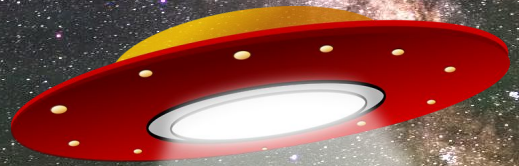
Data - X

Search for E.T.



Powered by -

BERKELEY SETI
RESEARCH CENTER

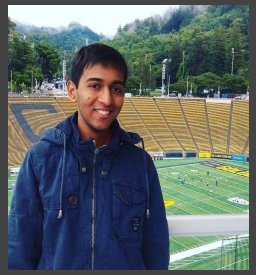
The Team

Dr. Steve Croft
Mentor
Berkeley SETI Research Center, California

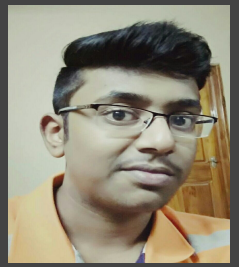
Tin

Anshul

Diego



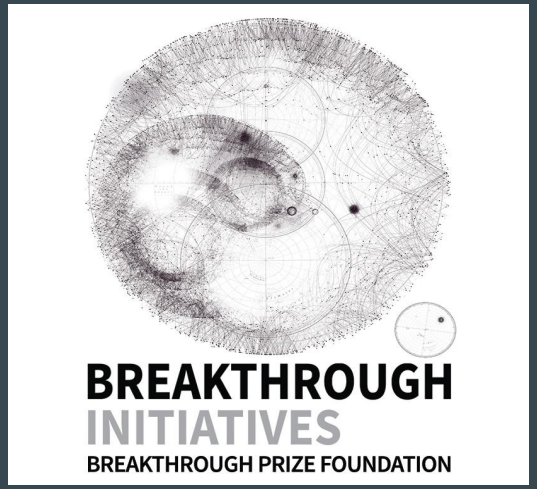
Arundhishaan



Pushya Mitra

Problem we are solving

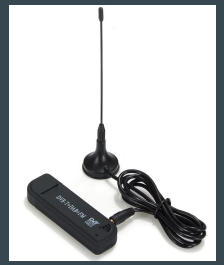
Finding signals from the unknown



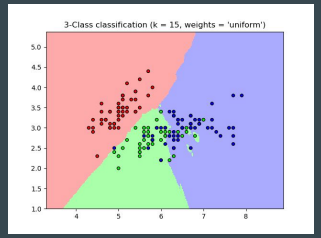
Many have and are trying different approaches towards extra-terrestrial Exploration

- **UKSRN**
- **Hawking's Starshot program**

Approach taken to solve the problem



Use Antenna to record the signal and save it as csv



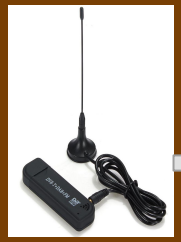
Use sklearn's KNeighborsClassifier to train a model that can classify FM signal

Go through every FM station we found and record them for 1 second

Use tensorflow to train a model that can classify the sample is music or not music

Compile the result as a list and present that to user

Architecture of solution



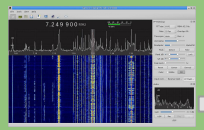
Use Antenna to record the signal

.csv file

Open with Pandas

Use sklearn's KNeighbor Classifier to identify FM station

List of FM station frequencies



Use gqrx to convert singal to .wav

.wav file

Use tensorflow's neural network to identify music

Interactive Report And Radio receiver

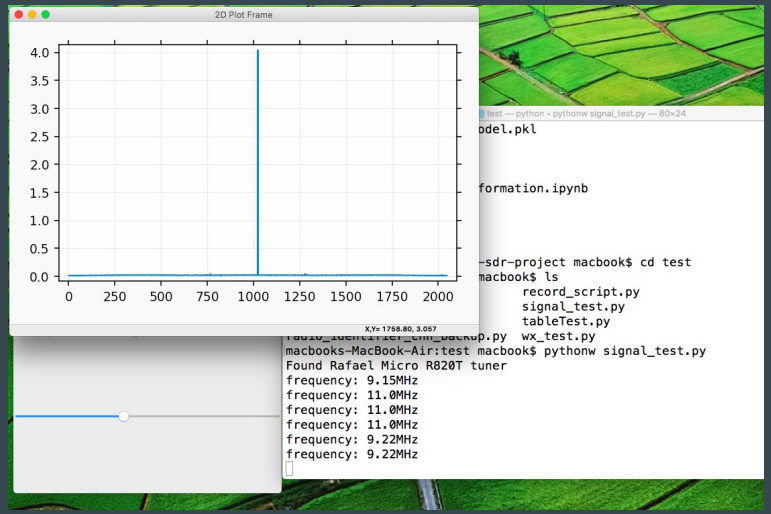
Load pretrained model (Pickled file)

Data processing/ machine learning

Load pretrained model (tensorflow)

The story of our learning path

Our first crappy UI



Building an app using matplotlib with dynamic updates

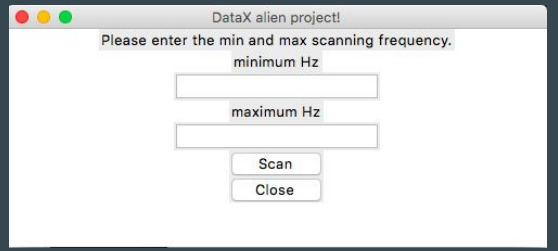
Exploring Music Classification

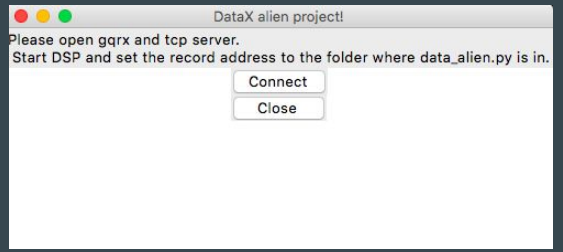
Record 4200 of Audio samples

Training the Dataset

Connecting GQRX TCP Server with our Interface to record audio

Our Present UI





The story of our learning path

Playing with RTL-SDR

- **Receives frequencies from 60 MHz up to 2400 MHz**
- **We started learning to capture signals using the SDR Dongle, with the GQRX Software**

COMMAND -> rtl_power -f min:max:bin -g gain -i interval -e runtime filename.ext

where min is initial frequency, max is terminal frequency

bin is frequency interval, interval in seconds

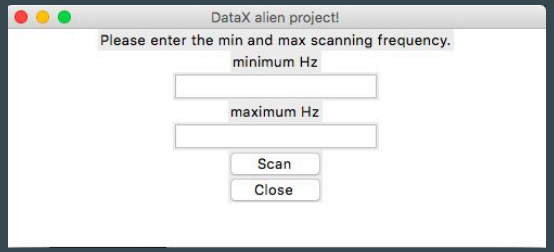
COMMAND We USED -

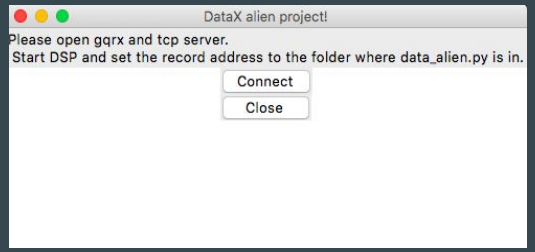
rtl_power -f 87M:108M:1k -g 20 -i 10 -e 5m logfile.csv

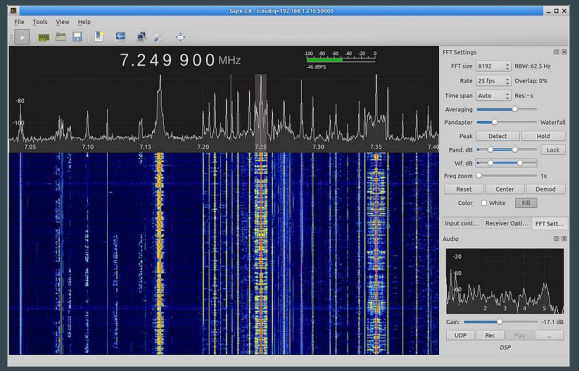
All the data is stored in a csv file logfile.csv.

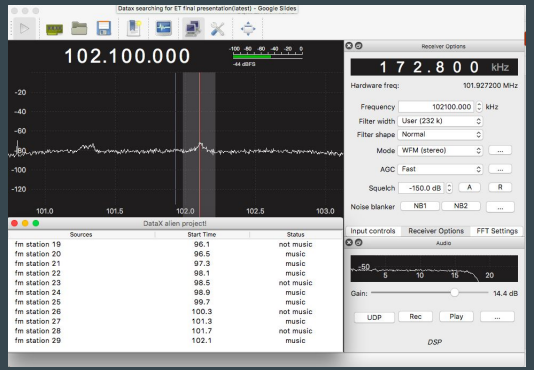
User Interface (demo!)

User Interface (Intended)









Final results

Acheived ~ 88% accuracy in the FM signal classifier

In [80]:
```
l3 =[89.7, 89.9, 90.1, 90.2, 90.3, 90.5, 90.7, 91.1, 91.5, 91.7, 92.1, 92.3, 92.7, 93.3, 94.1, 94.5, 94.8, 94.9, 95.3, 95.7, 96.1
```

In [81]:
```
#Bay Area FM Radio Station
radio = [87.9, 88.1, 88.5, 89.1, 89.3, 89.5, 89.7, 89.9, 90.1, 90.3, 90.5, 90.7, 91.1, 91.5, 91.7, 92.1, 92.3, 92.7, 93.3, 94.1,
```

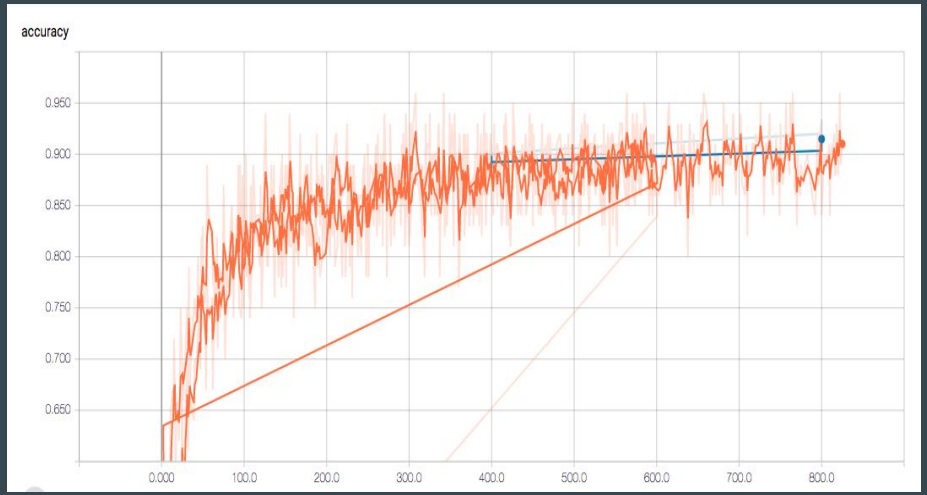
In [82]:
```
ctr=0
for i in range (0,len(l3)):
    if(l3[i] in radio):
        ctr=ctr+1
print("KNN model found", len(l3),"radio stations out of which actual Bay Area FM Radio Station are",ctr,"\nAnd",ctr,"stations were
```

```
KNN model found 33 radio stations out of which actual Bay Area FM Radio Station are 29
And 29 stations were found out of 46 Bay Area FM Radio Station
KNN model accuracy - 87.87878787878788 %
```

Final results

Acheived ~ 90% accuracy in music classifier

```
speech_commands — -bash — 80×24
INFO:tensorflow:Step #581: rate 0.001000, accuracy 86.0%, cross entropy 0.307873
INFO:tensorflow:Step #582: rate 0.001000, accuracy 89.0%, cross entropy 0.233574
INFO:tensorflow:Step #583: rate 0.001000, accuracy 89.0%, cross entropy 0.324120
INFO:tensorflow:Step #584: rate 0.001000, accuracy 92.0%, cross entropy 0.221804
INFO:tensorflow:Step #585: rate 0.001000, accuracy 95.0%, cross entropy 0.207853
INFO:tensorflow:Step #586: rate 0.001000, accuracy 91.0%, cross entropy 0.235727
INFO:tensorflow:Step #587: rate 0.001000, accuracy 93.0%, cross entropy 0.192298
INFO:tensorflow:Step #588: rate 0.001000, accuracy 89.0%, cross entropy 0.258026
INFO:tensorflow:Step #589: rate 0.001000, accuracy 88.0%, cross entropy 0.237599
INFO:tensorflow:Step #590: rate 0.001000, accuracy 91.0%, cross entropy 0.272120
INFO:tensorflow:Step #591: rate 0.001000, accuracy 84.0%, cross entropy 0.378407
INFO:tensorflow:Step #592: rate 0.001000, accuracy 89.0%, cross entropy 0.245126
INFO:tensorflow:Step #593: rate 0.001000, accuracy 92.0%, cross entropy 0.267026
INFO:tensorflow:Step #594: rate 0.001000, accuracy 86.0%, cross entropy 0.297942
INFO:tensorflow:Step #595: rate 0.001000, accuracy 86.0%, cross entropy 0.340704
INFO:tensorflow:Step #596: rate 0.001000, accuracy 92.0%, cross entropy 0.192765
INFO:tensorflow:Step #597: rate 0.001000, accuracy 88.0%, cross entropy 0.281181
INFO:tensorflow:Step #598: rate 0.001000, accuracy 90.0%, cross entropy 0.235033
INFO:tensorflow:Step #599: rate 0.001000, accuracy 89.0%, cross entropy 0.277639
INFO:tensorflow:Step #600: rate 0.001000, accuracy 92.0%, cross entropy 0.205572
INFO:tensorflow:Saving to "/tmp/speech_commands_train/conv.ckpt-600"
INFO:tensorflow:Step #601: rate 0.001000, accuracy 84.0%, cross entropy 0.361573
^CTraceback (most recent call last):
  File "train.py", line 431, in <module>
```



Used trained model to find FM stations from 60-2400MHz

```
KNN model found 53 radio stations out of which actual Bay Area FM Radio Station are 40
And 40 stations were found out of 46 Bay Area FM Radio Station
KNN model accuracy - 75.47169811320755 %
```

Future of our project

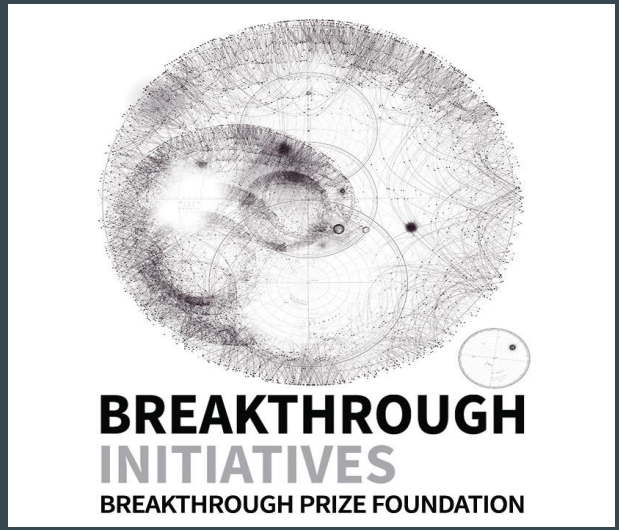
- ❏ We are now part of this open platform at the SETI Research center and we intend to demystify the unknown called Aliens.

Sight in the Future

- Identifying other signals and clustering them in real-time
- Learning about unknown signals and extra-terrestrial exploration

BREAKTHROUGH
LISTEN

- January **2016,** with **100 million** foundation
- A science-based program to **search for extraterrestrial communications** in the Universe.



Summary

We successfully applied what we have learned in Data-X!

- ❑ Worked with GQRX Software and rtl-sdr to capture signals
- ❑ Used python, numpy, pandas
- ❑ Used two Machine Learning api (sklearn and tensorflow)
- ❑ Used KNeighborsClassifier in sklearn to identify FM signals
- ❑ Connected our User Interface to the GQRX TCP server
- ❑ Controlled the GQRX to record the audio .wav file
- ❑ Pre-trained music data
- ❑ Used CNN in tensorflow to differentiate between music and not music

Thank you Professors and Staff!!

Thank You
and Remember...

Github link: https://github.com/sith0726/datax-sdr-project