



YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

YAPAY ZEKA 2. ÖDEVİ

20011607 Zeynep Çolak

19011089 Elif Ayanoğlu

Giriş

Bu projede amacımız, giyim kategorisine ait müşteri yorumlarının olumlu mu olumsuz mu olduğunun tespit edilmesidir. Bunun için Gradient Boosting, Support Vector Machines, Logistic Regression, K-Nearest Neighbors ve Random Forest algoritmaları ayrı ayrı denenmiştir. Verisetini anlamlı bir şekilde kullanabilmek ve modelin sonuçlarının güvenilirliğini artırmak için KFold çapraz doğrulama kullanılmıştır. Bunun yanı sıra, algoritmaların başarılarına olan etkilerini gözlemek için özellik seçimi yöntemlerinden “chi squared” tekniği, “Principal Component Analysis (PCA)” ve normalizasyon tekniklerinden “maximum absolute scaling” yöntemleri ayrı ayrı denenmiş ve başarıya olan etkileri gözlemlenmiştir. En son tahminleyicilerin performansları arasında anlamlı bir fark olup olmadığını test etmek için “t-test” ile ölçüm yapılmıştır.

Veri Kümesi

Bu projede kullanılan veri seti, Trendyol sitesinde yayınlanan müşteri yorumlarından çekilerek hazırlanmıştır. Sadece giyim kategorisine ait ürünlerin yorumları çekilmiştir. Trendyol’un sitesinde yorumlar için yapılan değerlendirmelerde kullanıcılar 1 ile 5 arasında yıldız vermektedir. Biz de proje kapsamında yorumları etiketlerken 1 ve 2 yıldızlı yorumlar için olumsuz, 4 ve 5 yıldızlı yorumlar için de olumlu etiket verdik. Veri seti içerisinde toplam 3000 yorumun 1500 tanesi olumlu olup, “1” ile 1500 tanesi de olumsuz olup “0” ile etiketlenmiştir.

En Başarılı Sonuçlar

En başarılı sonuçlar grafikleri ve ayrıntılarıyla aşağıda sunulmuştur.

En başarılı GB algoritması sonucu Ki-kare k=800

GradientBoosting

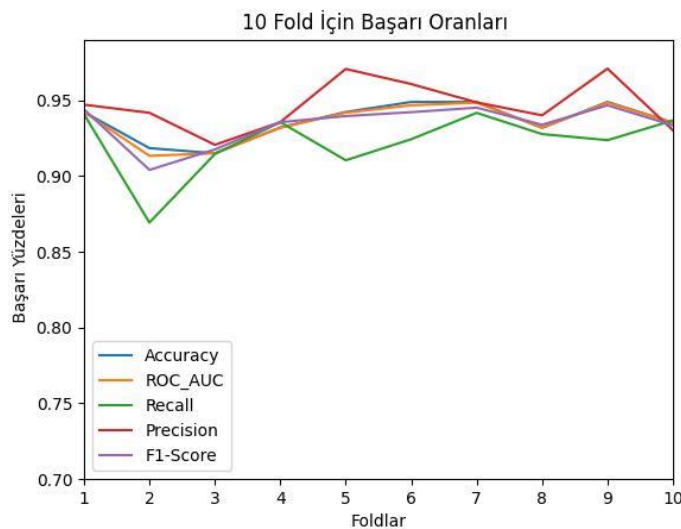
Tüm Foldların Ortalama Accuracy Skoru : 0.94

Tüm Foldların Ortalama F1 Skoru : 0.93

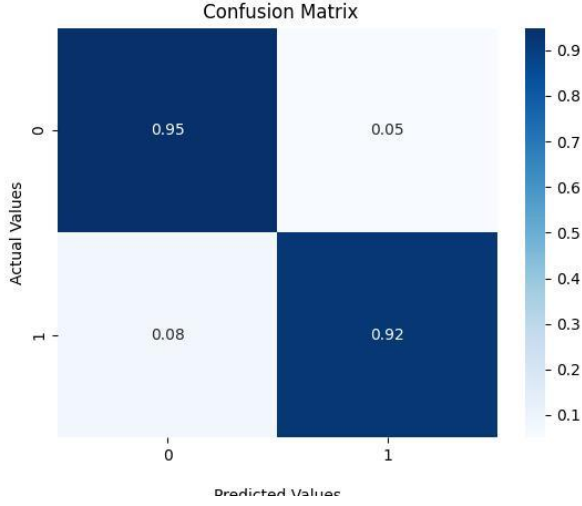
Tüm Foldların Ortalama Recall Skoru : 0.92

Tüm Foldların Ortalama ROC-AUC Skoru : 0.94

Tüm Foldların Ortalama Precision Skoru : 0.95



Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



Confusion Matrisi yukardaki gibidir.

En başarılı SVM algoritması sonucu Ki-kare k=800

SVC

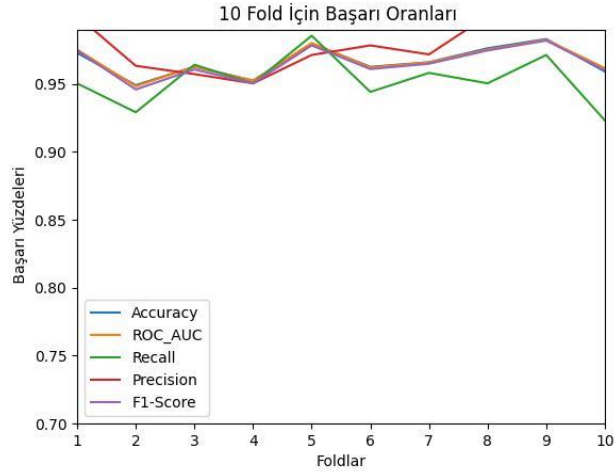
Tüm Foldların Ortalama Accuracy Skoru : 0.97

Tüm Foldların Ortalama F1 Skoru : 0.97

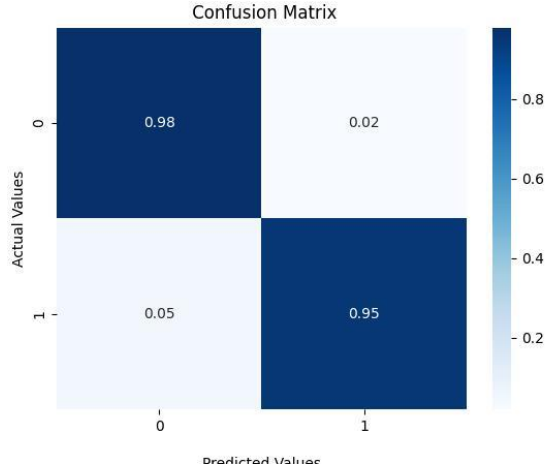
Tüm Foldların Ortalama Recall Skoru : 0.95

Tüm Foldların Ortalama ROC-AUC Skoru : 0.97

Tüm Foldların Ortalama Precision Skoru : 0.98



Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



Confusion Matrisi yukardaki gibidir.

En başarılı Random Forest algoritması sonucu Ki-kare k=2000

RandomForest

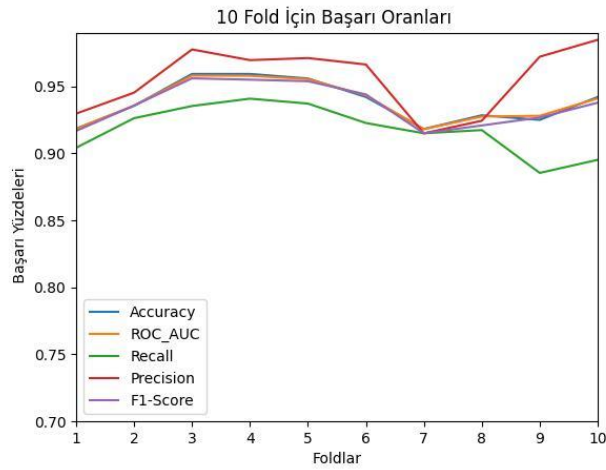
Tüm Foldların Ortalama Accuracy Skoru : 0.94

Tüm Foldların Ortalama F1 Skoru : 0.94

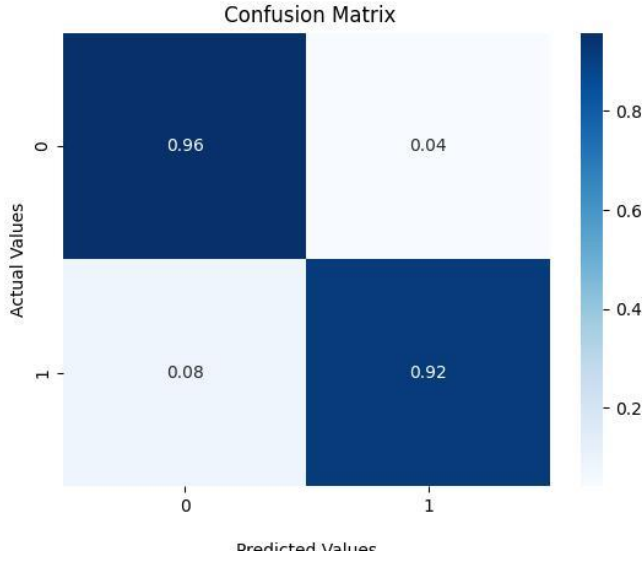
Tüm Foldların Ortalama Recall Skoru : 0.92

Tüm Foldların Ortalama ROC-AUC Skoru : 0.94

Tüm Foldların Ortalama Precision Skoru : 0.96



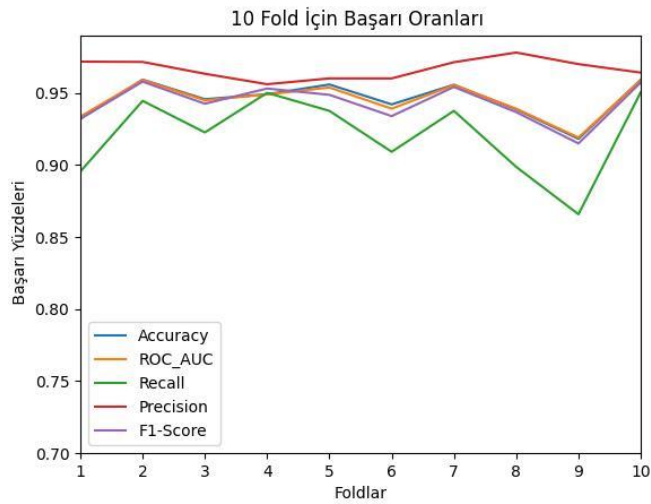
Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



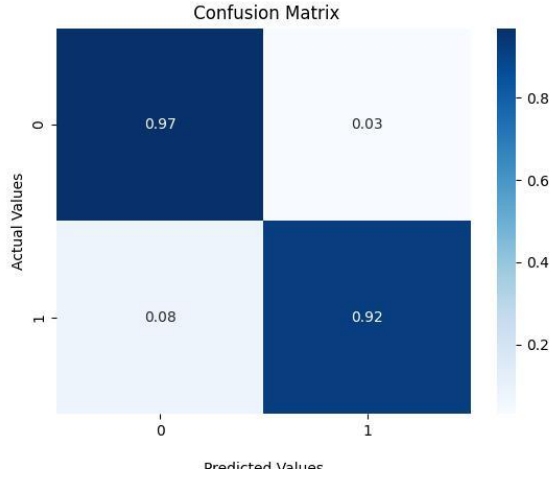
Confusion Matrisi yukardaki gibidir.

En başarılı Logistic Regression algoritması sonucu PCA N = 150

Tüm Foldların Ortalama Accuracy Skoru : 0.95
Tüm Foldların Ortalama F1 Skoru : 0.94
Tüm Foldların Ortalama Recall Skoru : 0.92
Tüm Foldların Ortalama ROC-AUC Skoru : 0.95
Tüm Foldların Ortalama Precision Skoru :0.97



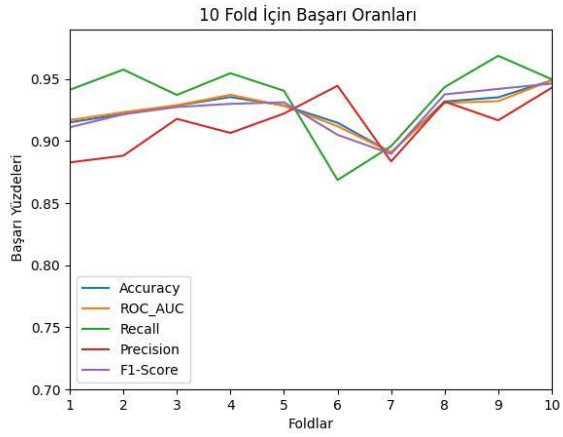
Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



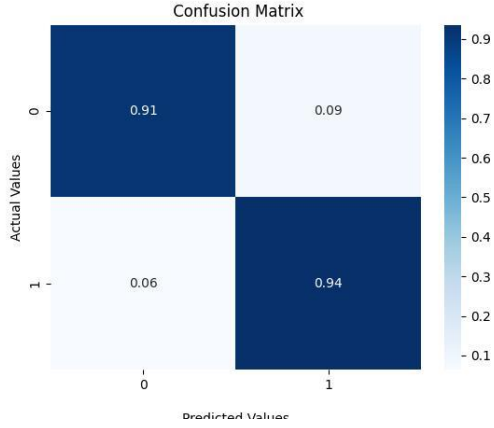
Confusion Matrisi yukardaki gibidir.

En başarılı KNN algoritması sonucu PCA N = 900

Tüm Foldların Ortalama Accuracy Skoru : 0.93
Tüm Foldların Ortalama F1 Skoru : 0.92
Tüm Foldların Ortalama Recall Skoru : 0.94
Tüm Foldların Ortalama ROC-AUC Skoru : 0.92
Tüm Foldların Ortalama Precision Skoru :0.91



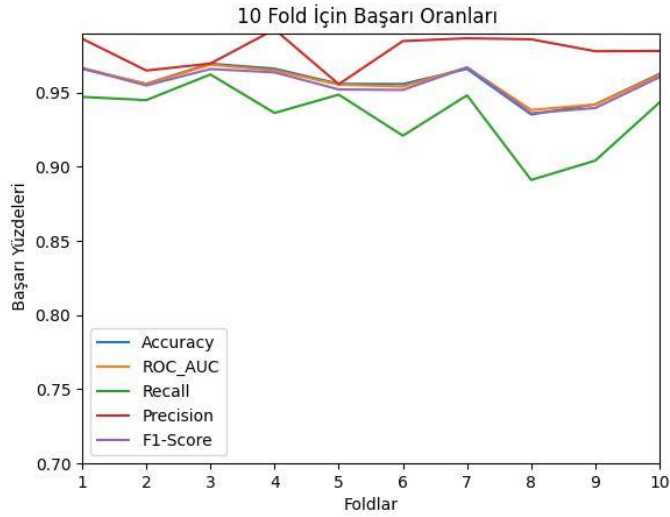
Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



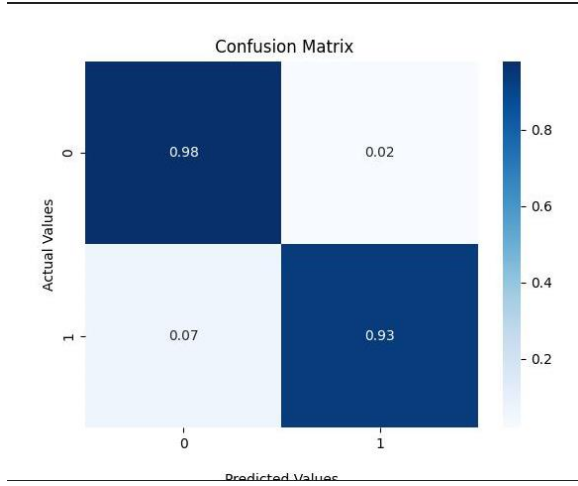
Confusion Matrisi yukardaki gibidir.

Herhangi bir teknik uygulanmadan elde edilen en başarılı sonuç SVM algoritmasında alınmıştır.

Tüm Foldların Ortalama Accuracy Skoru : 0.96
Tüm Foldların Ortalama F1 Skoru : 0.96
Tüm Foldların Ortalama Recall Skoru : 0.93
Tüm Foldların Ortalama ROC-AUC Skoru : 0.96
Tüm Foldların Ortalama Precision Skoru :0.98



Başarı metriklerinin foldlar üzerindeki etkileri yukardaki grafikte görülebilir.



Confusion Matrisi yukardaki gibidir.

Sonuçlar ve Yorumlar

Bu başlık altında deney sonuçlarımızdan elde ettiğimiz “accuracy” değerleri ve “confusion matrix” değerleri tablolaştırılarak sunulmuştur. Her tablonun altında o tablodan çıkarılan sonucu “yorum” başlığı altında bulabilirsiniz.

“PCA” Tekniği Uygulanmış Sonuçlar ve Yorumları

N/ Algoritmalar	GradientBstn	SVM	L-Regression	KNN (k=5)	RndmForest
N=30	0.93	0.93	0.92	0.90	0.92
N=40	0.93	0.94	0.93	0.90	0.92
N=50	0.93	0.94	0.93	0.90	0.93
N=100	0.93	0.95	0.94	0.89	0.93
N=150	0.93	0.95	0.95	0.88	0.93
N=200	0.93	0.95	0.95	0.85	0.93
N=700	0.92	0.96	0.95	0.90	0.92
N=800	0.91	0.96	0.95	0.91	0.91
N=900	0.91	0.96	0.95	0.93	0.91
N=1000	0.89	0.96	0.95	0.93	0.90

Tablo : PCA için N değeri artarken “Accuracy” metriğinin değişimi

ÖNEMLİ NOT: N değeri 200 – 600 arasında iken de test edilmiştir ancak bütün 5 algortmada da sonuçların bu aralıkta sabit kaldığı gözlemlenmiştir. Bu yüzden tabloda bu aralığa yer verilmemiştir.

PCA'da, özellik sayısı azaltıldığı için veri daha az boyutlu hale gelir. Bu, modelin daha az parametre içermesi ve daha az overfitting yapması anlamına gelir. Bununla birlikte, aşırı boyut azaltma önemli bilgi kaybına neden olabilir ve bu da modelin daha düşük bir performans göstermesiyle sonuçlanabilmektedir. Yani bazı durumlarda, PCA uygulaması modelin performansını artırırken, bazı durumlarda performansı düşürebilir.

Bu çalışma kapsamındaki denemelerimiz sonucunda PCA'nın 5 algortmada da farklı değerlerde farklı değişimlere sebebiyet verdiği gözlemlenmiştir. TabloX'de de görüldüğü üzere **GBC algortimasında**

200 değerine kadar en yüksek başarı değeri olan 0.93 değeri elde edilirken 700 değeri itibariyle başarı kademeli olarak düşmeye başlamıştır. Yani bu algortimada özellik sayısını 700'den az tutmak modelin başarısına olumlu etki etmiştir. **SVM algoritmasında**, 30'dan 1000'e kadar kademeli bir artış görülmektedir ancak bu artışın 700'den sonra sabit kaldığı ve daha da arttırmaya gerek olmadığı açıkça gözlemlenebilir. **Logistic Regression algoritmasında** da SVM ile aynı şekilde kademeli bir artış mevcuttur ancak 150'den sonra özellik sayısının artması başarıya etki etmemiştir. **KNN algoritmasında** diğerlerine göre daha dalgalı bir durum gözlemlenmiştir. Bu durum KNN algoritmasının, veri kümesindeki özelliklerin benzerliğine dayalı olarak sınıflandırma yapmasından dolayı meydana geldiği düşünülebilir. Son olarak **Random Forest algoritmasında**, 200 değerine kadar kademeli bir artış görülmüş ancak 200'den sonra kademeli bir düşüş gözlemlenmiştir. Bu durum 200'den fazla olan özellik sayısının başarıyı kötü etkilediğini göstermektedir.

N/ SVM Conf. M.	TN	FP	FN	TP
N=30	0.95	0.05	0.09	0.91
N=40	0.96	0.04	0.08	0.92
N=50	0.96	0.04	0.08	0.92
N=100	0.96	0.04	0.07	0.93
N=150	0.97	0.03	0.06	0.94
N=200	0.97	0.03	0.06	0.94
N=700	0.98	0.02	0.06	0.94
N=800	0.98	0.02	0.06	0.94
N=900	0.98	0.02	0.06	0.94
N=1000	0.98	0.02	0.06	0.94

Tablo : PCA için N değeri artarken SVM algoritmasının “Confusion Matris” değişimi

Tabloda görüldüğü üzere TP ve TN değerleri (pozitif ve negatif değerlerin doğru bulunduğu sütunlar) kademeli olarak artmıştır. Her seferinde negatif olanların doğru bulunma başarısı, pozitif olanların doğru bulunma başarısından yüksek çıkmaktadır. FP ve FN değerleri ise (pozitif ve negatif değerlerin yanlış bulunduğu sütunlar) kademeli olarak düşmüştür ve pozitif değerlerin yanlış bulunması negatif değerlerin yanlış bulunması durumundan daha düşüktür. Bu durumlar değerlendirildiğinde PCA uygulamasının SVM algoritmasının başarısına olumlu etki ettiği gözlemlenmiştir.

N/ LR Conf. M.	TN	FP	FN	TP
N=30	0.96	0.04	0.11	0.89
N=40	0.96	0.04	0.10	0.90
N=50	0.96	0.04	0.09	0.91
N=100	0.97	0.03	0.09	0.91
N=150	0.97	0.03	0.08	0.92
N=200	0.97	0.03	0.07	0.93
N=700	0.98	0.02	0.07	0.93
N=800	0.97	0.03	0.07	0.93
N=900	0.97	0.03	0.07	0.93
N=1000	0.98	0.02	0.07	0.93

Tablo : PCA için N değeri artarken Logistic Regression algoritmasının “Confusion Matris” değişimi

Tabloda görüldüğü üzere TP ve TN değerleri (pozitif ve negatif değerlerin doğru bulunduğu sütunlar) kademeli olarak artmıştır. Her seferinde negatif olanların doğru bulunma başarısı, pozitif olanların doğru bulunma başarısından yüksek çıkmaktadır. FP ve FN değerleri ise (pozitif ve negatif değerlerin yanlış bulunduğu sütunlar) kademeli olarak düşmüştür ve pozitif değerlerin yanlış bulunması negatif değerlerin yanlış bulunması durumundan daha düşüktür. Bu durumlar değerlendirildiğinde PCA uygulamasının Logistic Regression algoritmasının başarısına olumlu etki ettiği gözlemlenmiştir.

N/ GBC Conf. M.	TN	FP	FN	TP
N=30	0.94	0.06	0.08	0.92
N=40	0.94	0.06	0.07	0.93
N=50	0.94	0.06	0.07	0.93
N=100	0.94	0.06	0.07	0.93
N=150	0.94	0.06	0.07	0.93
N=200	0.94	0.06	0.08	0.92
N=700	0.94	0.06	0.11	0.89
N=800	0.94	0.06	0.11	0.89
N=900	0.94	0.06	0.13	0.87
N=1000	0.93	0.07	0.14	0.86

Tablo : PCA için N değeri artarken Gradient Boosting Classifier algoritmasının “Confusion Matris” değişimi

TN değerleri genel olarak sabit kalırken TP değerleri belli bir noktaya kadar sabit kalıp bir noktadan sonra düşmektedir. FP değeri genel olarak sabit kalırken FN değerleri belli bir noktaya kadar sabit kalıp bir noktadan sonra düşmektedir. Bu durumlar değerlendirildiğinde PCA uygulamasının n=200 değerinden sonra **Gradient Boosting Classifier algoritması için uygulanması uygun değildir**. Bu değerden sonra başarı değerlerine etkisi olumsuz olmuştur.

N/ KNN Conf. M.	TN	FP	FN	TP
N=30	0.93	0.07	0.13	0.87
N=40	0.94	0.06	0.13	0.87
N=50	0.94	0.06	0.13	0.87
N=100	0.91	0.09	0.14	0.86
N=150	0.89	0.11	0.11	0.89
N=200	0.80	0.20	0.08	0.92
N=700	0.85	0.15	0.05	0.95
N=800	0.88	0.12	0.05	0.95
N=900	0.91	0.09	0.06	0.94
N=1000	0.92	0.08	0.05	0.95

Tablo : PCA için N değeri artarken KNN algoritmasının “Confusion Matris” değişimi

TN değerlerinin değişimi dalgalı şekilde seyretmektedir. TP değerleri ise kademeli olarak artmaktadır. FP değeri belli bir noktaya kadar artarken bir noktadan sonra düşmüştür. FN değerleri genel bir düşüş eğilimi göstermiştir. Bu durumlar değerlendirildiğinde PCA uygulamasının **KNN algoritması için uygulanması pek uygun değildir**.

N/ RF Conf. M.	TN	FP	FN	TP
N=30	0.93	0.07	0.08	0.92
N=40	0.93	0.07	0.08	0.92
N=50	0.93	0.07	0.08	0.92
N=100	0.93	0.07	0.07	0.93
N=150	0.94	0.06	0.08	0.92
N=200	0.94	0.06	0.08	0.92
N=700	0.95	0.05	0.12	0.88
N=800	0.95	0.05	0.13	0.87
N=900	0.95	0.05	0.14	0.86
N=1000	0.96	0.04	0.15	0.85

Tablo : PCA için N değeri artarken Random Forest algoritmasının “Confusion Matris” değişimi

Tabloda görüldüğü üzere TN değerleri kademeli olarak artmıştır. TP değerleri ise kademeli olarak düşmüştür. Her seferinde negatif olanların doğru bulunma başarısı, pozitif olanların doğru bulunma başarısından yüksek çıkmaktadır. FP değerleri kademeli olarak düşmüş ve FN değerleri kademeli olarak artmıştır. Pozitif değerlerin yanlış bulunması negatif değerlerin yanlış bulunması durumundan daha düşüktür. Bu durumlar değerlendirildiğinde PCA uygulamasının **Random Forest algoritmasının başarısına olumlu etki ettiği gözlemlenmiştir**. Uygun bir değer seçilip uygulanabilir.

“chi squared” Tekniği Uygulanmış Sonuçlar ve Yorumları

K/ Algoritmalar	GradientBstn	SVM	L-Regression	KNN (k=5)	RndmForest
K=10	0,85	0,85	0,85	0,70	0,82
K=25	0,90	0,89	0,90	0,84	0,88
K=50	0,92	0,92	0,91	0,87	0,91
K=100	0,92	0,93	0,93	0,90	0,92
K=200	0,93	0,95	0,94	0,90	0,93
K=400	0,93	0,96	0,95	0,86	0,93
K=800	0,94	0,97	0,95	0,76	0,93
K=1200	0,93	0,96	0,95	0,81	0,93
K=1500	0,93	0,96	0,95	0,79	0,93
K=1800	0,93	0,96	0,95	0,80	0,93
K=2000	0,94	0,96	0,95	0,79	0,94
K=2500	0,94	0,96	0,95	0,74	0,94
K=3000	0,93	0,96	0,95	0,70	0,94
K=3500	0,93	0,96	0,95	0,70	0,94
K=4000	0,93	0,96	0,95	0,69	0,94
K=4500	0,93	0,96	0,95	0,71	0,94

Tablo : Özellik seçimi k değeri artarken “Accuracy” metriğinin değişimi

K = 4772	0,92	0,96	0,95	0,93	0,95
----------	------	------	------	------	------

Ki kare tekniği kullanılmadan tüm özellikleri ile çalıştırılan algoritmaların sonuçları

Öznitelik seçimi, veri seti içerisinde en yararlı öznitelikleri seçme ve bulma sürecidir. Çok fazla değişken kullanmak modelin performansını düşürebilir. En iyi performans sonucu elde etmek için, en optimum değişkenleri belirleyip onlar ile model kurmak gerekir. Bu işlem Makine Öğrenmesi modelinin performansını çok fazla etkilemektedir. Veriseti üzerinde çalıştırılan 5 algoritma'ya farklı k değerleri verilerek ki kare tekniği kullanılmıştır. Ede edilen sonuçlar Tablo X'de gösterilmiştir. Sonuçların yorumları ise aşağıdadır.

GB algoritmasında k değeri küçükken başarı düşük olmuş k=800'den sonra %94 değeri görülmüştür. Bu değer özellik seçimi yapılmadığında elde edilen başarıdan %2 yüksektir. Özellik seçimi bu algortmada **olumlu yönde etki göstermiştir**.

SVM algoritması da düşük k değerlerinde düşük başarı göstermiş ve k=800'de en yüksek başarıya ulaşmıştır. Bu değer ki kare tekniği kullanılmadığında gösterilen başarıdan %1 daha yüksektir. Bu arada SVM ve GB algoritmasının en yüksek başarıya ulaştığı k değerlerinin aynı olması da gözden kaçmamalıdır.

Logistik Regresyon algoritması k=400'den sonra sabitlenen bir başarı göstermiştir. 300-4772 değerleri arasında başarı hep aynı kalmış. Bu değer Ki kare tekniği kullanılmadığında alınan başarıyla aynıdır. Anlaşıldığı üzere özellik seçimi yapmak bu algortmada **başarıyı yükseltecek yönde bir etki göstermemiştir**.

KNN algoritması için özellik seçiminin etkisi **olumsuz yönde ve yüksektir**.

Random Forest algoritmasının sonuçları da **SVM algoritmasının** sonuçlarına benzerdir. **Kayda değer olumlu bir etki görülmemiştir**.

K/ GB Conf. M.	TP	FN	FP	TN
K=10	0,77	0,23	0,07	0,93
K=25	0,84	0,16	0,05	0,95
K=50	0,88	0,12	0,05	0,95
K=100	0,88	0,12	0,04	0,96
K=200	0,90	0,10	0,04	0,96
K=400	0,95	0,05	0,02	0,98
K=1200	0,94	0,06	0,07	0,93
K=1500	0,94	0,06	0,07	0,93
K=2000	0,94	0,06	0,07	0,93
K=3000	0,92	0,08	0,07	0,93
K=4500	0,92	0,08	0,07	0,93

Tablo : Özellik seçimi k değeri artarken GB algoritmasının "Confusion Matris" değişimi

Tabloda gösterilen değişimlere bakılarak GB algoritmasında k=400 iken Ki kare tekniği başarıyı yükseltmiştir. Anlaşıldığı üzere özellik seçimi başarıya olumlu etki etmiştir ve bu etki hem true pozitifler için hem de true negatifler için de olumlu yönde olmuştur. Bu demek oluyor ki hem olumlu sınıfları tahmini hem olumsuz sınıfların tahminine olumlu yönde etki etmiştir.

K/ SVM Conf. M.	TP	FN	FP	TN
K=10	0,78	0,22	0,08	0,92
K=25	0,84	0,16	0,05	0,95
K=50	0,87	0,13	0,04	0,96
K=100	0,90	0,10	0,04	0,96
K=200	0,93	0,07	0,03	0,97
K=400	0,95	0,05	0,02	0,98
K=1200	0,95	0,05	0,02	0,98
K=1500	0,94	0,06	0,02	0,98
K=4500	0,94	0,06	0,02	0,98

Tablo : Özellik seçimi k değeri artarken SVM algoritmasının “Confusion Matris” değişimi

NOT: K değeri 1500 – 4500 arasında iken test edilen sonuçlarda değişiklik gözlemlenmediğinden tabloda yer verilmemiştir.

Tabloya bakıldığında SVM algoritması sonucunun TN değeri bir yerden sonra sabitlenmesine rağmen TP değeri k=1500’de düşme yaşandığı görülür. Buna bakılarak k=400-1200 aralığında en iyi sonucu verdiği söylenebilir. Bu sonuçlar özellik seçimi olmadan elde edilen sonuçlara benzerdir.

K/ LR Conf. M.	TP	FN	FP	TN
K=10	0,77	0,23	0,07	0,93
K=25	0,83	0,17	0,04	0,96
K=50	0,87	0,13	0,04	0,96
K=100	0,90	0,10	0,04	0,96
K=200	0,91	0,09	0,03	0,97
K=400	0,92	0,08	0,03	0,97
K=1200	0,93	0,07	0,03	0,97
K=1500	0,93	0,07	0,03	0,97
K=3000	0,93	0,07	0,03	0,97
K=4500	0,93	0,07	0,02	0,98
K=4772	0,94	0,06	0,05	0,95

Tablo : Özellik seçimi k değeri artarken LR algoritmasının “Confusion Matris” değişimi

Özellik seçimi uygulandığında Logistik Regresyon algoritması en iyi sonucu k=4500’de göstermiş. Hem TP hem TN değerleri bu aşamada yüksek çıkmış. Bu değer ki kare uygulamadığında elde edilen sonuçlarla çok farklı değildir. Ancak TN sütunu incelendiğinde özellik seçiminin olumsuz sınıfları tahmin etmede olumlu yönde etki gösterdiği görülebilir (%98).

K/ KNN Conf. M.	TP	FN	FP	TN
K=10	0,95	0,05	0,55	0,45
K=25	0,94	0,06	0,26	0,74
K=50	0,90	0,10	0,15	0,85
K=100	0,91	0,09	0,11	0,89
K=200	0,93	0,07	0,13	0,87
K=400	0,97	0,03	0,23	0,77
K=800	0,97	0,03	0,44	0,56
K=1200	0,77	0,23	0,15	0,85
K=1500	0,91	0,9	0,33	0,67
K=3000	0,98	0,02	0,58	0,42
K=4500	0,96	0,04	0,53	0,47

Tablo : Özellik seçimi k değeri artarken KNN algoritmasının “Confusion Matris” değişimi

KNN algoritmasına özellik seçimi uygulamak True Negatif değerini oldukça etkilemiş. Yüksek k değerleri TN değerlerini oldukça düşürmüştür. Ancak TP değerleri TN değerleri kadar etkilenmemiş. Hatta arasında büyük bir fark ortaya çıkarmıştır. Genel başarıya bakılırsa k=100’de TN en yüksek, k=3000’de TP en yüksek değerine ulaşmıştır.

K/ RF Conf. M.	TP	FN	FP	TN
K=10	0,73	0,27	0,09	0,91
K=25	0,82	0,18	0,06	0,94
K=50	0,88	0,12	0,07	0,93
K=100	0,90	0,10	0,06	0,94
K=200	0,91	0,09	0,03	0,97
K=400	0,94	0,06	0,07	0,93
K=1200	0,91	0,09	0,05	0,95
K=1500	0,91	0,09	0,04	0,96
K=3000	0,92	0,08	0,04	0,96
K=4500	0,92	0,08	0,04	0,96

Tablo : Özellik seçimi k değeri artarken RandomForest algoritmasının “Confusion Matris” değişimi

Random Forest algoritmasında TP ve TN, en yüksek başarılarını farklı k değerlerinde almışlardır. K=400’de TP en yüksek değeri alırken k=200’de TN en yüksek değerini almıştır. Başarıları farklı k değerlerinde aldıkları için özellik seçiminin hangi k değerinde en yüksek olduğu bu tabloya bakarak tam anlaşılamaz.

T-TEST Sonuçları

T-test, iki grup arasındaki sayısal verilerin ortalamaları arasındaki farkın anlamlı olup olmadığını test etmek için kullanılan bir hipotez testidir. T-testi kullanarak, iki farklı popülasyonun örneklem verileri arasındaki farkın tesadüfi mi yoksa gerçekten anlamlı mı olduğunu ölçebilirsiniz. T-testi sonucunda elde edilen p-value değeri, ön hipotezin doğruluğu hakkında bilgi verir. Eğer p-value değeri, belirli bir anlamlılık düzeyine (genellikle %5) göre küçükse, ön hipotez reddedilir ve alternatif hipotez kabul edilir. Aşağıda 5 algoritmanın birbirleriyle karşılaştırılmak için elde edilen t-test sonuçlarını görebilirsiniz. N değerleri PCA sonuçlarını, K değerleri Ki kare Tekniği sonuçlarını göstermektedir.

N = 30 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark yok.
GBC - LR:
İstatistiksel olarak anlamlı bir fark yok.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark yok.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 40 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark yok.
GBC - LR:
İstatistiksel olarak anlamlı bir fark yok.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark yok.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 50 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark yok.
GBC - LR:
İstatistiksel olarak anlamlı bir fark yok.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark yok.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 100 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark var.
GBC - LR:
İstatistiksel olarak anlamlı bir fark var.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark var.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 150 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark var.
GBC - LR:
İstatistiksel olarak anlamlı bir fark yok.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark var.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 200 :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark var.
GBC - LR:
İstatistiksel olarak anlamlı bir fark yok.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark var.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

N = 700 :

N = 800 :

N = 1000 :

```
GBC - SVM:
İstatistiksel olarak anlamlı bir fark var.
GBC - LR:
İstatistiksel olarak anlamlı bir fark var.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark var.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.
```

K=100:

```
T-test Sonuçları:
GB - SVM:
İstatistiksel olarak anlamlı bir fark yok.
GB - LR:
İstatistiksel olarak anlamlı bir fark yok.
GB - KNN:
İstatistiksel olarak anlamlı bir fark var.
GB - RF:
İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark yok.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:
İstatistiksel olarak anlamlı bir fark yok.
```

K=800:

```
T-test Sonuçları:
GB - SVM:
İstatistiksel olarak anlamlı bir fark var
GB - LR:
İstatistiksel olarak anlamlı bir fark var
GB - KNN:
İstatistiksel olarak anlamlı bir fark var
GB - RF:
İstatistiksel olarak anlamlı bir fark yok
SVM - LR:
İstatistiksel olarak anlamlı bir fark var
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var
SVM - RF:
İstatistiksel olarak anlamlı bir fark var
LR - KNN:
İstatistiksel olarak anlamlı bir fark var
LR - RF:
İstatistiksel olarak anlamlı bir fark yok
KNN - RF:
İstatistiksel olarak anlamlı bir fark var
```

K=4000:

T-test Sonuçları:	T-test Sonuçları:	T-test Sonuçları:
GB - SVM:	GB - SVM:	GB - SVM:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
GB - LR:	GB - LR:	GB - LR:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
GB - KNN:	GB - KNN:	GB - KNN:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
GB - RF:	GB - RF:	GB - RF:
İstatistiksel olarak anlamlı bir fark yok.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark yok.
SVM - LR:	SVM - LR:	SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.	İstatistiksel olarak anlamlı bir fark yok.	İstatistiksel olarak anlamlı bir fark var.
SVM - KNN:	SVM - KNN:	SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
SVM - RF:	SVM - RF:	SVM - RF:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
LR - KNN:	LR - KNN:	LR - KNN:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.
LR - RF:	LR - RF:	LR - RF:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark yok.	İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:	KNN - RF:	KNN - RF:
İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.	İstatistiksel olarak anlamlı bir fark var.

Normalizasyon

Bu çalışma kapsamında normalizasyon işlemi için MaxAbsScaler algoritması kullanılmıştır. Normalizasyon işlemi, SVM – LR ve LR – RF algoritmaları arasında istatistiksel olarak anlamlı bir fark oluşmasını sağlamıştır ancak GBC – SVM algoritmaları arasında bir fark oluşturamamıştır. Aşağıda bu normalizasyonlu halinin ve normalizasyonsuz halinin t-test sonuçlarını inceleyebilirsiniz.

Normalizasyonlu:

GBC - SVM:
İstatistiksel olarak anlamlı bir fark yok.
GBC - LR:
İstatistiksel olarak anlamlı bir fark var.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark var.
GBC - RF:
İstatistiksel olarak anlamlı bir fark var.
SVM - LR:
İstatistiksel olarak anlamlı bir fark var.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark var.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.

Normalizasyon uygulanmadan :

GBC - SVM:
İstatistiksel olarak anlamlı bir fark var.
GBC - LR:
İstatistiksel olarak anlamlı bir fark var.
GBC - KNN:
İstatistiksel olarak anlamlı bir fark yok.
GBC - RF:
İstatistiksel olarak anlamlı bir fark var.
SVM - LR:
İstatistiksel olarak anlamlı bir fark yok.
SVM - KNN:
İstatistiksel olarak anlamlı bir fark var.
SVM - RF:
İstatistiksel olarak anlamlı bir fark var.
LR - KNN:
İstatistiksel olarak anlamlı bir fark var.
LR - RF:
İstatistiksel olarak anlamlı bir fark yok.
KNN - RF:
İstatistiksel olarak anlamlı bir fark var.