# Project by Sudha Ramanantham

Monitor and analyze real time data from remote IoT connected devices.

* Use remote Raspberry Pi to collect data for temperature and humidity sensor.

     Measurement and collect multiple datasets.

* Create channels on ThingSpeak for real time data uploading from Raspberry Pi
* Plot data with Python/Pandas
* Perform linear regression for data analysis and check model.

**Introduction:**

* In this project, I receive data from a Raspberry Pi that uses VNC to connect to my laptop and measure the temperature in San Jose.
* The data was then exported to a CSV file, and I used it to work in Jupyter Notebook.
* Following that, I use Python's Pandas library to clean, prepare, and analyze the data.
* The machine was then trained and tested, and the predicted temperature was obtained.
* I plotted the actual temperature vs. predicted temperature graph using Matplotlib to show how the machine was trained to predict the temperature.

**Overview of project:**

* In this project, I monitor and analyze real-time data that tracks San Jose's temperature.
* To interact with the Raspberry Pi Internet of Things device, I set up VNC and ThingSpeak accounts on my PC.
* I next downloaded the VNC Viewer so I could see the Raspberry Pi on my computer.
* I then signed in to ThingSpeak, navigated to the devices section, and clicked on MQTT to obtain the credentials (username, ClientID, and password) needed to access the Raspberry Pi's measurement of the temperature in San Jose through Virtual Network Computing (VNC) Viewer.
* I was able to access the Raspberry Pi script through VNC viewer, enter my login information, and run it to set up my channel on ThingSpeak so I could monitor data and graphs on my Desktop.

**ML models:**

In this project, the supervised machine learning technique known as linear regression is used to identify the best fit linear line between the independent and dependent variables.

**Block Diagram:**

Raspberry Pi

MATLAB Visualization of Temperature

ThingSpeak Channels

Run

Given

MQTT

See

VNC Viewer

View through

Connected

My computer

Credentials – Username, Password, ClientID

Professor Computer computer

**Solutions:**

* I import LinearRegression from sklearn.linear model to execute linear regression on the data gathered through feeds.
* Finally, using train test split from sklearn.model selection, the machine was trained to predict the temperature and the data was tested.
* Next, I used Seaborn to plot the Residual graph to examine how closely the temperature prediction matched the actual temperature.
* Then I use Matplotlib visualization to view the graph of actual temperature versus anticipated temperature and get the best matched linear line.
* To prepare, clean, and analyze the data, I utilized Pandas.
* To check the accuracy, I import the mean squared error, mean absolute error, and r2 score values of the actual and projected temperature from sklearn.metrics.

**Summary:**

Monitor and analyze real time data from remote IoT connected devices using Raspberry Pi and ThingSpeak. The acquired San Jose temperature data is then plotted using pandas in Python. The data analysis in Matplotlib was then performed using linear regression to display the data visualization graph.

**References:**

<https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b>

<https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html>