Leveraging Machine Learning Models for Climate Data Interpretation and Forecasting

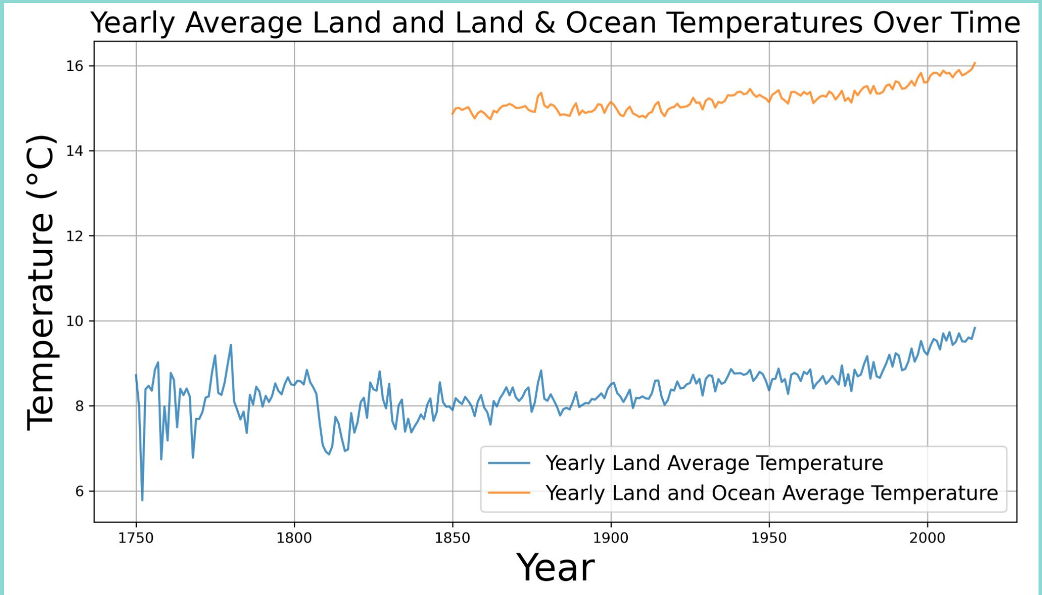
Edwin Peraza
Maxwell Lebda
Cade Duncan

Introduction

- Climate change impacts ecosystems, agriculture, and society.
- Predicting temperature trends is critical for mitigation and adaptation strategies.

Objective

Use machine learning models to analyze historical temperature data and forecast future trends.



Dataset and features

Dataset: Berkeley Earth Surface Temperature data.

Features:

- Target Variable: AverageTemperature
- Temporal: Date (Year, Month, Day)
- Spatial: Latitude, Longitude
- Excluded: City and Country (to avoid redundancy)

Preprocessing:

- ➔ Remove data points without AverageTemperature
- ➔ Remove data where the year is earlier than 1870
- ➔ Split date feature into day, month and year

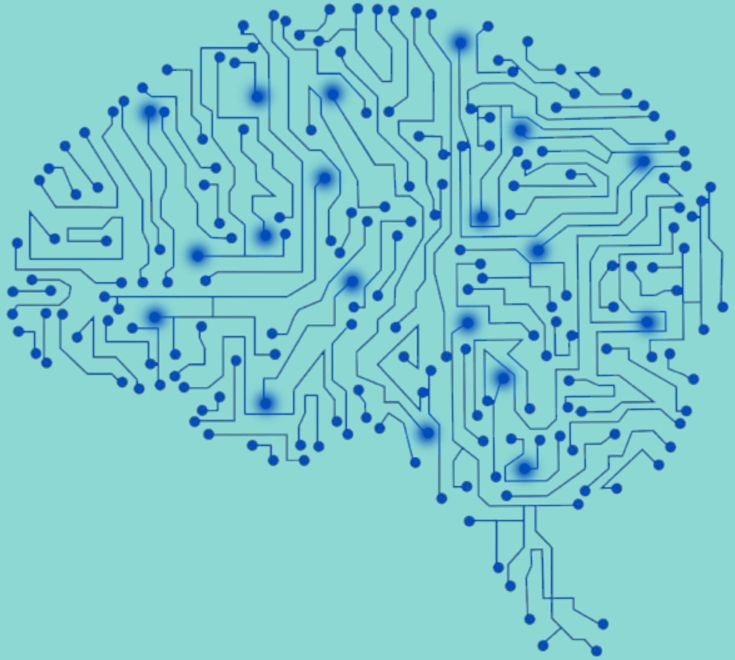
Results:

- ➔ Before: 239177 data points
- ➔ After: 171125 data points

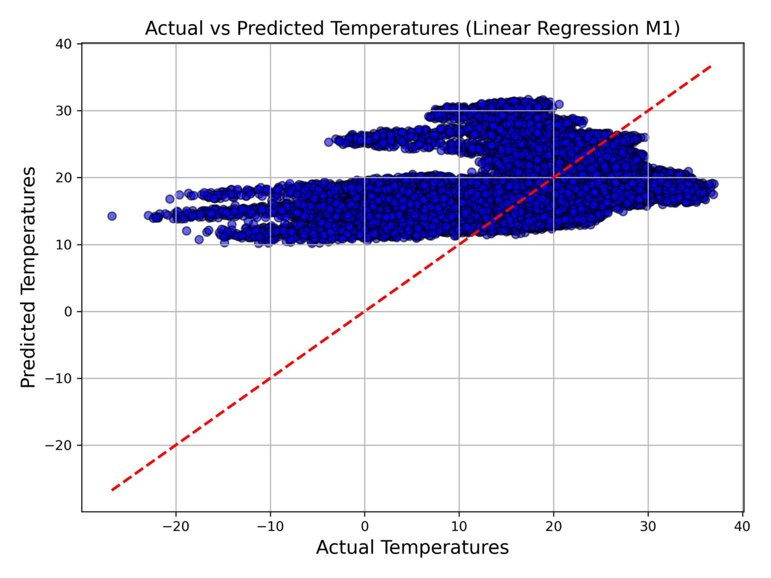
	AverageTemperature	AverageTemperatureUncertainty	Latitude	Longitude	Year	Month
0	21.425	0.437	4.02	-76.34	1926	10
1	22.05	0.343	4.02	-76.34	1967	3
2	21.577	0.461	4.02	-76.34	1910	6
3	21.109	0.502	4.02	-76.34	1913	5
4	21.432	1.062	4.02	-76.34	1888	10

Machine Learning Models

1. Linear Regression
2. Random Forest
3. SVM (Support Vector Machine)
4. KNN (K-Nearest Neighbours)

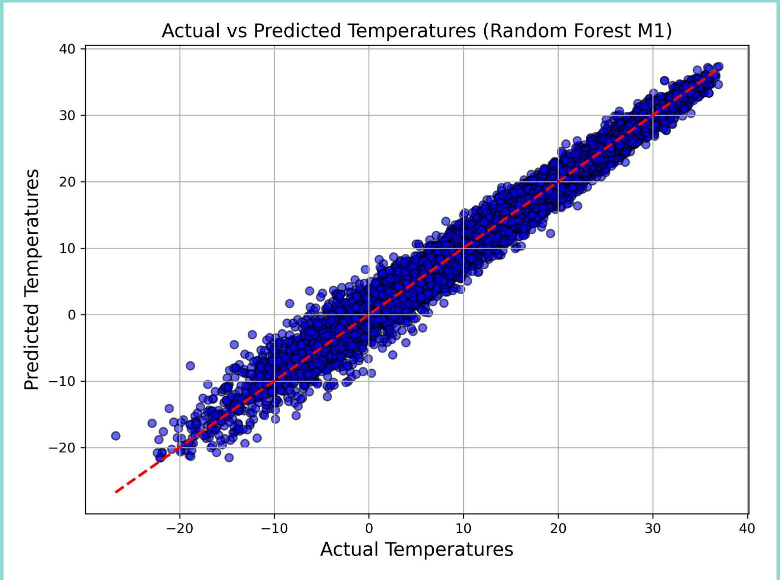


Linear Regression



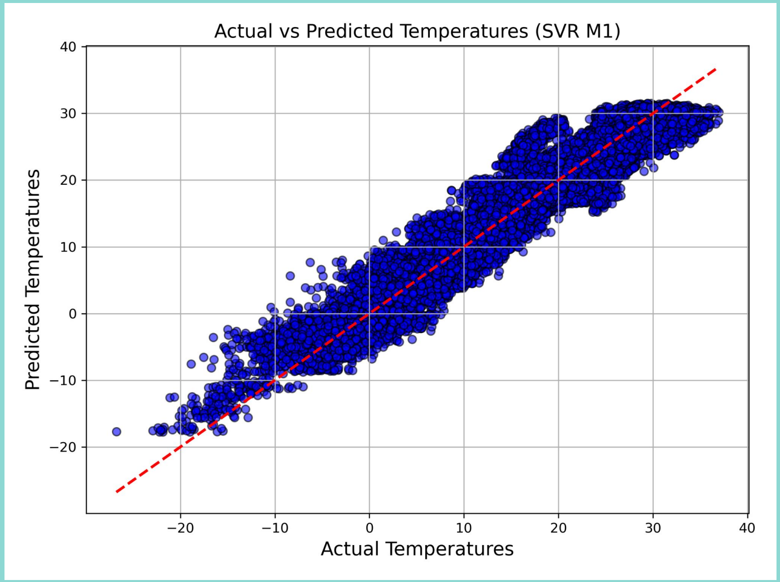
- ➔ Random train-test split (80/20)
- ➔ Why?
  - ◆ **Baseline model**
  - ◆ Simplicity and interpretability
- ➔ Experiment configuration:
  - ◆ Tested with final features and reduced features
- ➔ Results:
  - ◆ **Performed poorly** due to inability to model non-linear relationships
  - ◆ Provided a guidance for future experiments

Random Forest



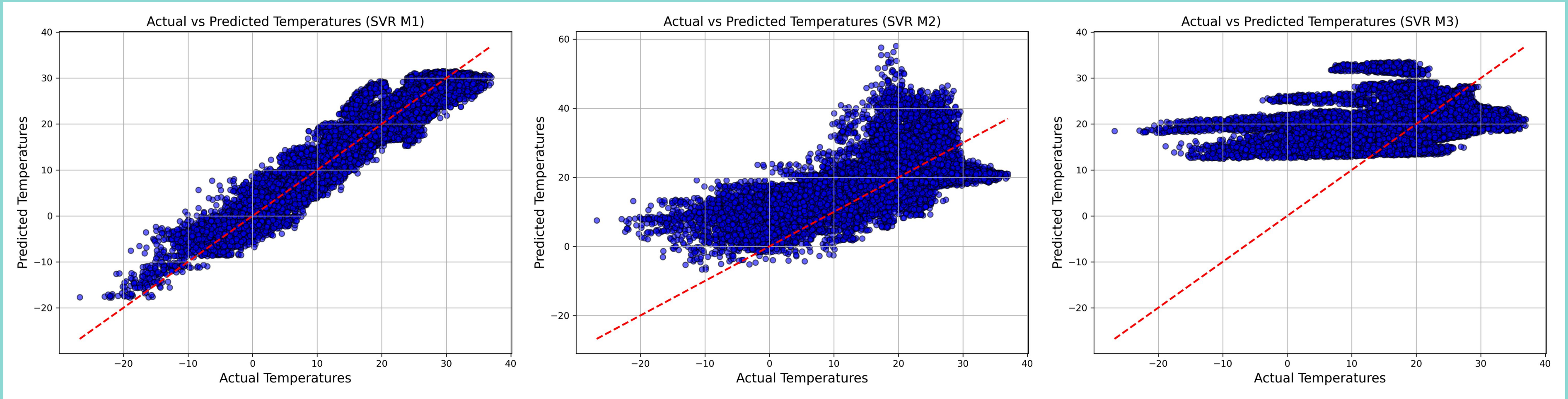
- ➔ Random train-test split (80/20)
- ➔ Why?
  - ◆ **Captures non-linear relationship**
  - ◆ Robust Against Overfitting
- ➔ Experiment configuration:
  - ◆ With and without country encoding
- ➔ Results:
  - ◆ Best R2 Score: **0.985694**
  - ◆ Performance almost unchanged with encoded country variable

SVM



- ➔ Random train-test split (80/20)
- ➔ Why?
  - ◆ **Handles non-linear patterns effectively**
  - ◆ Robust to outliers
- ➔ Experiment configuration:
  - ◆ Kernels tested: Linear, Polynomial, RBF
- ➔ Results:
  - ◆ Polynomial and Linear kernel performed poorly
  - ◆ RBF kernel had one of the highest R2 Score: **0.9179**
  - ◆ Slower training and prediction times compared to other models

Comparison of SVM

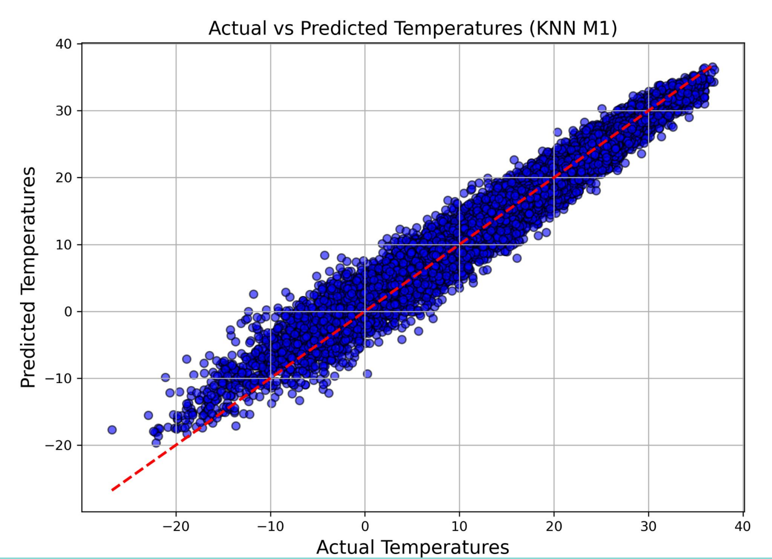


Radial Basis Function (RBF)

Polynomial

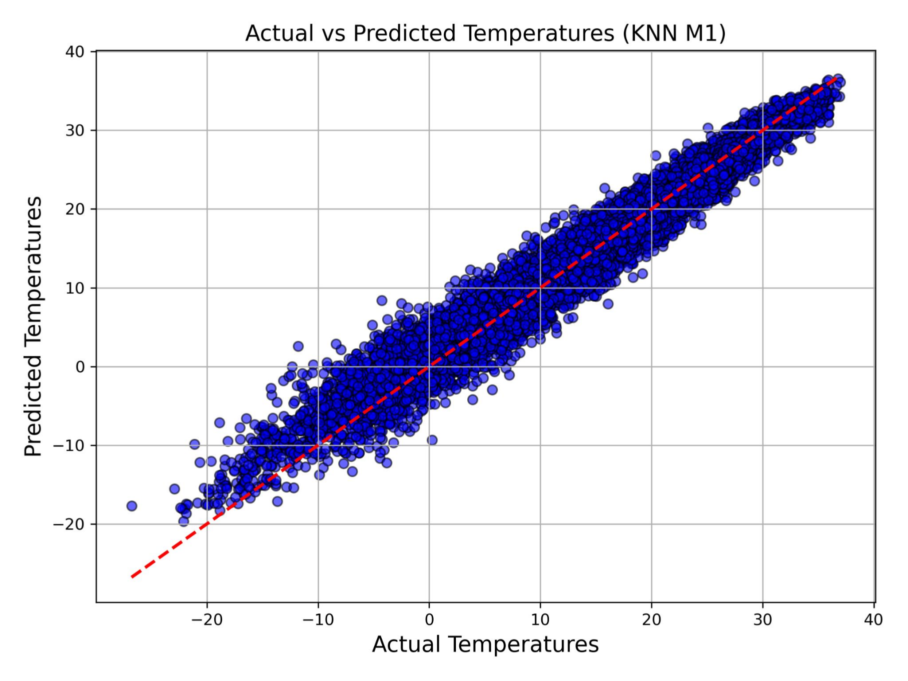
Linear

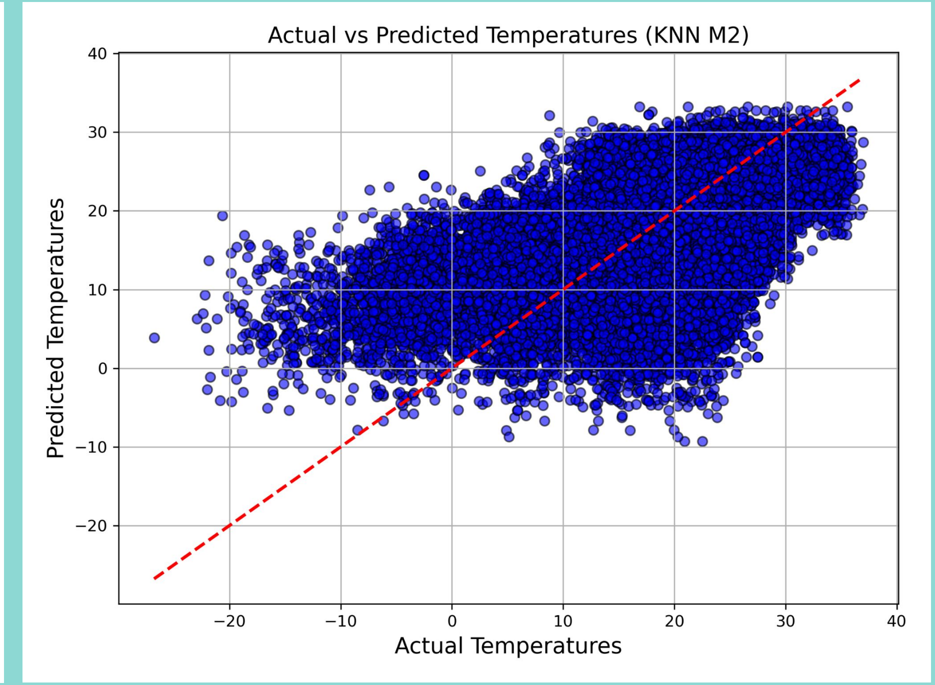
KNN



- ➔ Random train-test split (80/20)
- ➔ Why?
  - ◆ Simplicity of implementation
  - ◆ **Effective for capturing local, spatial and temporal patterns**
- ➔ Experiment configuration:
  - ◆ Tested reduced feature set
  - ◆ Optimal **k = 5**
- ➔ Results:
  - ◆ Performed poorly on reduced features
  - ◆ Second highest R2 Score: **0.970539**

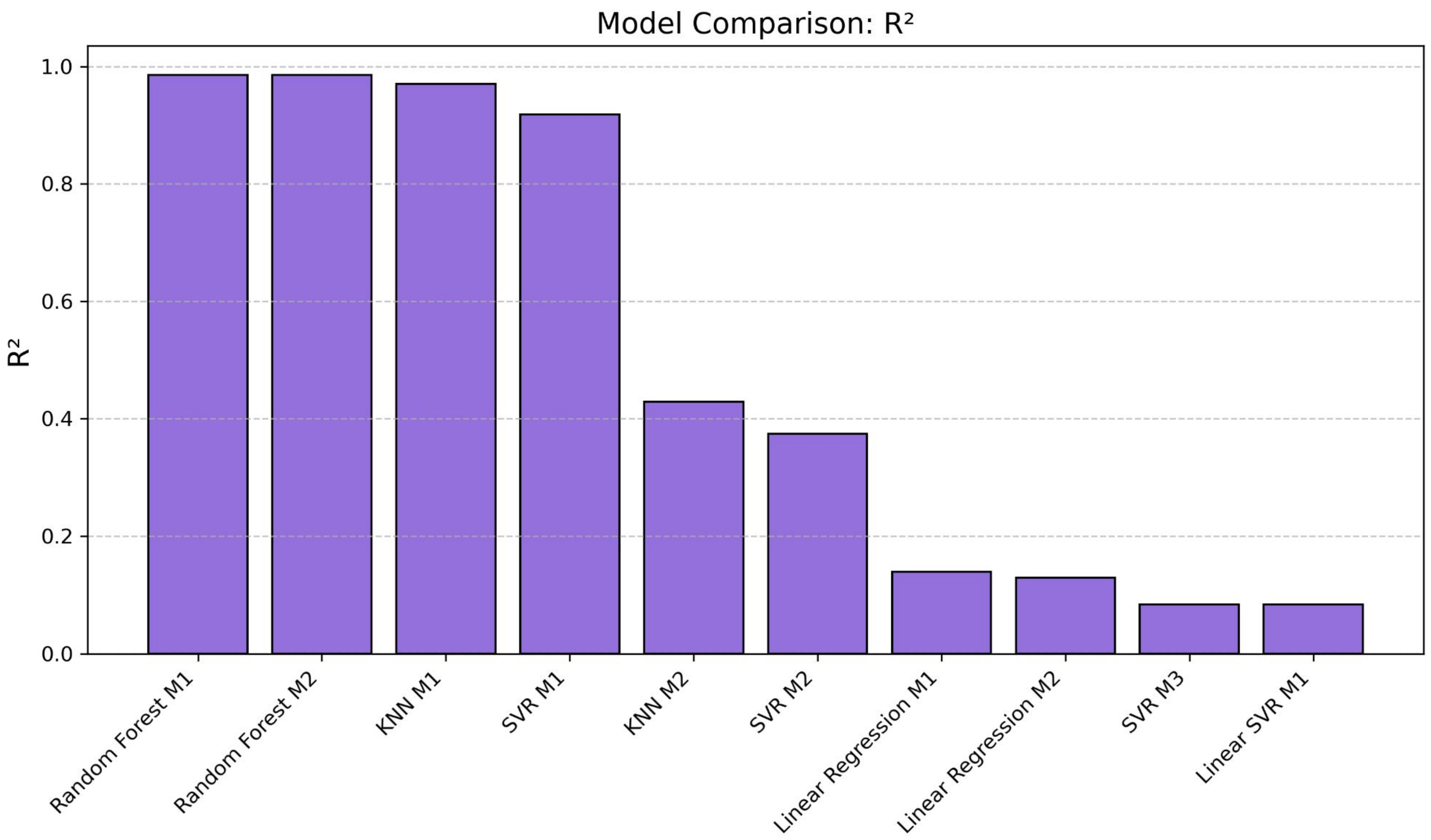
Comparison of KNN





Results

Model	MAE	MSE	RMSE	R^2
Random Forest	0.7750	1.3258	1.1514	**0.9857**
KNN (Final Features)	1.1471	2.7303	1.6524	0.9705
KNN (Reduced Features)	5.2274	52.9441	7.2763	0.4287
SVR (RBF Kernel)	2.0081	7.6086	2.7584	0.9179
SVR (Poly Kernel)	5.9241	58.0018	7.6159	0.3741
SVR (Linear Kernel)	6.8501	84.9333	9.2159	**0.0835**
Linear Regression	7.0842	79.7846	8.9322	0.1391

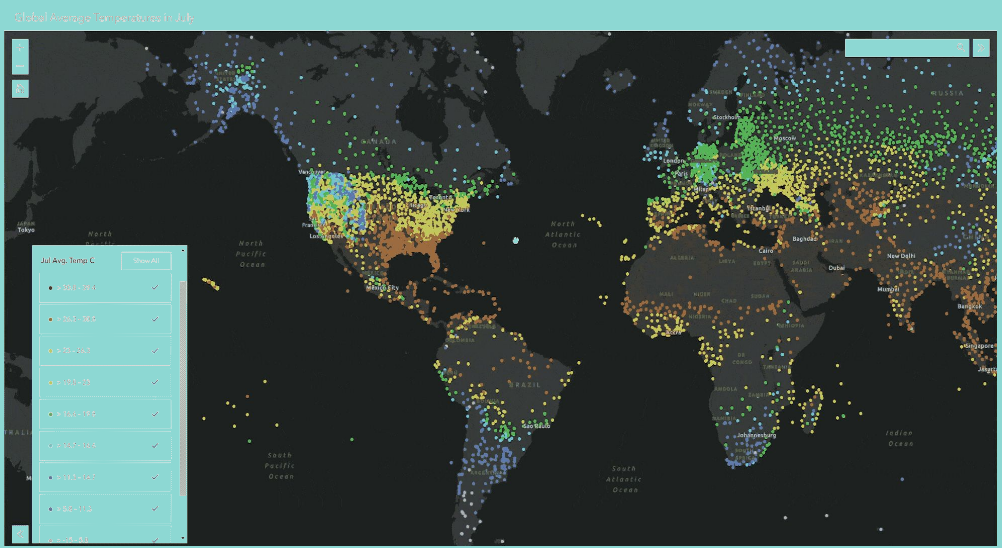


Model Comparison: R^2
R^2
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest M1
Random Forest M2
KNN M1
SVR M1
KNN M2
SVR M2
Linear Regression M1
Linear Regression M2
SVR M3
Linear SVR M1

Conclusion

- ❑ Machine learning models, specifically Random Forest and KNN, demonstrated exceptional capability in capturing spatial and temporal climate patterns.
- ❑ **Random Forest** achieved the **highest accuracy** with an R2 score of **0.9857**, highlighting its robustness and predictive power.
- ❑ The results **underscore** the critical role of non-linear models in addressing the complexity of climate data.
- ❑ **Linear models** like Linear Regression and SVR with a linear kernel **performed poorly** due to the complex nature of climate data.

Future Work



- **Map integration** using arcGIS
- Utilize **more models** and **merge more data** for more accurate predictions
- Integrate with a **front end** allow for users to see various metric of data from our models

Questions