Examples for kNN have relatively low transferability to
Ensemble to take advantage of low transferability

Find some adversarial input
does not know target’s model internals or have the
does know target’s architecture and has own dataset

Adversarial Examples generated on one system tend to
generalize well to other systems

Transferability:
● Adversarial Examples generated on one system tend to
generalize well to other systems

Motivation

Ensembling Defense
Adversarial examples built for some model transfer best to
other models of same type
● Examples for kNN have relatively low transferability to
CNN and vice versa
● Ensemble to take advantage of low transferability

Ensembling Defense

Attack Setting
Attacker Knowledge:
● does not know target’s model internals or have the
target database
● does know target’s architecture and has own dataset

Target System
Blackbox Target Model

Attacker System
Substitute Model

Convoluted Neural Network (CNN):
● Substitute model is simple two convolutional layers
followed by two fully connected layers
● Target model is two max pool layers followed by two
fully connected layers, trained with dropout

Ensemble Model (Ens):
● Learns a parameter \( \alpha \) and score examples with
\[ s_{ensemble}(x) = a_{kNN}(x) + (1 - \alpha)s_{CNN}(x) \]

Model

k-Nearest Neighbors (kNN):
● \( k = 5 \) in target, \( k = 3 \) in substitute, both with \( l_2 \)-distance
● In substitute, gradient is approximated with soft-min:
\[ s_k(x) = \frac{\sum_{z \in S(x)} e^{-\|z - x\|^2}}{\sum_{z \in S(x)}} \]

Convolutional Neural Network (CNN):
● Substitute model is simple two convolutional layers
followed by two fully connected layers
● Target model is two max pool layers followed by two
fully connected layers, trained with dropout

Model

Results

Adversarial Success % / Partial Success %

<table>
<thead>
<tr>
<th>Target</th>
<th>kNN</th>
<th>CNN</th>
<th>Ens</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 18.7% / 30.1% )</td>
<td>( 12.0% / 24.3% )</td>
<td>( 11.4% / 19.9% )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Substitute</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>kNN</td>
<td>96.3%</td>
</tr>
<tr>
<td>CNN</td>
<td>96.3%</td>
</tr>
<tr>
<td>Ens</td>
<td>96.1%</td>
</tr>
</tbody>
</table>

Adversarial examples \( x^* \) created for substitute
● success if model(\( x^* \)) = \( t_{adversarial}^* \)
● partial success if model(\( x^* \)) \( \neq t_{true}^* \)

Results show that Ensemble model is more robust to
adversarial examples crafted for kNN and for CNN
Furthermore, Ensemble model is more robust to adversarial
examples crafted for other Ensemble models
Ensembling achieves this without sacrificing test accuracy

Future Work
Future work include examining ensembling as a defense for
adversarial examples crafted using other attack generation
methods, effectiveness with other datasets (CIFAR10),
investigating other ensemble models, and how ensembling
complements other defences like distillation.

References