Makine Öğrenimi Modeli Saldırıları için Otokodlayıcı ile Saldırı Dayanıklılığının Arttırılması
Incrementing Adversarial Robustness with Autoencoding for Machine Learning Model Attacks

Samed Sivaslıoğlu
TÜBİTAK BİLGEM
Kocaeli, Türkiye
samed.sivaslioglu@tubitak.gov.tr

Ferhat Özgür Çatak
TÜBİTAK BİLGEM
Siber Güvenlik Enstitüsü
Kocaeli, Türkiye
ozgur.catak@tubitak.gov.tr

Ensar Gül
Bilgi Güvenliği Mühendisliği
Istanbul Şehir Üniversitesi
İstanbul, Türkiye
ensargul@sehir.edu.tr

***Özetçe* —Günümüzde makine öğrenimi çok geniş bir alanda kullanılmaya başlamıştır. Makine öğrenimini yanıltmaya yönelik bir çok saldırı da ortaya çıkmaya başlamıştır. Hatalı sınıflandırma, karar mekanizmalarını bozma, filtrelerden kaçma gibi bir çok sonuca sebep olan makine öğrenimi modeli saldırılarına karşı yapılmış olan bu çalışmada, otokodlama ile makine öğrenim modellerinin nasıl dayanıklı hale getirileceği, Mnist veri kümesi ile eğitilmiş modele hedeflenmemiş saldırılar yapılarak gösterilmiştir. Bu çalışma kapsamında, makine öğrenimi güvenlik bileşenlerine en yaygın ve önemlisi olan hedeflenmemiş saldırıya karşı elde edilen sonuçlar ve gelişimi sunulmuştur.**

Anahtar Kelimeler—Otokodlayıcı, Makine öğrenimi, Saldırı dayanıklılığı, Düşmancıl saldırılar.

***Abstract*—Nowadays, machine learning is being used widely. There have also been attacks towards machine learning process. In this study, robustness against machine learning model attacks which cause many results such as misclassification, disruption of decision mechanisms and avoidance of filters has been shown by autoencoding and with non-targeted attacks to a model trained with Mnist dataset. In this work, the results and improvements for the most common and important attack method, non-targeted attack are presented.**

Keywords—Autoencoder, Machine learning, Adversarial robustness, Adversarial attacks

I. GİRİŞ

Makine öğrenimi günümüzde kendini bir çok alanda geliştirmiştir. Günlük hayatımızda görsel sınıflandırma, tavsiye verme, ses ve yüz tanımlama, oyun oynama ve insanların yaptığı davranışları diğer bir çok davranışı sergileyerek yer almaktalardır. Makine öğreniminin böyle başarılar elde edip yaygınlaşması da bu konuda olan çalışmaları motive ettiği gibi, bu konuya karşı saldırı çalışmalarını da motive etmektedir. Olumsuz veri örneklerin varlığı ya da modele yapılmış bir saldırı, en gelişmiş modellerin bile yapacakları tahmin ve sınıflandırmalarda yanlış sonuçlar üretmelerine sebep olmuştur.

Büyük veriler ile çalışılan bu teknolojide, siber güvenliğin diğer alanlarına göre saldırıları insanın gözle görebilmesi de zordur. Bu yüzden bu saldırılara karşı dayanıklı makine öğrenimi bileşenleri oluşturmak oldukça önemlidir. Bu alanda uzun çalışmalar yapılmış ve bu çalışmalar saldırılara karşı dayanıklılıkların çok dirençli olmadıkları görülmüştür [1], [2]. Öyle ki bu çalışmalarda başarılı olanlar bile belirli bir saldırı yöntemi kümesine karşı başarı göstermiş iken genel olarak tam bir koruma sağlayamamışlardır [3].

Bu alanın öneminden de bahsetmek istersek günümüzdeki makine öğrenimi hızla yaygınlaşırken; istenmeyen ve kimlik çalma amaçlı e-postaların filtrelerinden kaçma, şirketlerin kurduğu siber güvenlik filtrelerine takılmama gibi durumlardan kendini süren bir arabanın ya da uçakların sensör verilerini zehirlemeye kadar bir çok tehlike ortaya çıkmaktadır. Bu sistemlerde gidilebilecek olan yol parametrelerine göre yapılan değerlendirmeler bozulduğunda ya da bu parametrelerin çeşitli varyasyonlarını çıkartarak, sistemin kullandığı algoritma tarafından gidilebilecek bir yol gibi gözüken ama tehlikeli olan güzergahlar kurulması gibi felaket senaryoları ortaya çıkabilir.

Araştırmacılar makine öğrenimindeki bu sorunu çözmek için yaptıkları araştırmaları "Düşmancıl Makine Öğrenimi" adı altında topladılar ve çeşitli öneriler sundular [4]. Bu çalışmalarda araştırmacılar, algoritmaların tasarımlarındaki temel denge ve bu dengeyi etkileyecek olumsuzluklara karşı dirençli yeni algoritmalar ve yöntemler üzerinde uğraşmışlardır.

Biz de bu çalışmamızda, bahsedildiği gibi sadece spesifik saldırılara karşı değil, saldırılara karşı genel bir direnç gösterecek bir yöntem bulmayı amaçladık. Ve bu amaç doğrultusunda gelecek olan verileri otokodlayıcıdan geçecek şekilde eğitim modeline gitmesini sağladık. Bu modele yapılacak olan saldırılar ise "hedeflenmemiş" türde olacak olan saldırılardır. Otokodlayıcının yaptığı işlemleri insan gözü tarafından da anlaşılacağı şekilde gözlemleyebilmek için insanların el yazılarıyla yazılmış olan rakamlardan oluşan Mnist veri kümesini kullanmayı seçtik.

II. İLGİLİ ÇALIŞMALAR

Günümüzde saldırıların artmasıyla bu konuda önlem oluşturma amaçlı da bir çok çalışma yapılmıştır. Düşmancıl Makine Öğrenimi tanımlanırken karşı önlemler olarak veri steril-

978-1-7281-1904-5/19/$31.00

liği ve öğrenim dayanıklılığı tavsiye edilmiştir [4]. Bu doğrultudaki çalışmaların büyük bir bölümü konulara odaklanarak yapılmıştır. Bo Li ve Yevgeniy Vorobeychik'in yapmış olduğu çalışma ikili tabanlara ve sınıflandırmalara odaklanmıştır. Stackelberg Oyunu Çoklu Düşmancıl Model algoritmasıyla ve düşmancıl örneklerle modeli yeniden eğitme algoritmasıyla çalışmışlardır [5]. Aynı şekilde Xiao ve arkadaşları yönlendirilmiş doğrusal birime (RELU) karşı dayanıklılık eğitiminin hızını arttırmaya çalışmışlardır [6]. Yu ve arkadaşları ise sinirsel ağın özelliklerini, düşmancıl saldırılar altında doğruluğunu düşmancıl saldırılar ile test edilmeden değerlendirebilecek bir çalışma yapmışlardır. Bu çalışmada girdi uzayı ile düşmancıl örnekler arasında bağlantı olduğu gösterilmiş ve sinir ağının düşmancıl sağlamlığının bir göstergesi olarak ağ sağlamlığı ile karar yüzeyi geometrisi arasındaki bağlantıyı belirmişlerdir [7].

Mardy, Makelov ve arkadaşları ise yapay sinir ağları düşmancıllığa karşı optimizasyonlar ile dayanıklı hale getirmeye çalışmışlardır ve doğruluk oranlarını farklı yöntemlerle arttırmışlardır [3]. Pinto ve arkadaşları destekli öğrenme yöntemiyle bu sorunu çözmeye çalışmışlardır. Yaptıkları çalışmada öğrenimi sıfır toplamlı, minimum hedef fonksiyonu olarak formüle etmişlerdir . Düşmancıllığa dayanıklı destekli öğrenme ile eğitim başlangıcında dayanıklı, daha iyi genellemeler yapan ve eğitim ile test koşulları arasındaki değişikliklerden az etkilenen, test ortamında eğitim sırasında modellemesi zor olan rahatsızlıklara karşı dayanıklı olduğunu göstermişlerdir [8] . Carlini ve Wagner hedeflenmiş ve hedeflenmemiş saldırılar üzerinden çalışarak güçlü bir saldırı ile makine öğrenim modelinin dayanıklılığının öz mantığının yenilebileceğini, ve bu tipteki saldırıların genellikle potansiyel savunmaların etkinliğini değerlendirmek için kullanılabileceğini göstermişlerdir. Makine öğrenimindeki düşmancıl saldırılara karşı bahsettikleri güçlü saldırı yanı sıra güvenli olmayan bir modelden üretilmiş yüksek güvenilirliğe sahip düşmancıl örnekler oluşturarak, güvenli modele aktarmada başarısız olacakları sunmayı tavsiye etmişlerdir [1] . Harding ve arkadaşları da benzer şekilde hedeflenmiş ve hedeflenmemiş saldırılardan üretilen düşmancıl örneklerin insanların kararlarına etkilerini incelemişlerdir. Ve hedeflenmemiş örneklerin, insan algısı ve kategorizasyon kararlarına hedefli örneklerden daha fazla müdahale ettiğini göstermişlerdir [9].

III. Önbilgiler

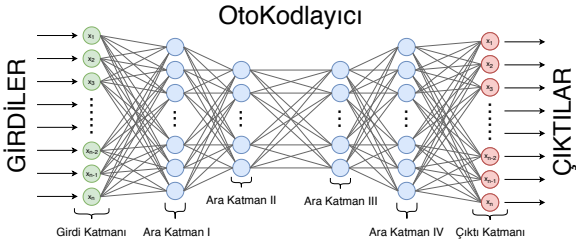
A. Hedeflenmemiş Saldırılar

Hedeflenmemiş saldırıların odağı, sınıflandırıcının yanlış sonuç vermesidir. Diğer bir deyişle, saldırı yapmak için savunmasız bileşenlerin varlığı yeterlidir. Belirli bir grup ya da sistem belirtilmeksizin yapılan bu saldırılar iyi geliştirdikleri taktirde çok daha fazla kümeye tehdit oluşturduğu için daha tehlikelidir. Bu sebepten dolayı da araştırmalarda daha çok tercih edilmiştir. Biz de bu çalışmada hedeflenmemiş saldırılara odaklanmayı seçtik.

B. Beyaz Kutu Testi

Açık kutu olarak da bilinen beyaz kutu testleri, bileşenin veya sistemin iç yapısının analizine dayanarak test etme üzerinedir. Test edilen öğenin iç yapısı, tasarımı, uygulaması bilinmektedir. Programlama bilgisi ve uygulama bilgisi esastır. Beyaz kutu testleri, hem iç hem de dış güvenlik açıklarının kapsamlı bir değerlendirmesini sağlar ve hesaplamalı testler için en iyi seçimdir. Biz de bu sebepten ötürü çalışmamızda bu yöntemi seçtik.

C. Otokodlayıcı

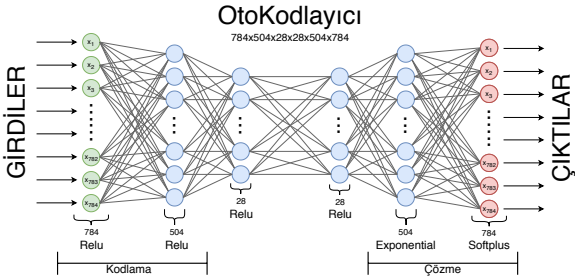


Şekil 1: Otokodlayıcı Katman Yapısı

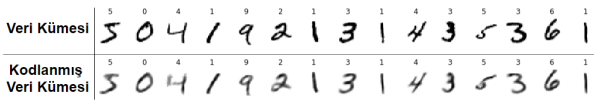
Bir otomatik kodlayıcı sinir ağı, hedef değerleri girdilere eşit olacak şekilde ayarlayan, geri yayılma uygulayan denetimsiz bir öğrenme algoritmasıdır [10] . Otokodlayıcılar, tür olarak üretici modellerdir. Aldığı girdiler ile bu girdilerin sonuçlarını bilmeden üretim sağlayabilirler. Normal bir modelin kullanımı, `model.fit(X,Y)` şeklinde olurken otokodlayıcıda `model.fit(X,X)` şeklinde verilmektedir. Otokodlayıcı, **x** girdilerine uygun olan **x** çıktılarını almak için kimlik fonksiyonu ile çalışmaktadır. Kimlik fonksiyonu, öğrenmeye çalışmak için özellikle önemsiz bir işlev gibi görünmektedir; ancak ağ üzerindeki gizli birimlerin sayısını sınırlamak gibi kısıtlamalar koyarak, verilerle ilgili ilginç bir yapı oluşmaktadır [10]. Biz de bu çalışmamızda otokodlayıcı kullanarak girdilerden yeni girdiler ürettik, ve bu girdiler üzerinden hedeflenmemiş saldırıyı gerçekleştirdik.

IV. Yöntem

A. Otokodlayıcı Modelini Oluşturma

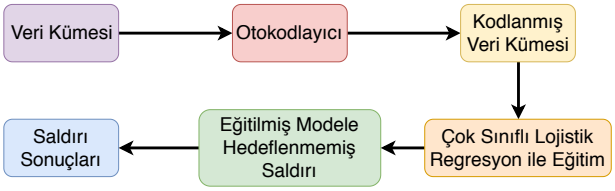


Şekil 2: Otokodlayıcı Aktivasyon Fonksiyonları



Şekil 3: Normal ve Kodlanmış Veri Karşılaştırması

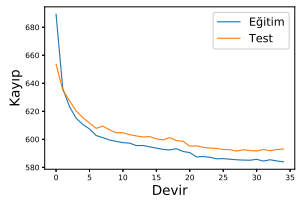
Mnist veri kümesiyle çalışıldığı için otokodlayıcı modelindeki katman yapısı Mnist veri kümesine uyması amacıyla 28 katları olarak seçilmiştir. Bu yapı Şekil 2'de gösterilmiştir.



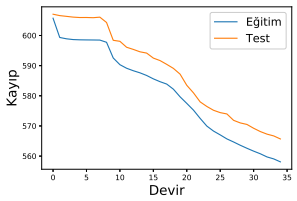
Şekil 4: Süreç Diyagramı

Otokodlayıcı sayesinde değişmiş olan Mnist verilerimiz Şekil 3'te gösterilmiştir. Eğitimde Mnist veri kümesi yerine, bu kodlanmış veriler kullanılmışlardır. Eğitim yöntemi olarak çok sınıflı lojistik regresyon yöntemi seçilmiş ve hedeflenmemiş saldırı bu modele gerçekleştirilmiştir. Süreç diyagramı Şekil 4'te verilmiştir.

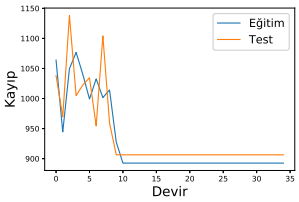
B. Aktivasyon Fonksiyonu Seçimi



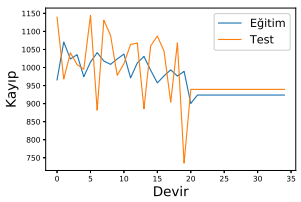
(a) Relu Kayıp Tarihçesi



(b) Sigmoid Kayıp Tarihçesi



(c) Softsign Kayıp Tarihçesi



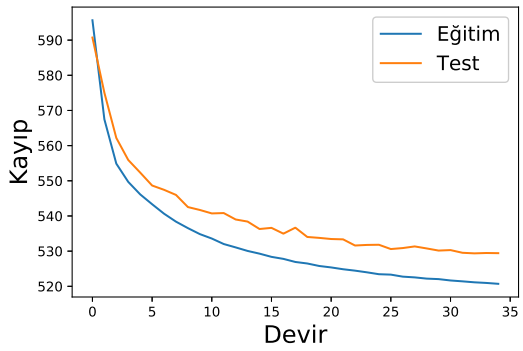
(d) Tanh Kayıp Tarihçesi

Şekil 5: Farklı aktivasyon fonksiyonlarının kayıp tarihçeleri

Farklı aktivasyon fonksiyonlarıyla kayıp değerleri karşılaştırılmıştır. Karşılaştırma sonuçları Şekil 5'te gösterilmiştir. Bu değerler arasında sigmoid ve relu en iyi sonuçları vermektedir. Sigmoid düşük devirlerde relu'ya göre daha çok kayıp vermiştir fakat sonrasında daha iyi sonuçlar vermiştir. Bu yüzden her iki katmanda bulunan aktivasyon fonksiyonunun en iyi sonucuna ulaşmak hedeflenmiştir. En az kayıp değerine sahip olan model, relu fonksiyonuyla kodlama kısımlarının yapılması ve çözümleme kısmında ise sırasıyla exponential ve softplus fonksiyonlarının kullanılması şeklinde olmuştur. Şekil 6'da bu kayıp fonksiyonunun sonucu, Şekil 2'de de aktivasyon fonksiyonlarıyla modelin yapısı gösterilmiştir.

V. Deneysel Sonuçlar

Çalışmamızda, NIPS 2017'de geliştirilmiş olan çok sınıflı sınıflandırma modellerine karşı yapılan saldırı yöntemi kullanılmıştır [11]. Yapılan saldırı şiddetini belirlemek amacıyla bir epsilon değeri kullanılmaktadır. Otokodlayıcı olmadan, Mnist veri kümesi tarafından eğitilmiş ve çok sınıflı lojistik regresyon ile eğitilmiş bir modele gerçekleşmiş bir saldırı durumunu ele aldığımızda Şekil 7'de gösterilen karışıklık matrisi elde edilmektedir.



Şekil 6: Optimize edilmiş Relu Kayıp Tarihçesi

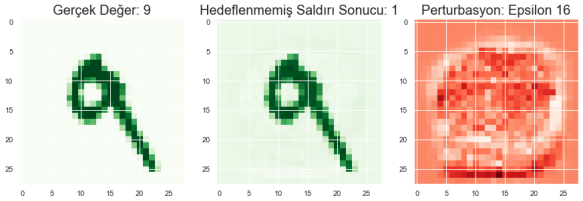
		Tahmin Değerleri									
		0	1	2	3	4	5	6	7	8	9
Gerçek Değerler	0	276	0	17	31	10	62	20	15	9	3
	1	0	0	25	6	0	11	0	11	21	16
	2	61	47	96	70	225	35	277	297	203	177
	3	29	248	217	37	167	84	31	192	480	179
	4	1	0	24	36	52	60	20	16	2	151
	5	467	67	53	332	49	10	389	204	64	194
	6	71	0	90	42	61	73	103	4	34	1
	7	29	294	66	15	61	46	48	53	43	18
	8	16	486	339	389	38	424	28	73	22	266
	9	57	9	72	68	312	45	48	156	119	5

Şekil 7: Otokodlayıcı olmadan modele yapılmış saldırının karışıklık matrisi

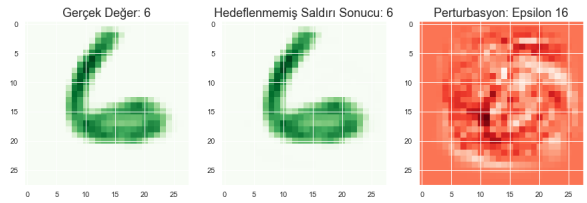
		Tahmin Değerleri									
		0	1	2	3	4	5	6	7	8	9
Gerçek Değerler	0	998	0	0	1	0	1	0	1	0	0
	1	0	1145	2	0	0	0	0	1	1	0
	2	0	0	1015	0	0	2	0	1	0	0
	3	0	0	1	967	0	19	0	0	0	0
	4	0	0	2	0	970	0	0	2	0	2
	5	0	0	0	0	0	847	0	0	0	0
	6	1	0	1	0	0	6	962	0	1	0
	7	0	0	0	0	0	0	0	1004	0	0
	8	1	0	8	10	0	22	0	1	967	0
	9	0	0	0	1	5	2	0	7	4	1019

Şekil 8: Otokodlayıcıdan geçmiş veriden eğitilen modele yapılmış saldırının karışıklık matrisi

Şekil 11'de verilmiş olan Epsilon değer grafiğinde, Epsilon 16 olarak seçilip saldırı sonucu verinin değişimini ve perturbasyonu Şekil 9'da gösterilmiştir.

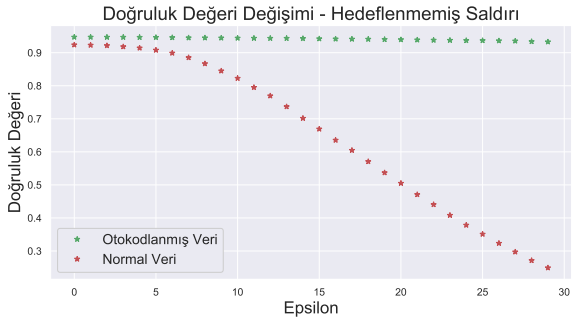


Şekil 9: Otokodlayıcı olmadığında yapılan saldırı sonucunda değer değişimi ve perturbasyon

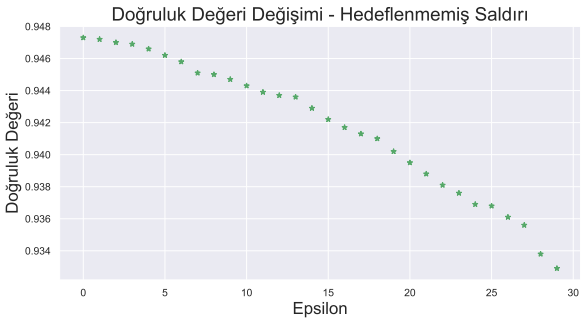


Şekil 10: Otokodlayıcı olduğunda yapılan saldırı sonucunda değer değişimi ve perturbasyon

Otokodlayıcı tarafından üretilen verilere yapılan saldırı ile Mnist veri kümesinin verilerine yapılmış hedeflenmemiş saldırı sonuçları ile otokodlayıcılı verilere yapılmış saldırının detaylandırılmış grafiği Şekil 11 - 12'de verilmiştir. Otokodlayıcı sonrası Mnist veri kümesindeki değişimler de Şekil 3'te gözlemlenebilir. Otokodlayıcının olduğu, saldırıda epsilon 16 değerindeki bir verinin değer değişimine ve perturbasyonuna Şekil ÞermutasyonAE'den gözlemleyebiliriz.



Şekil 11: Otokodlanmış ve Kodlanmamış Veri kümesi karşılaştırması



Şekil 12: Otokodlanmış Veri Kümesi Detaylı Grafiği

Şekil 4'te gösterilmiş olan süreçteki olduğu gibi otokodlayıcıyı kullanıldığında, veri kümesini önce otokodlayıcıdan geçirilip sonrasında çıkan kodlanmış veri kümesi kullanılarak çok sınıflı lojistik regresyon eğitimiyle sınıflandırma modeli oluşturulmakatdır. Bu oluşan modele hedeflenmemiş saldırı gerçekleştirildiği zaman oluşan iyileşme, Şekil 8'de karışıklık matrisinde gösterilmektedir. Karışıklık matrisi incelendiği zaman sınıflandırma peformansında çok fazla değişiklik olmadığı gözlemlenmiştir.

VI. Sonuç ve Gelecek Çalışmalar

Otokodlayıcıların kullanımıyla elde edilen deneysel sonuçları incelediğimizde, Bölüm V'de anlatılan Google Brains [11] yayınında bulunan hedeflenmemiş saldırıya karşı başarılı olduğu gözlemlenmektedir. Bu çalışma kapsamında kullanılan veri kümesi şu ana kadar karşılaştırma için kullanılan Mnist veri kümesidir. Gelecek çalışmalarımızda, farklı veri kümeleri üzerinde önerilen yöntemin test edilmesi ve otokodlayıcıların diğer düşmancıl makine öğrenimi model saldırılarına karşı ne kadar etkili olduğu araştırılacaktır. Hedefli saldırılara karşı alınması gereken yöntemler de incelenecektir.

Kaynaklar

[1] Towards Evaluating the Robustness of Neural Networks Nicholas Carlini, David Wagner *arXiv:1608.04644*, 2016

[2] Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples Anish Athalye, Nicholas Carlini, and David Wagner *arXiv:1802.00420*, 2018

[3] Towards Deep Learning Models Resistant to Adversarial Attacks Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, Adrian Vladu *arXiv:1706.06083*, 2017

[4] Adversarial machine learning Huang, Joseph, Nelson, Rubinstein, Tygar *doi:10.1145/2046684.2046692*, 2011

[5] Evasion-Robust Classification on Binary Domains BO LI, University of California, Berkeley Yevgeniy Vorobeychik, Vanderbilt University *doi:10.1145/3186282*, 2018

[6] Training for Faster Adversarial Robustness Verification via Inducing ReLU Stability Kai Y. Xiao, Vincent Tjeng, Nur Muhammad Shafiullah, Aleksander Madry *arXiv:1809.03008*, 2018

[7] Interpreting Adversarial Robustness: A View from Decision Surface in Input Space Fuxun Yu, Chenchen Liu, Yanzhi Wang, Liang Zhao, Xiang Chen *arXiv:1810.00144*, 2018

[8] Robust Adversarial Reinforcement Learning Lerrel Pinto, James Davidson, Rahul Sukthankar, Abhinav Gupta *arXiv:1703.02702*, 2017

[9] Human Decisions on Targeted and Non-Targeted Adversarial Samples Samuel M. Harding, Prashanth Rajivan, Bennett I. Bertenthal, Cleotilde Gonzalez, 2018

[10] Understanding Autoencoders with Information Theoretic Concepts Shujian Yu, Jose C. Principe *arXiv:1804.00057*, 2018

[11] Adversarial Attacks and Defences Competition Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, Alan Yuille, Sangxia Huang, Yao Zhao, Yuzhe Zhao, Zhonglin Han, Junjiajia Long, Yerkebulan Berdibekov, Takuya Akiba, Seiya Tokui, Motoki Abe *arXiv:1804.00097*, 2018