URBAN CLIMATE MAPPING FOR SHIMLA

Remote Sensing, GIS, Machine Learning and MATLAB Workflow

Proposed technical framework for Shimla City / Shimla Municipal Corporation

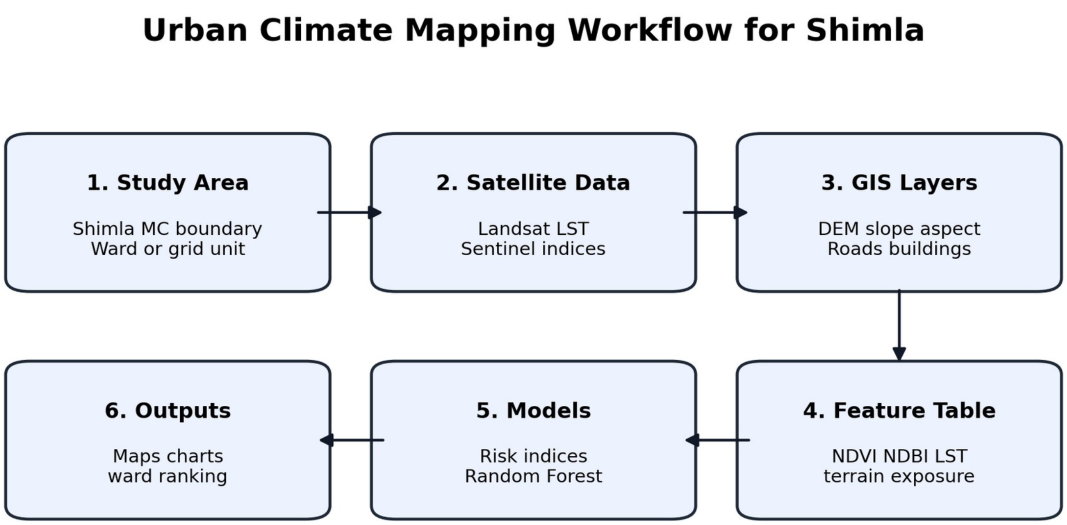

Figure 1. Complete workflow for urban climate mapping and ML-based risk assessment.

Note: The maps included in this document are illustrative sample outputs created for methodology demonstration. Final research maps must be generated from real GEE/GIS data for the Shimla boundary.

1. Project Overview

The proposed work develops an urban climate mapping framework for Shimla using remote sensing, GIS, terrain analysis and machine learning. The main purpose is to map spatial variation in heat stress, terrain-controlled rainfall vulnerability, cloudburst-impact susceptibility and final urban climate risk at grid or ward scale.

Shimla is a suitable study area because it is a Himalayan hill city where urban climate is controlled by both urban morphology and topography. Built-up intensity, vegetation cover, road density and population exposure interact with elevation, slope, aspect, drainage and rainfall concentration.

Component	Purpose
Remote sensing	Derive LST, NDVI, NDBI, NDWI, albedo and land-cover indicators.
GIS and DEM analysis	Generate elevation, slope, aspect, hillshade, flow accumulation and drainage proximity.
Machine learning	Predict LST and classify vulnerable climate zones using environmental and urban features.
Dashboard/frontend	Visualize maps, charts, risk ranking and location-specific prediction.

2. Research Objectives

- Prepare spatial maps of Land Surface Temperature, vegetation cover, built-up intensity and terrain parameters for Shimla.
- Quantify the relationship between LST, NDVI, NDBI, elevation, slope and urban density.
- Develop heat vulnerability, cloudburst-impact susceptibility and final urban climate risk indices.
- Train machine learning models for LST prediction and risk classification.
- Create a GIS/ML workflow that can be converted into a frontend decision-support dashboard.

3. Data Requirements and Technical Sources

Dataset	Spatial role	Main features	Suggested source
Landsat 8/9 Collection 2 Level-2	Thermal and multispectral raster	LST, NDVI, NDBI, NDWI, albedo	Google Earth Engine / USGS
Sentinel-2 MSI	High-resolution multispectral raster	Vegetation, built-up, land cover	Copernicus / GEE
SRTM DEM 30 m	Terrain raster	Elevation, slope, aspect, hillshade	NASA/USGS/GEE
NASA POWER / ERA5-Land	Climate time series/grid	Rainfall, air temperature, humidity, wind, solar radiation	NASA / Copernicus
OpenStreetMap	Vector urban form	Roads, buildings, drainage proxy	OSM / QGIS
Shimla ward boundary	Administrative boundary	Ward-wise aggregation and planning	Municipal/GIS source

Technical note: Landsat Collection 2 Level-2 provides surface reflectance and surface temperature products; Sentinel-2 provides high-resolution multispectral imagery useful for vegetation and land-cover mapping; SRTM provides 30 m elevation data for terrain derivatives; NASA POWER provides solar and meteorological data through API services. See References [1]-[5].

4. Study Area and Mapping Units

The preferred study area is the Shimla Municipal Corporation boundary rather than the full Shimla district. The district includes rural and forest areas, while the urban climate question requires a city or ward-scale analysis.

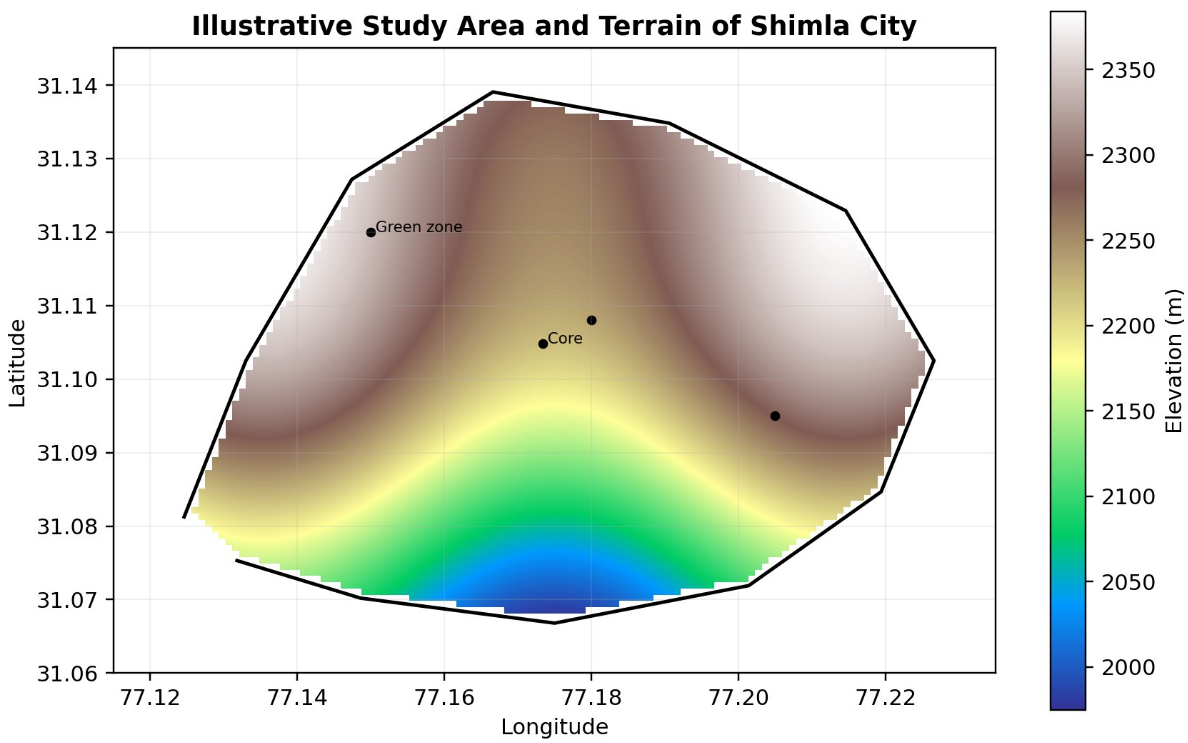

Figure 2. Illustrative Shimla study area and terrain layer. Replace with official boundary and DEM-derived elevation for final analysis.

Mapping unit	Best use	Recommended output
Ward level	Urban planning and administration	Ward-wise vulnerability ranking
Grid level, e.g. 30 m or 100 m	Machine learning and raster risk map	Pixel/grid-based climate risk class
Sample points	MATLAB modelling and validation	CSV feature table for ML

5. Remote Sensing and GIS Methodology

5.1 Satellite preprocessing

1. Define Shimla boundary in GEE/QGIS.
2. Filter Landsat images for pre-monsoon/summer period, such as April-June.
3. Apply QA_PIXEL cloud and cloud-shadow masking.
4. Create seasonal median composite to reduce cloud and noise effects.
5. Clip raster layers to the Shimla boundary and export as GeoTIFF/CSV.

5.2 Feature engineering equations

Feature	Equation / derivation	Interpretation
NDVI	(NIR - Red) / (NIR + Red)	Higher vegetation and cooling capacity

NDBI	(SWIR - NIR) / (SWIR + NIR)	Higher built-up or impervious surface
NDWI	(Green - NIR) / (Green + NIR)	Surface moisture or water influence
LST	ST_B10 x scale factor + offset - 273.15	Surface temperature in deg C

5.3 Terrain and drainage layers

Layer	Why it matters in Shimla
Elevation	Temperature often decreases with altitude and ridge-valley position affects ventilation.
Slope	Steep slopes affect surface heating, runoff and landslide sensitivity.
Aspect	North-facing and south-facing slopes receive different solar exposure.
Hillshade	Shows topographic shadow and illumination patterns.
Flow accumulation	Identifies natural runoff collection zones.
Drainage proximity	Helps map flash-flood and cloudburst-impact susceptible zones.

6. Sample Climate Layers

The following figures show the expected map outputs. These are illustrative sample maps created from synthetic Shimla-like data to demonstrate the workflow. Final figures should be generated from real Landsat, Sentinel, DEM and urban GIS layers.

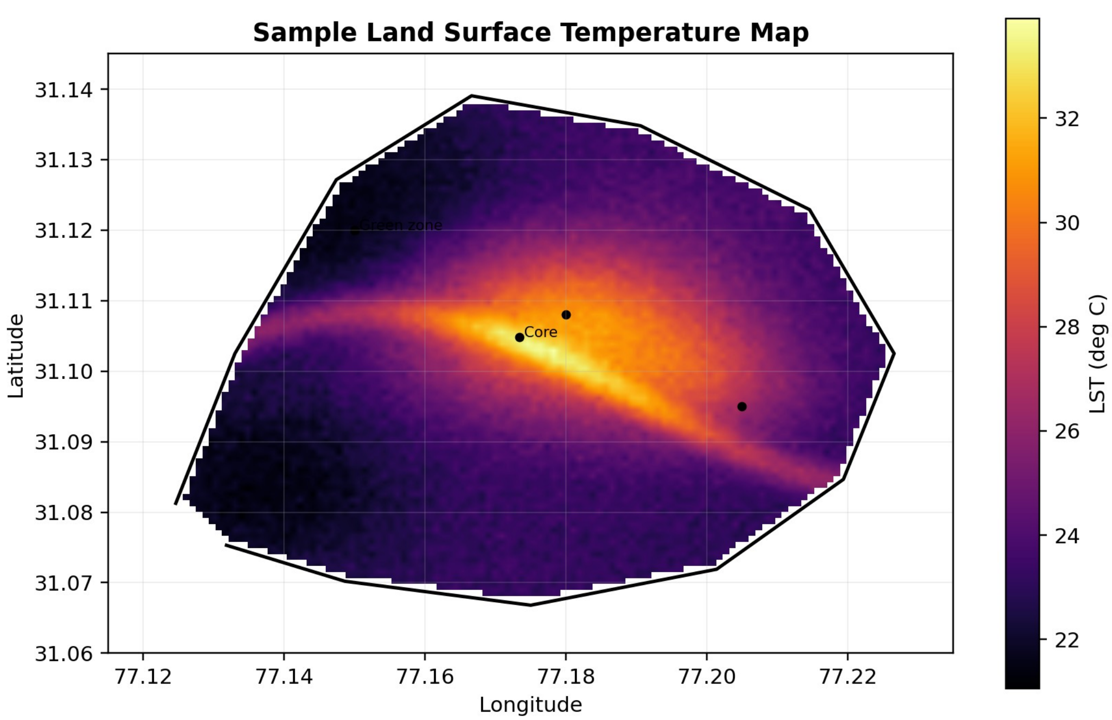

Figure 3. Land Surface Temperature layer.

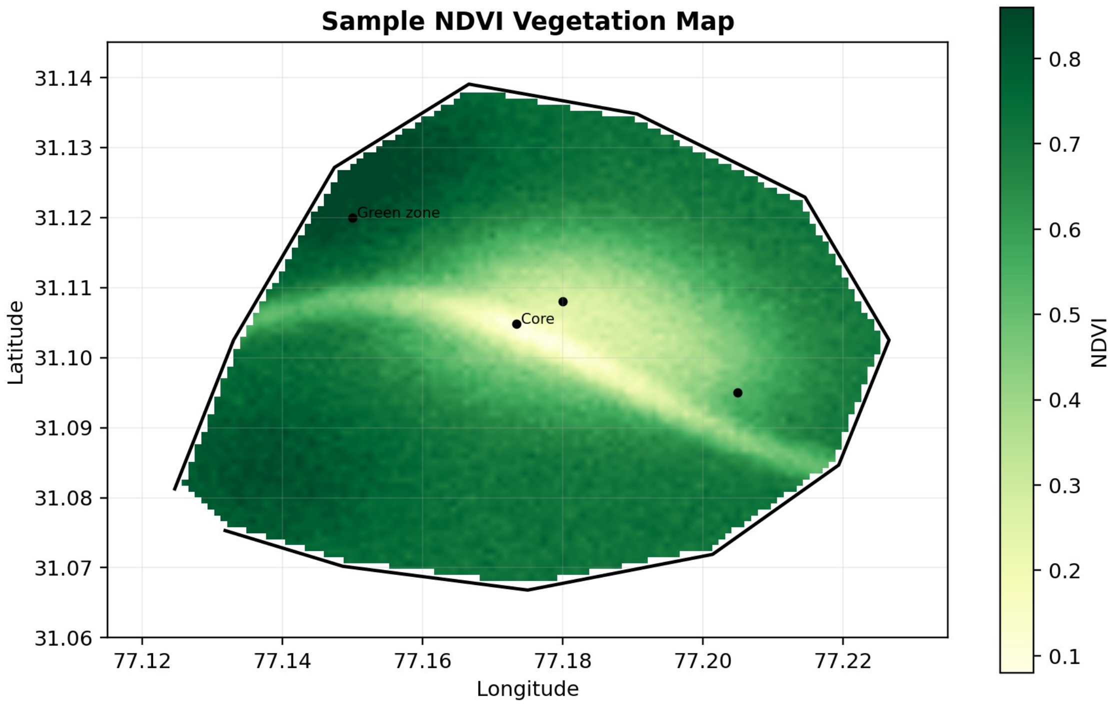

Figure 4. NDVI vegetation layer.

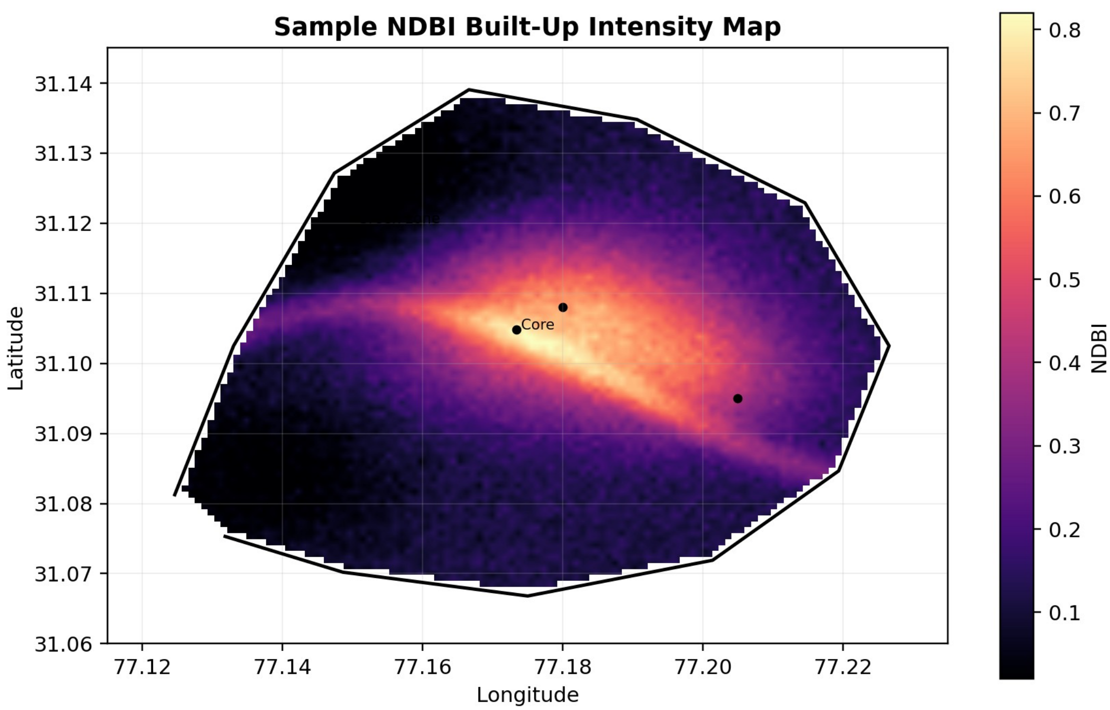

Figure 5. NDBI built-up intensity layer.

7. Urban Climate Risk Indices

7.1 Heat vulnerability index

Heat vulnerability combines thermal hazard, built-up intensity, road/building exposure and vegetation-based cooling capacity.

```
Heat Vulnerability = 0.40*LST_norm + 0.25*NDBI_norm + 0.15*RoadDensity_norm + 0.10*BuildingDensity_norm - 0.10*NDVI_norm
```

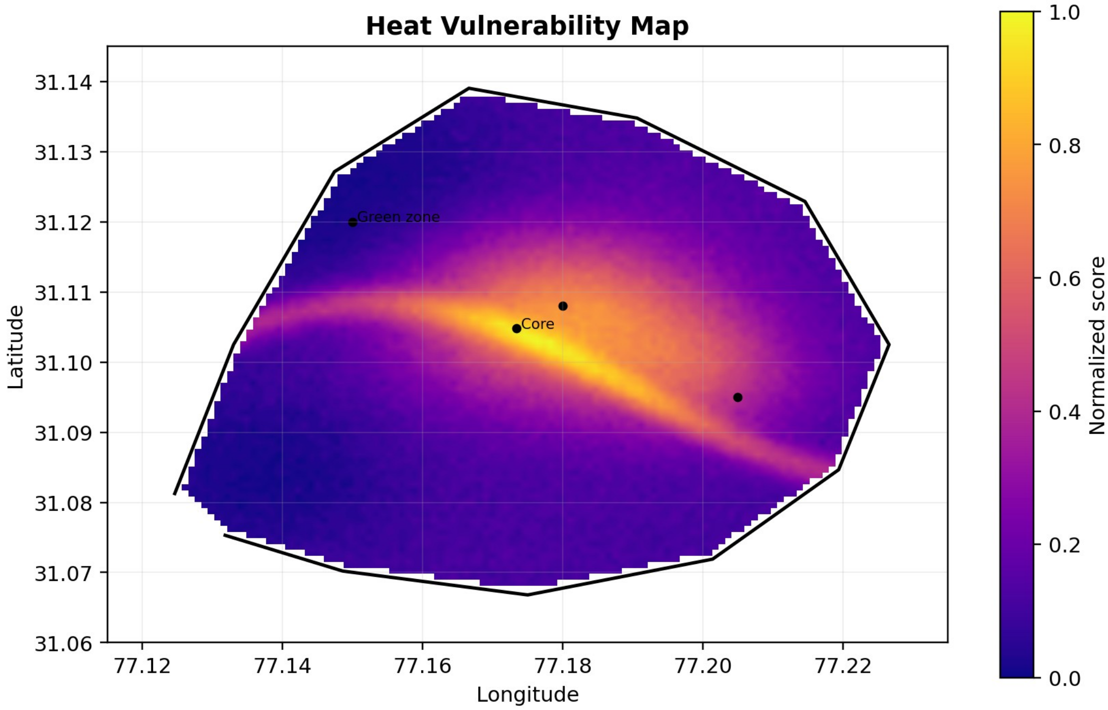

Figure 6. Heat vulnerability map.

7.2 Cloudburst impact susceptibility index

This module does not claim exact cloudburst prediction. It maps areas that may be more susceptible to impacts from short-duration extreme rainfall, such as flash flooding, drainage overflow, road disruption and slope failure. IMD-related official material commonly describes cloudburst conditions as rainfall near 10 cm/hour or more over a small area; exact nowcasting requires dense radar/satellite/ground observations. See References [6]-[7].

```
Cloudburst Susceptibility = 0.25*Rainfall_norm + 0.20*Slope_norm + 0.15*FlowAccumulation_norm + 0.10*RoadDensity_norm +
0.10*BuildingDensity_norm + 0.10*NDBI_norm - 0.10*NDVI_norm
```

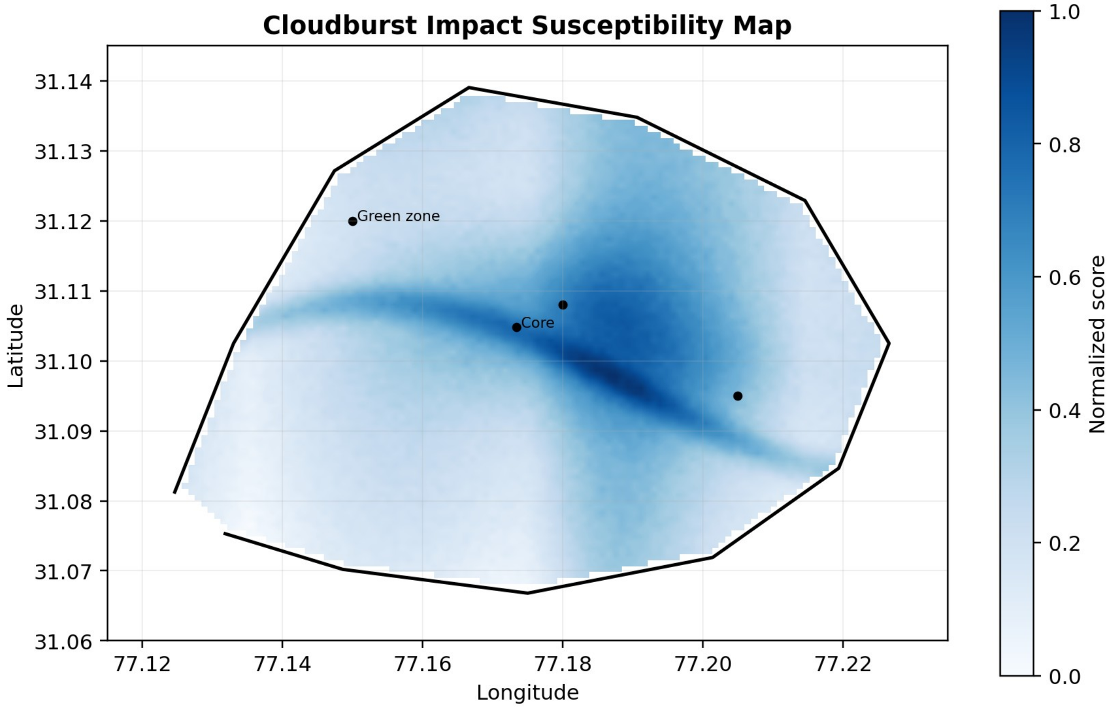

Figure 7. Cloudburst impact susceptibility map.

7.3 Final urban climate risk

```
Final Urban Climate Risk = 0.45*HeatVulnerability + 0.35*CloudburstSusceptibility + 0.20*ExposureIndex
```

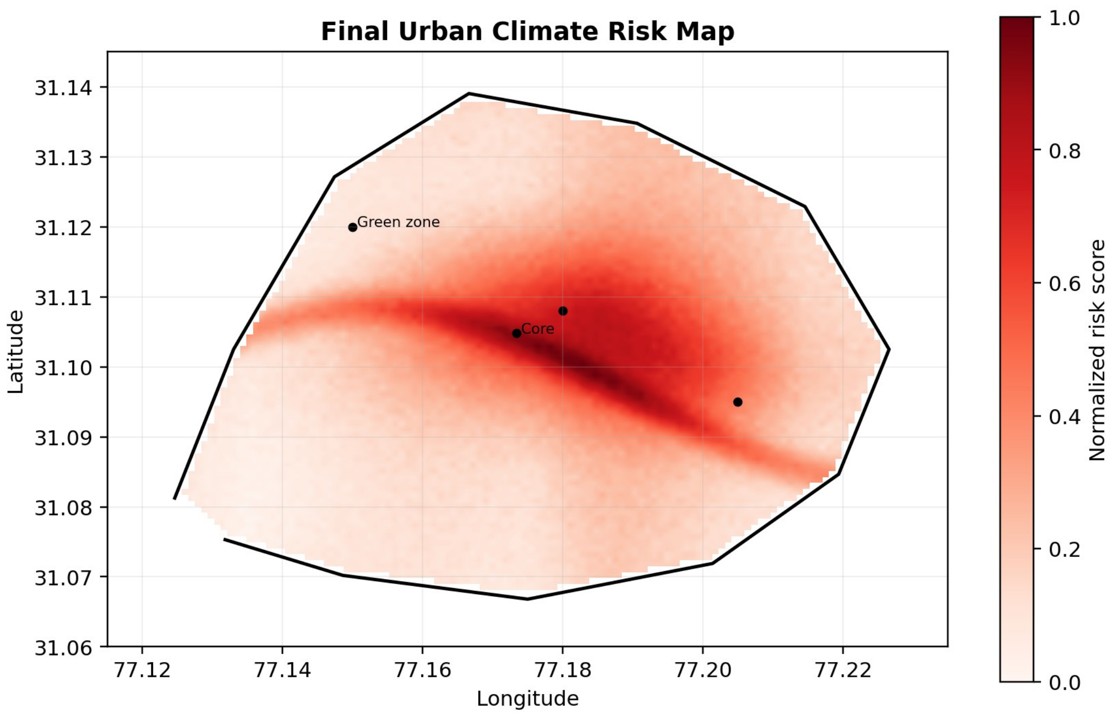

Figure 8. Final urban climate risk map.

Normalized score	Risk class
0.00 - 0.25	Low
0.25 - 0.50	Moderate

0.50 - 0.75	High
0.75 - 1.00	Very High

8. Machine Learning Model

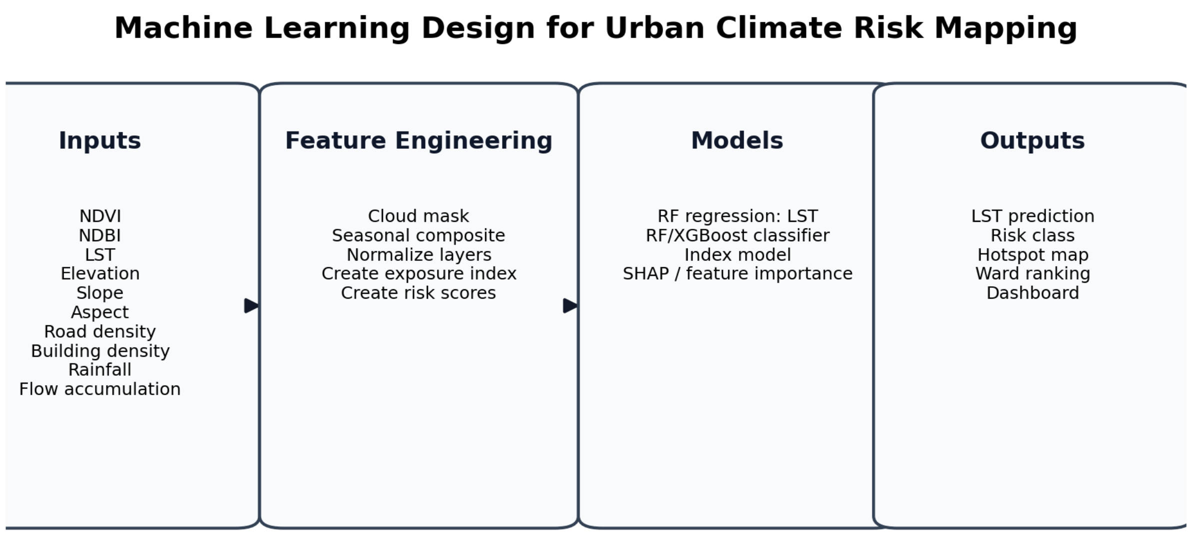

Figure 9. Machine learning structure for LST prediction and risk classification.

The first ML task is regression: predict LST using vegetation, built-up, terrain and exposure variables. The second task is classification: classify each grid or ward into Low, Moderate, High or Very High risk. Random Forest is recommended for the initial model because it captures nonlinear relationships and provides feature importance. XGBoost can be used later for improved accuracy.

Task	Input variables	Output
LST regression	NDVI, NDBI, elevation, slope, road density, building density, rainfall, flow accumulation	Predicted LST in deg C
Heat risk classification	LST, NDVI, NDBI, road/building density, population exposure	Heat risk class
Cloudburst impact susceptibility	Rainfall, slope, flow accumulation, drainage density, built-up, roads, NDVI	Susceptibility score/class
Feature explanation	Trained model features	Variable importance / SHAP effect
Validation	Train-test split, spatial cross-validation, field sensors if available	RMSE, MAE, R2, confusion matrix

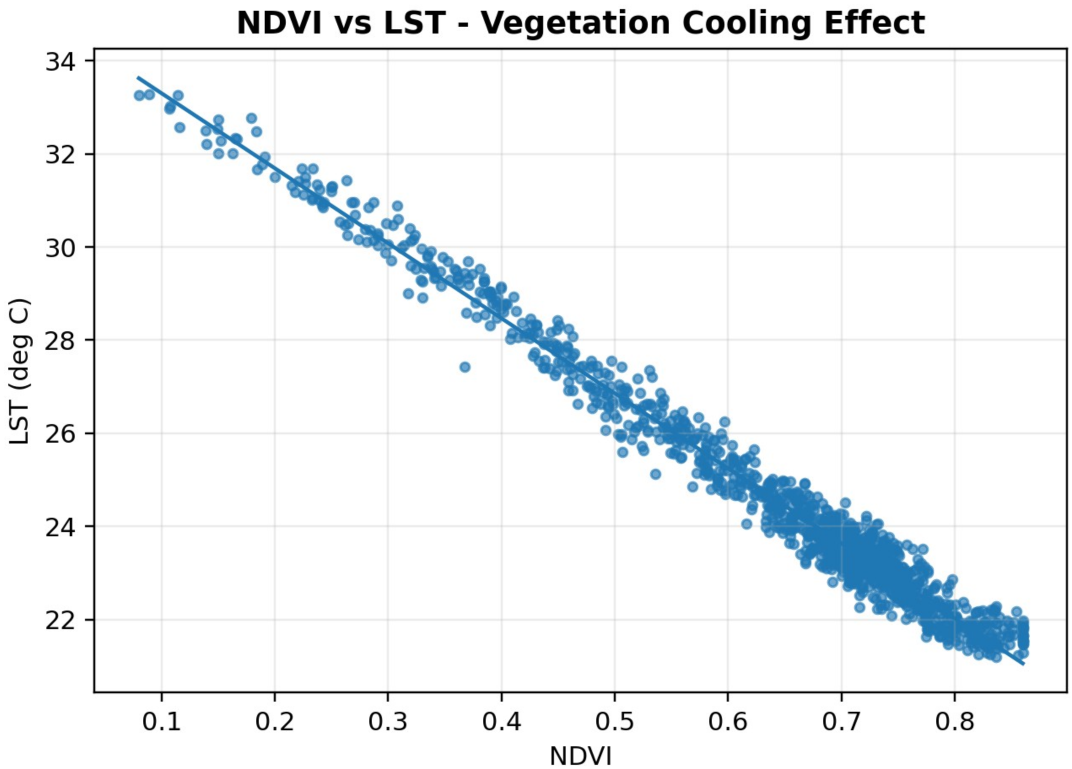

Figure 10. Expected negative relationship between NDVI and LST.

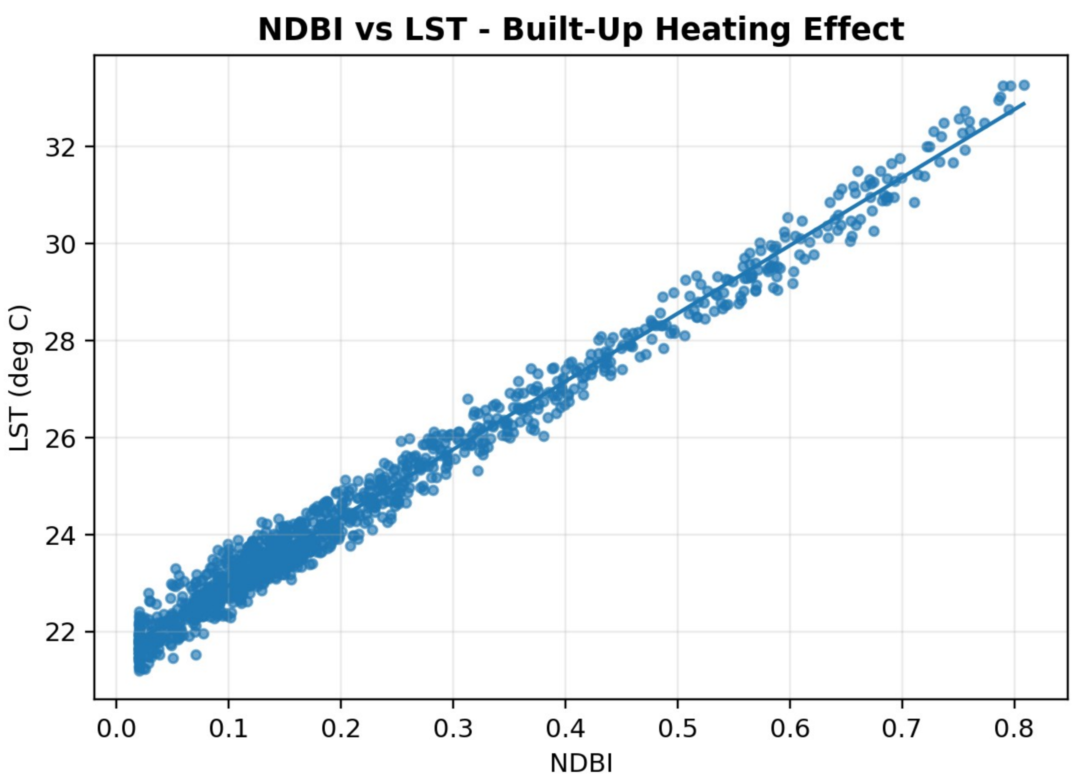

Figure 11. Expected positive relationship between NDBI and LST.

9. MATLAB Code for Urban Climate Mapping

The complete MATLAB script is included in the package as code/shimla_urban_climate_mapping.m. A shortened preview is shown below.

```
clc;
clear;
close all;

%% URBAN CLIMATE MAPPING FOR SHIMLA
% Input CSV fields required:
% Latitude, Longitude, NDVI, NDBI, LST_C, Elevation, Slope,
% RoadDensity, BuildingDensity, Rainfall, FlowAccumulation

%% 1. Load data
data = readtable('sample_shimla_urban_climate_data.csv');

lat = data.Latitude;
lon = data.Longitude;
NDVI = data.NDVI;
NDBI = data.NDBI;
LST  = data.LST_C;
Elevation = data.Elevation;
Slope = data.Slope;
RoadDensity = data.RoadDensity;
BuildingDensity = data.BuildingDensity;
Rainfall = data.Rainfall;
FlowAccumulation = data.FlowAccumulation;

%% 2. Normalization
normalize = @(x) (x - min(x)) ./ (max(x) - min(x));
NDVI_n = normalize(NDVI);
NDBI_n = normalize(NDBI);
LST_n  = normalize(LST);
Slope_n = normalize(Slope);
RoadDensity_n = normalize(RoadDensity);
BuildingDensity_n = normalize(BuildingDensity);
Rainfall_n = normalize(Rainfall);
FlowAccumulation_n = normalize(FlowAccumulation);

%% 3. Heat vulnerability index
HeatVulnerability = 0.40*LST_n + 0.25*NDBI_n + 0.15*RoadDensity_n + ...
    0.10*BuildingDensity_n - 0.10*NDVI_n;
HeatVulnerability = normalize(HeatVulnerability);

%% 4. Cloudburst impact susceptibility index
CloudburstSusceptibility = 0.25*Rainfall_n + 0.20*Slope_n + ...
    0.15*FlowAccumulation_n + 0.10*RoadDensity_n + ...
    0.10*BuildingDensity_n + 0.10*NDBI_n - 0.10*NDVI_n;
CloudburstSusceptibility = normalize(CloudburstSusceptibility);

%% 5. Exposure index
ExposureIndex = 0.40*BuildingDensity_n + 0.35*RoadDensity_n + 0.25*NDBI_n;
ExposureIndex = normalize(ExposureIndex);

%% 6. Final urban climate risk
```

```
FinalRisk = 0.45*HeatVulnerability + 0.35*CloudburstSusceptibility + 0.20*ExposureIndex;
FinalRisk = normalize(FinalRisk);

%% 7. Risk classification
RiskClass = strings(height(data),1);
for i = 1:height(data)
   if FinalRisk(i) < 0.25
      RiskClass(i) = "Low";
   elseif FinalRisk(i) < 0.50
      RiskClass(i) = "Moderate";
   elseif FinalRisk(i) < 0.75
      RiskClass(i) = "High";
   else
      RiskClass(i) = "Very High";
   end
end

%% 8. Export result
Result = table(lat, lon, NDVI, NDBI, LST, Elevation, Slope, ...
   HeatVulnerability, CloudburstSusceptibility, ExposureIndex, FinalRisk, RiskClass);
writetable(Result, 'shimla_urban_climate_risk_output.csv');
disp(Result(1:min(10,height(Result)),:));

%% 9. Spatial scatter maps
figure;
scatter(lon, lat, 45, LST, 'filled');
colorbar; xlabel('Longitude'); ylabel('Latitude');
title('Land Surface Temperature Map of Shimla'); grid on;

figure;
scatter(lon, lat, 45, HeatVulnerability, 'filled');
% ... continued in package code file
```

10. Google Earth Engine Code for Feature Extraction

The complete Google Earth Engine script is included in the package as code/shimla_feature_extraction_gee.js. Replace the rectangular geometry with the official Shimla Municipal Corporation boundary before final analysis.

```
// GOOGLE EARTH ENGINE SCRIPT: SHIMLA URBAN CLIMATE FEATURE EXTRACTION
// Replace the geometry below with the official Shimla Municipal Corporation boundary.

var shimla = ee.Geometry.Rectangle([77.115, 31.060, 77.235, 31.145]);
Map.centerObject(shimla, 11);

// Landsat 8/9 Collection 2 Level-2 preprocessing
function maskL8sr(image) {
  var qa = image.select('QA_PIXEL');
  var cloudShadowBitMask = (1 << 4);
  var cloudsBitMask = (1 << 3);
  var mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0)
      .and(qa.bitwiseAnd(cloudsBitMask).eq(0));
  return image.updateMask(mask);
}
```

```
function addIndices(image) {
  var sr = image.select(['SR_B2','SR_B3','SR_B4','SR_B5','SR_B6','SR_B7'])
      .multiply(0.0000275).add(-0.2);
  var blue = sr.select('SR_B2');
  var green = sr.select('SR_B3');
  var red = sr.select('SR_B4');
  var nir = sr.select('SR_B5');
  var swir1 = sr.select('SR_B6');
  var swir2 = sr.select('SR_B7');

  var ndvi = nir.subtract(red).divide(nir.add(red)).rename('NDVI');
  var ndbi = swir1.subtract(nir).divide(swir1.add(nir)).rename('NDBI');
  var ndwi = green.subtract(nir).divide(green.add(nir)).rename('NDWI');
  var albedo = blue.multiply(0.356).add(red.multiply(0.130))
      .add(nir.multiply(0.373)).add(swir1.multiply(0.085))
      .add(swir2.multiply(0.072)).subtract(0.0018).rename('Albedo');

  // Landsat C2 L2 surface temperature conversion to Kelvin and Celsius.
  var lstK = image.select('ST_B10').multiply(0.00341802).add(149.0);
  var lstC = lstK.subtract(273.15).rename('LST_C');

  return image.addBands([ndvi, ndbi, ndwi, albedo, lstC]);
}

var collection = ee.ImageCollection('LANDSAT/LC08/C02/T1_L2')
    .filterBounds(shimla)
    .filterDate('2024-04-01', '2024-06-30')
    .filter(ee.Filter.lt('CLOUD_COVER', 30))
    .map(maskL8sr)
    .map(addIndices);

var composite = collection.median().clip(shimla);

// Terrain features
var dem = ee.Image('USGS/SRTMGL1_003').clip(shimla);
var terrain = ee.Terrain.products(dem);
var elevation = dem.rename('Elevation');
var slope = terrain.select('slope').rename('Slope');
var aspect = terrain.select('aspect').rename('Aspect');

var featureImage = composite.select(['LST_C','NDVI','NDBI','NDWI','Albedo'])
    .addBands([elevation, slope, aspect]);

Map.addLayer(composite.select('LST_C'), {min: 18, max: 40, palette: ['blue','yellow','red']}, 'LST C');
Map.addLayer(composite.select('NDVI'), {min: 0, max: 0.8, palette: ['brown','yellow','green']}, 'NDVI');
Map.addLayer(composite.select('NDBI'), {min: -0.2, max: 0.6, palette: ['green','white','purple']}, 'NDBI');
Map.addLayer(slope, {min: 0, max: 45, palette: ['white','orange','brown']}, 'Slope');

// Sample gridded points for MATLAB/ML.
var samples = featureImage.sample({
  region: shimla,
  scale: 30,
  numPixels: 5000,
  geometries: true,
  seed: 42
});
```

```
Export.table.toDrive({
  collection: samples,
  description: 'shimla_landsat_terrain_features',
  fileFormat: 'CSV'
});
```

11. Frontend / Dashboard Design

The frontend can be implemented using Streamlit. It should allow CSV upload, show LST and risk maps, plot NDVI-vs-LST and NDBI-vs-LST relationships, display feature importance and provide single-location prediction. A skeleton file is included as code/streamlit_dashboard_skeleton.py.

Dashboard component	Function
LST map	Display thermal hotspots and cool zones.
Heat vulnerability map	Display heat risk pattern.
Cloudburst susceptibility map	Display rainfall/terrain impact-prone zones.
Final risk map	Integrated urban climate risk layer.
Ward-wise ranking	Prioritize interventions at administrative scale.
Prediction form	Estimate risk for user-entered NDVI, NDBI, terrain and exposure values.

12. What Else Can Be Added

- Use official Shimla ward boundaries and generate ward-wise risk scores.
- Use multi-year analysis from 2014 to 2026 to detect urban expansion and UHI change.
- Compare urban core with rural/forest reference areas to calculate UHI intensity.
- Add field validation using handheld/mobile temperature and humidity sensors.
- Add rainfall nowcasting later using radar/satellite sequences if such data is available.
- Add SHAP explainability to show whether NDVI, NDBI, slope, elevation or road density controls model output most strongly.
- Create future planning scenarios such as 10 percent vegetation increase or 15 percent built-up expansion.

13. Professor-Facing Technical Summary

The proposed work develops a remote sensing, GIS and machine learning based urban climate mapping framework for Shimla city. Satellite-derived features such as Land Surface Temperature, NDVI, NDBI, NDWI and albedo are integrated with DEM-derived elevation, slope and aspect, as well as urban form indicators such as road density, building density and population exposure. A regression model predicts LST, while index-based and ML-based classification models generate heat vulnerability, cloudburst-impact susceptibility and integrated urban climate risk maps. Because Shimla is a Himalayan hill city, the method explicitly incorporates terrain and drainage features, making the work stronger than a conventional plain-city urban heat island study. The final output can support ward-wise climate-sensitive planning, hotspot identification and future green-infrastructure scenario analysis.

14. References and Data Source Notes

[1] Google Earth Engine Data Catalog, Landsat 8 Collection 2 Level-2: surface reflectance and surface temperature data.

[2] USGS, Landsat Collection 2 Surface Temperature: surface temperature in Kelvin and LST product description.

[3] Copernicus Data Space / ESA, Sentinel-2 mission: high-resolution multispectral imaging for land services.

[4] Google Earth Engine Data Catalog, NASA SRTM Digital Elevation 30 m: global DEM for terrain analysis.

[5] NASA POWER API documentation: solar and meteorological data from satellite observations and models.

[6] India Meteorological Department / MAUSAM literature: cloudburst as extreme short-duration rainfall, commonly 100 mm or more in one hour.

[7] Parliament of India/official response citing IMD: cloudburst considered at 10 cm/hour or more over approximately 20-30 sq km.

15. Package Contents

File/folder	Content
Technical_Report.docx	Editable technical report with images and code preview.
Technical_Report.pdf	PDF version for sending to professor.
images/	All methodology and sample map figures in PNG format.
MATLAB script	Full urban climate mapping and ML code.
GEE script	Google Earth Engine extraction script.
sample CSV	Sample dataset to test MATLAB/frontend workflow.