California Wildfire Prediction:

Comparative Analysis of Machine Learning Models

Cynthia M. Widjaja

California Science and Technology University

MB/CSE604 Machine Learning Fundamentals

Dr.Yongchang Feng & Dr. Glen Qin

29 September 2024

California Wildfire Prediction:

Comparative Analysis of Machine Learning Models

1. Introduction

The increase in wildfires over recent decades has caused severe damage to ecosystems, property, and human life. As part of efforts to mitigate the impact of these fires, predictive models can be valuable in understanding wildfire patterns and improving response strategies. This project aims to develop predictive models for forest wildfires using machine learning techniques. We utilized historical datasets from 1992 to 2015. The models developed include Linear Regression, Polynomial Regression, Random Forest, and ARIMA models. The objective is to predict the number of wildfires for the years 2023, and evaluate the performance of these models on each dataset.

2. Datasets

The dataset used in this analysis consists of historical wildfire data collected between 1992 and 2015, including key variables such as wildfire counts, locations, and source of fire.. The target variable for prediction is the number of wildfires occurring in each year.

3. Objective

The aim is to explore the accuracy of four models below in making the predictions:

- Linear Regression: A basic statistical model that fits a linear relationship between the independent variables and the target variable.

- Polynomial Regression: A non-linear regression model that fits a polynomial function to the data.
- Random Forest: An ensemble learning model based on decision trees, capable of handling non-linear relationships and complex interactions in the data.
- ARIMA: A time series model that combines autoregressive and moving average components to capture temporal dependencies.

4. Exploratory Data Analysis (EDA)

Before applying machine learning models to the wildfire dataset, Exploratory Data Analysis (EDA) was conducted to better understand the structure, trends, and key variables influencing wildfire occurrences. The EDA process involves analyzing the data's distribution, patterns, and relationships between variables to guide model selection and preparation.

- **Data Overview**
    - Year: The year of wildfire occurrence.
    - State: The state in which the wildfire occurred.
    - Number of Wildfires: The target variable representing the number of wildfires reported per year.
- **Data Cleaning**
    - Handling Missing Data: Any missing values in the dataset were identified and addressed (either imputed based on averages or removed)

- Outlier Detection: Some extreme values were retained as they represented significant wildfire years, while others were excluded if they were considered noise.

5. Model Evaluation Metrics

The following metrics were used to evaluate model performance:

- Mean Absolute Error (MAE)
  - This metric measures the average magnitude of the errors in a set of predictions, without considering their direction.
  - It represents the average absolute difference between predicted and actual values, with lower values indicating better model performance.
- Mean Squared Error (MSE)
  - This measures the average of the squares of the errors—i.e., the average squared difference between predicted and actual values.
  - MSE penalizes larger errors more than smaller ones, making it useful for highlighting significant prediction deviations.
- R^2 Score
  - This statistic indicates the proportion of the variance in the dependent variable that can be explained by the independent variables in the model.
  - R^2 values range from 0 to 1, where higher values signify a better fit of the model to the data.

6. Analysis

These are the predictions for the year 2023 generated by the four models.

Model	MSE	MAE	R^2	Predicted Wildfires (2023)
Linear Regression	1,104,351.84	825.93	0.15	6,671.01
Polynomial Regression	1,043,445.40	833.33	0.19	8,589.72
Random Forest	248,196.56	453.72	0.81	7,271.24
ARIMA Model	152,717.31	348.66	0.19	7,454.25

From the four models above, we analyzed the following:

- The Linear Regression model performed the worst in terms of fit, indicated by a low R^2 value of 0.15 and Random Forest model demonstrated superior performance with a high R^2 value of 0.81.
- The Linear Regression model and the Polynomial Regression model are not suitable for predicting the wildfire due to high MSE and MAE and low R^2 value.
- The Random Forest model has the highest value of R^2 and comparable lower value for MSE and MAE, indicating its ability to minimize prediction errors effectively.
- The ARIMA model produced the best prediction in terms of MAE and MSE, indicating its effectiveness in time-series forecasting, however R^2 value is comparable to the Polynomial Regression model.
- The ARIMA model has limitations in its assumption of linear relationships, where wildlife data can be influenced by numerous nonlinear factors.

7. Conclusion

Based on the evaluation of the four models, the Random Forest model outperformed the others in predicting wildfires for the year 2023, achieving the highest accuracy with the lowest errors. The ARIMA model also performed well, particularly in capturing the time-series nature of the data, but Random Forest provided a more robust solution when considering various predictors. Further fine-tuning of the Random Forest model could potentially enhance its predictive capability.

Future work could involve incorporating more detailed features such as monthly fire data, weather conditions, land usage patterns, and human activities, which could further improve the predictive accuracy of the models.

References

Frost, Jim. (2024). *Choosing the Correct Type of Regression Analysis*. Statistics By Jim. https://statisticsbyjim.com/regression/choosing-regression-analysis/

Kane, Michael J. (2014). Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks.. https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-15-276

Shi, Liurui. (2022). TMachine Learning for Stock Prediction by Different Models. https://doi.org/10.2991/978-94-6463-036-7_48

Tatman, Rachel (2020). 1.88 Million US Wildfires. https://www.kaggle.com/datasets/rtatman/188-million-us-wildfires/data

Appendix A: Additional Tables and Figures

Figure 1. Raw data [Total Fires by Year]

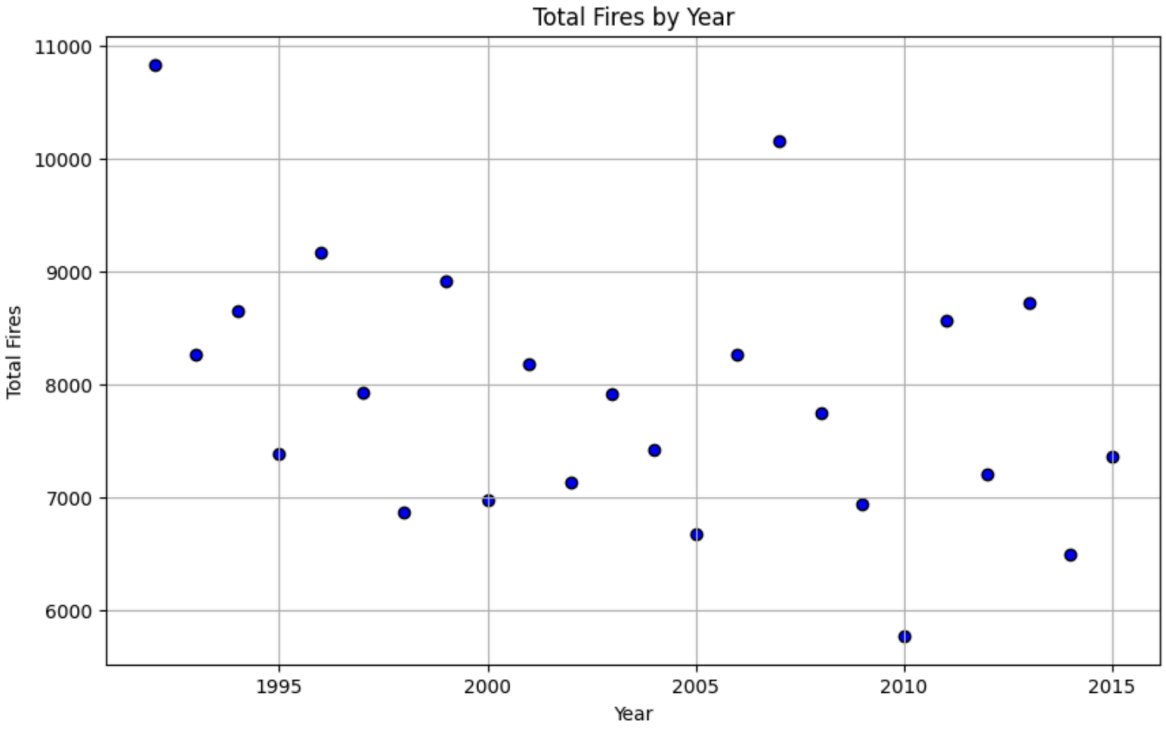


Figure 2. Linear Regression

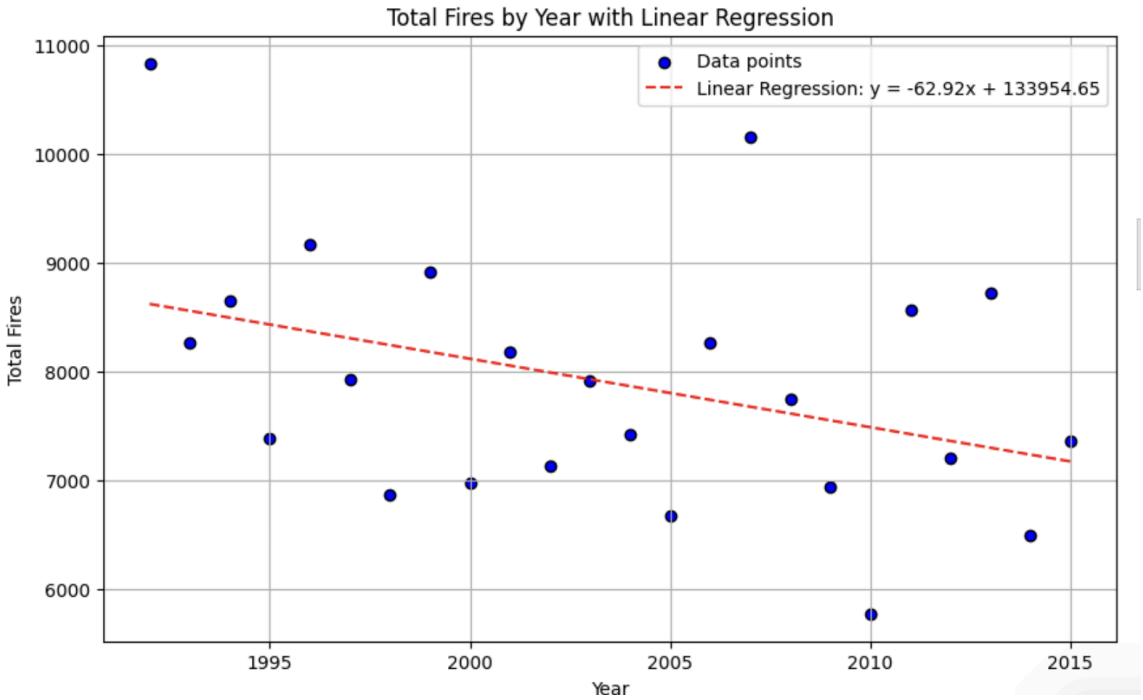


Figure 3. Polynomial Regression

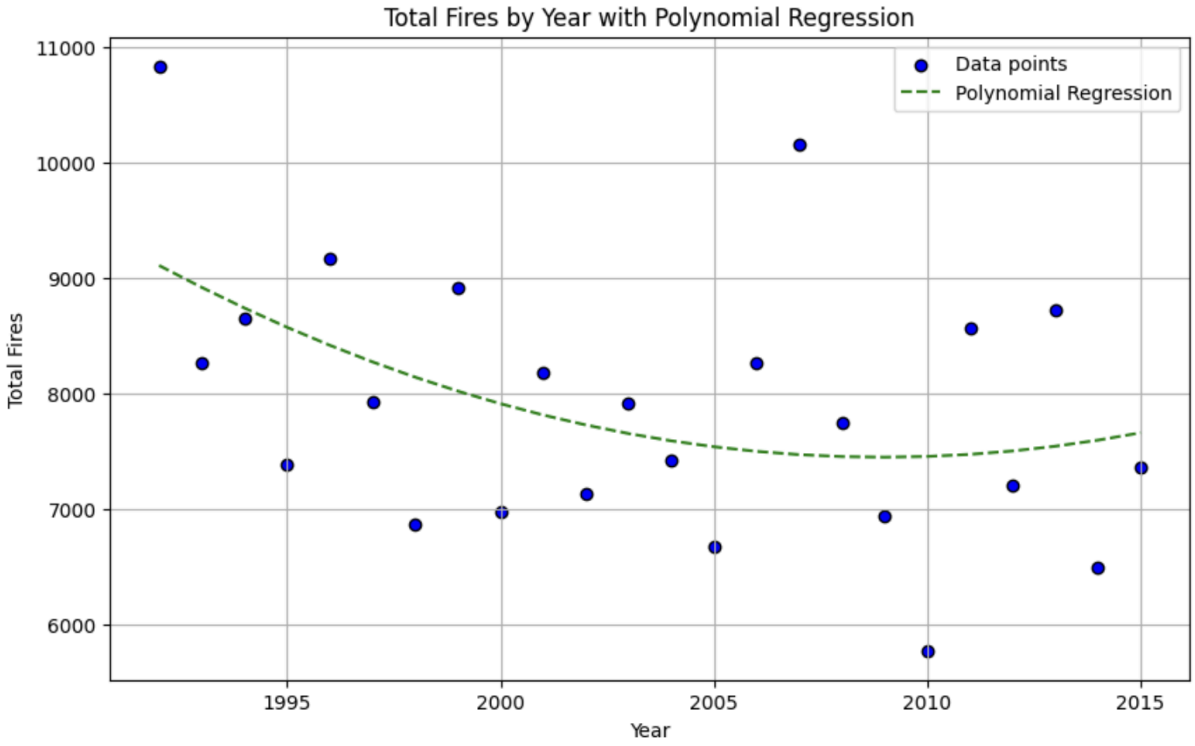


Figure 4. Random Forest

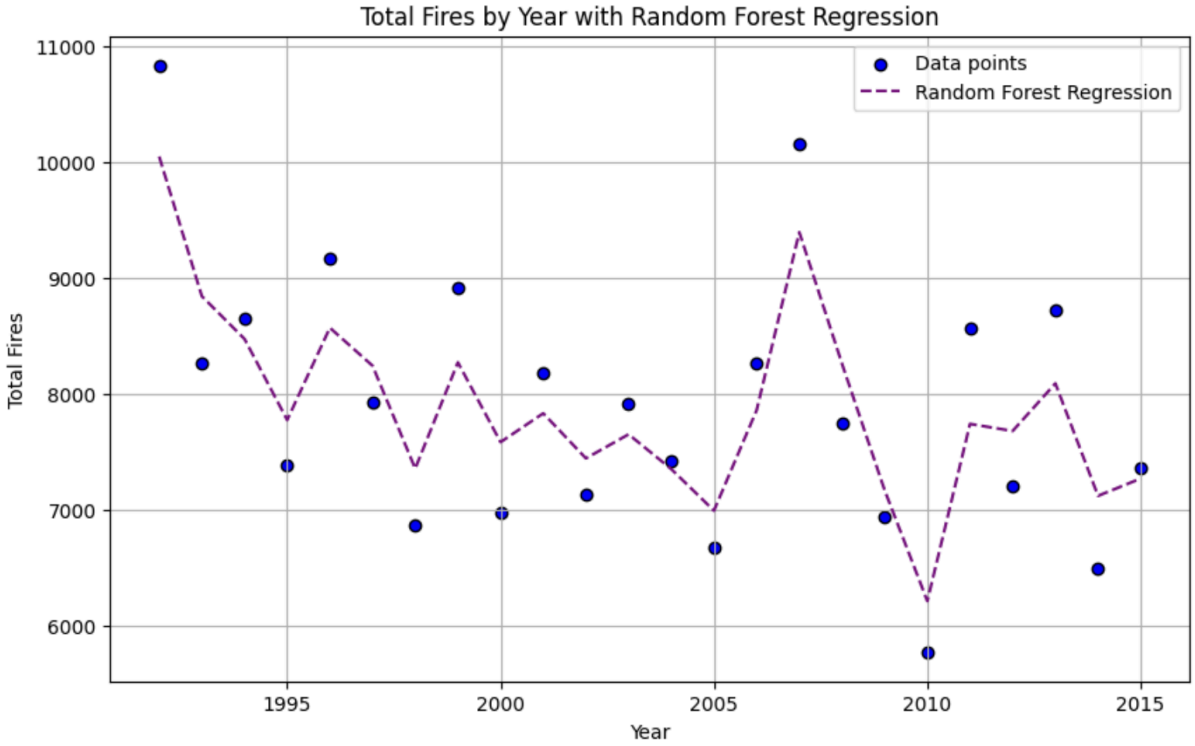
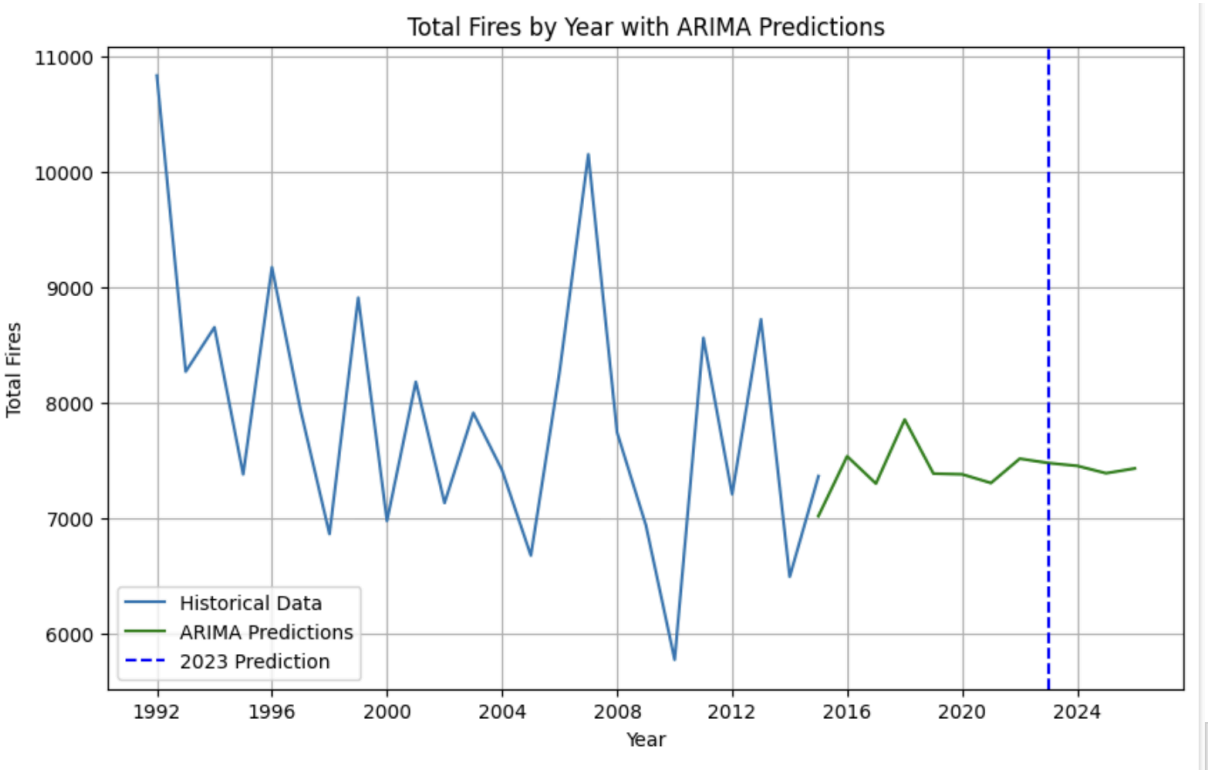


Figure 5. ARIMA Predictions



Appendix B: Code Snippets and Model Configurations

Snippet 1. Linear Regression

```
# Perform linear regression
slope, intercept, r_value, p_value, std_err = linregress(t['FIRE_YEAR'], t['total_fires'])

# Create regression line
regression_line = slope * t['FIRE_YEAR'] + intercept

# Predict the number of fires for the year 2023
year_to_predict = 2023
predicted_fires = slope * year_to_predict + intercept

# Print the prediction
print(f"Predicted number of fires in {year_to_predict}: {predicted_fires:.2f}")
```

Snippet 2. Polynomial Regression

```
# Polynomial features
poly = PolynomialFeatures(degree=2)  # Change degree as needed
X_poly = poly.fit_transform(X)

# Fit polynomial regression model
model = LinearRegression()
model.fit(X_poly, y)

# Predict for 2023
year_to_predict_poly_3 = np.array([[2023]])
year_to_predict_poly_poly_3 = poly.transform(year_to_predict_poly_3)
predicted_fires_poly_3 = model.predict(year_to_predict_poly_poly_3)
print(f"Polynomial Regression Prediction for 2023: {predicted_fires_poly_3[0]:.2f}")
```

Snippet 3. Random Forest

```
# Fit Random Forest model
model_rf = RandomForestRegressor(n_estimators=100)
model_rf.fit(X, y)

# Predict for 2023
year_to_predict_rf_3 = np.array([[2023]])
predicted_fires_rf_3 = model_rf.predict(year_to_predict_rf_3)
print(f"Random Forest Prediction for 2023: {predicted_fires_rf_3[0]:.2f}")
```

Snippet 4. ARIMA Predictions

```
# Fit ARIMA model
model_arima = ARIMA(y, order=(5, 1, 0))  # Example ARIMA parameters (p, d, q)
model_fit = model_arima.fit()

# Forecast fires for years beyond the dataset (2023)
start_year = '2014-01-01'
end_year = '2025-01-01'

# Forecasting from the last known year in the dataset to 2023
predictions_arima = model_fit.predict(start=start_year, end=end_year, typ='levels')

# Since ARIMA returns dates with monthly granularity by default, we'll extract the nearest y
predictions_arima.index = pd.to_datetime(predictions_arima.index)

# Convert predictions to yearly (taking the closest values to 2023)
predictions_arima_yearly = predictions_arima.resample('Y').mean()

# Extract predictions for  2023
predicted_fires_2023 = predictions_arima_yearly.loc['2023'].values[0]
```

Snippet 5. MAE, MSE and R^2 Calculation

```
# Calculate MSE, MAE, and R²
mse = mean_squared_error(t['total_fires'], regression_line)
mae = mean_absolute_error(t['total_fires'], regression_line)
r2 = r2_score(t['total_fires'], regression_line)

# Print the error metrics
print(f"Mean Squared Error (MSE): {mse:.2f}")
print(f"Mean Absolute Error (MAE): {mae:.2f}")
print(f"R-squared (R²): {r2:.2f}")
```