ASES ON-CHAIN PROTOCOL

METHODOLOGY FOR THE ISSUANCE OF VERIFIED WATER CREDITS

IV. Methodologies V2.0

June 2023

www.ases-eco.com

CONTENTS

Introduction 4

I. Definitions 5

II. Applicability conditions 5

III. Methodological considerations 6

III.1. Application of methodology 6

III.2. Project boundary 6

IV. Baseline scenario 7

Additionality 7

Quantification 8

Remote sensing approach 8

Linear Spectral Mixture Analysis 9

CN determination 9

Calculate surface runoff (Q) and infiltration (F) 10

Calculate evapotranspiration (ET) 11

Calculate change in groundwater storage (S) 12

Field observations and machine learning 12

Field Measurement of Infiltration Rates 13

Machine Learning Models for infiltration Rates Prediction 14

Model Performance Assessment 15

Cross-check of field data-machine learning model with remote sensing model 15

IV. Monitoring 17

Data and parameters used in both validation and monitoring 17

Description of the Monitoring Plan 24

 Sample design 24

 Monitoring plan 25

V. Calculation of water credits 25

Example of calculation of water credits for a specific project 25

Annex 1. Field technique for infiltration measurement with ring infiltrometer 28

Annex 2. Algorithms for infiltration assessment through machine learning (python programming language) 30

Random Forest Regressor Model (RF) 30
Hyperparameters of the Random Forest Model 33
Support Vector Machine (SVM) 34
Hyperparameters of the Support Vector Machine..36
Artificial Neural Network (ANN)...36
Hyperparameters of the Artificial Neural Network 39

INTRODUCTION

The assessment of groundwater recharge and its enhancements through Nature-Based Solutions (NBS) projects plays a vital role in sustainable water resource management. To effectively evaluate the impact of these projects, the aOCP provides a methodology that leverages digital technology, including satellite images and water balance modeling. This innovative approach enables a detailed monitoring of land cover changes and provides valuable insights into the evolution of ecosystem restoration projects as they mature.

The methodology incorporates digital tools to calculate the Curve Number (CN), a key parameter used to estimate infiltration and surface runoff, which directly influences groundwater recharge. By utilizing satellite images, the CN can be determined with improved accuracy, capturing the land cover characteristics and their spatial distribution within the project area. This satellite-based assessment enhances the precision of recharge estimations, enabling a more comprehensive understanding of groundwater dynamics.

Moreover, the integration of the CN calculation with a water balance model strengthens the methodology's analytical capabilities. The water balance model considers various hydrological components such as precipitation, evapotranspiration, surface runoff, and groundwater recharge, providing a sound framework to assess the impacts of NBS projects on groundwater resources.

One notable advantage of this methodology is its ability to monitor and track land cover changes as reforestation projects progress. By regularly analyzing satellite images, the evolution of vegetation cover and related land surface modifications can be closely monitored.

This document's goal is to outline the requirements and provide rationale for the aOCP's use of the methodology and baseline monitoring for Verified Water Credits (VWCs).

This methodology outlines the steps to follow prior to and once projects are registered and Verified Water Credits (VWCs) are issued under the aOCP.

I. DEFINITIONS

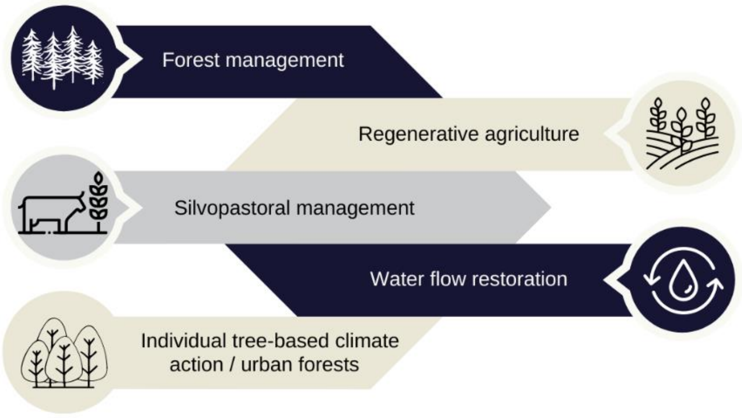
The following definitions also apply to this technique in addition to those in the most recent edition of the Program Definitions:

- **Erosion:** Process in which the top layer of soil, which provides plants with most of the nutrients and water they need, is lost. When this fertile layer is displaced, the productivity of the land decreases
- **Runoff:** Physical process that consists of the runoff of rainwater through the drainage network until it reaches the fluvial network.
- **Precipitation:** process by which water in the atmosphere condenses and falls to the Earth's surface in various forms, such as rain, snow, sleet, or hail.
- **Infiltration rate:** the rate at which water penetrates or seeps into the soil surface.
- **Evapo-transpiration:** combined process of water evaporation from the Earth's surface, including soil, water bodies, and vegetation, as well as the transpiration of water through plant stomata. It represents the loss of water from the land surface and vegetation to the atmosphere.
- **Groundwater recharge:** process by which water infiltrates into the subsurface and replenishes the groundwater reservoir. It is the amount of water that enters the aquifer system through natural or artificial means, such as precipitation, surface runoff, or irrigation.
- **Potential maximum retention:** maximum amount of water that a watershed can retain or store before generating any runoff. It represents the storage capacity of the watershed's soil and vegetation to absorb and retain rainfall.

II. APPLICABILITY CONDITIONS

This methodology is applicable under the following conditions:

a) The type of Project belongs to one of the following types:



b) The Project complies with the standards of the aOCP Program;
c) The Project was developed less than 24 months ago;
d) The Project area has not been degraded, deforested or burned in the last 24 months;

e) If a project area does not meet requirement "d," the project proponent must provide a technical reason arguing that ecological restoration is necessary because the area's biodiversity and environmental services are vulnerable.

f) The Project is likely to produce increase in ground water recharge of at least 1 m^3 in the first 5 years.

III. METHODOLOGICAL CONSIDERATIONS

III.1. APPLICATION OF METHODOLOGY

The projects that are eligible to the application of this aOCP methodology are listed in the following table. These projects correspond to those that will directly or indirectly benefit ecosystems, improving infiltration and hence increasing groundwater recharge.

TABLE 1. APPLICATION OF METHODOLOGY BY PROJECT

Type of project	Use of methodologies				
	Carbon in vegetation	Carbon in soil	Biodiversity	Soil conservation and restoration	Water
Regenerative agriculture					✓
Forest management					✓
Silvopastoral					✓
Urban forest					✓
Water flow restoration					✓

III.2. PROJECT BOUNDARY

The physical delineation and/or geographic area of the project activity shall include adjoining polygons that allow for comparison of project impacts and consideration of natural variation beyond the Project area (figure 1). These polygons are:

- Microbasin where the Project is located,
- Limits of the parcel with land ownership,

- Site of implementation of Project activities,
- Areas where restoration is needed (regardless if it is inside or outside of the land ownership polygon), within the microbasin.

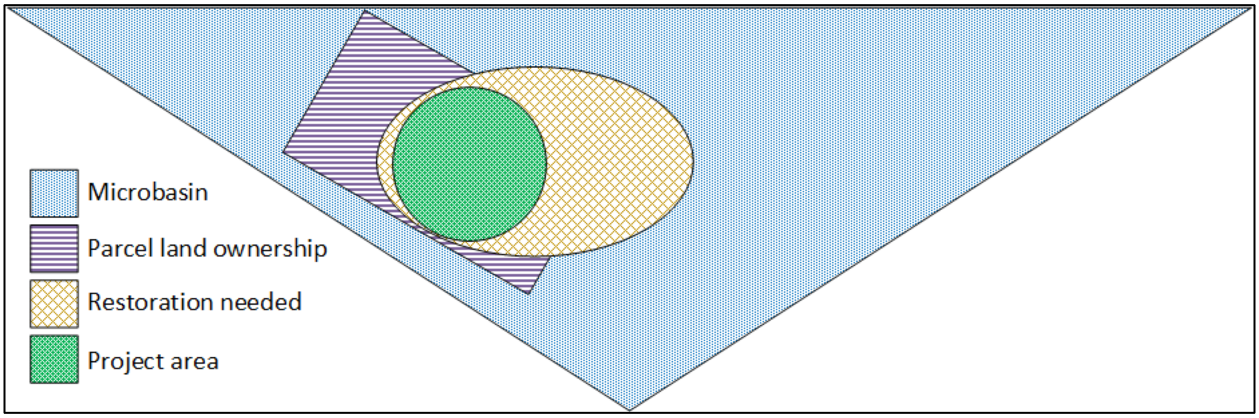


FIGURE 1. POLYGONS TO INCLUDE AS PART OF THE PROJECT BOUNDARY

IV. BASELINE SCENARIO

The baseline scenario represents the expected outcome if the Project activities were not implemented. This baseline scenario should consider factors such as existing land use practices, regulatory requirements, and environmental conditions. It serves as a reference against which the project's impact can be measured. The four polygons comprised within the project boundary will be assessed at the following periods:

- Before deforestation (if it occurred and if satellite images are available for this period)
- Before project implementation.

A counterfactual analysis is conducted to assess what would have happened in the absence of the project. Baseline will be surveyed synchronically via the remote monitoring approach along the life of the project. This will be done in areas within the microbasin with similar conditions at the beginning of the project and which do not undergo anthropogenic land use/land cover change. This will allow the comparison of the natural evolution of the ecosystem hydrologic conditions in the absence of restoration activities.

ADDITIONALITY

Additionality of nature-based solution projects consist in the determination of the genuine environmental benefits resulting from the project's implementation. This assessment ensures that the project's impacts are accurately measured, providing a solid basis for evaluating its effectiveness and supporting Verified Nature Positive Credits issuance.

Additionality can be evidenced by combining the applying the following approach:

- The first step is to establish the baseline scenario.

- By comparing the expected outcomes of the counterfactual scenario with the actual project outcomes, the additional environmental benefits brought about by the nature-based solution project can be determined.
- Additionality assessment can include both quantitative and qualitative indicators. Quantitative indicators may involve measuring changes in groundwater recharge rates, land cover, or other relevant environmental parameters. Qualitative indicators can include social and economic considerations, such as community engagement, job creation, or ecosystem services provided. These indicators help capture the multifaceted impacts of the project and determine if the achieved benefits go beyond what would have occurred naturally or through other interventions.
- Engage with stakeholders and experts to gather their perspectives and insights regarding the additionality of the project. This may involve conducting consultations, expert reviews, or third-party evaluations. Stakeholder input and expert opinions provide valuable perspectives on the project's uniqueness, its contributions to environmental goals, and the extent to which the project goes beyond business-as-usual practices.

QUANTIFICATION

The *aOCP Methodology for groundwater recharge assessment* encompasses two components. The first one leverages remote sensing techniques and utilizes the Curve Number method derived from satellite imagery, coupled with the Thornthwaite-Mather water balance method. By employing these remote assessment tools, it becomes possible to estimate groundwater recharge based on parameters such as land cover, precipitation, and evapotranspiration. This approach enables a cost-effective and efficient assessment of groundwater recharge changes across large areas, facilitating effective monitoring and evaluation of projects’ impacts on groundwater resources.

In addition, to improve the accuracy and precision of recharge estimates, the second part of the methodology integrates field observations into a machine learning model for infiltration estimation. By collecting field data, such as infiltration rates and soil characteristics, and training the machine learning model using these inputs, groundwater recharge estimates obtained via the remote approach can be evaluated.

REMOTE SENSING APPROACH

The data analysis uses the SCS-CN (Soil Conservation Service-Curve Number) method (Mishra & Singh, 2003)to evaluate surface runoff volume, the Turc method (Gudulas et al., 2013)to evaluate evaporation, and the Thornthwaite-Mather water balance method (Pranoto et al., 2019) to evaluate groundwater recharge. The recommended satellite images are from Sentinel-2, since these offer the best spatial resolution available at open source; higher spatial resolution images are also accepted. This method assumes that groundwater recharge is equal to evapotranspiration and surface runoff subtracted from precipitation. The procedure is the following:

LINEAR SPECTRAL MIXTURE ANALYSIS

Obtain V-I-S proportions

Use the LSMA method (Wang et al., 2017) to generalize urban land use types into three basic elements based on the V-I-S model, vegetation, impervious surface, and bare soil. The proportion of impervious surface, vegetation and soil of each pixel will be used to calculate the CN values.

Assess accuracy

The accuracy of the proportions of vegetation, impervious surface and soil components, shall be verified by selecting 50 random points within an area of 300 m × 300 m.

For each sample point, visually interpret on high-resolution images from Google Earth the vegetation, impervious surface and soil. Accuracy of the vegetation, impervious surface and soil maps is assessed by comparing the visual interpretation proportions from Google Earth and LSMA results. Root mean square error (RMSE) is computed to evaluate the accuracy of the un-mixing results. RMSE is a commonly used method for evaluating the difference between simulated and measured values. RMSE can be expressed by:

$$RMSE = \sqrt{\frac{\sum_{\mathrm{i=1}}^{\mathrm{N}}(Xi-Yi)^2}{N}} \quad [1]$$

where *Xi* represents the estimated impervious surface, vegetation, and soil fractions of sample *i* from Sentinel-2 by LSMA; *Yi* is the digitized proportion of *i* from the high-resolution image; and *N* is the number of samples.

CN DETERMINATION

Curve number (CN) is an index developed by the Natural Resource Conservation Service (NRCS), to represent the potential for storm water runoff within a drainage area. The CN method proposed by Fan et al., (2013) is used to calculate composite CN . Each 10 m × 10 m pixel was assumed to be an independent drainage area comprising impervious surface, vegetation and soil only. The composite CN value for each pixel is determined as the area-weighted average of the CN values associated with the impervious surface, vegetation, and soil. The calculation of the composite CN is carried out using the following formula:

$$CNc = Si\ \times CNi + Sv\ \times CNv + Ss\ \times CNs \quad [2]$$

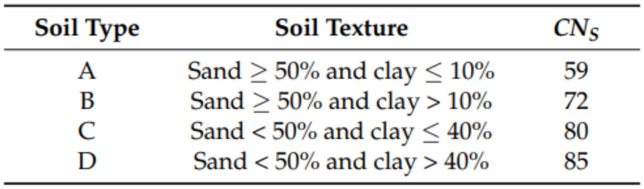
where *CNc* is the composite CN value; *Si* , *Sv,* and *Ss* are fractions of impervious surface, vegetation and soil extracted by the LSMA, respectively; and *CNi* , *CNv*, and *CNs* are the initial CN values of impervious surface, vegetation and soil, respectively.

The composite CN was calculated under the dry antecedent moisture condition (AMC-I).

CNi : a unique value of 98 is assigned to impervious surfaces, according to the lookup table of Technical Release 55 (TR-55) (USACE Hydrologic Engineering Center, n.d.).

CNs: the soil is classified into four hydrologic soil groups (A, B, C, and D) based on the proportion of sand and clay, as shown in Table 1.

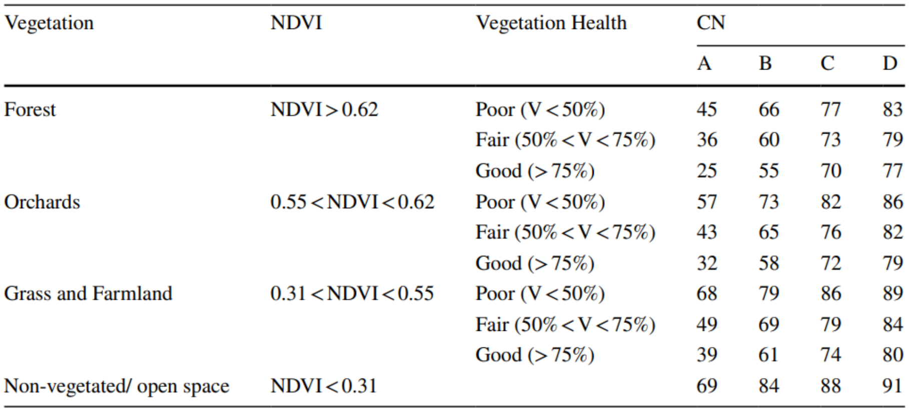
TABLE 1. SOIL TEXTURE CLASSIFICATION AND VALUES OF SOIL CURVE NUMBER (CNS) IN AMC-I (CHUNLIN ET AL., 2018).



Soil Type	Soil Texture	CN_S
A	Sand $\geq$ 50% and clay $\leq$ 10%	59
B	Sand $\geq$ 50% and clay > 10%	72
C	Sand < 50% and clay $\leq$ 40%	80
D	Sand < 50% and clay > 40%	85

CNv : First, calculate the NDVI (Normalized Difference Vegetation Index). Second, vegetation is classified into four categories according to values of NDVI, as shown in Table 2. Select CN for the hydrologic soil group defined in Table 1.

TABLE 2. CURVE NUMBER FOR VEGETATION (CNV) CLASSIFICATION (BERA ET AL., 2021).



Vegetation	NDVI	Vegetation Health	CN			
			A	B	C	D
Forest	NDVI > 0.62	Poor (V < 50%)	45	66	77	83
		Fair (50% < V < 75%)	36	60	73	79
		Good (> 75%)	25	55	70	77
Orchards	0.55 < NDVI < 0.62	Poor (V < 50%)	57	73	82	86
		Fair (50% < V < 75%)	43	65	76	82
		Good (> 75%)	32	58	72	79
Grass and Farmland	0.31 < NDVI < 0.55	Poor (V < 50%)	68	79	86	89
		Fair (50% < V < 75%)	49	69	79	84
		Good (> 75%)	39	61	74	80
Non-vegetated/ open space	NDVI < 0.31		69	84	88	91

If the study region is situated in a hilly terrain, besides the LULC, slope is also a driving factor for surface runoff. Slope correction is performed using the following equation as defined by (Huang et al., 2006)

$$CNsc = \frac{CNc \times (322.79 + 15.63 \times SL)}{SL + 323.52} \qquad [3]$$

Where *CNsc* is slope corrected composite curve number, *CNc* is composite curve number and *SL* is slope rise (in percentage).

CALCULATE SURFACE RUNOFF (Q) AND INFILTRATION (F)

Once curve number is determined, proceed to calculate surface runoff (Q) and infiltration rate (F) by SCS-CN method. The equations used are as follows:

$$Q = \frac{(P - 0.2S)^2}{P + (0.8S)} \qquad [4]$$

$$S = \frac{25400}{CNcs} - 254 \qquad [5]$$

$$Ia = 0.2S \qquad [6]$$

$$F = (P - Ia) - Q \qquad [7]$$

According to SCS-CN method, *Q* is estimated as zero if *P* ≤ *Ia*.

S: groundwater storage, which depends on land cover and soil hydrologic group, using CNsc calculated in the previous section.

Ia: initial abstraction, which is water held up in soil granules at the beginning of rain before infiltration and runoff take place.

F: infiltration rate, which is the addition of water to the soil that occurs after the initial abstraction process.

CALCULATE EVAPOTRANSPIRATION (ET)

ET is the second largest component (after precipitation) of the terrestrial water cycle at the global scale, since ET returns more than 60% of precipitation on land back to the atmosphere and thereby conveys an important constraint on water availability at the land surface. In addition, ET is an important energy flux since land ET uses up more than half of the total solar energy absorbed by land surfaces (Mu et al., 2013).

The FAO Penman-Monteith method is FAO's recommended as the sole ETo method for determining reference evapotranspiration. The algorithm used for the MOD16 data product collection is based on the logic of the Penman-Monteith equation, which includes inputs of daily meteorological reanalysis data along with MODIS remotely sensed data products such as vegetation property dynamics, albedo, and land cover. The total daily ET is the sum of evaporation from the wet canopy surface, the transpiration from the dry canopy surface and the evaporation from the soil surface. Google Earth Engine provides access to this dataset, identified as *The Terra Moderate Resolution Imaging Spectroradiometer (MODIS) MOD16A2GF Version 6.1 Evapotranspiration/Latent Heat Flux (ET/LE) product*.

When MOD16 data is not available or accessible, real evapotranspiration can be calculated from Turc (Gudulas et al., 2013), with the formula:

$$ET = \frac{P}{\sqrt{0.9 + \frac{P^2}{L^2}}} \qquad [8]$$

Where; ET: annual actual evapotranspiration (mm/year), P: annual rainfall (mm/year), t: mean annual temperature (°C), L: thermal indicator, defined by the following equation:

$$L = 300 + 25t + 0.05t^3 \quad [9]$$

CALCULATE CHANGE IN GROUNDWATER STORAGE (S)

The model used to calculate change in ground water recharge is the Thornthwaite-Mather water balance method (Pranoto et al., 2019) with the following equation:

$$P = Q + ET \pm \Delta R \quad [10]$$

Which is derived into:

$$\Delta R = P - Q - ET \quad [11]$$

Where *ΔR* is the change of groundwater storage (mm), *P* is rainfall (mm), *Q* is runoff (mm) and *ET* is evapotranspiration (mm).

A negative *ΔR* indicates deficit, i.e. loss of groundwater, while a positive *ΔR* indicates surplus, i.e. recharge.

FIELD OBSERVATIONS AND MACHINE LEARNING

This procedure integrates field measurements of infiltration rates and other soil parameters (table 2) with machine learning (ML) models to evaluate infiltration in the study area. To calibrate and train machine learning models and more precisely estimate infiltration rates, data from field observations are essential.

In order to predict infiltration rates, the machine learning (ML) models implemented combine the acquisition of remote sensing indexes as well as various soil physical and chemical properties influencing soil infiltration capacity, including soil moisture index (SMI), normalized difference vegetation index (NDVI), bulk density, among others. The soil properties incorporated in the model and their sources are listed in Table . Parameters other than remote sensing indexes shall be determined through analysis performed by a certified laboratory. The Soil Grids data can be used as a reference only, but for the calculation of credits, field measurements shall be used.

TABLE 2. REMOTE SENSING INDEXES AND SOIL PHYSICOCHEMICAL PROPERTIES INCLUDED IN ML MODELS.

No.	Soil property	Units	Source
1	Soil Moisture Index (SMI)	Unitless	Remote Sensing Derived
2	Topographic Wetness Index (TWI)	Unitless	Remote Sensing Derived
3	Normalized Vegetation Index (NDVI)	Unitless	Remote Sensing Derived
4	Bulk Density	cg/cm^3	SoilGrids / Field Observation
5	Sand Content	g/kg	SoilGrids / Field Observation
6	Silt Content	g/kg	SoilGrids / Field Observation
7	Clay Content	g/kg	SoilGrids / Field Observation

8	Coarse Fragment Content	cm^3/dm^3	SoilGrids / Field Observation
9	Cation Exchange Capacity (CEC)	mmol(c)/kg	SoilGrids / Field Observation
10	Organic Carbon Density	cg/kg	SoilGrids / Field Observation
11	Soil Organic Carbon Content	%	SoilGrids / Field Observation
12	Nitrogen Content	cg/kg	SoilGrids / Field Observation

SoilGrids: global predictions for standard numeric soil properties (Poggio et al., 2021).

Field Measurement of Infiltration Rates

Soil Sampling

Semi-stratified sampling technique is used to select sampling locations within the study area. While there is no ideal number of sample size for any given area as it depends mainly on heterogeneity of the study area with respect to soil physico-chemical characteristics, topography and purpose of study, however as a rule of thumb, a minimum of 20 samples per hectare is recommended.

It is important to note the higher the number of soil samples, the higher accuracy in results obtained.

Infiltration Assessment

The mini-disc infiltrometer or ring infiltrometer can be used for infiltration rate assessment. The choice of assessment device however depends on the objective of the study and resources available. Refer to annex 1 for detailed method of infiltration measurement with ring and mini-disc infiltrometer.

Infiltration Rate Computation from Ring Infiltrometer Measurements

To compute infiltration rates from the experiment, the volume of water used will have to be converted to water depth (h), and then divided by the elapsed time of infiltration.

Calculate the area of the infiltrometer used.

$$AI = \pi \times r^2 \tag*{[12]}$$

Where, AI: area of infiltrometer (cm^2); r: radius of infiltrometer (cm);

Calculate the depth (h) of water in cm.

$$h = \frac{V}{AI} \tag*{[13]}$$

Where; h: depth of water (cm); V: =Volume of water infiltrated (cm^3)

Infiltration Rate (cm/sec)* is then computed.

$$Ir = \frac{h}{t} \tag*{[14]}$$

Where; Ir: infiltration Rate (cm/s); t: recorded time for the water to infiltrate (s).

* other units can be used, such as mm/h.

Interpretation of Infiltration Rates

Infiltration Rates obtained after calculation may be compared with literature references based on soil texture category.

Machine Learning Models for Infiltration Rates Prediction

Three machine learning models; Random Forest Regressor, Support Vector Machine, and Artificial Neural Network were developed and tested on the data for prediction of infiltration rates based on 12 input parameters listed in table 1 above. Figure 2 below shows a flowchart of the methodology used.

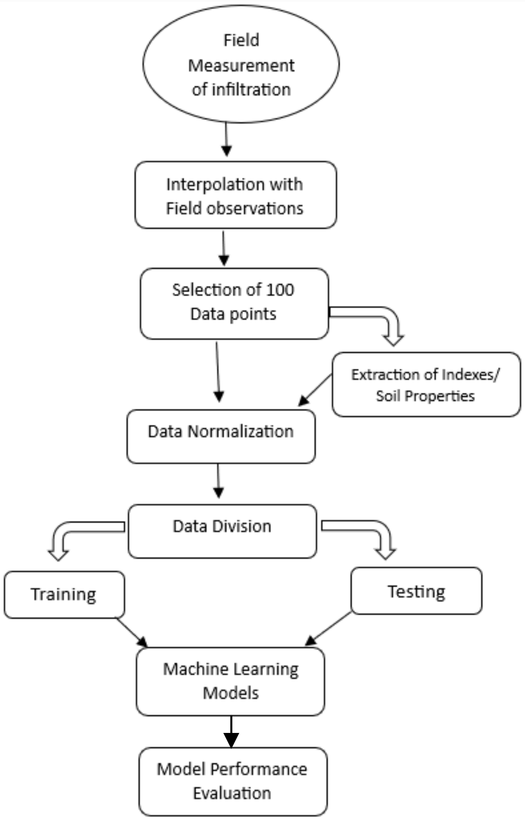

Field Measurement of infiltration
Interpolation with Field observations
Selection of 100 Data points
Extraction of Indexes/ Soil Properties
Data Normalization
Data Division
Training
Testing
Machine Learning Models
Model Performance Evaluation

Figure 2. Flowchart of infiltration Assessment Methodology

Inverse Distance Weighting (IDW) interpolation is carried out with the field observation data to obtain infiltration rates across the entire study area including unsampled points. A total of 100 sample points in addition to the field observations is extracted from the interpolated surface for infiltration rates. For each sample point, the corresponding NDVI, TWI, SWI, as well as the other soil properties listed in table 1 were extracted. Data were then transferred to an excel file for treatment after which it was converted to a comma separated value (.csv) before importation into the models. Data normalization was executed on the data set to eliminate any bias by transforming the variables to give them the same order of magnitude. The data was then divided with 60% used for model training and 40% for model testing. All three models are fitted on the data set and their prediction accuracy is assessed with four main error metrics: Root Mean Square Error (RMSE), Mean Square Error (MAE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2).

Model development was conducted with python programming language. Annex 2 contains the python codes used to develop and execute the model commands.

MODEL PERFORMANCE ASSESSMENT

Four main methods of assessment were implemented in this study. The Root Mean Square Error (RMSE), Mean Square Error (MAE), Mean Absolute Error (MAE), and Coefficient of Determination (R2).

Root Mean Square Error (RMSE)

RMSE measures the average magnitude of the difference between predicted values and the actual values. It is calculated by taking the square root of the average of squared differences between the predicted and actual values. RMSE values range from 0 to infinity where lower values indicate better predictive accuracy and 0 represents a perfect prediction.

Mean Square Error (MAE)

MSE is the average of squared differences between predicted values and actual values. It is calculated by taking the average of squared errors. Similar to RMSE, value ranges from 0 to infinity where lower values of MSE indicate better predictive accuracy.

Mean Absolute Error (MAE)

MAE measures the average absolute difference between predicted values and the actual values. It is calculated by taking the average of absolute differences between the predicted and actual values. The resulting values range between 0 and infinity where lower values of MAE indicate better predictive accuracy.

Coefficient of Determination (R^2)

R^2 measures the proportion of the variance in the dependent variable that can be explained by the independent variables in a regression model. It ranges from 0 to 1, where 0 indicates the model does not explain any variance, and 1 represents a perfect fit. It is important to note that R^2 does not determine the model's accuracy in making predictions. It primarily assesses the goodness-of-fit of the model to the observed data.

CROSS-CHECK OF FIELD DATA-MACHINE LEARNING MODEL WITH REMOTE SENSING MODEL

This stage consists of using the infiltration rates obtained from the machine learning model derived from field observations to calculate runoff, in substitution of the curve number method derived from satellite images. The formula used is:

$$Q = P - I \qquad [15]$$

The accuracy of the calculations obtained through the SCS-CN modified method can be assessed using the RMSE as described in section 2.1.1.

IV. Monitoring

Data and parameters used in both validation and monitoring

Parameter	*Ss*
Data unit	%
Description	Soil cover percentage
Equations	2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	LMSA performed on a composite Sentinel-2 image (median of the monitoring period)
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	*Sv*
Data unit	%
Description	Vegetation cover percentage
Equations	2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	LMSA performed on a composite Sentinel-2 image (median of the monitoring period)
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so

QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	*Si*
Data unit	%
Description	Impervious cover percentage
Equations	2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	LMSA performed on a composite Sentinel-2 image (median of the monitoring period)
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	NDVI
Data unit	Unitless
Description	Normalized Difference Vegetation Index
Equations	Table 2
Source of data	Calculated for each monitoring period. The least cloudy image within a 2-week window at the end of each season will be selected, i.e. days 7-21 in March, June, September and December.

Calculation method or default value applied	NDVI performed on a composite Sentinel-2 image.
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of vegetation curve number
Comments	

Parameter	CNv
Data unit	Unitless
Description	Vegetation Curve Number
Equations	Eq.2 and Table 2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Classification according to tables 1 and 2.
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	CNs
Data unit	Unitless
Description	Soil Curve Number

Equations	Table 1 and Eq.2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Classification according to table 1.
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	CNi
Data unit	Unitless
Description	Impervious Curve Number
Equations	Eq.2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Default value of 98, according to the SCS TR-55 Table 2-2a – Runoff curve numbers for urban areas[1]
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

[1] https://www.hec.usace.army.mil/confluence/hmsdocs/hmstrm/cn-tables

Parameter	CNi
Data unit	Unitless
Description	Impervious Curve Number
Equations	Eq.2
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Default value of 98, according to the SCS TR-55 Table 2-2a – Runoff curve numbers for urban areas[2]
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of composite curve number
Comments	

Parameter	S
Data unit	Millimeters (mm)
Description	Maximum potential storage
Equations	Eq. 4, 5 and 6
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Equation 5.
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation

[2] https://www.hec.usace.army.mil/confluence/hmsdocs/hmstrm/cn-tables

Purpose of data	Input for the calculation of runoff (Q)
Comments	

Parameter	Ia
Data unit	Millimeters (mm)
Description	Initial abstraction
Equations	Eq. 6 and 7
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Equation 6.
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of infiltration (Q)
Comments	

Parameter	F
Data unit	Millimeters (mm)
Description	Infiltration
Equations	Eq. 7 and 14
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Equation 7
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so

QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of runoff (Q)
Comments	

Parameter	P
Data unit	Millimeters (mm)
Description	Rainfall
Equations	Eq. 4, 7, 8, 10, 11 and 15
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Calculated on Google Earth Engine from the “CHIRPS Daily: Climate Hazards Group InfraRed Precipitation With Station Data (Version 2.0 Final)” dataset (Funk et al., 2015).
Frequency of monitoring/recording	Calculate once and use the same value for the project lifecycle. The period shall be the 30 years before the start of the project. For instance, if the project starts on June 2023, the dataset will comprise 01-01-1992 to 31-12-2022.
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of runoff (Q), infiltration (F) and groundwater recharge (ΔR)
Comments	

Parameter	Q
Data unit	Millimeters (mm)
Description	Runoff
Equations	Eq. 4, 7, 10, 11 and 15

Source of data	Calculated for each monitoring period
Calculation method or default value applied	Equation 4
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of infiltration (F) and groundwater recharge (ΔR)
Comments	

Parameter	ET
Data unit	Millimeters (mm)
Description	Evapotranspiration
Equations	Eq. 8, 9, 10 and 11
Source of data	Calculated for each monitoring period
Calculation method or default value applied	Equation 8
Frequency of monitoring/recording	Quarterly, or different if the monitoring plan establishes so
QA/QC procedures to be applied	Technical verification by repetition of the calculation
Purpose of data	Input for the calculation of change of groundwater storage (ΔR)
Comments	

DESCRIPTION OF THE MONITORING PLAN

- **SAMPLE DESIGN**

The remote sensing approach implies the assessment of the whole Project area and the comparison with the whole microbasin given that GIS are used to analyze the satellite images and other layers in raster format.

Sampling for the field measurement approach is explained in section 7.2.1.

- **MONITORING PLAN**

Remote sensing monitoring is to be conducted quarterly in alignment to the aOCP strategy for close follow up of the evolution of projects and quick decision making in case of unintended events.

Field measurements will be performed on a yearly basis in order to run the machine learning model and compare with the results issued from the remote sensing approach.

V. CALCULATION OF WATER CREDITS

The potential for generation of water credits is calculated based on the expected change in groundwater recharge or infiltration (ΔR). Assuming that the project leads to the restoration of the Project area to optimal conditions, the potential improvement in ΔR, is calculated as the difference between ΔR before project implementation and ΔR when the project reaches the expected results and comparing it to the expected outcome in the absence of Project activities. One way to forecast the expected results is using the values of ΔR in areas within the region where ecosystem is in optimum condition and/or the state of the Project area before it was degraded. The following section presents an example of calculation of water credits for a project.

EXAMPLE OF CALCULATION OF WATER CREDITS FOR A SPECIFIC PROJECT

If the project leads to the restoration of the Project area in 30 years, it will improve from the current -351 mm up to -336 mm per year. This calculation is based on a linear progression. Each mm in ΔR equals 1 L/m^2 or 1 m^3/ha. In this example, ΔR is negative because there is a deficit, meaning that evapo-transpiration and run-off are higher than precipitation.

Table 3 presents the calculation of water credits along the life of the project. Column *Infiltration Project (mm = L/m^2)* shows the modelled evolution in the project area. Column *Infiltration NoProject (mm = L/m^2)* shows modelled evolution in the counterfactual area, that is an area were no project activities. Column *Water gained (m^3/ha)* shows the difference in water infiltration in the water area respect to the counterfactual area, indicating the additional volume of water that has infiltrated, presumably due to the implementation of project activities. This gives the number of credits to issue each year per hectare. In this example, the accumulated volume of water (in m^3) and, therefore, the number of credits along the 30-year period of the Project will be 11,698 per hectare.

If infiltration remains constant from year 31 and on, the project will generate 780 credits yearly. However, if the project improves further the Groundwater recharge, the number of yearly credits will be higher.

TABLE 3. EXAMPLE OF CALCULATION OF WATER CREDITS

Year	Infiltration Project (mm = L/m^2)	Infiltration NoProject (mm = L/m^2)	Water gained (m^3/ha)
0	-451	-451	0
1	-451	-451	0
2	-447	-449	27
3	-443	-448	54
4	-439	-447	81
5	-435	-445	108
6	-431	-444	134
7	-427	-443	161
8	-423	-442	188
9	-419	-440	215
10	-415	-439	242
11	-411	-438	269
12	-407	-437	296
13	-403	-435	323
14	-399	-434	350
15	-395	-433	376
16	-391	-432	403
17	-387	-430	430
18	-383	-429	457
19	-379	-428	484
20	-375	-427	511
21	-372	-425	538
22	-368	-424	565
23	-364	-423	592
24	-360	-422	619
25	-356	-420	645
26	-352	-419	672
27	-348	-418	699
28	-344	-416	726
29	-340	-415	753
30	-336	-414	780
TOTAL			11698

BIBLIOGRAPHY

Funk, C., Peterson, P., Landsfeld, M., Pedreros, D., Verdin, J., Shukla, S., Husak, G., Rowland, J., Harrison, L., Hoell, A., & Michaelsen, J. (2015). The climate hazards infrared precipitation with stations—a new environmental record for monitoring extremes. *Scientific Data 2015 2:1*, *2*(1), 1–21. https://doi.org/10.1038/sdata.2015.66

Gudulas, K., Voudouris, K., Soulios, G., & Dimopoulos, G. (2013). Comparison of different methods to estimate actual evapotranspiration and hydrologic balance. *New Pub: Balaban*, *51*(13–15), 2945–2954. https://doi.org/10.1080/19443994.2012.748443

Mishra, S. K., & Singh, V. P. (2003). *Soil Conservation Service Curve Number (SCS-CN) Methodology*. *42*. https://doi.org/10.1007/978-94-017-0147-1

Mu, Q., Zhao, M., & Running, S. W. (2013). *MODIS Global Terrestrial Evapotranspiration (ET) Product*.

Poggio, L., De Sousa, L. M., Batjes, N. H., Heuvelink, G. B. M., Kempen, B., Ribeiro, E., & Rossiter, D. (2021). SoilGrids 2.0: Producing soil information for the globe with quantified spatial uncertainty. *SOIL*, *7*(1), 217–240. https://doi.org/10.5194/SOIL-7-217-2021

Pranoto, R., Endang Hadi, R., Sopandi, M., Harumansah, R., Saptomo, S. K., & Darmawan, A. (2019). Water Balance Prediction as Impact Land Use Change By GIS Based SCS-CN and Thornhtwaite-Mather Method. *2019 5th International Conference on Computing Engineering and Design (ICCED)*. https://doi.org/10.1109/ICCED46541.2019.9161101

USACE Hydrologic Engineering Center. (n.d.). *SCS Curve Number Loss Model*. HEC-HMS Technical Reference Manual. Retrieved June 19, 2023, from https://www.hec.usace.army.mil/confluence/hmsdocs/hmstrm/infiltration-and-runoff-volume/scs-curve-number-loss-model

Wang, J., Wu, Z., Wu, C., Cao, Z., Fan, W., & Tarolli, P. (2017). Improving impervious surface estimation: an integrated method of classification and regression trees (CART) and linear spectral mixture analysis (LSMA) based on error analysis. *Https://Doi.Org/10.1080/15481603.2017.1417690*, *55*(4), 583–603. https://doi.org/10.1080/15481603.2017.1417690

ANNEX 1. FIELD TECHNIQUE FOR INFILTRATION MEASUREMENT WITH RING INFILTROMETER

1. At each sample location, clip any plants on the site down to ground level, being careful not to disturb the soil.
2. The soil should typically be pre-wetted to a moisture level throughout the profile prior to conducting the experiment. Pre-wetting the soil ensures that it is at or near field capacity, meaning it is adequately moist but not saturated/waterlogged.

   In dry soils, water naturally infiltrates rapidly. This may cause an overestimation of infiltration rates.

   This step can be skipped for already moist soils which experienced some irrigation or rainfall prior to conduction of the experiment.
3. The metal ring should be marked at regular intervals (at least 2 markings) on the inside to ensure ease of measuring the drop of water height at each point and a 15 cm. mark from the bottom on the outside of the ring.
4. Insert the ring, until it reaches the 15 cm. depth mark in the soil. If the terrain has slope, the 15 cm mark shall be at the level of the soil on the lowest side of the ring (see figure 1).

   This is because flow may move laterally especially if the rings are set only a short depth into the soil.
5. Seal any large gaps along the exterior edges of the ring with soil taking care not to disturb the surface of soil inside the ring.
6. Gently fill the ring with water, being careful not to stir up the soil, until the level reaches the upper line drawn on the inside of the ring.
7. Measure with the help of a stopwatch and record the time taken for the water level to drop to each line marked.
8. Refill the ring with water and repeat the measurements several times until the time of infiltration is the same as on the previous measurement.

   At least two infiltration tests should be carried out at each sample point to make sure accurate results are obtained.

   As more water replaces the air in the pores, the water from the soil surface infiltrates more slowly and eventually reaches a steady rate from which the basic infiltration rate of the soil can be obtained.

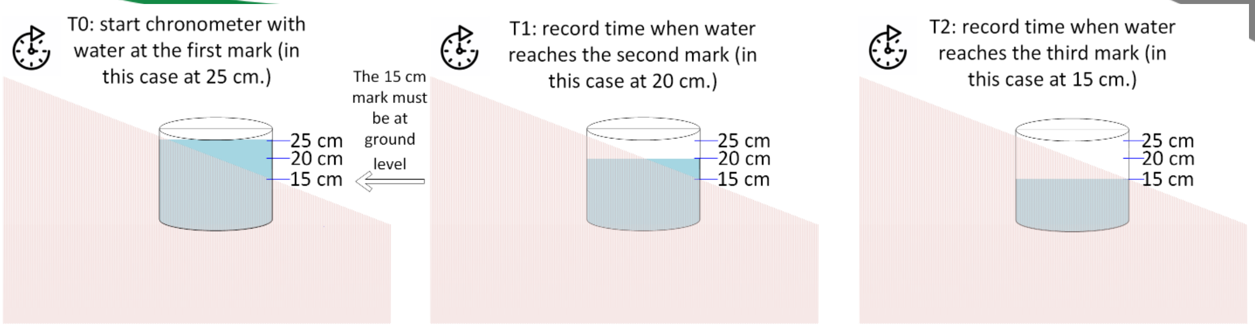

T0: start chronometer with water at the first mark (in this case at 25 cm.)
25 cm
20 cm
15 cm
The 15 cm mark must be at ground level
T1: record time when water reaches the second mark (in this case at 20 cm.)
25 cm
20 cm
15 cm
T2: record time when water reaches the third mark (in this case at 15 cm.)
25 cm
20 cm
15 cm

Figure 1. Procedure for measuring water infiltration on the field.

ANNEX 2. ALGORITHMS FOR INFILTRATION ASSESSMENT THROUGH MACHINE LEARNING (PYTHON PROGRAMMING LANGUAGE)

RANDOM FOREST REGRESSOR MODEL (RF)

```
#Importation of relevant Libraries#
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.ensemble import RandomForestRegressor
from sklearn import preprocessing
================================================================================
=====
#importation of dataset
df = pd.read_csv(‘filepath.csv', delimiter= ",")
df
# Normalization of input parameters
scaler = preprocessing.MinMaxScaler((0,1))
df['TWI']= scaler.fit_transform(df[['TWI']])
df['NDVI']= scaler.fit_transform(df[['NDVI']])
df['SMI']= scaler.fit_transform(df[['SMI']])
df['Bulk_Density']= scaler.fit_transform(df[['Bulk_Density']])
df['Sand']= scaler.fit_transform(df[['Sand']])
df['Silt']= scaler.fit_transform(df[['Silt']])
df['Clay']= scaler.fit_transform(df[['Clay']])
df['Coarse_Frag']= scaler.fit_transform(df[['Coarse_Frag']])
df['CEC']= scaler.fit_transform(df[['CEC']])
df['Nitrogen']= scaler.fit_transform(df[['Nitrogen']])
df['SOC_Stock']= scaler.fit_transform(df[['SOC_Stock']])
df['Organic_Carbon']= scaler.fit_transform(df[['Organic_Carbon']])
#Extraction of independent (input) and dependent (output) variables into X and Y colums respectively.
```

```
x = df.iloc[:,:-1]

y = df.iloc[:,-1]

#To visualize the first few columns of the input variables.

x.head()

#Data division into train and test splits.

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 1.0/3.0, random_state = 10)

#Defining hyperparameters of the model.

model = RandomForestRegressor(n_estimators=100, max_depth= 10, max_features=’sqrt’, criterion='squared_error')

=======================================================================================

##This block of codes for Gridsearch algorithm may be included to help find optimum parameters for tuning of model.

# Defining the parameter grid for Grid Search

param_grid = {

   'n_estimators': [ 300,400,600],  # Number of trees in the forest

   'max_depth': [None, 5, 10],  # Maximum depth of the trees

   'min_samples_split': [ 4, 6,8,10,20],  # Minimum number of samples required to split a node

   'min_samples_leaf': [ 10, 6,8,4],  # Minimum number of samples required at each leaf node

}

model = RandomForestRegressor( max_features='sqrt', criterion='squared_error')

#model.fit(x_train,y_train)

# Perform Grid Search to find the best hyperparameters

grid_search = GridSearchCV(estimator=model, param_grid=param_grid, cv=5)

grid_search.fit(x_train, y_train)


# Retrieve the best hyperparameters and the best model

best_params = grid_search.best_params_

best_model = grid_search.best_estimator_

# Evaluate the best model on the test set

score = best_model.score(x_test, y_test)
```

```
print("Best Model Score:", score)
print("Best Hyperparameters:", best_params)
==============================================================================

#fitting the model on the training dataset
model.fit(x_train,y_train)
#importation of libraires for model performance assessment
from sklearn import metrics
from sklearn.metrics import r2_score
#Prediction test on test dataset
y_pred = model.predict(x_test)
#Model Performance Assessment with R2, MAE, MSE, and RMSE methods.
R_square = r2_score(y_test, y_pred)
print('Coefficient of Determination', R_square)
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
#imports libraries to assess the importance of each input variable.
from sklearn.inspection import permutation_importance
from matplotlib import pyplot as plt
# To visualize the importance of each input variable.
variable_importance = model.feature_importances_
variable_importance
sorted_indices = np.argsort(variable_importance)[::-1]
sorted_indices
features = df.columns
#plots a graph of variable importance with the following attributes ie. Title, color etc.
plt.title('Variable Importance')
plt.barh(range(len(sorted_indices)),variable_importance[sorted_indices],color='b',align='center')
plt.yticks(range(len(sorted_indices)),[features[i] for i in sorted_indices])
plt.xlabel('Relative Importance')
plt.show
#Percentage error computation
errors = (y_test - y_pred) / y_test * 100
#To plot the results of the error assessment for visualization purposes.
plt.scatter(range(len(errors)), errors)
plt.axhline(y=0, color='g', linestyle='--')
plt.axhline(y=np.mean(errors), color='b', linestyle='-')
plt.axhline(y=np.mean(errors) + np.std(errors), color='r', linestyle='-.')
plt.axhline(y=np.mean(errors) - np.std(errors), color='r', linestyle='-.')
plt.xlabel('Test Sample')
plt.ylabel('Error (%)')
plt.title(' Random Forest Model Evaluation')
plt.show()
```

Hyperparameters of the Random Forest Model

- n_estimators; This parameter controls the number of decision trees in the random forest. Generally, a higher number of trees can improve performance, but at the cost of increased computation time for larger datasets.
- max_depth; This parameter sets the maximum depth of each decision tree in the random forest. A deeper tree can capture more complex relationships but may also lead to overfitting of model. The default value is set to None, None specifies that the nodes inside the tree will continue to grow until all leaves become pure.
- max_features; This parameter determines the number of features to consider when looking for the best split at each node. The choice of this parameter depends also on the dimensionality of the input data (n_features). It can take five values “auto“, “sqrt“, “log2”, int or float and None.

auto: considers max_features = sqrt(n_features)

sqrt: considers max_features = sqrt(n_features), it is same as auto

log2: considers max_features = log2(n_features)

None: considers max_features = n_features

For regression problems, a common practice is to use the square root (‘sqrt’) or logarithm of the total number of features.

- min_samples_split: The minimum number of samples required to split an internal node. It helps control the tree's depth and prevents overfitting. An optimum range for this parameter is typically between 2 and 20.
- random_state : This parameter controls both the randomness of the bootstrapping of the samples used when building trees and the sampling of the features to consider when looking for the best split at each node.

Hyperparameter tuning is about finding a set of optimal hyperparameter values which maximizes the model's performance, minimizes loss and produces better outputs.

Support Vector Machine (SVM)

```
#Importation of relevant Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn import preprocessing
from sklearn.svm import SVR
from sklearn import metrics
#importation of dataset
df = pd.read_csv('D:\Vee_Internship\ML_Models\Infiltration_ML_Data_3.csv', delimiter= ",")
df
# Normalization of input parameters
scaler = preprocessing.MinMaxScaler((0,1))
df['TWI']= scaler.fit_transform(df[['TWI']])
df['NDVI']= scaler.fit_transform(df[['NDVI']])
df['SMI']= scaler.fit_transform(df[['SMI']])
df['Bulk_Density']= scaler.fit_transform(df[['Bulk_Density']])
df['Sand']= scaler.fit_transform(df[['Sand']])
df['Silt']= scaler.fit_transform(df[['Silt']])
df['Clay']= scaler.fit_transform(df[['Clay']])
df['Coarse_Frag']= scaler.fit_transform(df[['Coarse_Frag']])
df['CEC']= scaler.fit_transform(df[['CEC']])
df['Nitrogen']= scaler.fit_transform(df[['Nitrogen']])
```

```
df['SOC_Stock']= scaler.fit_transform(df[['SOC_Stock']])

df['Organic_Carbon']= scaler.fit_transform(df[['Organic_Carbon']])

#Extraction of independent (input) and dependent (output) variables into X and Y colums respectively.

x = df.iloc[:,:-1]

y = df.iloc[:,-1]

#To visualize the first few columns of the input variables.

x.head()

#Data division into train and test splits.

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 1.0/3.0, random_state = 7)

#Fitting the SVR model to the training dataset#

regressor = SVR(kernel = 'rbf' , C=1.0, gamma=0.01)

regressor.fit(x_train, y_train)

#Prediction test on test dataset

y_pred = regressor.predict(x_test)

y_pred

#Model Performance Assessment with R2, MAE, MSE, and RMSE methods.

MAE = metrics.mean_absolute_error

MSE = metrics.mean_squared_error

print('Mean Absolute Error:', MAE(y_test, y_pred))

print('Mean Squared Error:', MSE(y_test, y_pred))

print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))

from sklearn.metrics import r2_score

R_square = r2_score(y_test, y_pred)

print('Coefficient of Determination', R_square)
```

HYPERPARAMETERS OF THE SUPPORT VECTOR MACHINE

- kernel (Kernel Function): The kernel function specifies the type of non-linear transformation applied to the input features. Some kernel functions include;

Linear Kernel: Uses linear regression. To apply this, set kernel='linear'.

Polynomial Kernel: Transforms features using polynomial functions. To apply this, set kernel='poly' and tune degree (degree of polynomial) and coef0 (constant term) hyperparameters.

Radial Basis Function (RBF) Kernel: Transforms features using radial basis functions. To apply this, set kernel='rbf' and tune gamma (kernel coefficient) hyperparameter.

- C (Regularization Parameter): C controls the trade-off between achieving a low training error and a low-margin hyperplane. Higher values of C result in a smaller margin but may lead to overfitting. Lower values of C increase the margin but may result in underfitting. Common values to try are in the range of 0.1 to 100.

- gamma (Kernel Coefficient): The gamma parameter defines how far the influence of a single training example reaches, with low values meaning ‘far’ and high values meaning ‘close’. Typical values for gamma ranges between 0.0001 < gamma < 10. It is important to note that lower values of gamma result in models with lower accuracy and the same as the higher values of gamma. It is the intermediate values of gamma which gives a model with good decision boundaries.

ARTIFICIAL NEURAL NETWORK (ANN)

```
#Importation of relevant Libraries#

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

from sklearn.preprocessing import MinMaxScaler

from sklearn.model_selection import train_test_split

from sklearn import preprocessing

from sklearn import metrics

# Neural Net modules

from keras.models import Sequential

from keras.layers import Dense, Dropout

from keras.callbacks import EarlyStopping

from keras.optimizers import Adam

#importation of dataframe
```

```
df = pd.read_csv('D:\Vee_Internship\ML_Models\Infiltration_ML_Data_3.csv', delimiter= ",")

df

# Normalization of input parameters

scaler = preprocessing.MinMaxScaler((0,1))

df['TWI']= scaler.fit_transform(df[['TWI']])

df['NDVI']= scaler.fit_transform(df[['NDVI']])

df['SMI']= scaler.fit_transform(df[['SMI']])

df['Bulk_Density']= scaler.fit_transform(df[['Bulk_Density']])

df['Sand']= scaler.fit_transform(df[['Sand']])

df['Silt']= scaler.fit_transform(df[['Silt']])

df['Clay']= scaler.fit_transform(df[['Clay']])

df['Coarse_Frag']= scaler.fit_transform(df[['Coarse_Frag']])

df['CEC']= scaler.fit_transform(df[['CEC']])

df['Nitrogen']= scaler.fit_transform(df[['Nitrogen']])

df['SOC_Stock']= scaler.fit_transform(df[['SOC_Stock']])

df['Organic_Carbon']= scaler.fit_transform(df[['Organic_Carbon']])

#Extraction of independent (input) and dependent (output) variables into X and Y colums respectively.

x = df.iloc[:,:-1]

y = df.iloc[:,-1]

#To visualize the first few columns of the input variables.

x.head()

#Data division into train and test splits.

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 1.0/3.0, random_state = 7)

#Addition of a split for validation purposes because of the early stopping we will incorporate.

x_train, x_val, y_train, y_val = train_test_split(x_train, y_train, test_size=0.2, random_state=42)

# Build the Neural Network model#

model = Sequential()

model.add(Dense(128, input_shape=(x_train.shape[1],), activation='relu')) ## (input node features,)

model.add(Dense(64, activation='relu'))  #hidden layer
model.add(Dense(32, activation='relu')) #hidden layer
model.add(Dense(1, activation='linear')) # output node
model.summary() # see what your model looks like
# Compile the model and selecting hyperparameters#
model.compile(optimizer='rmsprop', loss='mse', metrics=['mae'])
# early stopping callback
es = EarlyStopping(monitor='val_loss',
             mode='min',
             patience=50,
             restore_best_weights = True)
# fitting model to train dataset, selecting parameters and storing results in a variable ‘history’
# to visualize the learning curves
history = model.fit(x_train, y_train,
              validation_data = (x_test, y_test),
              callbacks=[es],
              epochs=100,
              batch_size=10,
              verbose=1)
#Model Performance Evaluation #
# Visualizing the training and validation accuracy by epoch
history_dict = history.history
loss_values = history_dict['loss']
val_loss_values = history_dict['val_loss']
epochs = range(1, len(loss_values) + 1) # range of X (no. of epochs)
#plots a graph of training and validation losses against the number of epochs to visualize model learning curve with the following attributes ie. Title, color etc.
plt.plot(epochs, loss_values, 'bo', label='Training loss')
plt.plot(epochs, val_loss_values, 'orange', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
#Prediction test on test dataset
y_pred = model.predict(x_test)
y_pred
#Model Performance Assessment with R2, MAE, MSE, and RMSE methods.
MAE = metrics.mean_absolute_error
MSE = metrics.mean_squared_error
print('Mean Absolute Error:', MAE(y_test, y_pred))
print('Mean Squared Error:', MSE(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
from sklearn.metrics import r2_score
R_square = r2_score(y_test, y_pred)
print('Coefficient of Determination', R_square)
```

Hyperparameters of the Artificial Neural Network

- Number of Hidden Layers and Neurons: A neural network has input layer(s), hidden layer(s), and output layer(s). The hidden layers perform computations and transform the input data allowing the model to learn non-linear relationships and solve complex problems. For less complex data with fewer dimensions or features, neural networks with 1 to 2 hidden layers should work however for more complex data with many dimensions, 3 to 5 hidden layers can be used to obtain an optimum solution.

- Activation Function: Activation functions introduce non-linearity to the neural network, allowing it to learn complex patterns. Examples of activation functions include ReLU (Rectified Linear Unit), sigmoid, and tanh. The choice of activation function depends on the problem at hand. However, ReLU is happens to be a popular choice due to its simplicity and effectiveness in many scenarios.

- Number of Training Epochs: The number of training epochs specifies how many times the network will iterate over the entire training dataset. Too few epochs may result in underfitting, while too many epochs may lead to overfitting. The optimal number of epochs is often determined through experimentation and monitoring of validation performance.

- Batch Size: The batch size determines the number of training samples seen by the network before weight updates are performed during training.

- Regularization Techniques: These techniques help to minimize the adjusted loss function and prevent overfitting or underfitting. Examples of Regularization techniques include L1 and L2 regularization (weight decay), dropout, and early stopping. It is important to note that these techniques introduce additional hyperparameters that need to be tuned, such as the regularization strength or the dropout rate.

- Optimization Algorithm: The choice of optimization algorithm affects how the network learns and updates its weights. Examples of optimization algorithms include stochastic gradient descent (SGD), Adam, and RMSprop.

DOCUMENT HISTORY		
Version	**Date**	**Comments**
V2.0	25/06/2023	Second version released for review by the aOCP Steering Committee under the aOCP Version 2. Machine learning algorithms and Penman-Monteith ET Method were added.
V1.0	15/03/2023	First version Second version released for review by the aOCP Steering Committee under the aOCP Version 1.