



Makine Öğrenmesi Tabanlı Kayıp Satışların İncelenme Sistemi

BİL 496

Son Sunum

Burak AKTEN

Proje Danışmanı: Dr. Öğr. Üyesi Yakup GENÇ

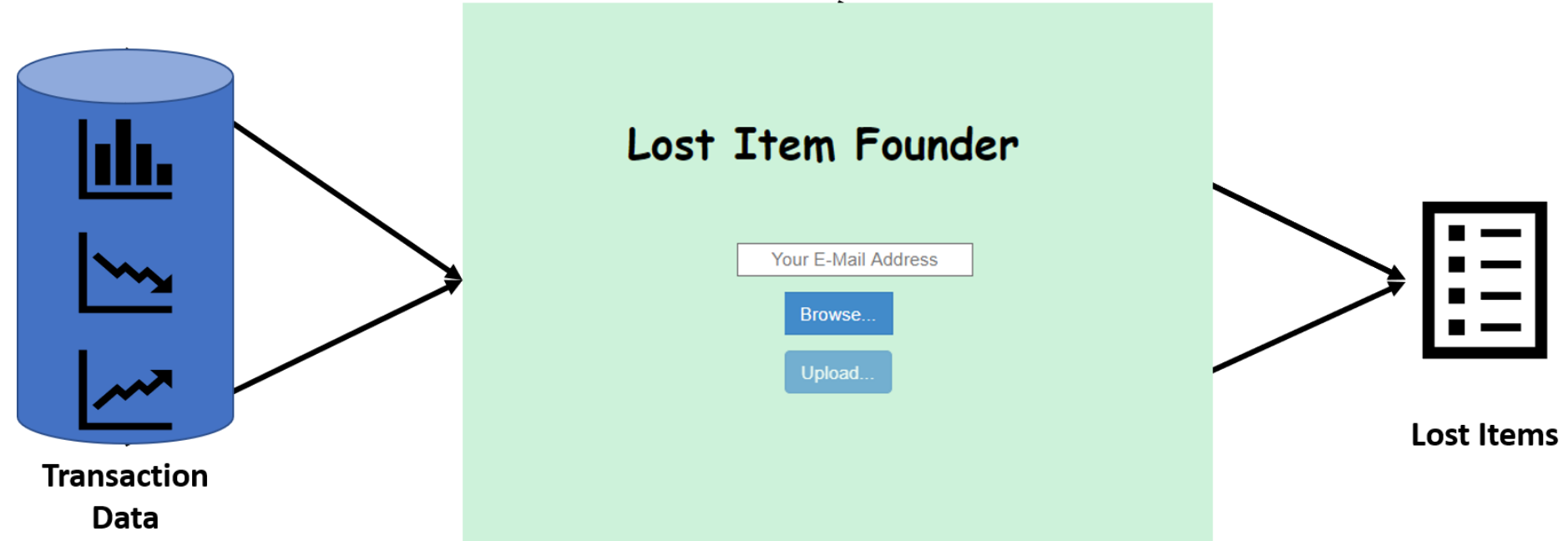
Mayıs 2019



- Projenin Şeması ve Tanımı
- Proje Tasarım Planı
- Yapılanlar
- Sonuçlar
- Başarı Kriterleri Değerlendirilmesi
- Kaynaklar



Proje Şeması ve Tanımı

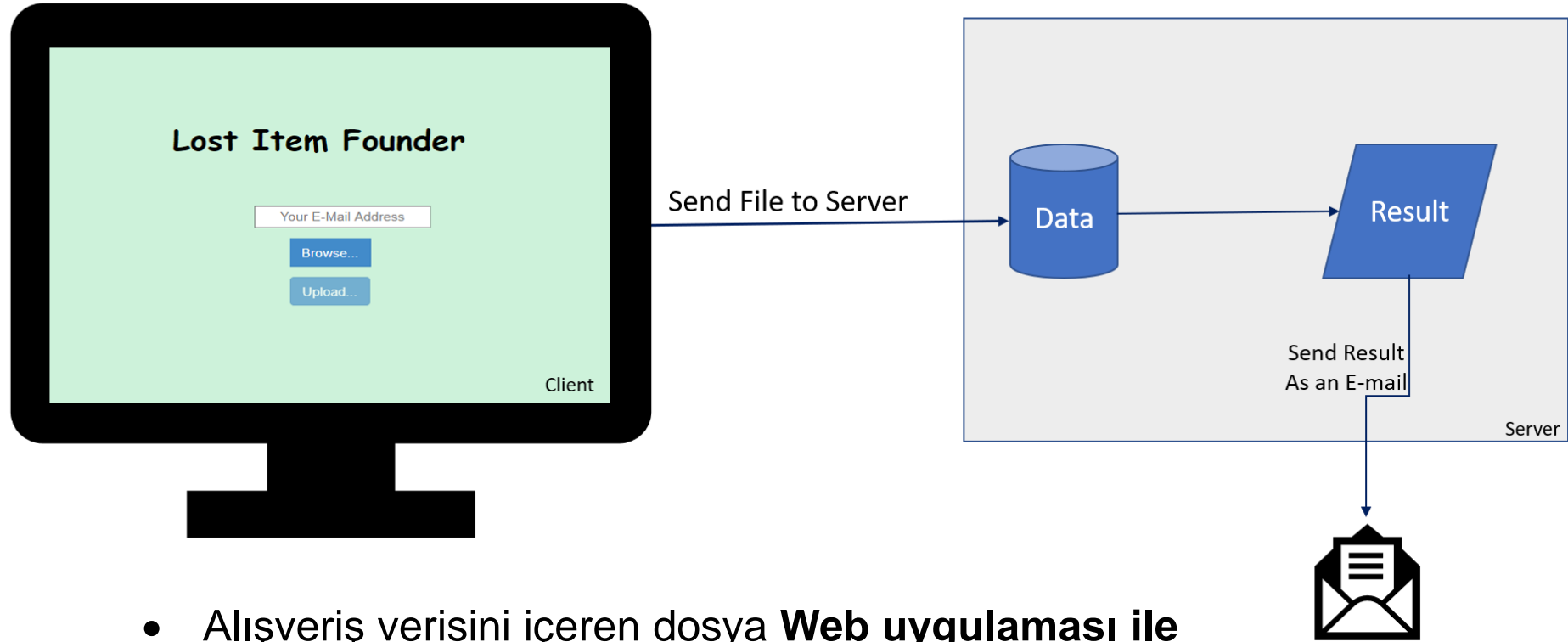


Proje nedir?

- Bu projede ticaret yapan şirketlerin **alışveriş verisi** (transaction) kullanılacaktır.
- Kullanılan veri üzerinde **makine öğrenmesi** algoritmaları kullanarak kayıp satışların tespiti gerçekleştirilecektir.



Proje Tasarım Planı



- Alışveriş verisini içeren dosya **Web uygulaması ile bilgisayardan seçilecektir.**
- Bu dosya da bulunan veri **sunucuya gönderilecek** ve orada makine öğrenmesi algoritmaları ile bu veri üzerinde işlem yapılacaktır.
- Daha sonra tespit edilen ürünler kullanıcıya rapor halinde **mail** olarak dönecektir.

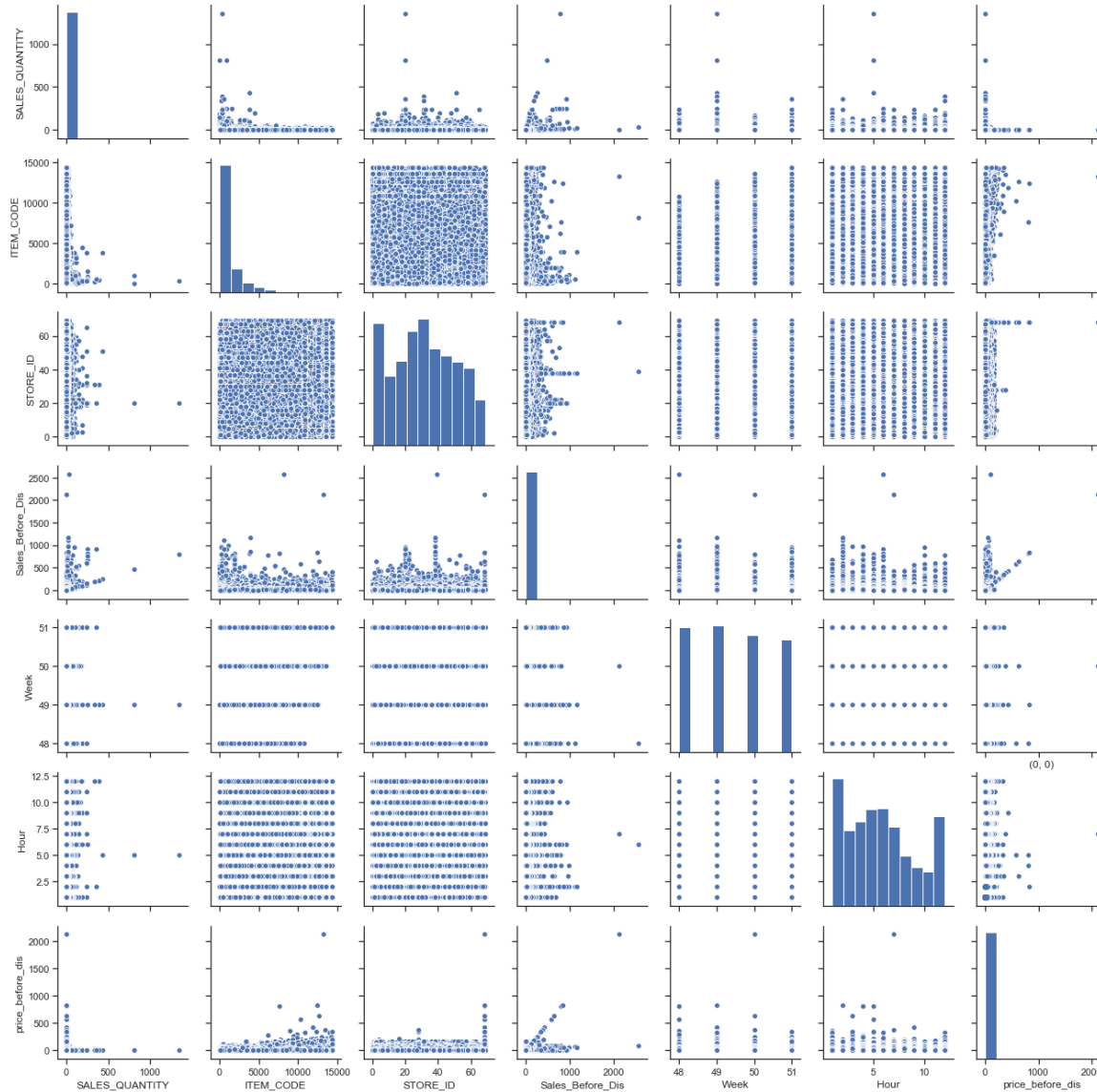


- Projenin client tarafı için bir **Web uygulaması** yapıldı.
- Bu uygulama ile kullanıcı test için elinde bulunan transaction verisini **local diskinden seçip** upload edebiliyor.
- Bu uygulama test datası ile işlem yaptıktan sonra oluşan sonucu bir rapor halinde kullanıcıya iletilmesi için kullanıcıdan **E-mail adresini** input olarak girmesini de istemektedir.

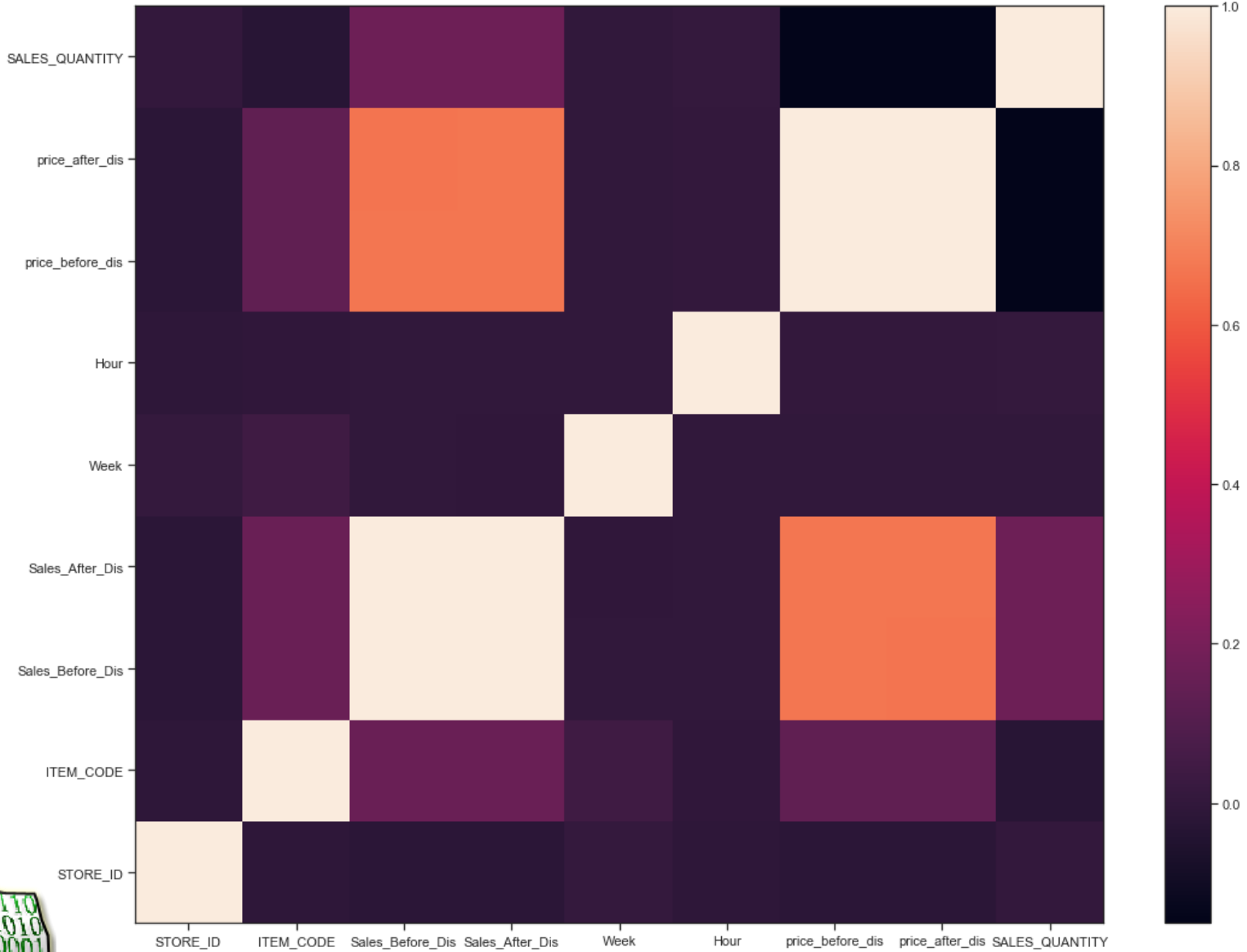


- Eğitim işlemini gerçekleştirmek için bir firmadan **4 haftalık alışveriş verisi** temin edildi. Bu veri içerisinde ilk başta belirtilen bilgiler bulunmaktaydı.
 - DWH_TRX_NO
 - ALISVERIS_TARIHI
 - STORE_ID
 - SALES_AMOUNT_BEFORE_DISCOUNT
 - SALES_AMOUNT_AFTER_DISCOUNT
 - SALES_QUANTITY
- Daha sonrasında veri üzerinde yapılan **ön işlemler** sonucunda yeni bilgiler eklendi.
 - Week
 - Hour
 - price_after_dis
 - price_before_dis
- Ek olarak da “**Store_ID**” ve “**ITEM_CODE**” vektörleri büyük integer değerler olduğundan matematiksel işlemlerde işimizi kolaylaştırması açısından küçük değerlerle map edildi.





Yapılanlar



- Eğitim için verilen alışveriş verisini eğitmek için **Tensor Factorization** olarak adlandırılan matematiksel işlem implement edildi.
- Bu implementi gerçekleştirmek için **Python** içerisinde bulunan **sktensor, numpy ve sklearn** gibi kütüphanelerden yararlandım.
- Yazılan **Web uygulaması, Send_Mail** modülü ve **Create_Report** modülü birbirine entegre edildi. Bu sayede en batada belirtildiği gibi kullanıcı mail adresini ve veriyi uygulama üzerinden gönderebilecek ve sonuç oluşur oluşmaz rapor hazırlanıp kullanıcıya mail yolu ile iletelebilmektedir.
- Implementasyonu gerçekleştirilen Tensor Factorization methoduna farklı şekilde verilen parametreler ile farklı sistemler oluşturuldu.



- Şekilde gösterilen sistemin oluşması için içersinde 500 adet alışveriş verisi olan train ve test datası kullanıldı ve 5 iterasyon sonucu oluşan sistemin Ortalama Kare Hata değeri **2.8851** olarak hesaplanmıştır.

```
1 Xtest, core, U = main()
[INFO] ----- TF -----
[INFO] Data: Number of types for each feature = [70, 10809, 69, 69, 50, 13, 93, 93]
[INFO] Data: Training Size = 500. Testing Size = 500
[INFO] Settings: K = [5, 5, 5, 5, 5, 5, 5, 5]. reg = 0.01. regS = 0.01. lr = 0.0001. lrS = 0.001
[INFO] Iter 1/5. Time: 25.1
[INFO] Training Loss: 2.01 (Testing Loss: 2.01%). Testing MSE: 2.8850
[INFO] Iter 2/5. Time: 25.0
[INFO] Training Loss: 2.00 (Testing Loss: 2.00%). Testing MSE: 2.8851
[INFO] Iter 3/5. Time: 24.8
[INFO] Training Loss: 1.99 (Testing Loss: 1.99%). Testing MSE: 2.8851
[INFO] Iter 4/5. Time: 25.0
[INFO] Training Loss: 1.98 (Testing Loss: 1.98%). Testing MSE: 2.8851
[INFO] Iter 5/5. Time: 25.0
[INFO] Training Loss: 1.97 (Testing Loss: 1.97%). Testing MSE: 2.8851
[INFO] Total Time: 125.0s
```



- Şekilde gösterilen sistemde 5000 adet eğitim için 500 adet test için alışveriş verisi kullanılmıştır.
- Diğer parametrelerde fazla bir değişiklik yapılmamıştır. Oluşan systemin Mean Squared Error değeri **2.8832** olarak hesaplanmıştır.

```
1 Xtest, core, U = main()
```

```
[INFO] ----- TF -----  
[INFO] Data: Number of types for each feature = [70, 10809, 102, 102, 50, 13, 116, 116]  
[INFO] Data: Training Size = 5000. Testing Size = 500  
[INFO] Settings: K = [5, 5, 5, 5, 5, 5, 5, 5]. reg = 0.01. regS = 0.001. lr = 0.001. lrS = 0.0001  
[INFO] Iter 1/5. Time: 245.5  
[INFO] Training Loss: 0.48 (Testing Loss: 0.40%). Testing MSE: 2.8859  
[INFO] Iter 2/5. Time: 244.6  
[INFO] Training Loss: 0.48 (Testing Loss: 0.40%). Testing MSE: 2.8857  
[INFO] Iter 3/5. Time: 245.1  
[INFO] Training Loss: 0.48 (Testing Loss: 0.40%). Testing MSE: 2.8853  
[INFO] Iter 4/5. Time: 245.5  
[INFO] Training Loss: 0.48 (Testing Loss: 0.40%). Testing MSE: 2.8846  
[INFO] Iter 5/5. Time: 245.1  
[INFO] Training Loss: 0.48 (Testing Loss: 0.40%). Testing MSE: 2.8832  
[INFO] Total Time: 1225.8s
```



- Şekilde gösterilen sistemde bir önceki sistemle aynı boyutta aynı veri kullanılmıştır.
- Diğer parametrelerde Birkaç değişiklik yapılmıştır. Oluşan systemin Mean Squared Error değeri **2.8753** olarak hesaplanmıştır.

```
: 1 Xtest, core, U = main()

[INFO] ----- TF -----
[INFO] Data: Number of types for each feature = [70, 10809, 102, 102, 50, 13, 116, 116]
[INFO] Data: Training Size = 5000. Testing Size = 500
[INFO] Settings: K = [4, 4, 4, 4, 4, 4, 4, 4]. reg = 0.01. regS = 0.001. lr = 0.001. lrS = 1e-06
[INFO] Iter 1/5. Time: 21.5
[INFO] Training Loss: 0.84 (Testing Loss: 0.36%). Testing MSE: 2.8814
[INFO] Iter 2/5. Time: 21.7
[INFO] Training Loss: 0.84 (Testing Loss: 0.36%). Testing MSE: 2.8808
[INFO] Iter 3/5. Time: 21.5
[INFO] Training Loss: 0.84 (Testing Loss: 0.36%). Testing MSE: 2.8798
[INFO] Iter 4/5. Time: 21.3
[INFO] Training Loss: 0.84 (Testing Loss: 0.36%). Testing MSE: 2.8782
[INFO] Iter 5/5. Time: 21.9
[INFO] Training Loss: 0.84 (Testing Loss: 0.36%). Testing MSE: 2.8753
[INFO] Total Time: 107.9s
```



- Son olarak eğitim için bir haftalık very olan yani 1.279.084 adet alışveriş verisi kullanılarak bir system oluşturulmuştur.
- Oluşan systemin Mean Squared Error değeri **2.8883** olarak hesaplanmıştır.

```
1 Xtest, core, U = main()
[INFO] ----- TF -----
[INFO] Data: Number of types for each feature = [70, 10809, 2577, 2577, 50, 13, 812, 812]
[INFO] Data: Training Size = 1279084. Testing Size = 500
[INFO] Settings: K = [4, 4, 4, 4, 4, 4, 4, 4, 4]. reg = 0.1. regS = 0.1. lr = 0.001. lrS = 0.0001
[INFO] Iter 1/5. Time: 4918.6
[INFO] Training Loss: 52708362.74 (Testing Loss: 15180.22%). Testing MSE: 2.8883
[INFO] Iter 2/5. Time: 4956.5
[INFO] Training Loss: 16764679098976.55 (Testing Loss: 4826993966.87%). Testing MSE: 2.8883
[INFO] Iter 3/5. Time: 4913.8
[INFO] Training Loss: 4928312612913712128.00 (Testing Loss: 1418991385986565.75%). Testing MSE: 2.8883
[INFO] Iter 4/5. Time: 4935.6
[INFO] Training Loss: 1340984678194298220445696.00 (Testing Loss: 386104911874224029696.00%). Testing MSE: 2.8883
[INFO] Iter 5/5. Time: 4944.8
[INFO] Training Loss: 338207308227397486730171908096.00 (Testing Loss: 97378818014682012459204608.00%). Testing MSE: 2.8883
[INFO] Total Time: 24669.3s
```



- Oluşturulacak modelin train süresi maximum **5 saat** olması bekleniyor. (100 ürün – 100 bayi için)
 - Bu süre parametrelere göre değişiklik gösterebilmektedir.
- Oluşturulan modelin haftalık veri için çalışma süresi en fazla **1 saat** olması bekleniyor. (100 ürün – 100 bayi için)
 - Test için iki yöntem var var olan sistemi kullanmak veya data için yeni system oluşturmak. İlk durumda saniyeler içinde sonuç alınabilir. İkinci durumda ise ilk başarı kriterindeki durum söz konusudur.
- Sistemin **Ortalama Kare hata değeri (Mean Squared Error) %20** olması bekleniyor.
 - Bu değer oluşturulan bütün system için %80 civarındadır.



1. Veri Bilimcisi. (2019). *MSE, RMSE, MAE, MAPE ve Diğer Metrikler*. [online] Available at: <https://veribilimcisi.com/2017/07/14/mse-rmse-mae-mape-metrikleri-nedir/> [Accessed 5 Mar. 2019].
2. Towards Data Science. (2019). *Machine Learning – Towards Data Science*. [online] Available at: <https://towardsdatascience.com/machine-learning/home> [Accessed 5 Mar. 2019].
3. Kaggle.com. (2019). *Walmart Sales*. [online] Available at: <https://www.kaggle.com/anshg98/walmart-sales> [Accessed 5 Mar. 2019].
4. Medium. (2019). *Tensor Decomposition in Python*. [online] Available at: <https://medium.com/@mohammadbashiri93/tensor-decomposition-in-python-f1aa2f9adb4> [Accessed 31 Mar. 2019].

