Säkerhet i AI-system (DV2607)

[Rapport för inlämningsuppgift]

Namn student 1: Ahmad Tarshehani
E-post student 1: ahta19@student.bth.se

Namn student 2: Samer Albardan
E-post student 2: saab19@student.bth.se

Del 1: Data poisoning attacker

1.1) Baseline-prestanda

Model classification performance report	
Algorithm	Random Forest
Last trained	2023-11-7 21:07
Trained on	40200 entries
Accuracy	81.26%
Precision	79.31%
Recall	81.26%
F1-score	79.44%
ROC-AUC	65.38%

1.2) Attack

1.2.1) Beskrivning

I vårt försök att utföra en dataförgiftningsattack på ML-modellen analyserade vi datauppsättningen strategiskt för att identifiera det mest effektfulla attributet att manipulera. Analysen syftade till att upptäcka vilken modifiering som skulle få de mest betydande återverkningarna på modellens prestanda. Baserat på vår bedömning fastställde vi att inriktning på attributet "default payment next month" skulle vara den mest effektiva metoden.

Vårt strategiska tillvägagångssätt är att släppa alla kolumner för 'x' och sedan manipulera 'y_pred'-arrayen genom att invertera labels för "default payment next month" göra träningen på 100 beslutsträd och två beslutsnivåer med RandomForestClassifier. Detta tillvägagångssätt har en betydande inverkan på inlärningsprocessen, och betonar att även enkla strategier effektivt kan störa modellens förståelse.

1.2.2) Resultat

Model classification performance report	
Algorithm	Random Forest
Last trained	2023-11-26 16:12
Trained on	40200 entries
Accuracy	28.98%
Precision	61.69%
Recall	28.98%
F1-score	26.2%
ROC-AUC	47.66%

Attacken har framgångsrikt lagt till 20 100 instanser till träningsdatan utan några misslyckade tillägg.

1.3) Skyddsåtgärder mot data Poisoning

De två mest lämpade skyddsåtgärderna tagit från artikeln ” Machine Learning Security against Data Poisoning: Are We There Yet”.

- **Datarengöring**
  Datarengöring är en viktig process för att förbättra datakvaliteten och minska risken för data poisoning attack. Det finns många stegar för att ha väldigt bra datarengöring som:
  a. Identifiera felaktiga datapunkter genom statistisk analys och det tillämpad genom att analysera datapunkternas statistiska egenskaper, t.ex. medelvärde, median, standardavvikelse och interkvartilomfång. Identifiera datapunkter som ligger utanför det normala intervallet och sen visualisera datan med hjälp av diagram och histogram för att identifiera avvikande datapunkter och använd domänkunskap för att identifiera datapunkter som inte är rimliga eller troliga.
  b. Ta bort felaktiga datapunkter genom att identifiera och ta bort felaktiga datapunkter manuellt eller att använd automatiserade metoder för att identifiera och ta bort felaktiga datapunkter.
  c. Validering av datarengöring genom att analysera datasetet efter datarengöring för att säkerställa att inga legitima datapunkter har tagits bort.

  Fördelar med datarengöring:
  - Datarengöring kan förbättra datakvaliteten och göra den mer tillförlitlig.
  - Datarengöring kan leda till bättre prestanda för maskininlärningsmodeller.
  - Datarengöring kan minska risken för att maskininlärningsmodeller påverkas av dataförgiftningsattacker.
- **Robusta inlärningsalgoritmer**
  Robusta inlärningsalgoritmer är utformade för att vara motståndskraftiga mot data poisoning attack. De kan uppnå detta genom att använda olika tekniker, till exempel:
  a. Identifiering av avvikande datapunkter och det händer genom använd algoritmer till exempel som Local Outlier Factor (LOF), Isolation Forest och One-Class Support Vector Machines (SVM) för att identifiera avvikande datapunkter. Sedan, analysera datapunkternas fördelning och identifiera datapunkter som ligger utanför den normala fördelningen. Använd ensemble learning-metoder för att kombinera flera

modeller och identifiera datapunkter som de flesta modellerna anser vara avvikande.

b. Dataförstärkning genom att minska antalet datapunkter i överrepresenterade klasser genom att ta bort datapunkter slumpmässigt eller med hjälp av viktade metoder.
c. Det viktig och ha en ensemble learning för få bättre skydd och det händer genom att träna flera modeller på olika delmängder av datan och kombinera deras prediktioner eller träna modeller sekventiellt och vikta varje modells prediktion baserat på dess prestanda på den föregående modellen.
säkerställa att inga legitima datapunkter har tagits bort.

Fördelar med robusta inlärningsalgoritmer:

- Kan ge starkt skydd mot dataförgiftningsattacker.
- Kräver ingen manuell inblandning.
- Olika tekniker kan användas för att skydda mot specifika attacker.
- En av de bästa tekniker som finns nu för att detektera poisoned data

1.4) Riskanalys av ML-system

Risknamn	Riskvärde	Sannolikhet	Konsekvens	Beskrivning	Skyddsmekanism
Hårdvarufel	12	3	4	Hårdvaran skadas och modellen blir otillgänglig.	Fysisk säkerhet, redundans.
Programvarufel	10	2	5	Programvaran innehåller en bugg som leder till felaktiga prediktioner.	Säkerhetskodning, testning, uppdatering.
Dataförgiftning	16	4	4	Angriparen manipulerar träningsdata för att påverka modellens prestanda.	Dataförberedelse, robusta inlärningsalgoritmer, modellövervakning.
Bias i data	12	3	4	Träningsdata innehåller oavsiktlig bias som påverkar modellens prestanda.	Datainsamlingsprocesser, dataförberedelse, algoritmval.
Bristande datakvalite	8	2	4	Träningsdata innehåller fel, inkonsekvenser eller saknade värden.	Datainsamlingsprocesser, dataförberedelse, datakvalitetskontroll.
Överanpassning	8	2	4	Modellen lär sig för mycket av träningsdatan och presterar dåligt på ny data.	Regularisering, dropout, tidig stoppning.

Underanpassning	8	2	4	Modellen lär sig inte tillräckligt av träningsdatan och presterar dåligt på all data.	Mer data, bättre funktioner, mer komplexa modeller.
Algoritmfel	10	2	5	Algoritmen som används i modellen är felaktig eller olämplig.	Algoritmval, modellvalidering, expertgranskning.

Del 2: Adversarial input attacker

Del 2.1) Beskrivning av adversarial input attacker

Vad är Adversarial Input Attack?

En **Adversarial Input Attack** är en teknik där man lurar en maskininlärningsmodell genom att ge den manipulerade indata. Dessa manipulerade indata, även kallade **adversariella exempel**, ser oskyldiga ut för en människa men kan få modellen att göra felaktiga förutsägelser.

Till exempel en självkörande bil som använder en maskininlärningsmodell för att identifiera trafiksignaler. En angripare kan skapa ett adversariellt exempel genom att lägga till små, omärkliga förändringar till en bild av ett stoppskylt. Dessa förändringar kan få modellen att misstolka stoppskylten som en fartbegränsningsskylt, vilket kan leda till en olycka.

Det finns två huvudsakliga typer av Adversarial Input Attacks:

- **White box attacker:** I en white box attack har angriparen fullständig information om maskininlärningsmodellen, inklusive dess arkitektur, parametrar och träningsdata. Detta gör det möjligt för angriparen att skapa mer sofistikerade adversariella exempel som är mer benägna att lura modellen.
- **Black box attacker:** I en black box attack har angriparen inte tillgång till någon information om maskininlärningsmodellen. Istället måste de använda sig av trial-and-error-metoder för att skapa adversariella exempel.
- **Targeted attacks:** Dessa attacker är riktade mot en specifik klass av indata. Målet är att få modellen att felaktigt klassificera ett indata av den specifika klassen.
- **Untargeted attacks:** Dessa attacker är inte riktade mot en specifik klass av indata. Målet är att få modellen att göra felaktiga förutsägelser oavsett indata.

Adversarial Input Attacks Typer

- **Limited-memory BFGS (L-BFGS):** En optimeringsalgoritm baserad på gradienten som minimerar antalet störningar som läggs till i bilder för att lura modellen.
- **FastGradient Sign Method (FGSM):** En enkel och snabb metod baserad på gradienten som lägger till små störningar till indata för att orsaka felklassificering.
- **Deepfool Attack:** Syftar till att minimera det euklidiska avståndet mellan stört prov och originalprov, och orsakar felklassificering med färre störningar.

Skydd mot Adversarial Input Attacks

För att skydda mot Adversarial Input Attacks som Limited-memory BFGS (L-BFGS), FastGradient Sign Method (FGSM), och Deepfool Attack kan flera strategier implementeras.

- **Adversarial Training:** Adversrial training är en träningsprocessen kan modellen exponeras för adversarial examples som genererats med L-BFGS, FGSM, eller Deepfool. Genom att göra detta kan modellen lära sig att identifiera och motstå dessa attacker. Dessa skydd kan öka modellens robusthet mot adversarial attacks genom att göra den mer mottaglig för små förändringar i indata.

- **Robust Loss Function:** den metoden kan tillämpas genom att Välja förlustfunktioner som inte bara fokuserar på minimering av fel på legitima data, utan också på att förbättra modellens robusthet. Denna skydd hjälper till att minska modellens vulnerability för attacker genom att göra det svårare för attacker att utnyttja små förändringar i indata.
- **Defensive Distillation:** Den metoden tillämpas genom att träna en student modell på utdata från en master modell som är robust mot adversarial attacks, kan man utnyttja master modellens förmåga att identifiera och motstå attacker, samtidigt som student modellen är mer robust mot sådana attacker. Denna skydd kan ge en bra balans mellan robusthet och prestanda, eftersom student modellen är mer robust mot attacker samtidigt som den behåller en god prestanda på legitima data.

Del 2.3) Säkerhetsåtgärder

För att skydda mot Boundary Attack, en effektiv metod är att använda en kombination av robusta försvarsmekanismer som inkluderar adversarial training, regularisering, och robusta loss funktioner. Dessa tekniker är viktiga eftersom de hjälper till att förbättra modellens förmåga att identifiera och motstå attacker som manipulerar gränserna för prediction intervallen.

Adversarial training är en metod där modellen tränas på korrupt exempel (adversarial examples) för att göra den mer robust mot attacker. Genom att inkludera korrupt exempel i träningsuppsättningen eller genom att använda tekniker som Mixup eller Fast Gradient Sign Method (FGSM) för att generera skadlig exempel under träningen, kan modellen lära sig att känna igen och motstå dessa attacker. Detta är särskilt effektivt mot Boundary Attack, eftersom det fokuserar på att förbättra modellens förmåga att identifiera och motstå attacker som manipulerar gränserna för förutsägelseintervallen (Madry et al., 2017).

Regularisering är en annan viktig försvarsmekanism som kan hjälpa till att förhindra överanpassning, vilket är en vanlig svaghet i maskininlärningsmodeller. Genom att tvinga modellen att ha enklare representationer av data kan regularisering minska modellens känslighet för små förändringar i inmatningen, vilket gör den mindre vulnerable för Boundary Attack. Tekniker som L1 eller L2-regularisering kan effektivt användas för att uppnå detta (Srivastava et al., 2014).

Robusta robusta loss funktioner är utformade för att vara mindre känsliga för små förändringar i inmatningen, vilket gör dem mer robusta mot attacker som Boundary Attack. Genom att använda en robust loss funktion kan modellen fortsätta att ge korrekta predictions även när inmatningen manipuleras. Exempel på sådana loss funktioner inkluderar Total Variation Distortion (TVD) eller den robusta loss funktion som introducerades av Madry et al. (2017).

Referenser

1. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. In Proceedings of the International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1706.06083
2. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research. https://jmlr.org/papers/v15/srivastava14a.html
3. Goodfellow, I., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Proceedings of the IEEE International Conference on Computer Vision. https://arxiv.org/abs/1412.6572