

Yapay Sinir Ağı

Tolga Sağlam
Bilgisayar Mühendisliği
18011064
tolga.saglam@std.yildiz.edu.tr

Ömer Talha Baysan
Bilgisayar Mühendisliği
18011103
talha.baysan@std.yildiz.edu.tr

I. GİRİŞ

A. Amacımız

Projemizdeki amacımız bir bilgisayar çıktığında o bilgisayara puan vererek müşterilere satın alma konusunda yardımcı olmak.

B. Yöntemimiz

Epey adlı her türlü karşılaştırmaların yapılabildiği bir sitedeki puanlandırılmış bilgisayarları, python kullanarak Webscraping ile çektik. Sonrasında Yapay Sinir Ağları kullanarak mümkün olduğunca doğru puanlama yapmak.

II. YAPAY SINIR AĞLARI(ANN)

Yapay sinir ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir.

Yapay sinir ağları insan beyni örnek alınarak, öğrenme sürecinin matematiksel olarak modellenmesi sonucu ortaya çıkmıştır. Beyindeki biyolojik sinir ağlarının yapısını, öğrenme, hatırlama ve genelleme kabiliyetlerini taklit eder. Yapay sinir ağlarında öğrenme işlemi örnekler kullanılarak gerçekleştirilir. Öğrenme esnasında giriş çıkış bilgileri verilerek, kurallar koyulur.

Sinir ağları, bilgisayarların sınırlı insan desteğiyle akıllı kararlar almasına yardımcı olur. Bunun nedeni, doğrusal olmayan ve karmaşık girdi ve çıktı verileri arasındaki ilişkileri öğrenip modelleyebilmeleridir. Örneğin, sinir ağları aşağıdaki görevleri gerçekleştirebilir.

A. Yapay Sinir Ağlarının Avantajları

-Yapay Sinir Ağları bir çok hücreden meydana gelir ve bu hücreler eş zamanlı çalışarak karmaşık işleri gerçekleştirir.

-Öğrenme kabiliyeti vardır ve farklı öğrenme algoritmalarıyla öğrenebilirler.

-Görülmemiş çıktılar için sonuç (bilgi) üretebilirler. Gözetimsiz öğrenim söz konusudur.

-Örüntü tanıma ve sınıflandırma yapabilirler. Eksik örüntüleri tamamlayabilirler.

-Hata toleransına sahiptirler. Eksik veya belirsiz bilgiyle çalışabilirler. Hatalı durumlarda dereceli bozulma (graceful degradation) gösterirler.

-Paralel çalışabilmekte ve gerçek zamanlı bilgiyi işleyebilmektedirler.

B. Yapay Sinir Ağlarının Kullanım Alanları

-Hesaplamalı Finans(Computational finance): Kredi skorlaması(credit scoring), Algoritmik Ticaret(algorithmic trading)

-Görüntü işleme ve bilgisayarla görü(image processing and computer vision): Yüz tanıma(face recognition), hareket tanıma(motion detection), nesne tanıma(object detection)

-Hesaplamalı biyoloji(Computational biology): Tümör bulma(tumor detection), İlaç keşfi(drug discovery), DNA dizilimi(DNA sequencing)

-Enerji üretimi (Energy production): Fiyat ve yük tahmini (price and load forecasting)

-Otomotiv, havacılık ve üretim (Automotive, aerospace and manufacturing): Öngörücü bakım (predictive maintenance)

-Doğal dil işleme(Natural language processing): Sesli Asistan(voice assistant), Duygu analizi(emotion analysis)

C. Yapay Sinir Ağı Yapısı

Perceptron: Yapay sinir ağının en küçük parçası olarak bilinen perceptron, aşağıdaki gibi lineer bir fonksiyonla ifade edilmektedir ve ilk defa 1957 yılında Frank Rosenblatt tarafından tanımlanmıştır.

$$y = W \times x + b$$

y: x'in değerine bağlı olduğundan bağımlı değişkendir. Girdiye ait skoru verir.

x: bağımsız değişken, girdi.

W: ağırlık parametresi

b: bias değeri

YSA ya da Derin Öğrenme modelinde yapılan temel işlem; modelin en iyi skoru vereceği W ve b parametrelerinin hesabını yapmaktır.

Giriş değerimiz ve ağırlığımız(W) çarpıldıktan sonra(W.x) sinir hücresine iletilir ve sinir hücresinde bu çarpma işlemi yapılıyor ve tüm dentritlerden gelen ağırlık ile giriş çarpımları toplanır. Yani ağırlıklı toplama işlemi yapılır. Ardından bir bias(b) ile toplandıktan sonra aktivasyon fonksiyonu ardından çıkışa aktarılır. Bu çıkış nihai çıkış olabileceği gibi bir başka hücrenin girişi olabilir. Matematiksel olarak ağırlıklar

ile girişler çarpılır artı bir bias eklenir. Böylelikle basit bir matematiksel model elde edilir.

Yapay Sinir Ağlarında yapılan temel işlem; modelin en iyi skoru vereceği w(ağırlık parametresi) ve b(bias değeri) parametrelerinin hesabını yapmaktır.

Her bir sinir hücresi aynı şekilde hesaplanır ve bunlar birbirine seri ya da paralel şekilde bağlanır.

III. GELİŞTİRME SÜRECİ

A. Webscrapping

Projeye internetteki verileri Python ile çekmeye çalışarak başladık. Bu adımda farklı farklı videolar izledik ve bunları sentezleyerek verileri çekmeyi başardık. Bu konuda en çok zorlandığımız şey ise linklerdeki html'lerde özel isimler yer almasıydı. Bu bir linkteki veriyi çekeceğimiz kodun diğerinde çalışamayacağı anlamına geliyordu. Videolardaki işlemler ise işe yaramıyordu fakat buna farklı bir yöntem bulduk. "find_next_sibling()" metodu ile istediğimiz bilgiye özel isim kullanmadan ulaşabiliyorduk. Başka zorlandığımız bir kısım ise her bilgisayarın istediğimiz bilgisi bulunmuyordu. Örneğin bazı bilgisayarların ssd'si yokken bazılarının ekran kartı yoktu. Bu hatayı da giderdikten sonra pek bir sorunla karşılaşmadık.

B. ANN uygulanması

İkinci adım olarak ANN'i uygulamaya çalıştık. Ama öncesinde verileri ANN'e uygun hale getirmek için ön işleme yaptık. Kategorik verilerin Onehotencoding ile dönüştürülmesi öneriliyor fakat bizim verilerimiz çok fazla farklılıklar içerdiği için sınırı aştı. Bu nedenle Labelencoding yöntemini kullandık.

IV. ÇALIŞTIRMA ÖRNEKLERİ

Webscrapping yaparak aşağıdaki dataframe'i elde ediyoruz ve bunu excel'e atıyoruz.

	İsim	Marka	İşlemci	İşlemci Çekirdeği	Ekran Çözünürlüğü	Ram	Bellek Türü	SSD	GPU Modeli	GPU Bellek	GPU Hz	Puan
0	Lenovo ThinkPad P15 G2 20YQS2T002A0 Notebook	Lenovo	Intel Core i9-11950H	8	1920 x 1080 Piksel	128	DDR4	1536	NVIDIA RTX A4000	8	8	73
1	Asus ROG Zephyrus G14 GA402ZY-N2023W Notebook	Asus	AMD Ryzen 9 7940HS	8	2560 x 1600 Piksel	32	DDR5	1024	NVIDIA GeForce RTX 4060 125W	16	16	73
2	MSI Vector GP77 13V-G-015TR Notebook	MSI	Intel Core i9-13900H	14	2560 x 1440 Piksel	32	DDR5	2048	NVIDIA GeForce RTX 4070 140W	8	8	70
3	Lenovo ThinkPad P17 G2 20YU001XTX047 Notebook	Lenovo	Intel Core i7-11800H	8	1920 x 1080 Piksel	128	DDR4	512	NVIDIA RTX A2000	4	4	66
4	HP OMEN 17-cm2001nt (T7959EA) Notebook	HP	Intel Core i7-12700HX	16	2560 x 1440 Piksel	16	DDR5	1024	NVIDIA GeForce RTX 4070	8	8	66
...
212	HP ProBook 450 G8 (Z1W8T0EA) Notebook	HP	Intel Core i3-1115G4	2	1366 x 768 Piksel	4	DDR4	256	GPU Yok	0	0	22
213	Lenovo V14 G2 82KAD01TTX061 Notebook	Lenovo	Intel Core i3-1115G4	2	1920 x 1080 Piksel	8	DDR4	600	GPU Yok	0	0	21
214	HP Z50 G7 (T75R9EA) Notebook	HP	Intel Core i5-1035G7	4	1366 x 768 Piksel	4	DDR4	0	NVIDIA GeForce MX110	2	2	21
215	Dell Latitude 3301 N015L330113EMEA_W Ultrabook	Dell	Intel Core i3-8145U	2	1920 x 1080 Piksel	4	LPDDR3	256	GPU Yok	0	0	21
216	Casper Nirvana X400 1021-SU00T-S Ultrabook	Casper	Intel Core i5-10210U	4	1366 x 768 Piksel	8	DDR4	256	GPU Yok	0	0	21

Sonrasında ise ANN'i çalıştırıyoruz. 10'ar nörona sahip 2 farklı gizli katman oluşturduk

ve ilk olarak 50 epoch ile çalıştırdık.

```
5/5 [=====] - 0s 11ms/step - loss: 0.0249 - val_loss: 0.0269
Epoch 45/50
5/5 [=====] - 0s 10ms/step - loss: 0.0245 - val_loss: 0.0266
Epoch 46/50
5/5 [=====] - 0s 14ms/step - loss: 0.0241 - val_loss: 0.0265
Epoch 47/50
5/5 [=====] - 0s 15ms/step - loss: 0.0238 - val_loss: 0.0264
Epoch 48/50
5/5 [=====] - 0s 10ms/step - loss: 0.0234 - val_loss: 0.0262
Epoch 49/50
5/5 [=====] - 0s 15ms/step - loss: 0.0231 - val_loss: 0.0262
Epoch 50/50
5/5 [=====] - 0s 14ms/step - loss: 0.0228 - val_loss: 0.0260
```

-loss 0.0228
-validation loss 0.026
-R2 score 0.53
-Mean Absolute Error 0.13
Sonuçlarını aldık.

Sonrasında aynı ANN yapısında 100 epoch ile çalıştırdık.

```
5/5 [=====] - 0s 18ms/step - loss: 0.0126 - val_loss: 0.0147
Epoch 95/100
5/5 [=====] - 0s 12ms/step - loss: 0.0125 - val_loss: 0.0143
Epoch 96/100
5/5 [=====] - 0s 12ms/step - loss: 0.0124 - val_loss: 0.0140
Epoch 97/100
5/5 [=====] - 0s 10ms/step - loss: 0.0123 - val_loss: 0.0140
Epoch 98/100
5/5 [=====] - 0s 10ms/step - loss: 0.0122 - val_loss: 0.0141
Epoch 99/100
5/5 [=====] - 0s 12ms/step - loss: 0.0121 - val_loss: 0.0143
Epoch 100/100
5/5 [=====] - 0s 13ms/step - loss: 0.0121 - val_loss: 0.0141
```

-loss 0.0121
-validation loss 0.0141
-R2 score 0.72
-Mean Absolute Error 0.12
Sonuçlarını aldık. Bu şekilde çok daha iyi değerlere sahip

olmuş olduk.

Sonrasında ise ilk gizli katman 7, ikinci gizli katman 4 nöron içerecek şekilde modeli oluşturup 100 epoch ile çalıştırdık.

```
Epoch 96/100
5/5 [=====] - 0s 10ms/step - loss: 0.0148 - val_loss: 0.0211
Epoch 97/100
5/5 [=====] - 0s 10ms/step - loss: 0.0148 - val_loss: 0.0210
Epoch 98/100
5/5 [=====] - 0s 14ms/step - loss: 0.0148 - val_loss: 0.0211
Epoch 99/100
5/5 [=====] - 0s 10ms/step - loss: 0.0147 - val_loss: 0.0210
Epoch 100/100
5/5 [=====] - 0s 12ms/step - loss: 0.0147 - val_loss: 0.0210
```

-loss 0.0147
-validation loss 0.021
-R2 score 0.57
-Mean Absolute Error 0.12
Sonuçlarını aldık. Bu denemede ise değerler daha kötü

çıktı. Sonrasında ise bir katman daha ekleyerek çalıştırdık.

Bu katmanı da 2 nörondan oluşacak şekilde ayarladık.

```
Epoch 96/100
5/5 [=====] - 0s 10ms/step - loss: 0.0763 - val_loss: 0.0668
Epoch 97/100
5/5 [=====] - 0s 10ms/step - loss: 0.0758 - val_loss: 0.0664
Epoch 98/100
5/5 [=====] - 0s 12ms/step - loss: 0.0752 - val_loss: 0.0660
Epoch 99/100
5/5 [=====] - 0s 10ms/step - loss: 0.0748 - val_loss: 0.0656
Epoch 100/100
5/5 [=====] - 0s 10ms/step - loss: 0.0744 - val_loss: 0.0653
```

-loss 0.0744
-validation loss 0.0653
-R2 score -0.15
-Mean Absolute Error 0.21

Sonuçlarını aldık. Bu aldığımız en kötü değerlerdi. Gizli katman sayısını artırmak sonucu kötüleştirdiği için gizli katman sayısını 1 yaptık. 10 nöron ile daha iyi sonuçlar aldığımız için nöron sayısını 10 yaptık.

```
Epoch 96/100
5/5 [=====] - 0s 12ms/step - loss: 0.0187 - val_loss: 0.0216
Epoch 97/100
5/5 [=====] - 0s 12ms/step - loss: 0.0187 - val_loss: 0.0216
Epoch 98/100
5/5 [=====] - 0s 16ms/step - loss: 0.0186 - val_loss: 0.0215
Epoch 99/100
5/5 [=====] - 0s 18ms/step - loss: 0.0185 - val_loss: 0.0214
Epoch 100/100
5/5 [=====] - 0s 13ms/step - loss: 0.0184 - val_loss: 0.0214
```

-loss 0.0184
-validation loss 0.0214
-R2 score 0.71
-Mean Absolute Error 0.1

Sonuçlarını aldık ve 2 gizli katmanlı 10'ar nöronlu ve 100 epoch ile çalıştırdığımız modelin değerlerinden düşük sonuçlar elde ettik. Bu şekilde en uygun modeli bulmuş olduk.

REFERENCES

-<https://www.veribilimiokulu.com/yapay-sinir-agiartificial-neural-network-nedir/>
-<https://aws.amazon.com/tr/what-is/neural-network/>
-<https://ayyucekizrak.medium.com/>
-<https://www.youtube.com/watch?v=o83WYkwNslUpp=ygUMd2ViIHNjcmFwaW5n>
-<https://www.youtube.com/watch?v=AftWcUOb6Z0>
-<https://www.kaggle.com/code/samnijin/house-price-prediction-ann>
-<https://www.udemy.com/course/python-ile-makine-ogrenimi/learn/lecture/28169474?start=1>