Plan:

Questions:

Does TMR work in a machine learning environment?
Does NMR increase reliability as we increase N, or does the stochastic nature of machine learning add more noise and thus elss reliability?
Can we improve the robustness of a network using learnings from above?

Hypothesis:

TMR does work in a machine learning environment
As we increase N we get better reliability
ADversarial attacks can be negated using TMR
We are able to increase the reliability of a CNN using what we learnt.

Approach:

Experiment 1:

Question: Does TMR work in a machine learning environment?

Solution:

- Generate 3 CNN networks architectures and train them on the same dataset.
- Generate 100 adversarial images for each of the networks using each of the adversarial generation techniques
- Group all the data into a single “Adversarial” dataset
- Run the data through each of the individual networks
- Run the data through a TMR network

Experiment 2:

Question: Does NMR increase the reliability of networks

Solution:

- Train N more CNN networks using different architectures
- Connect them in NMR fashion and measure their dependability using the same adversarial dataset.

Experiment 3:

Question: Can we improve the robustness of a CNN

Solution:

Current approaches suggest that data augmentation is the key to better robustness

Generate a network architecture A
Train A using a dataset
Run CNN-CERT on this network to get a certified robustness
Run data augmentation on dataset A
Train A using the augmented data
Run CNN-CERT on this network to get a certified robustness
Generate random images and feed them to the TMR system to generate labels and a much larger dataset
Run CNN-CERT on this network to get a certified robustness

Unsound Logic: Why do we assume random noise is a label suggested by the TMR system. We maybe generate another class called random noise and make that a second step. So Network A -> Network A with a random noise class -> Network A with a random noise class and augmented data -> TMR generated noise network

Analysis:

Our technique applied to the udacity network