**Assignment Cover Sheet**

**BSC Creative Computing**

To be completed **electronically** by the student and submitted with each piece of work. Please upload this completed cover sheet via Turnitin

**Assignment Title:** Machine Learning (Model Development- 60%)

**Tutor:** Usman Ahmad

**Student Name:** John Mhelvs Vitto

**Student Number:** 577910

**Date of Submission:** January 14, 2026

**Details of your submission**

Online submission: Enter the URL of where you project files can be accessed (e.g., Google Drive)

**GitHub Repository Link:** https://github.com/mjgrant/machinelearning.git

**Kaggle Link:** https://www.kaggle.com/code/mjgrant/a2-wildfire-prediction-gui

**YouTube Video Link:** https://youtu.be/2SBdg3Ebw7s

In submitting this assignment, I am confirming that I have read and understood the regulations for assessment, and I am aware of the seriousness with which the University regards unfair practice.

Signed: Date: January 14, 2026

## Introduction

Wildfires recently have become a common occurrence in certain areas that are very prone to it. Countries such as Australia or states like Los Angeles in the US are prime examples of said wildfire prone areas. These wildfires create serious societal, environment, and economic damage, and having the awareness to assess risky conditions can support better preparedness. This project develops a machine learning model that estimates the likelihood of wildfire occurrence using historical weather-related conditions. The main goal is to convert complex meteorological data into an easy-to-understand prediction that non-technical users can explore. Multiple classification algorithms were tested and compared using a consistent train/test split and standard evaluation metrics. The main contribution is a trained wildfire-likelihood classifier and an interactive Gradio GUI that allows users to adjust conditions such as temperature, humidity, and wind to see how the predicted risk changes.

## Problem Statement and Objectives

The problem being addressed is: Given observed weather conditions, can we predict whether a wildfire is likely to occur? Wildfire occurrence is influenced by multiple factors, and weather is one of the key drivers that affects dryness and spread conditions. However, it isn't so easy for typical users to understand raw meteorological values, and manual risk judgement is unreliable. This project aims to build a supervised classification model that maps weather inputs into a binary outcome: fire occurred (1) vs no fire (0).

Here are the objectives:

1. Prepare/Look for a clean dataset suitable for classification (handle missing values and remove unsuitable columns).
2. Train and test multiple algorithms (KNN, Random Forest, HistGradientBoosting) using an 80/20 split for fair comparison.
3. Evaluate models using accuracy and confusion matrices.
4. Select the best-performing model and deploy it in a simple interactive GUI so users can experiment with weather conditions and view predicted fire probability.

## Data and Features

The dataset used contained historical records combining wildfire occurrence labels with numerical weather-related attributes (e.g., temperature, humidity, wind speed, pressure, dew point, cloud cover, and fire weather index). The target variable (y) was the column called "occured" (0/1) which basically mentioned if a fire happened or not. Preprocessing included removing columns that are not suitable as predictors for risk estimation, such as the frp column, which can act as leakage since it relates to fire intensity rather than pre-fire conditions. Missing values were handled using median imputation for number features. Feature selection for the GUI focused on understandable weather controls (temperature, humidity, wind, etc.) rather than geographic coordinates.

## Model Development

This project followed a standard supervised learning pipeline: data loading, preprocessing, train/test splitting, model training, and evaluation. Initially, the dataset was loaded and inspected for missing values and target distributions. The output label "occured" was converted to integer format to support classification. Because some columns may directly correlate with an active fire rather than risk conditions, frp was removed to avoid leakage and to ensure that the model represents a realistic "risk from conditions" prediction scenario.

The dataset was split into training and testing subsets using 80/20 splits with a fixed random seed to support reproducibility. Stratified splitting was used to maintain a similar class balance in both sets. Three models were implemented and compared:

**K-Nearest Neighbors (KNN)** which was taught in class was used as a baseline model. KNN predicts based on similarity between feature vectors, so feature scaling was applied where needed to make sure that the columns with larger numeric ranges do not dominate distance calculations. A tuning step tested multiple values of k (number of neighbors) and selected the value that produced the best test accuracy.

**Random Forest** which was also taught in class was implemented as the main improved model. This model combines many decision trees trained on random subsets of data and features. This method was effective for non-linear decision boundaries and feature interactions common in

weather-based patterns. Hyperparameters included a large number of trees (n_estimators = 400) to stabilize results and improve generalization.

**HistGradientBoostingClassifier** was not taught in class but was used as one of the models anyway, since we were given the freedom of making use of any model. This model was also tested as an additional advanced model. Gradient boosting builds trees sequentially, focusing on correcting previous mistakes. A tuning step explored learning rate, depth, and iteration parameters to attempt performance improvement. Despite tuning, it did not outperform Random Forest on this dataset.

Evaluation focused on test accuracy and confusion matrices to understand the type of errors (false alarms vs missed fires). Lastly, the best-performing model (Random Forest) was integrated into a Gradio GUI, allowing users to adjust weather conditions through sliders and select day/night, then view a probability-based risk output.

## Results and Evaluation

Model performance was evaluated on the held-out test set using accuracy and confusion matrices. KNN achieved a baseline accuracy of approximately 0.648, and after tuning k it reached 0.650, showing only a minor improvement. Random Forest produced the best result, achieving a test accuracy of approximately 0.668. HighGradientBoosting achieved about 0.656, and tuning did not significantly improve it (~0.6560). A comparison chart was generated to clearly visualize the differences across models, confirming Random Forest as the highest-performing approach in this project.

The moderate accuracy was expected because wildfire occurrence is not determined by weather alone; ignition sources and environmental fuel conditions also contribute, leading to overlapping patterns between “fire” and “no fire” cases. The confusion matrix helped interpret real-world trade-offs between false positives (predicting fire when none occurs) and false negatives (missing a fire). The final GUI outputs a clear label and probability percentage to support user understanding.

## Conclusion and Future Work

This project was about wildfire likelihood classifiers using historical weather-related features and compared three algorithms. Random Forest achieved the best performance (~0.668 accuracy) and was selected as the final mode. A key contribution is the interactive Gradio GUI that allows non-technical users to adjust weather conditions and instantly see how predicted wildfire probability changes, making the model practical and easy to demonstrate. Future work could improve performance by adding better predictors such as vegetation/fuel dryness, land cover, elevation, and ignition-related information, and by using time-based or region-based validation for more realistic generalization. Additional improvements include probability calibration and threshold tuning to balance false alarms versus missed fires.

## References

Kaggle notebook where the dataset was taken: https://www.kaggle.com/datasets/vijayaragulvr/wildfire-prediction?utm_source=chatgpt.com