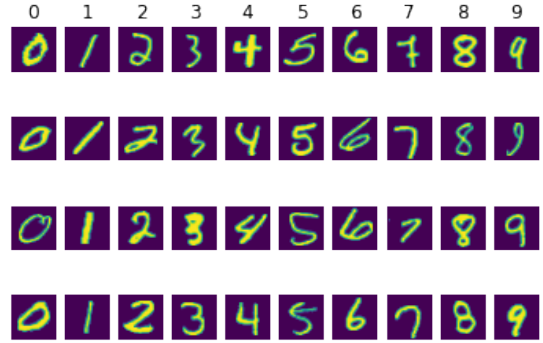


MNIST Rakam Tanıma ve Makine Öğrenimi Algoritmalarının Karşılaştırılması

Seyyid Hikmet Çelik
TOBB Ekonomi ve Teknoloji Üniversitesi, Türkiye
seyyidhikmetcelik@etu.edu.tr

Özet—Bu çalışmada verilen el yazısı rakam (MNIST) veri seti üzerinde makine öğrenmesi algoritmaları ile rakamların öğrenilmesi ve öğrenme tamamlandıktan sonra resimlere bağlı rakam değerlerini çıktı olarak verebilme amaçlanmıştır. Öğrenilen rakamların ne kadar doğrulukla tespit edilip edilemediği farklı yöntemler bazında karşılaştırılarak performans ölçümü gerçekleştirilmiştir. Veri setine <https://www.kaggle.com/competitions/digit-recognizer> adresinden, çalışmada yazılmış kodlara <https://github.com/celuc/machine-learning-digit-recognizer> adresinden, videosuna ise <https://youtu.be/QShwrZaymE4> adresinden ulaşılabilir.

Anahtar Kelimeler—Machine Learning, MNIST, Digit Recognizer



Şekil 1: Veri Seti

I. GİRİŞ

Projede, <https://www.kaggle.com/competitions/digit-recognizer> bağlantısında görülen MNIST el yazısı rakam veri seti kullanılacaktır. Her resim 28x28, toplam 784 pikselden oluşmaktadır. Veri setinde bu resimlerin aşağıdaki gibi piksel karşılıkları vardır.

```
000 001 002 003 ... 026 027 028 029 030
031 ... 054 055 056 057 058 059 ... 082
083 | | | ... | 728 729 730 731 ... 754
755 756 757 758 759 ... 782 783
```

En son model eğitildikten sonra çıktılar aşağıdaki gibi resim id'si ve denk tahmin edilen rakam etiketi şeklinde verilmelidir.

```
ImageId,Label 1,3 2,7 3,8 (27997 more
lines)
```

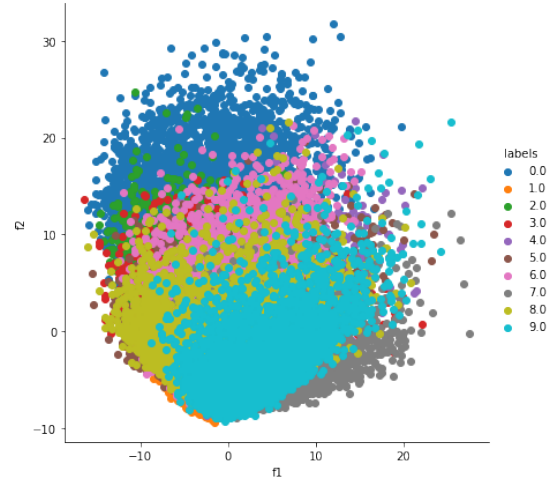
II. LİTERATÜR TARAMASI

Daha önceden yapılan çalışmalarda [1][2][3] daha çok derin öğrenme yöntemlerinin kullanıldığı görülmüştür. Bunun sebebi ise derin öğrenme yöntemlerinin diğer yöntemlerden daha iyi sonuç vermesi ve sürekli bu yöntemin üzerine konularak ya da veri setini iyileştirerek gidilmek istenmesindendir. Ayrıca melez modeller [3][4][5][6][7] kullananlar da vardır.

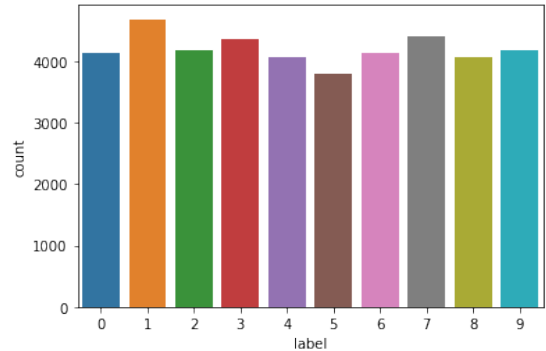
III. VERİ SETİ, ÖZELLİKLERİ VE ÖZNİTELİKLER

Veri setinin bir kısmı Şekil 1'de görülebilir. 10 rakam olduğu için öznitelik sayımız azdır.

Veri dağılımı Şekil 2, histogramı ise Şekil 3'te görülebilir.



Şekil 2: Veri Dağılımı



Şekil 3: Veri Histogramı

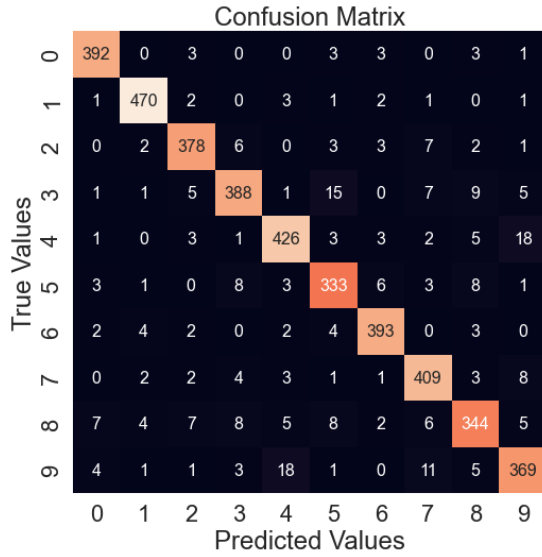
Görüleceği üzere verilerimiz güzel dağılmış ve temiz duruyor. Veri seti temiz olduğu için de çok fazla üzerinde işlem yapmaya gerek kalmıyor, sadece veriler karıştırılıp direkt olarak modellere verilebilir.

IV. KULLANILAN MODELLER VE TEST SONUÇLARI

Bölüm II'de bahsedildiği gibi el yazısı rakam veri seti için genel olarak derin öğrenme algoritmaları kullanılmaktadır. Bu yüzden KNN (K-Nearest Neighbor), Decision Tree, Random Forest, SVM (Support Vector Machine) modellerinin yanı sıra MLP (Multi-Layer Perceptron) ve CNN (Convolutional Neural Network) de kullanılmıştır. Bu modellerin kullanılmasının bir diğer sebebi ise popüler olmalarıdır. Modeller python üzerinde sklearn, tensorflow ve keras kütüphaneleri kullanılarak oluşturulmuş ve test edilmiştir.

A. KNN

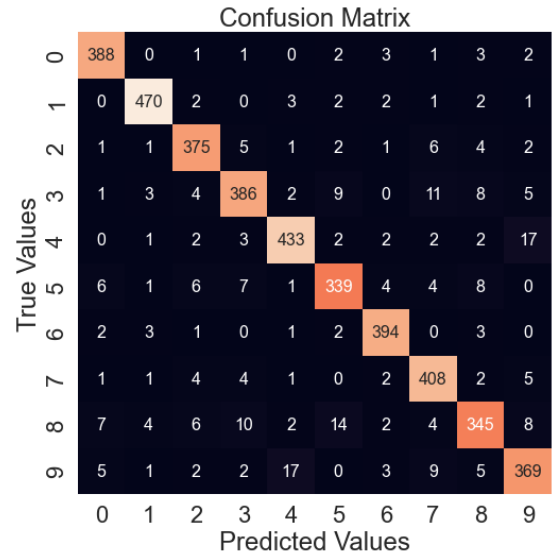
3 en yakın komşu modeli kullanılmıştır. Test accuracy değeri **%92.90** çıkmıştır. Confusion matrisi Şekil 4'te görülebilir.



Şekil 4: KNN Confusion Matrisi

B. Decision Tree

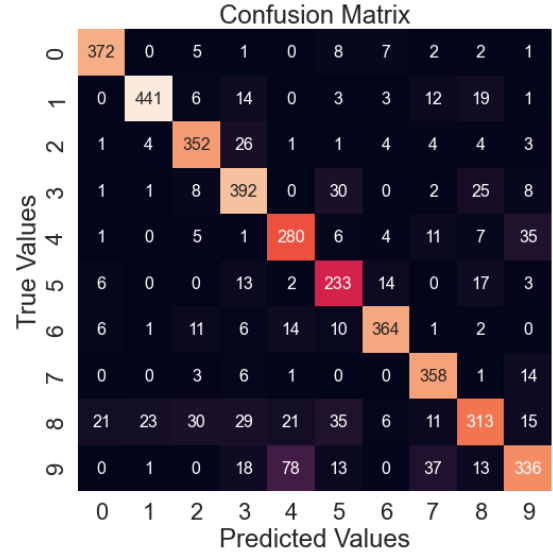
Karar ağacı modelinin test accuracy değeri **%92.73** çıkmıştır. Confusion matrisi Şekil 5'te görülebilir.



Şekil 5: Karar Ağacı Confusion Matrisi

C. Random Forest

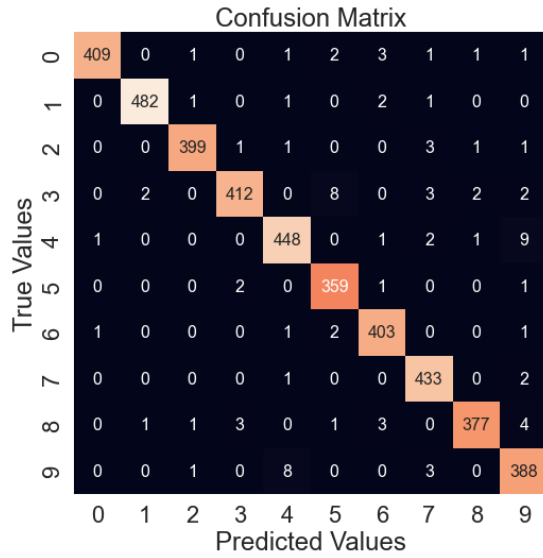
Random forest modelinin test accuracy değeri **%81.92** çıkmıştır. Confusion matrisi Şekil 6'da görülebilir.



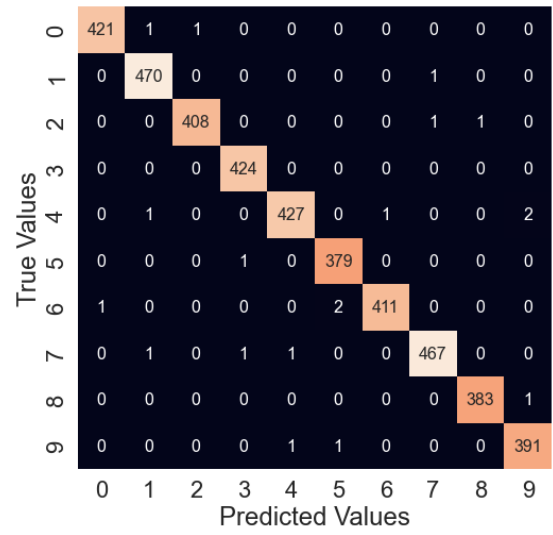
Şekil 6: Random Forest Confusion Matrisi

D. SVM

SVM modelinin test accuracy değeri **%97.85** çıkmıştır. Confusion matrisi Şekil 7'de görülebilir.



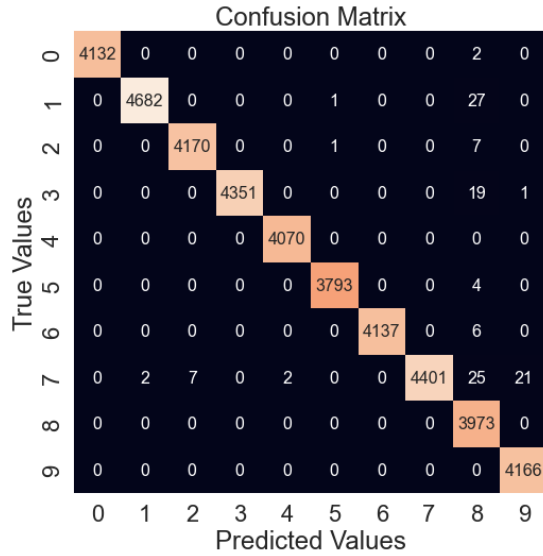
Şekil 7: SVM Confusion Matrisi



Şekil 9: CNN Confusion Matrisi

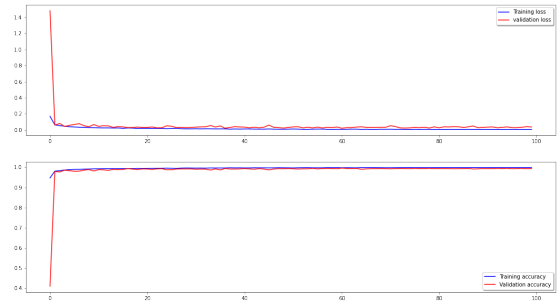
E. MLP

MLP modelinin test accuracy değeri **%99.70** çıkmıştır. Confusion matrisi Şekil 8'de görülebilir.



Şekil 8: MLP Confusion Matrisi

Şekil 10'da CNN için training-validation loss grafiği görülebilir.



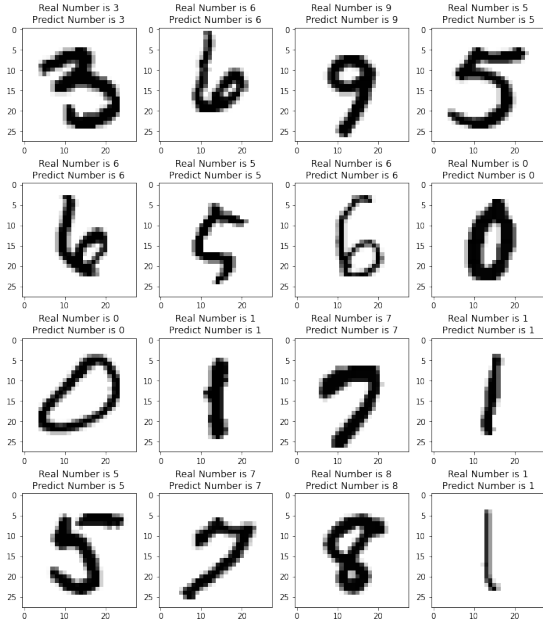
Şekil 10: CNN Training-Validation Kayıp Grafiği

Şekil 10'a bakıldığında eğitimin validationa çok yaklaştığı görülüyor ki bu da modelin çok iyi çalıştığını gösteriyor.

F. CNN

CNN modelinin test accuracy değeri **%99.54** çıkmıştır. Confusion matrisi Şekil 9'da görülebilir.

Şekil 11'de CNN tahmin sonuçları bir kısım el yazısı rakamlar için görülebilir.



Şekil 11: CNN El Yazısı Rakam Tahminleri

Kullanılan modellere ait tüm test accuracy Tablo 1’de en yüksekten düşüğe göre görülebilir.

TABLO I: Modellere Göre Accuracy Değerleri

Model	Accuracy
MLP	%99.70
CNN	%99.54
SVM	%97.85
KNN	%92.90
Decision Tree	%92.73
Random Forest	%81.92

En yüksek test accuracy değerlerini MLP ve CNN modellerinde görüyoruz ki bu da beklediğimiz bir şeydi çünkü ikisi de derin öğrenme modeli. Diğer çok bilinen modeller o kadar yaklaşmasa yine iyi sonuçlar veriyor fakat görülüyor ki Random Forest diğerlerinden çok çok daha düşük değer veriyor, bu da Random Forest modelinin veri setimize uygun bir model olmadığını gösteriyor.

V. SONUÇLAR

Genel olarak rakam tanıma veri setinin çok yüksek accuracy değerleri verdiğini ve derin öğrenme modellerinin bu veri seti için oldukça uygun olduğunu gördük. Gelecek çalışmalarda farklı melez modeller kullanılabilir ya da veri seti üzerinde oynamalar, kernel değiştirme işlemleri yapılabilir.

REFERANSLAR

- [1] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger *et al.*, “Comparison of learning algorithms for handwritten digit recognition,” in *International conference on artificial neural networks*, vol. 60, no. 1. Perth, Australia, 1995, pp. 53–60.

- [2] E. Kussul and T. Baidyk, “Improved method of handwritten digit recognition tested on mnist database,” *Image and Vision Computing*, vol. 22, no. 12, pp. 971–981, 2004.
- [3] S. An, M. Lee, S. Park, H. Yang, and J. So, “An ensemble of simple convolutional neural network models for mnist digit recognition,” *arXiv preprint arXiv:2008.10400*, 2020.
- [4] D. Hirata and N. Takahashi, “Ensemble learning in cnn augmented with fully connected subnetworks,” *arXiv preprint arXiv:2003.08562*, 2020.
- [5] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, “Deep, big, simple neural nets for handwritten digit recognition,” *Neural computation*, vol. 22, no. 12, pp. 3207–3220, 2010.
- [6] V. Mazzia, F. Salvetti, and M. Chiaberge, “Efficient-capsnet: Capsule network with self-attention routing,” *Scientific reports*, vol. 11, no. 1, pp. 1–13, 2021.
- [7] P. Jeevan, K. Viswanathan, and A. Sethi, “Wavemix-lite: A resource-efficient neural network for image analysis,” *arXiv preprint arXiv:2205.14375*, 2022.

VI. APPENDIX

Kodlara ulaşmak için GitHub sayfasını ziyaret edin.
<https://github.com/celuk/machine-learning-digit-recognizer>