Assessment 7

SIT 719

Attacks that are often encountered when we consider vulnerabilities due to adversarial techniques can be classified into two categories:

1. When the attack is done during Training phase, these attacks generally tend to alter the training data to corrupt the model.
2. When the attack occurs during inference (testing) phase, these attacks try to create a scenario where model's outcome is altered as compared to the expected outcome.

From these attacks **five** attacks are:

1. **Poisoning Attacks**: Poisoning attacks are also known as Causative attacks. These attacks tend to corrupt the model during training phase, this is generally done by manipulating the data because of which the model training finds samples which cause the training to yield in a low performing model. The data manipulation causes a poor boundary function to be modelled, which results in poorly trained model. Apart from altering or corrupting the training data, poisoning attacks can also change the underlying algorithm of a machine learning model (also known as Logic Corruption), this yields in unexpected model outcomes.

2. **Evasion Attacks**: These attacks are done when the model is deployed. These attacks rely on finding pattern in the output from the trained models hence they need the underlying model or a similar model which can provide information about the change in feature vectors for a given set of inputs. These inputs are then changed iteratively to find patterns leading to change in output till a misclassification is obtained. (This category of evasion attacks is known as gradient-based evasion attacks). These changes are generally not visible to human eye. Another subcategory is gradient free evasion attacks, these attacks just need the output probabilities from the model to find the example that can cause misclassification error.

3. **Extraction Attacks**: Extraction attacks are part of a subcategory of attacks known as Oracle attacks, these attacks are generally used as a steppingstone to build a substitute model which can replicate original model's predictions with output probabilities, this allows the attacker to use the substitute model for evasion attacks.

4. **Inversion Attacks**: These attacks are also part of Oracle attacks, these attacks target input on which the model has been trained. This allows the attacker to infringe on private or proprietary data.

5. **Membership Inference Attack**: These attacks utilize change in confidence values of a trained model. The end goal for these attacks is to find the nature of distribution of data for which the attack is being done, this is done to mimic the training data on which the model is trained.

Methods for defence from these attacks are:

Defence from Poisoning Attacks: Poisoning attacks are prevented by

1. Sanitizing the data, where samples which cause high error rates are rejected from the training set.
2. Another way to prevent these attacks is robust statistics, which utilizes regularization techniques which allow the model to be trained in expected way even though the data has been poisoned.

Also, because poisoning works on the training data, it is advisable to use traditional methods of data protection like encryption to safeguard data from poisoning attacks.

Defence from Evasion, Extraction, Inversion and Membership Inference Attacks: Although these attacks are done when a model is deployed (testing phase), the defence against these attacks is done during training phase. This includes methods like, **Adversarial Training** where a model is trained on adversarial samples i.e., samples which can be used in case of a potential attack, but these samples are used with correct labels. Due to this, the model learns to look beyond the changes that are introduced and does not misclassify the samples. Another defence is **Gradient Masking**, this method tries to reduce the model's sensitivity which leads to fewer misclassification error against adversarial samples. One more method would be to use **Ensemble methods** as opposed to stand alone models, as these models are more robust which makes it harder for an attacker to gain insights into trained model's parameters. A pre-processing step of **Feature Squeezing** can also be used to defend against these attacks as it can weed out potential adversarial noise from input data.

References:

[1] E. Tabassi, K. J. Burns, M. Hadjimichael, A. D. Molina-Markham, and J. T. Sexton, "A taxonomy and terminology of adversarial machine learning," 2019.