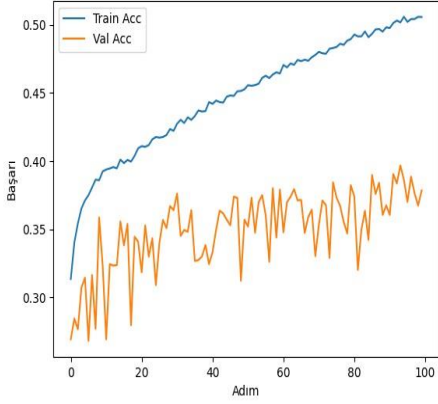


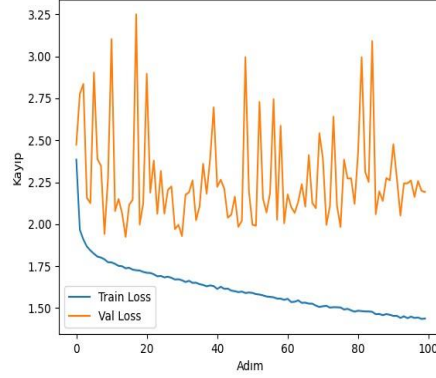
1. RMSprop

➤ Learning_rate : 1e-2

Train accuracy ve Validation Accuracy Grafiği

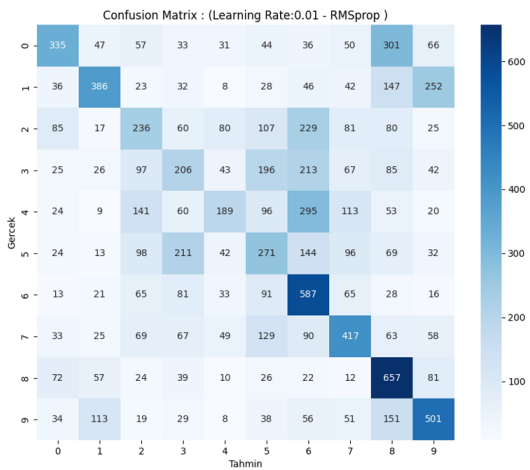


Train loss ve Validation loss Grafiği



Yukarıda ki RMSprop optimizör için grafikleri incelediğimde, 1e-2 öğrenme hızına sahip bu da 1e-6 ya göre çok daha hızlıdır. Cifar veriseti çok büyük bir set olduğu için bu öğrenme oranı bazı kolay olabilir. Fakat train ve validation ın başarı oranlarına baktığımda optimizörümün çok da başarılı olmadığını gözlemlemekteyim. Train seti için 100 adımda maksimum %50 başarı sağlayabilmiş. Bunu genel veri setine vurduğumuzda mac %40 a yakın sonuç vermiş Loss grafiğine baktığımda ise optimizörüm trainset validationa göre daha az kayıp vermiş çok başarılı bir sonuç değil. Aynı zamanda grafiği baktığımda burada overfitting olduğunu gözlemliyorum makine hızlı öğrenme sağladığı için aşırı öğrenmeye kaçmış ve genelde de hata yapmış. Aynılıyor ki Hypermetre seçimleri yanlış olmuş ve overfit engelleyici methodlar kullanılmalı

Confision Matrix



Results

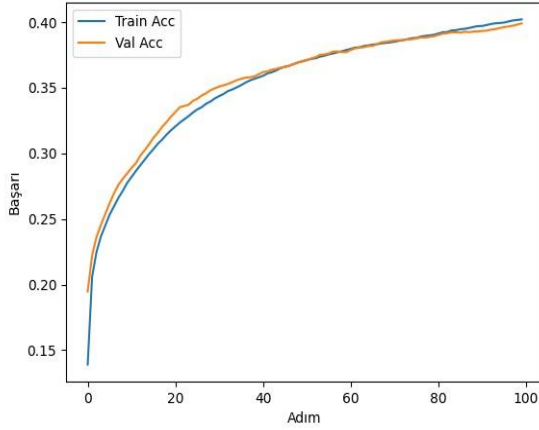
TP: 3785
Overall Accuracy: 37.8%

Class	n (truth) ①	n (classified) ②	Accuracy	Precision	Recall	F1 Score
1	681	1014	89.76%	0.33	0.49	0.40
2	714	1000	90.59%	0.39	0.54	0.45
3	829	1000	86.45%	0.24	0.28	0.26
4	818	1000	85.96%	0.21	0.25	0.23
5	493	1000	88.87%	0.19	0.38	0.25
6	1026	1000	85.18%	0.27	0.26	0.27
7	1732	1000	84.44%	0.59	0.34	0.43
8	994	1000	88.42%	0.42	0.42	0.42
9	1634	1000	86.82%	0.66	0.40	0.50
10	1093	1000	89.11%	0.50	0.46	0.48

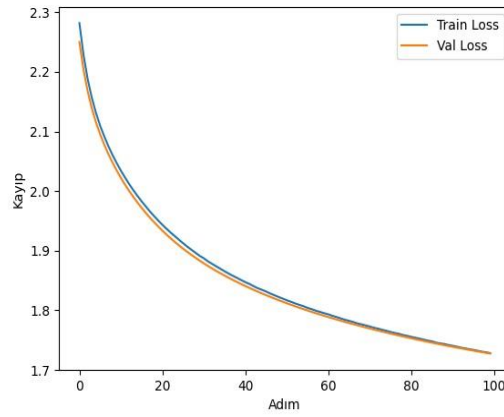
Burada yandaki tablodan confision matrisin başarımlarını ayrıntılı bir şekilde görebiliyorum. Cifar veri setinde 10 tane sınıf bulunmakta sınıfların kendi içlerinde acc genel olarak %85 üstü ve gayet iyi fakat overall acc %37,8 bu düşük bir oran bunun sebebinin overfitting olduğunu düşünüyorum. Hyperparametre ayarlarını yani batch size, epoch size, learning rate gibi çok uygun değer değerler olmadığını overall acc den anlamış oldum.

➤ Learning_rate : 1e-6

Train accuracy ve Validation accuracy Grafiği

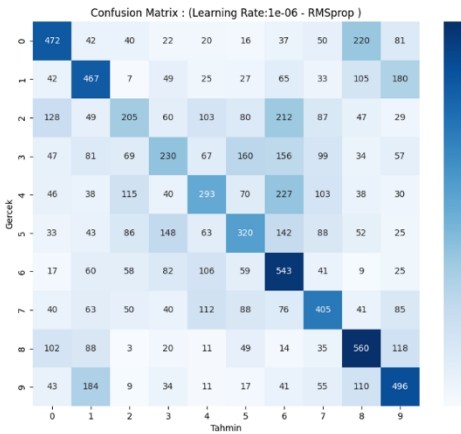


Train loss ve Validation loss Grafiği



Yukarıda ki RMSprop optimizier için grafikleri incelediğimde, 1e-6 öğrenme hızına sahip bu da 1e-2 ya göre çok daha yavaştır. Bu sayede grafikten görüyorum ki yavaş ama sağlam bir öğrenme gerçekleşmiş. Validation ve Train acc ve loss grafikleri neredeyse aynı modelim bu learning rate için düzgün ve başarılı bir genelleme yapmış aynı zamanda adım sayısı arttıkça kayıp da azalmış. Fakat başarı oranı 100 adım için maksimum %40 a erişmiş bu sonuç hala düşük.

Confision Matrix



RESULTS

TP: 3991
Overall Accuracy: 23.54%

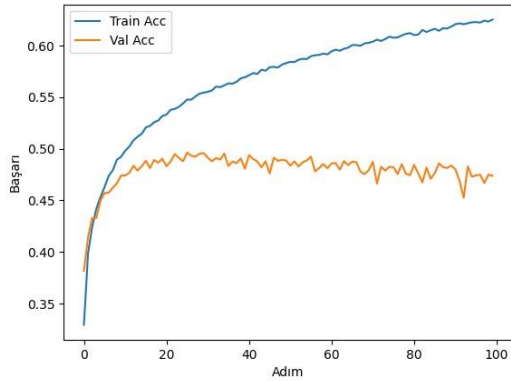
Class	n (truth) ①	n (classified) ②	Accuracy	Precision	Recall	F1 Score
1	970	7600	55.01%	0.062	0.49	0.11
2	1466	1000	90.96%	0.47	0.32	0.38
3	642	1000	92.73%	0.20	0.32	0.25
4	725	1000	92.54%	0.23	0.32	0.27
5	811	1351	90.7%	0.22	0.36	0.27
6	886	1000	92.65%	0.32	0.36	0.34
7	1513	1000	91.58%	0.54	0.36	0.43
8	796	1000	93%	0.41	0.41	0.41
9	1216	1000	93.33%	0.56	0.46	0.51
10	7776	1000	54.37%	0.50	0.044	0.11

Bu learning rate üzerinden baktığımızda yukarıdakine göre daha kötü sonuçlar verdiğini hem grafikten hem de confision matris sonuçlarından anlaşıyor. Daha yavaş öğrenme sergilediği için Burada overfitting yoktur. Veri seti büyük olması ve yavaş öğrenim sergilemesi başarıyı etkilemiştir. Matrice bakılırsa koyu renkler köşegenlerde birleşmiş bu başarı için güzel bir olay. En yüksek başarı ise 8. Sınıf predictinde gerçekleşmiş. Fakat hatalı tahminler de bulunmakta örneğin gerçekte 4 e ait olan veriler 6 gibi predict edilmiş.

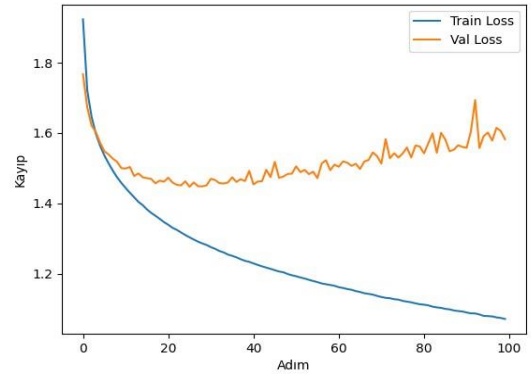
2. SGD

➤ Learning_rate= 1e-2

Train accuracy ve Validation accuracy Grafiği

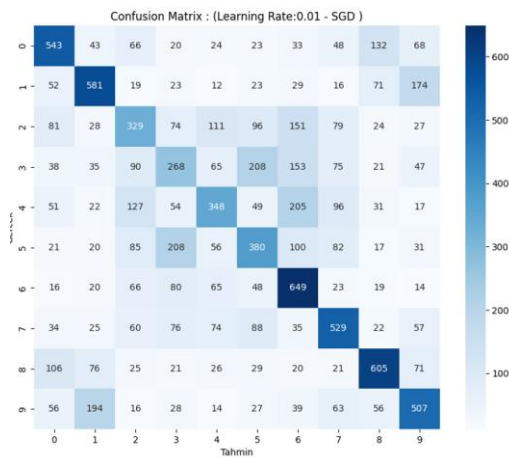


Train loss ve Validation loss Grafiği



Yukarıda ki SGD optimizier için grafikleri incelediğimde, 1e-2 öğrenme hızına sahip algoritması hızlı öğrenme sağlamıştır. Burada aşırı bir şekilde overfit olduğunu gözlemleyebiliyorum başarıım oranı %60 a kadar çıkması üsteki optimizierden daha iyi çalıştığı anlamına geliyor. Ayrıca Loss da daha düşük fakat buradaki overfit oranı çok daha büyük. Bunun sebebi yine hızlı öğrenmeden kaynaklıdır.

Confision Matrix

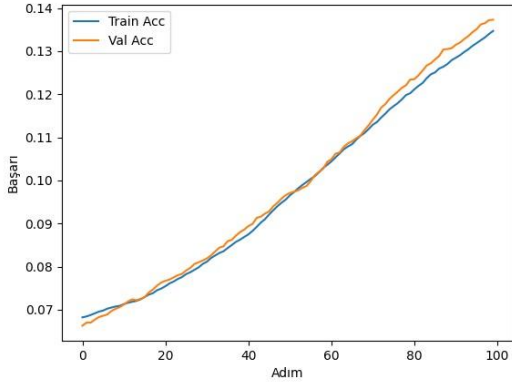


	Class	n (truth) ①	n (classified) ②	Accuracy	Precision	Recall	F1 Score
	1	998	1000	95.44%	0.54	0.54	0.54
	2	1044	1000	95.59%	0.58	0.56	0.57
	3	883	1100	93.38%	0.30	0.37	0.33
	4	852	10900	43.92%	0.025	0.31	0.046
	5	795	1000	94.51%	0.35	0.44	0.39
	6	971	1000	93.95%	0.38	0.39	0.39
	7	1414	1000	94.42%	0.65	0.46	0.54
	8	11032	1000	45.13%	0.53	0.048	0.088
	9	998	1000	96.06%	0.60	0.61	0.61
	10	1013	1000	95.01%	0.51	0.50	0.50

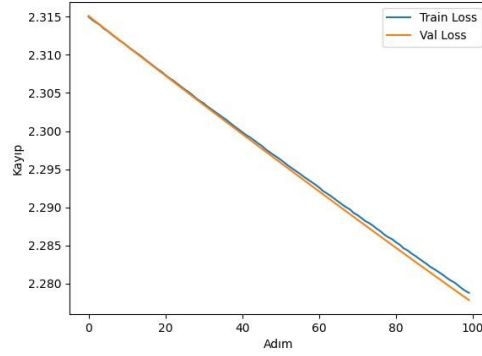
Başarıım oranı RMSprop a göre düşüktür. Daha yanlış sınıflandırma yaptığını tablodan okuyabiliyorum Yine sınıflandırmaların kendi içerisinde % 90'dan fazla olup overall acc %23,7'olması overfittingi gösteriyor. LR= 1e-2 Hızlı öğrenme sağladığı için LR=1e-6 ya göre çok daha başarılı sonuçlar veriyor fakat validation bunu desteklemiyor. Bu yüzden learning rate daha uygun değer bir değer seçilmeli. Başarıımın en koyu renk olan 6. Sınıfta olduğunu görebiliyorum.

➤ Learning_rate= 1e-6

Train accuracy ve Validation accuracy Grafiği

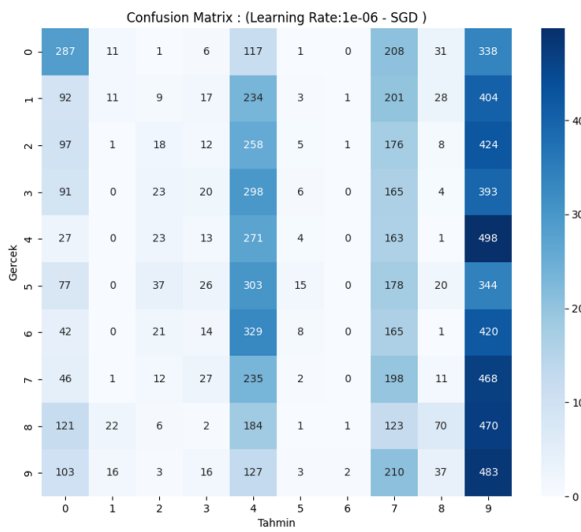


Train loss ve Validation loss Grafiği



Burada Validation ve train uyumu 1e-2 learning rate nazaran çok daha iyi ve overfit gözlenmez Fakat başarı 1e-2 de çok daha yüksek burada %15 lerce iken Yukarıda %60. Yukarıdaki overfit durumu göz önüne almama rağmen %15 çok az bir başarı ve arasında çok fark var LR seçerken bu yüzden daha 1e-2 ile 1e-6 arasında 1e-2 yakın değeri seçmek benim için daha mantıklı loss grafiğine baktığımda epoch sayısı arttıkça azalmış bu güzel bir durum fakat çok fazla bir azalış göstermemiş.

Confusion Matrix

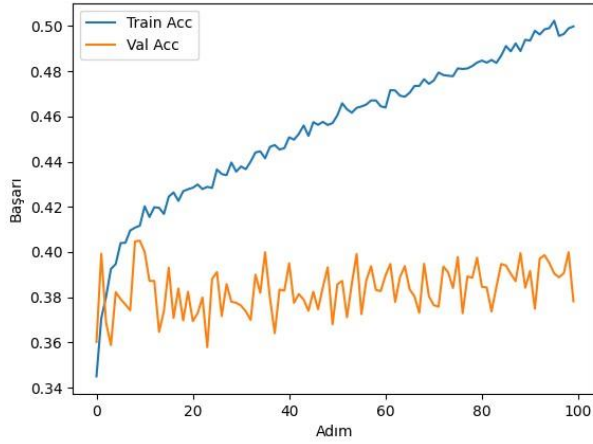


Yandaki matrixde koyu renkli taraf sağ tarafta yoğunlaşmış bu da modelimin başarısızlığını gösteriyor köşegenler koyu renk olmalıydı fakat en yüksek değer 498 bu 4. Sınıf modele verilmiş ve model 9. Sınıfmış gibi tahmin etmiş zaten başarının çok düşük olduğunu yukarıdaki grafiklerden de gözlemleyebiliyorum. Buradan baktığımda da modelim çoğu verilen veriyi 9 gibi tahmin ediyor.

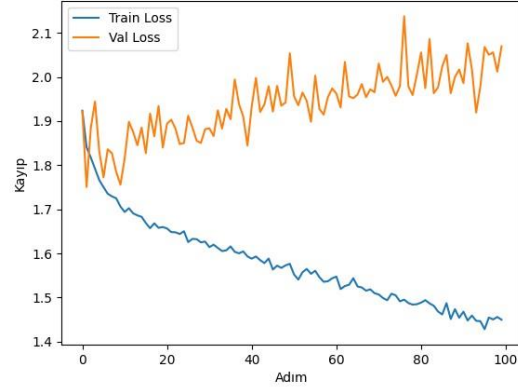
3. Adam

➤ Learning_rate= 1e-2

Train accuracy ve Validation accuracy Grafiği

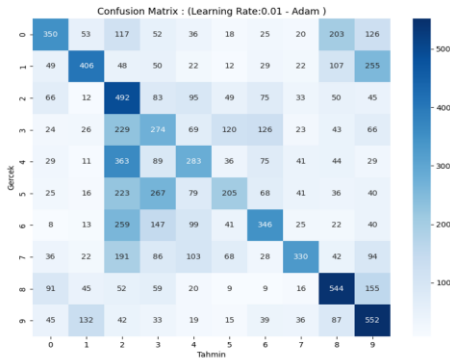


Train loss ve Validation loss Grafiği



Burada da yine tüm LR=1e-2 lerde olduğu gibi overfitting olduğu gözlemliyorum başarımlar oranı RMSpropa yakın fakar burada overfit daha fazla . Valiation ve trainingin loss grafiğinde overfit çok yüksek Stabil olmaması grafiğin yorum yapmamı zorlaştırıyor. Bu yüzden modelin hyperparametrelerinin değiştirilmesi veya early stop methodu gibi uygulamaların yapılması lazım.

Confision Matrix



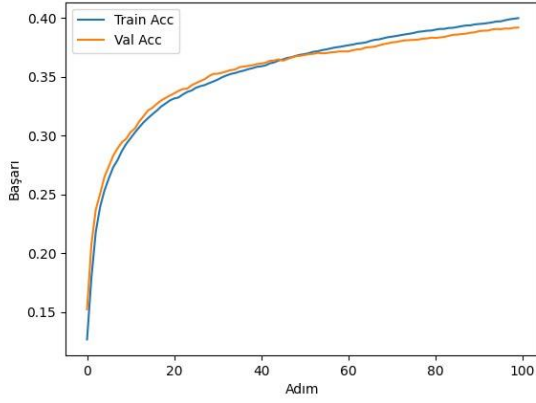
TP: 3752
Overall Accuracy: 37.63%

Class	n (truth) ①	n (classified) ②	Accuracy	Precision	Recall	F1 Score
1	723	1000	89.74%	0.35	0.48	0.41
2	736	1000	90.73%	0.41	0.55	0.47
3	2016	1000	79.62%	0.49	0.24	0.33
4	1140	1000	84.02%	0.27	0.24	0.26
5	825	1000	87.37%	0.28	0.34	0.31
6	573	1000	88.04%	0.20	0.36	0.26
7	820	1000	88.69%	0.35	0.42	0.38
8	587	1000	90.7%	0.33	0.56	0.42
9	1178	1000	89.07%	0.54	0.46	0.50
10	1372	970	86.96%	0.54	0.38	0.45

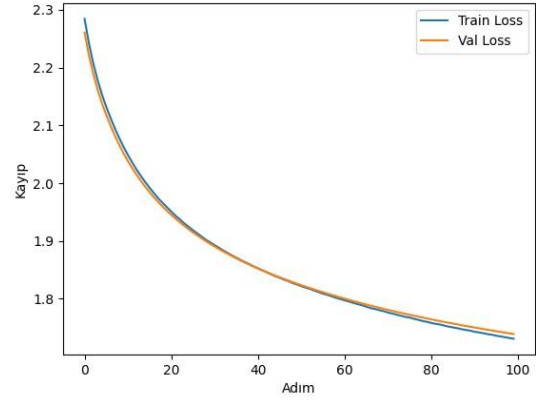
Burada değerler yine kendi içerisinde çok yüksek ve overall acc düşük overfitting olduğu aşikâr. Verilere ve grafiğe baktığımda 4.class Verileri sanki 2. Class gibi değerlendirilmiş tabloda ise 1. Den başladığı için 5. Sıra göstermekte. Başarımlar en yüksek 9. Sınıfta gözükmemekte en düşük ise 3. Sınıfta olmuş.

➤ Learning_rate= 1e-6

Train accuracy ve Validation accuracy Grafiği

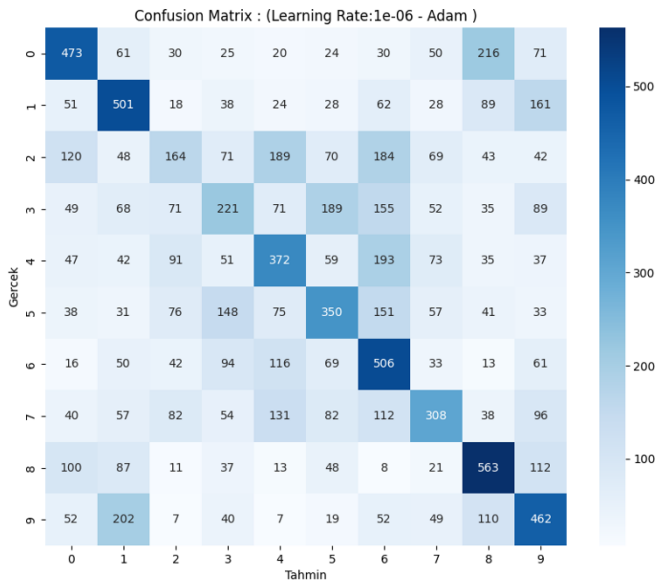


Train loss ve Validation loss Grafiği



Başarım oranı $1e-2$ göre daha az da olsa güzel v stabil bir öğrenme sağlamış overfit veya undefit olduğunu zannetmiyorum gerekli olan başarısızlık belki batch size veya step sayısı değiştirilerek düzelebilir. Loss grafiğinde ciddi bir azalma mevcut validation ve train birbirlerini destekliyor fakat başarım oranı istenildiği kadar yüksek değil.

Confision Matrix

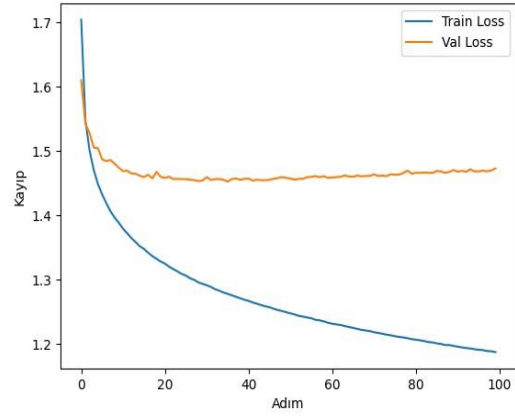
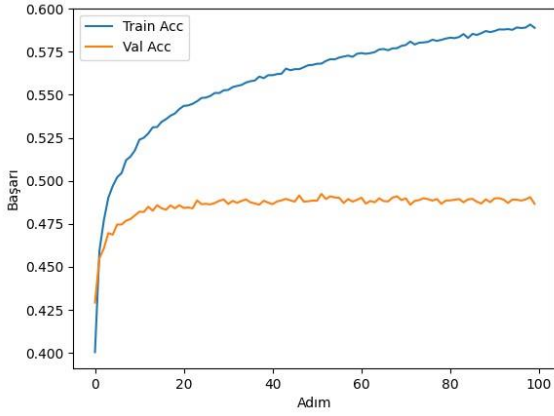


Yandaki matrixi incelediğimde koyu renkler köşegenlerde yoğunlaştığı için bu güzel bir model SGD'nin LR $1e-6$ kadar başarısız değil. En yüksek başarı rakamlara baktığımda 8. Sınıfı clasfier ederken elde edilmiş 2. Sınıfın sınıflandırması aralarında en kötüsü bazı sınıflandırmalarda köşegen hariç en koyu renk olan 216 yani gerçek sınıfı 0 iken 8. Sınıf olarak tahmin edilerek hata edilmiş

4. Adagrad

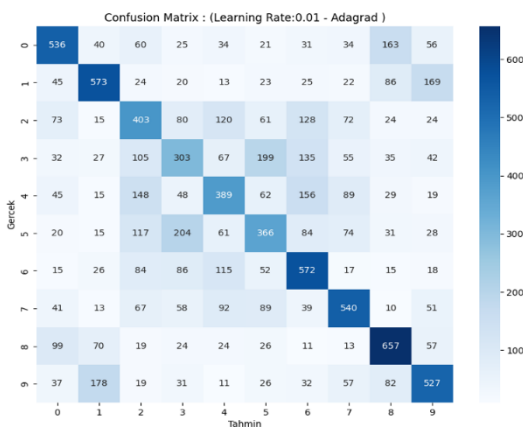
➤ Learning_rate= 1e-2

Train accuracy ve Validation accuracy Grafiği Train loss ve Validation loss Grafiği



Overfit tüm modellerde olduğu gibi burada da gözlemleniyor. Başarı oranı diğer modeller arasında yüksek overfit e baktığımda ise diğer modellere göre az olduğunu düşünüyorum tabi aynı LR değerinde gözlemlediğimde. Los değerinin en iyi başarıyı burada sanırsam çünkü en çok bu grafikte azalma var. Validation ile train yine aşırı öğrenmeden dolayı birbirlerini desteklemiyorlar

Confusion Matrix



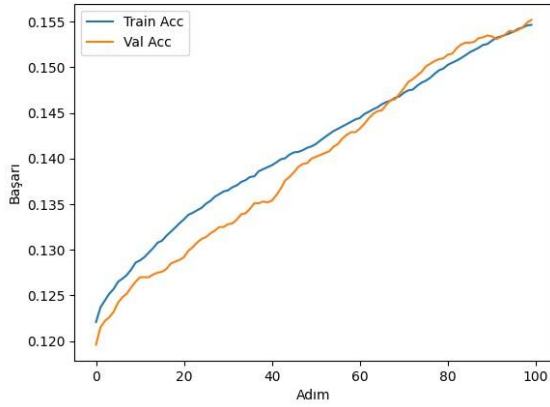
TP: 4566
Overall Accuracy: 47.07%

Class	n (truth) ①	n (classified) ②	Accuracy	Precision	Recall	F1 Score
1	643	700	91.02%	0.34	0.37	0.35
2	972	1000	91.46%	0.57	0.59	0.58
3	1046	1000	87.22%	0.40	0.39	0.39
4	879	1000	86.88%	0.30	0.34	0.32
5	926	1000	88.16%	0.39	0.42	0.40
6	925	1000	87.7%	0.37	0.40	0.38
7	1212	1000	88.96%	0.57	0.47	0.52
8	973	1000	90.79%	0.54	0.55	0.55
9	1132	1000	91.57%	0.66	0.58	0.62
10	991	1000	90.34%	0.53	0.53	0.53

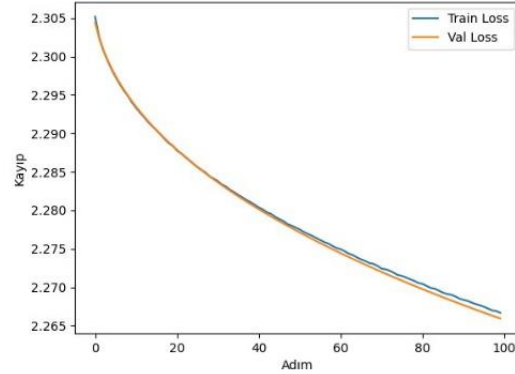
Yukarıdaki matrix ve sonuç tablosuna baktığımda görebiliyorum ki burada acc sınıflandırması kendi içerisinde çok başarılı olmasında rağmen %90 küsürlerde overfall ac %47 lerde bu yüzden overfitting olduğunu gözlemleyebiliyorum fakat over fitting değerini diğer modellere nazaran daha düşük gibi yorum yapabilirim sanırım çünkü diğer modellerin overfall acc çok daha düşük.

➤ Learning_rate= 1e-6

train accuracy ve Validation accuracy Grafiği

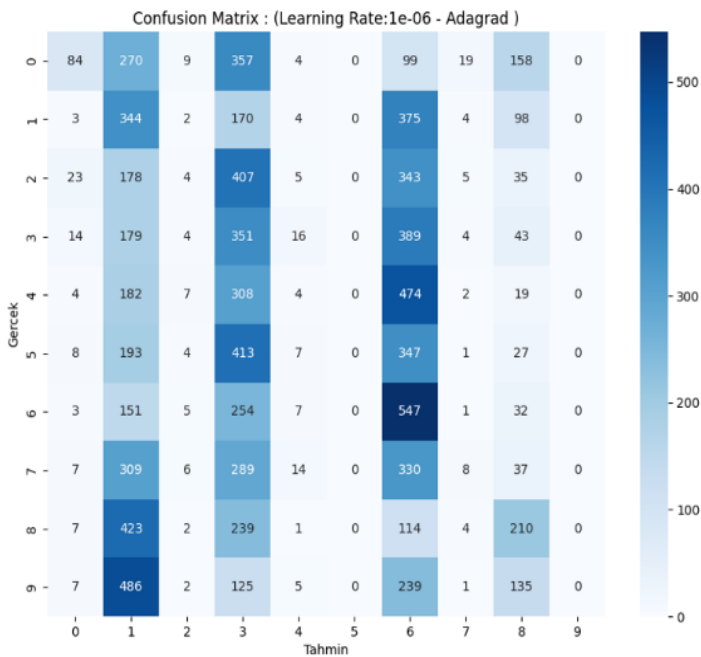


Train loss ve Validation loss Grafiği



Overfit veya underfit bu LR de gözlemlemiyorum fakat başarıım oranı çok kötü aynı şekilde loss func değeri de çok kötü. Bu model ve bu hyperparametreler çok kötü bir seçim olmuş. Diğer modellerin 1e-6 LR göre bu modelin acc grafiğinde train ve validation biraz farklı davranmış diğerlerinde başarıım oranı kötü olmasına rağmen stabil bir grafik gözükmekteydi bunda biraz dalgalanma olmuş.

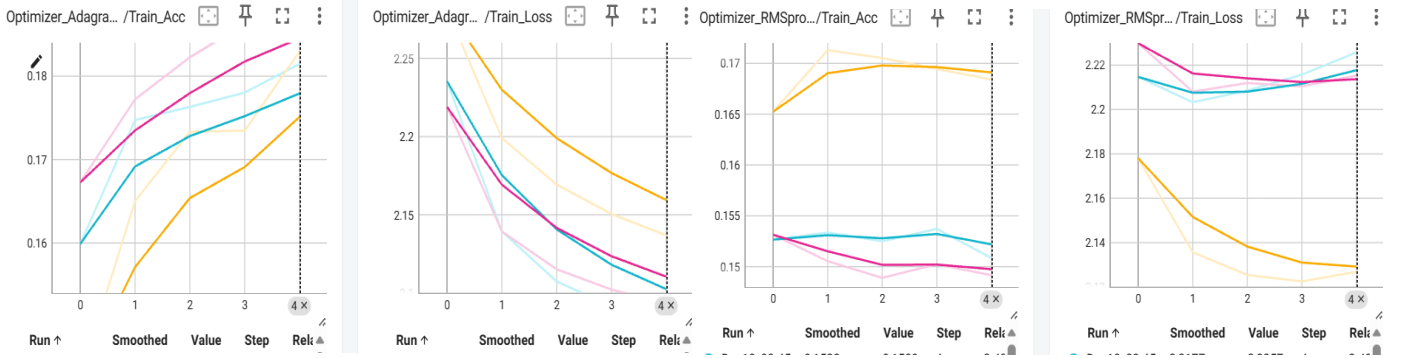
Confision Matrix



Yandaki matrise baktığımda başarısının çok daha düşük olduğunu gözlemleyebiliyorum. Sebebi ise koyu renklerin köşegenlerde yoğunlaşmaması olması , gördüğüm üzere çok fazla hatalı predict yapılmış Sadece gerçek sınıfı 6 olan veriler daha doğru classifier edildiğini görebiliyorum. Hatasından bahsetmek gerekirse, örneğin , gerçek sınıfı 9 olan veriler 1. Sınıf gibi predict edilmiş . Matrice baktığımda çoğunluk 1. 6. Ve 3. Sınıf gibi yanlış predict edilmiş. Bu kötü dağılım sgd nin lr= 1e-6 benzemekte buda çok yavaş öğrendiği için olduğunu düşünüyorum.

Sonuç ve Yorum

Yaptığım projede, aktivasyon kodu olarak `nn.LeakyReLU(negative_slope=0.01)` kullandım çünkü Relu sigmoid ve diğeri arasında 5 stepli bir tensorboard üzerinden deneme yaptım en iyi leaklyReu sonuç verdi. Optimizlerden optimizera göre değişiklik gösterdi aktivasyon fonk başarısı. Batch size olarak =32 gibi hyper parametre belirledim.



Yaptığım çalışmanın sonuçlarına göre, tüm optimizeler için learning rate değerini $1e-2$ olarak belirlemem, test başarısını artırmış gibi görünüyor. Ancak, bu yüksek başarıya rağmen, tüm optimizeler için $LR=1e-2$ uygulamasında test ve validasyon accuracy grafiklerinde belirgin bir overfitting gözlemlenmiştir. Overfitting'in nedeni, çok büyük bir veri seti için çok hızlı bir öğrenme oranı seçilmesinden kaynaklanmaktadır. Bu durumda, model eğitim verilerine aşırı derecede uyum sağlamış ve genel desenleri kaçırmış olabilir.

Diğer yandan, $LR=1e-6$ uygulamasında ise daha düzgün ve istikrarlı bir grafik elde edilmiştir. Bu durum, daha küçük bir öğrenme oranının genel desenlere daha iyi uyum sağladığını göstermektedir. Overfitting gözlemlenmemiş olsa da, bu durumda elde edilen test başarısı daha düşüktür. Çünkü çok küçük bir öğrenme oranı, modelin eğitim verilerinden yeterince öğrenememesine neden olabilir.

Optimizasyon algoritmalarını sıraladığımda, RMSprop ve Adam'ın performansının diğerlerine göre daha iyi olduğunu gözlemledim. Adagrad, $1e-2$ öğrenme oranında eğitim başarısında diğerlerine göre daha yüksek bir başarı elde etti, ancak $1e-6$ öğrenme oranında performansı çok düştü. Bu nedenle, optimizasyon algoritmalarını seçerken hem öğrenme oranını hem de diğer hiperparametreleri dikkatlice ayarlamak önemlidir.

Sonuç olarak, elde ettiğim veriler yüksek başarıyı temsil etmez, modelin genelleme yeteneğini artırmak ve overfitting'i engellemek için öğrenme oranı gibi hiperparametreleri daha dikkatli seçmeli ve uygun düzenleme tekniklerini uygulamalıyım.