// Future FD calculation

// Define the region of interest

var roi = ee.Geometry.Polygon([

[[-30, 30],

[-30, 75],

[60, 75],

[60, 30]]

]);

// Define the start and end dates

var startDate = ee.Date('2025-01-01');

var endDate = ee.Date('2026-01-01');

var dateFilter = ee.Filter.date(startDate, endDate);

// Define the list of models and scenario

var models = ['ACCESS-CM2', 'CMCC-ESM2', 'BCC-CSM2-MR', 'GISS-E2-1-G', 'MPI-ESM1-2-HR'];

var scenario = 'ssp585'; // Using a single scenario

// Create an empty image collection to store the model-averaged 'tasmin' data

var modelMeanCollection = ee.ImageCollection([]);

// Loop through each model

models.forEach(function(model) {

var filtered = ee.ImageCollection('NASA/GDDP-CMIP6')

.filterBounds(roi)

.filter(dateFilter)

.filter(ee.Filter.eq('model', model))

.filter(ee.Filter.eq('scenario', scenario))

.select('tasmin')

.map(function(image) {

// Convert Kelvin to Celsius

var tempCelsius = image.subtract(273.15);

// Mark days with temperature below 0°C

var freezeDay = tempCelsius.lt(0).rename('freeze\_day');

return freezeDay.set('system:time\_start', image.get('system:time\_start'));

});

// Calculate the annual cumulative freeze days for the current model

var modelFreezeDaysSum = filtered.sum();

modelMeanCollection = modelMeanCollection.merge(ee.ImageCollection([modelFreezeDaysSum]));

});

// Compute the average of all models

var freezeDaysAverage = modelMeanCollection.mean().clip(roi);

// Generate the filename string with the specific year

var fileName = 'AnnualFD\_585\_2025';

// Export to Google Drive

Export.image.toDrive({

image: freezeDaysAverage,

description: fileName, // Filename and description are the same

fileNamePrefix: fileName,

folder: 'FD585', // Google Drive folder for export

region: roi,

scale: 27830,

crs: "EPSG:4326",

maxPixels: 1e13

});

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

// Future MinT calculation

// Define the region of interest (ROI)

var roi = ee.Geometry.Polygon([

[[-30, 30],

[-30, 75],

[60, 75],

[60, 30]]

]);

// Define the list of models and scenario

var models = ['ACCESS-CM2', 'CMCC-ESM2', 'BCC-CSM2-MR', 'GISS-E2-1-G', 'MPI-ESM1-2-HR'];

var scenario = 'ssp585'; // Using a single scenario

// Define the range of years

var startYear = 2025;

var endYear = 2100;

// Loop through each year

for (var year = startYear; year <= endYear; year++) {

// Define the start and end dates

var startDate = ee.Date(year + '-01-01');

var endDate = ee.Date((year + 1) + '-01-01');

var dateFilter = ee.Filter.date(startDate, endDate);

// Create an empty image collection to store the model-averaged 'tasmin' data

var modelMeanCollection = ee.ImageCollection([]);

// Loop through each model

models.forEach(function(model) {

var filtered = ee.ImageCollection('NASA/GDDP-CMIP6')

.filterBounds(roi)

.filter(dateFilter)

.filter(ee.Filter.eq('model', model))

.filter(ee.Filter.eq('scenario', scenario))

.select('tasmin')

.map(function(image) {

// Convert Kelvin to Celsius

var tempCelsius = image.subtract(273.15);

return tempCelsius.set('system:time\_start', image.get('system:time\_start'));

});

// Group by month and compute the monthly mean minimum temperature

var monthlyMean = ee.ImageCollection(

ee.List.sequence(1, 12).map(function(month) {

var monthlyImages = filtered.filter(ee.Filter.calendarRange(month, month, 'month')); // Filter by month

return monthlyImages.mean().set('month', month); // Compute the mean for that month

})

);

// Merge the current model’s monthly mean minimum temperature into the collection

modelMeanCollection = modelMeanCollection.merge(monthlyMean);

});

// Compute the average annual monthly mean minimum temperature across all models

var finalMeanTasmin = modelMeanCollection.mean().clip(roi);

// Generate a filename string including the specific year

var fileName = 'AnnualMeanTasmin\_585\_' + year;

// Export to Google Drive

Export.image.toDrive({

image: finalMeanTasmin,

description: fileName, // Same as filename

fileNamePrefix: fileName,

folder: 'MinT\_585', // Google Drive folder for export

region: roi,

scale: 27830,

crs: "EPSG:4326",

maxPixels: 1e13

});

// Print progress for the current year

print('Exported: ' + fileName);

}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%