Introduction
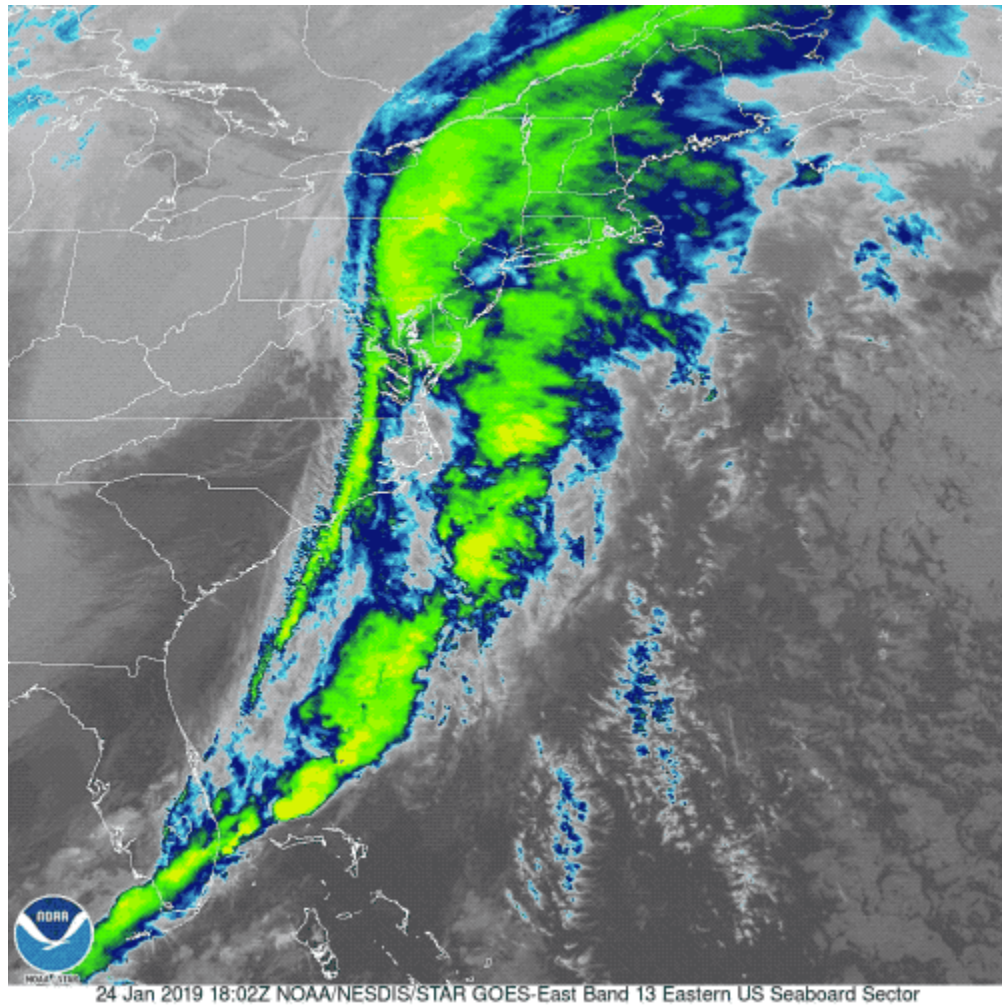
1) We will see how we can generate:

    a) Nowcast Test Data
    b) Synrad Test Data

2) How to use the Generated Test data to predict outcomes

    a) AnalyzeNowcast.ipynb
    b) AnalyzeSyntheticRadar.ipynb

24 Jan 2019 18:02Z NOAA/NESDIS/STAR GOES-East Band 13 Eastern US Seaboard Sector

Requirements:

1) [Make_nowcast_dataset.py](#)
2) [Make_synrad_dataset.py](#)
3) [Nowcast_generator.py](#)
4) [Synrad_generator.py](#)
5) [utils.py](#)

Resources:

1. Repository for the SEVIR nowcast and synrad: [SEVIR](#)
2. Links to the SEVIR research papers: [SEVIR](#), [SEVIR dataset](#)

Team 3: The Avengers

- Ankana Asit Baran Samanta (Wonder Woman) 🦸
- Sreepad Parigi (Iron Man) 🦸
- Parth Shah (Spider Man) 🕸
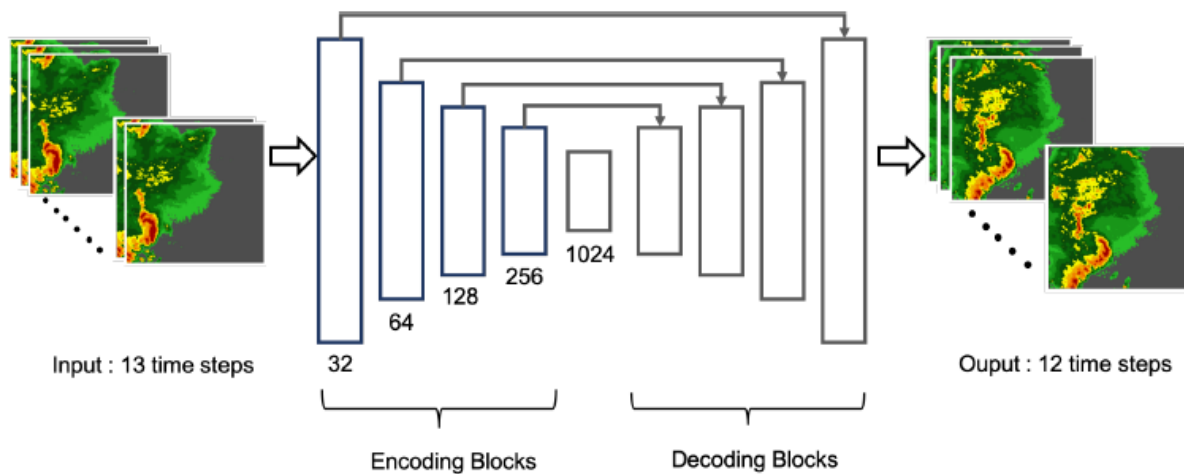
What is Nowcast and Synrad?

Short introduction about Radar nowcasting and Synthetic weather radar

1. Earth's weather is continuously monitored by sensors that collect TBs of data everyday. Weather data is collected through ground based radars NEXRAD and GOES satellites. But the amount of data collected by these sources is too large and isn't "machine learning ready". To address these challenges, SEVIR dataset was designed by 3 MIT scientists.

2. Two use cases of this SEVIR dataset are radar nowcasting(short term weather forecasts) and synthetic weather radar.

3. **Radar nowcasting** is high resolution, short-term weather forecasts of radar echos, precipitation, cloud coverage or other meteorological quantities widely used in public safety, air traffic control and many other areas.

4. This future prediction task takes 13 VIL images sampled at 5 minute intervals as input. The model takes this as input and it produces the next 12 images in the sequence corresponding to the next hour of weather. The

model input is of size Nx384x384x13 pixels and the output is of size Nx384x384x12 pixels.

5. **Synthetic weather radar** is generating radar like imagery of storm depictions using only satellite and lightning as inputs to the model. ir069, ir107 and lght image types are taken as input and these three image types are transformed into vil.

Nowcast workflow (Model architecture)

Input : 13 time steps   32   64   128   256   1024   Ouput : 12 time steps

Encoding Blocks   Decoding Blocks

Generating Test Data

Step 1

Pre-trained models for both nowcast and synrad are downloaded first using download_models.py file. The below code downloads the models:

```python
import pandas as pd
import urllib.request
import os

def main():
    model_info = pd.read_csv('model_urls.csv')
    for i,r in model_info.iterrows():
        print(f'Downloading {r.model}...')
        download_file(r.url,f'{r.application}/{r.model}')

def download_file(url,filename):
    os.system(f'wget -O {filename} {url}')

if __name__=='__main__':
    main()
```

```
Downloading gan_mae_weights.h5...
Downloading mse_vgg_weights.h5...
Downloading mse_weights.h5...
Downloading gan_generator.h5...
Downloading mse_and_style.h5...
Downloading style_model.h5...
Downloading mse_model.h5...
```

Step 2

1. We generate the synrad test dataset using the synrad data generator file to perform backtesting of the model. The synrad generator needs the h5 files of the image types: ir069, ir107, lght and vil. The h5 files have to be stored in the file format provided in catalog.csv.
2. The catalog.csv file has to be updated with the files for the image types: ir069, ir107, lght and vil.
3. Now, to generate the test data, the following code has to be run:

```
!python make_synrad_dataset.py --sevir_data /content/sample_data --sevir_catalog
/content/sample_data/catlog_data.csv --output_location /content/sample_data/Data
```

Note: In the make_synrad_dataset.py file the value of the default in n_chunks has to be changed from 20 to 10.

Step 3

Since the nowcast test dataset is missing, we generate the test dataset.

1. The nowcast generator needs the h5 file of the image type: vil. The file has to be stored as per the file format in catalog.csv.
2. The catalog.csv file has to be updated with the files of the image type vil.
3. To generate the test data, the following code has to be run:

```
!python make_nowcast_dataset.py --sevir_data /content/sample_data --sevir_catalog
/content/sample_data/catlog_data.csv --output_location /content/sample_data/Data
```
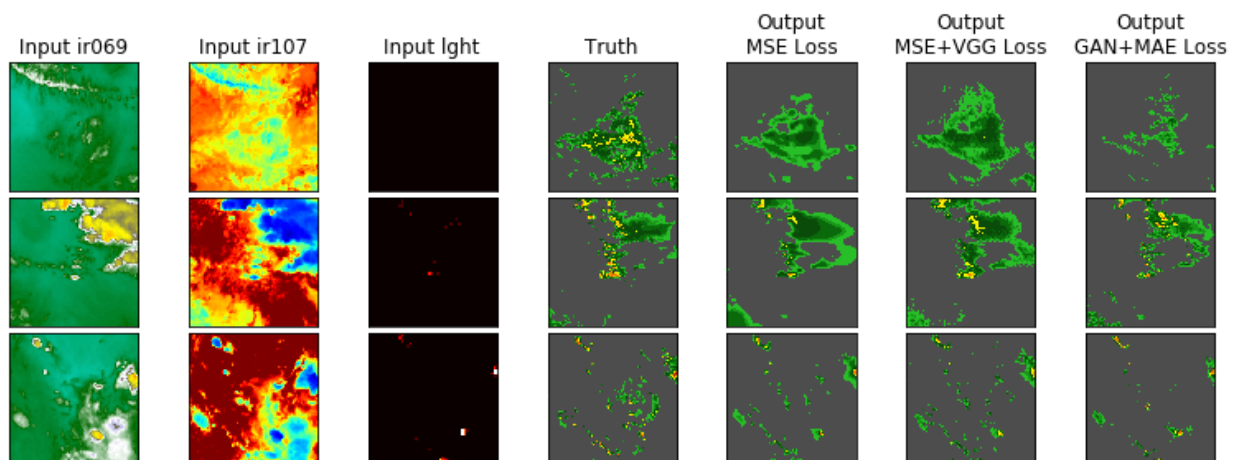
Synrad and Nowcast notebooks

Step 1

AnalyzeSyntheticRadar.ipynb has to be run with the sample dataset that was downloaded and then with the test dataset that we generated in step 2.

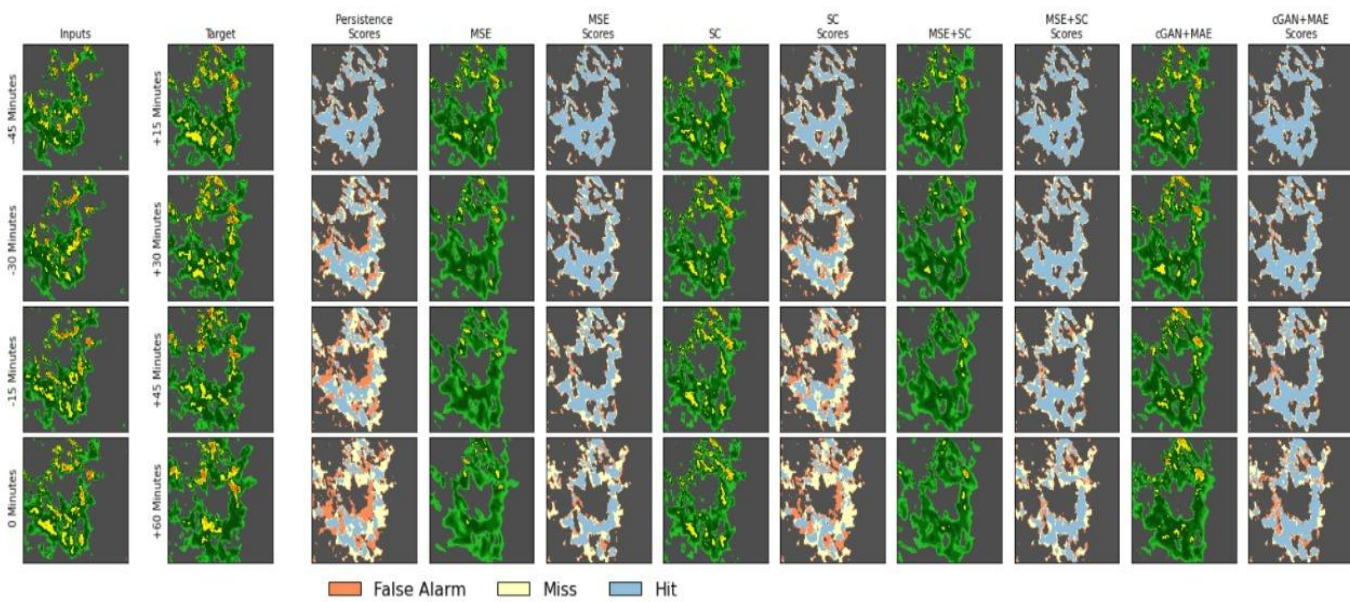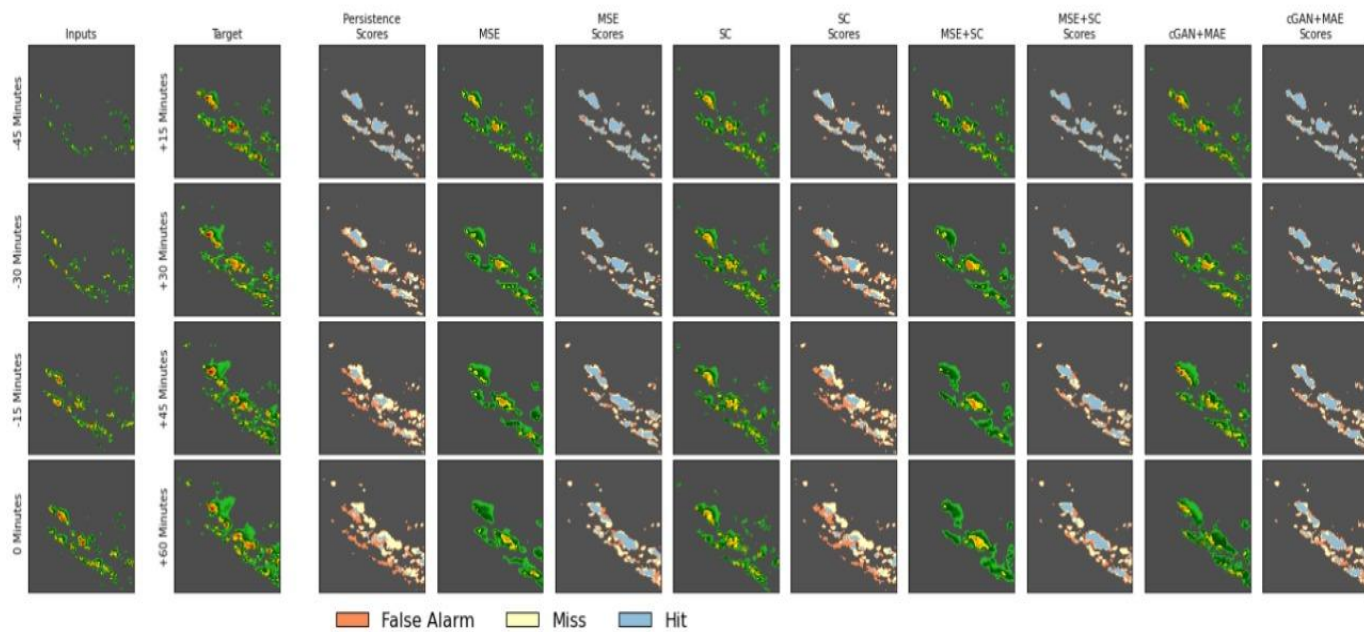**Output:** Visualization with the test data that we generated.
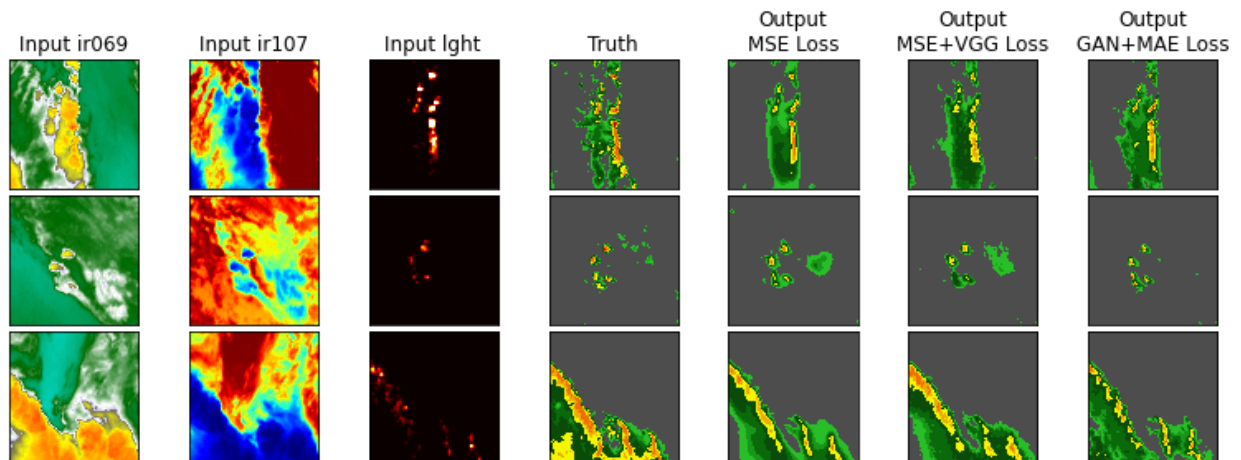
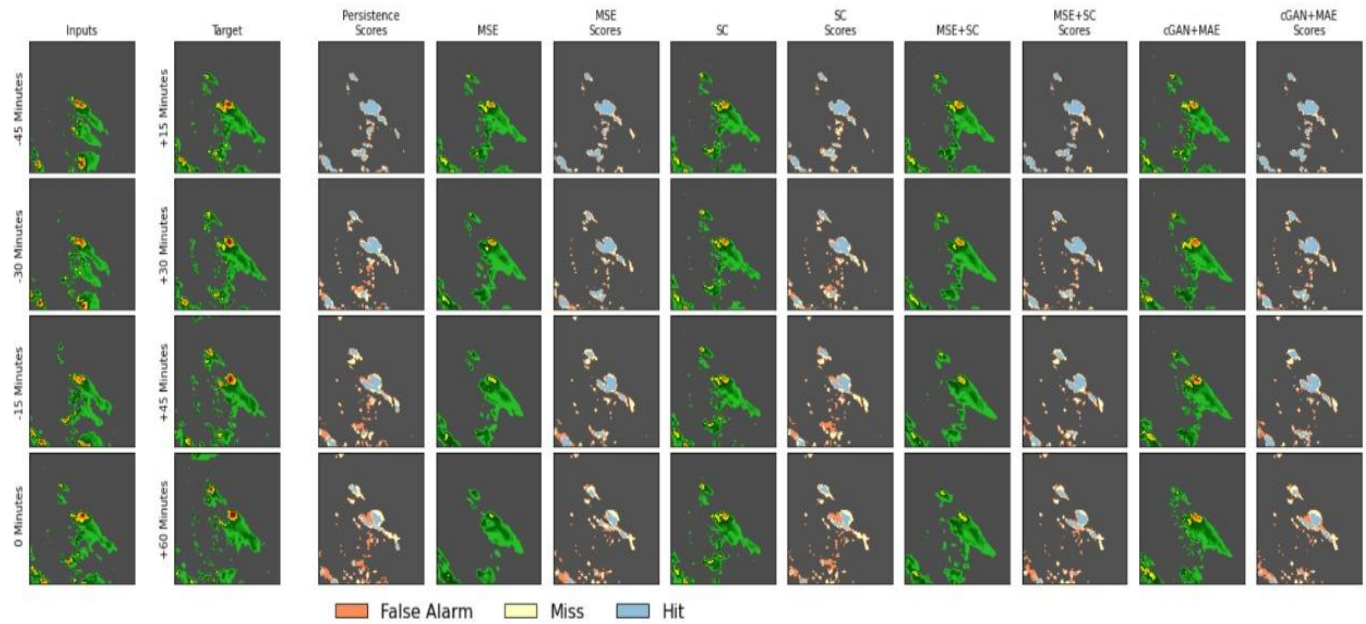**Output:** Visualization with the sample data downloaded from the notebook.

Step 2

Since the sample dataset for Nowcast is missing, we run **AnalyzeNowcast.ipynb** with the test dataset that we generated in step 3.

Output

Inputs | Target | Persistence Scores | MSE | MSE Scores | SC | SC Scores | MSE+SC | MSE+SC Scores | cGAN+MAE | cGAN+MAE Scores

-45 Minutes / +15 Minutes
-30 Minutes / +30 Minutes
-15 Minutes / +45 Minutes
0 Minutes / +60 Minutes

False Alarm   Miss   Hit

Inputs | Target | Persistence Scores | MSE | MSE Scores | SC | SC Scores | MSE+SC | MSE+SC Scores | cGAN+MAE | cGAN+MAE Scores

-45 Minutes / +15 Minutes
-30 Minutes / +30 Minutes
-15 Minutes / +45 Minutes
0 Minutes / +60 Minutes

False Alarm   Miss   Hit

The google document of this codelab can be found at

https://docs.google.com/document/d/1i0cKXjIWksTkFbtSPx2mH4Zioa
1o-oV1J3JXw49rNoE/edit?usp=sharing

The codelab link:

https://codelabs-preview.appspot.com/?file_id=1i0cKXjIWksTkFbtSPx2
mH4Zioa1o-oV1J3JXw49rNoE#3