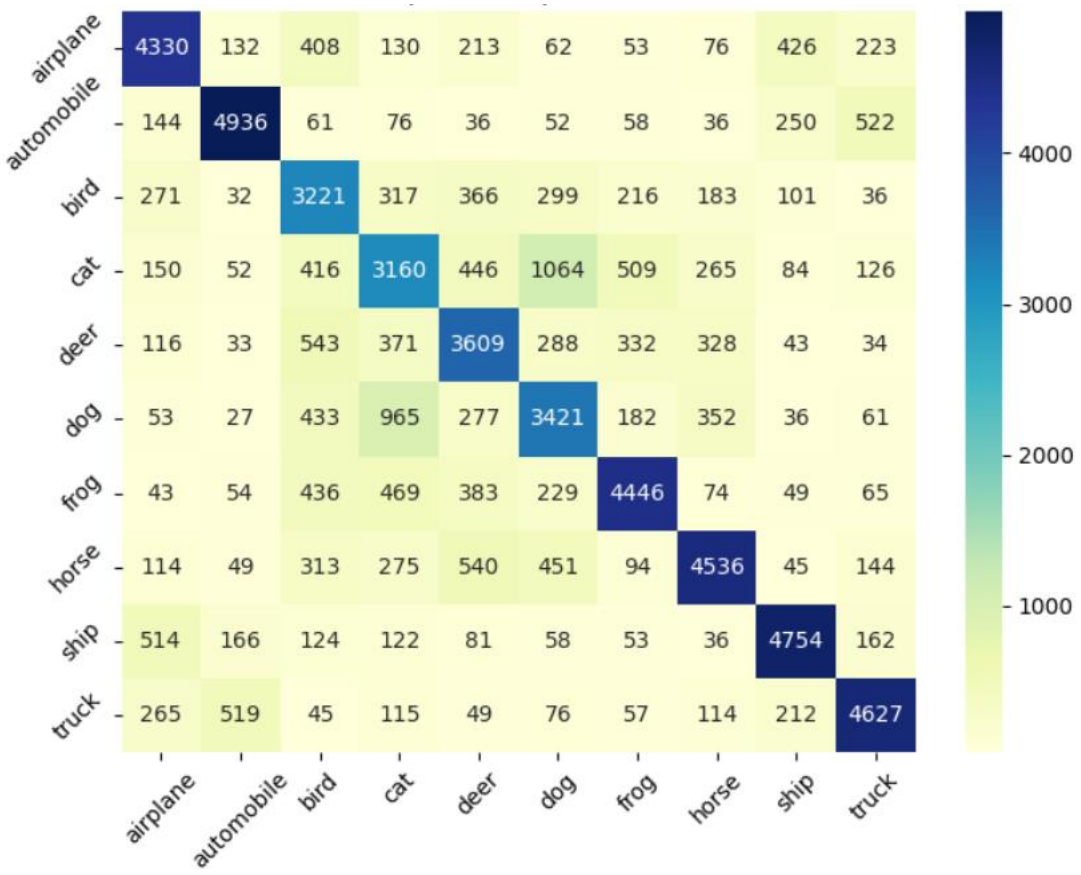
```
2023-12-12 14:54:16.964874 Epoch 1, Training loss 1.7979085021616552
2023-12-12 14:54:31.589961 Epoch 2, Training loss 1.4637582832590088
2023-12-12 14:54:59.820735 Epoch 4, Training loss 1.228781066282326
2023-12-12 14:55:28.471342 Epoch 6, Training loss 1.1067761255956976
2023-12-12 14:55:56.583463 Epoch 8, Training loss 1.0284692133631548
2023-12-12 14:56:24.816715 Epoch 10, Training loss 0.9709521866667911
2023-12-12 14:56:52.810036 Epoch 12, Training loss 0.9238295592462925
2023-12-12 14:57:20.902723 Epoch 14, Training loss 0.8835189616893564
2023-12-12 14:57:48.572475 Epoch 16, Training loss 0.84901308746594
2023-12-12 14:58:16.614381 Epoch 18, Training loss 0.8195060453451503
2023-12-12 14:58:44.443190 Epoch 20, Training loss 0.7933901263319928
2023-12-12 14:59:12.381190 Epoch 22, Training loss 0.7701474607295697
2023-12-12 14:59:40.260873 Epoch 24, Training loss 0.7490701191413128
2023-12-12 15:00:08.038122 Epoch 26, Training loss 0.729625809146925
2023-12-12 15:00:35.638639 Epoch 28, Training loss 0.7114967854355302
2023-12-12 15:01:03.381738 Epoch 30, Training loss 0.6945261056618313
```

```
2023-12-12 15:52:03.796146 Epoch 250, Training loss 0.09696116315586792
2023-12-12 15:52:31.393671 Epoch 252, Training loss 0.09551554993557199
2023-12-12 15:52:59.134853 Epoch 254, Training loss 0.094217654941675
2023-12-12 15:53:27.035755 Epoch 256, Training loss 0.09292600187651641
2023-12-12 15:53:54.873243 Epoch 258, Training loss 0.10875602820507058
2023-12-12 15:54:22.824513 Epoch 260, Training loss 0.10518154517278228
2023-12-12 15:54:50.649301 Epoch 262, Training loss 0.09388007683431744
2023-12-12 15:55:18.509272 Epoch 264, Training loss 0.0894095701787173
2023-12-12 15:55:46.243856 Epoch 266, Training loss 0.08756857704075859
2023-12-12 15:56:13.805296 Epoch 268, Training loss 0.08604711539986189
2023-12-12 15:56:41.482858 Epoch 270, Training loss 0.08476728640611061
2023-12-12 15:57:09.309187 Epoch 272, Training loss 0.0841083372407176
2023-12-12 15:57:37.081429 Epoch 274, Training loss 0.08261649457075636
2023-12-12 15:58:04.852035 Epoch 276, Training loss 0.08139216643222191
2023-12-12 15:58:32.761913 Epoch 278, Training loss 0.0802938675179201
2023-12-12 15:59:00.384448 Epoch 280, Training loss 0.07922483553819339
2023-12-12 15:59:28.521276 Epoch 282, Training loss 0.0782478787755246
2023-12-12 15:59:56.245773 Epoch 284, Training loss 0.0772211203039588
2023-12-12 16:00:23.986742 Epoch 286, Training loss 0.07638566999379402
2023-12-12 16:00:51.757557 Epoch 288, Training loss 0.07532905587150007
2023-12-12 16:01:19.522406 Epoch 290, Training loss 0.07437199000340632
2023-12-12 16:01:47.180026 Epoch 292, Training loss 0.07354304890441316
2023-12-12 16:02:15.115438 Epoch 294, Training loss 0.07259373265959303
2023-12-12 16:02:42.805888 Epoch 296, Training loss 0.07162352606577946
2023-12-12 16:03:10.598708 Epoch 298, Training loss 0.07074789561169303
2023-12-12 16:03:38.441055 Epoch 300, Training loss 0.07004670200921843
time: 1h 9min 35s (started: 2023-12-12 14:54:02 +00:00)
```

43]:

```
accuracy_batch_norm, predictions_batch_norm, expected_labels_batch_norm = va
```

```
Accuracy train: 0.70
Accuracy val: 0.62
time: 15.2 s (started: 2023-12-12 16:38:57 +00:00)
```



In [43]:
```
accuracy_batch_norm, predictions_batch_norm, expected_labels_batch_norm = validate(model_dropout, train_loader, val_loac
```

```
Accuracy train: 0.70
Accuracy val: 0.62
time: 15.2 s (started: 2023-12-12 16:38:57 +00:00)
```

Training time – 1h 9min 35sec
Training Loss – 0.070
Evaluation Accuracy – 62%

The model currently exhibits a lower training loss compared to other models, yet it does not surpass them in terms of evaluation accuracy. Nevertheless, its training process is considerably more rapid, marginally outpacing the speed of the ResNet dropout model with p=0.3.