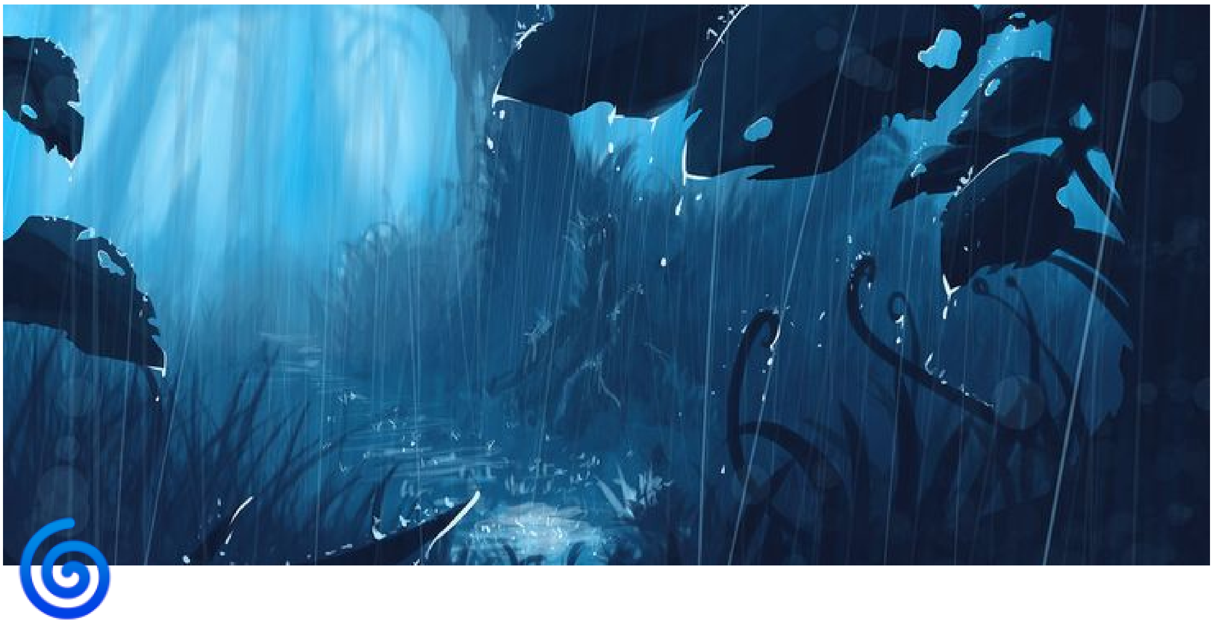


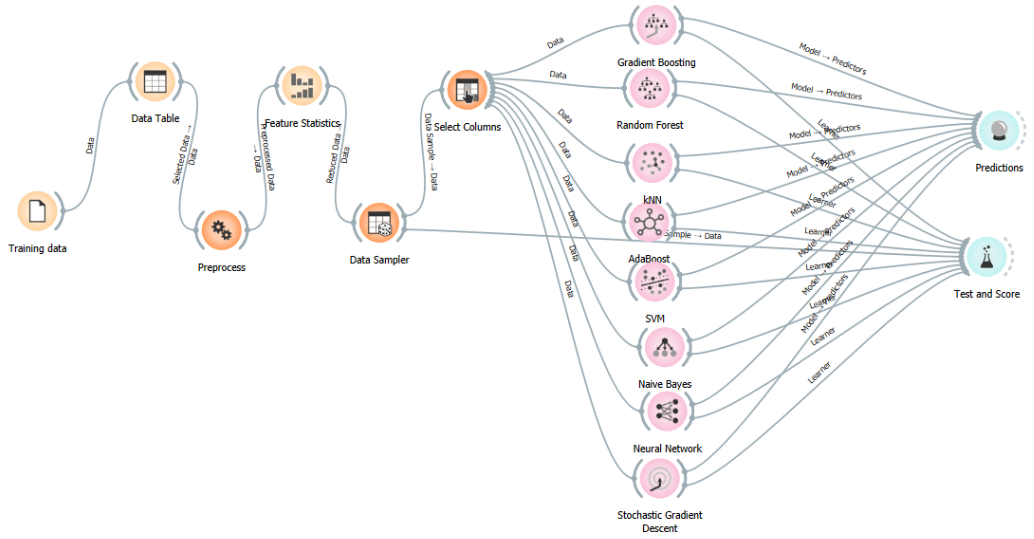
PhishGuard

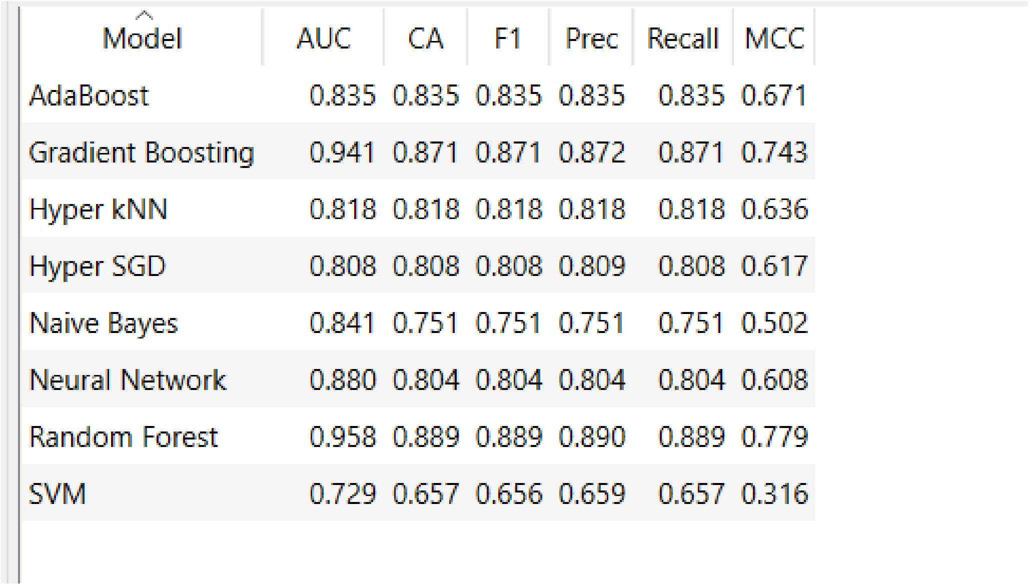
Predicting Malicious Links using AI

Dataset link

https://www.kaggle.com/datasets/shashwatwork/phishing-dataset-for-machine-learning

Number of variables :





Model	AUC	CA	F1	Prec	Recall	MCC
AdaBoost	0.835	0.835	0.835	0.835	0.835	0.671
Gradient Boosting	0.941	0.871	0.871	0.872	0.871	0.743
Hyper kNN	0.818	0.818	0.818	0.818	0.818	0.636
Hyper SGD	0.808	0.808	0.808	0.809	0.808	0.617
Naive Bayes	0.841	0.751	0.751	0.751	0.751	0.502
Neural Network	0.880	0.804	0.804	0.804	0.804	0.608
Random Forest	0.958	0.889	0.889	0.890	0.889	0.779
SVM	0.729	0.657	0.656	0.659	0.657	0.316

- **AUC (Area Under Curve)**: Measures the model's ability to distinguish between classes. Higher values indicate better classification performance.
- **CA (Classification Accuracy)**: Shows the percentage of correctly classified instances. A higher CA reflects better model accuracy.
- **F1 Score**: Balances precision and recall, particularly useful in imbalanced datasets.
- **Precision**: Measures the model's exactness or ability to avoid false positives.
- **Recall**: Reflects the model’s ability to capture all relevant instances (true positives).
- **MCC (Matthews Correlation Coefficient)**: Offers a balanced measure even with imbalanced classes, where 1 indicates a perfect prediction and 0 no better than random.

Problem and solution

- too much variables and meaningless variables
    - Make a table of alls of the variable so that we could determine which is meta, target and features.
    - Do PCA so we could reduce the dimensionality of the features
- How to deploy?

    - We use tableau to share our findings
- Whats our objective?
    - To ensure malicious links were detected before anyone could be a victims

Top 3?

1. Random Forest
2. Gradient Boosting
3. Neural Network

Random Forest: has the highest AUC and MCC, indicate best predictive ability and balanced performance on true positives and negatives

GB: Scores high at AUC, CA and F1 metrics. (effective classification and balance)

Neural Network: High AUC and respectable CA and F1 scores. (Strong predictive performance)

▼ **50 Variables**

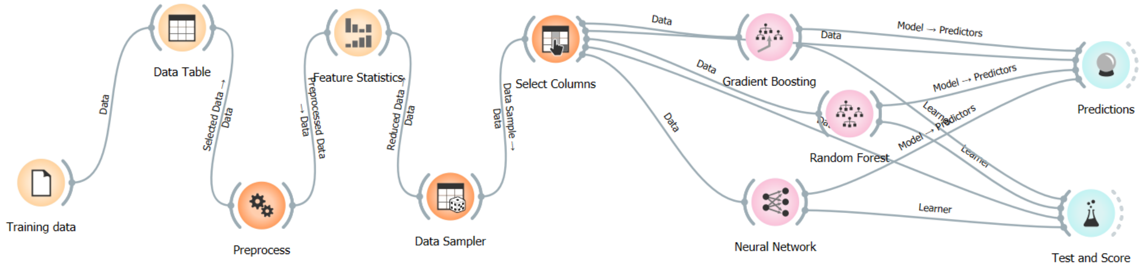
id	Unique identifier for each URL.
NumDots	Number of dots (.) in the URL.
SubdomainLevel	Level of subdomains in the URL (e.g., sub.sub.example.com has a higher subdomain level).
PathLevel	Depth level of the URL path (count of segments in the path after the domain).
UrlLength	Total length of the URL.
NumDash	Number of dash (-) symbols in the URL.
NumDashInHostname	Number of dashes in the hostname part of the URL.
AtSymbol	Indicates if the URL contains an "@" symbol, often used in phishing URLs.
TildeSymbol	Presence of the tilde (~) symbol in the URL.

NumUnderscore	Number of underscores (_) in the URL.
NumPercent	Number of percentage (%) symbols in the URL.
NumQueryComponents	Number of components in the URL query string.
NumAmpersand	Number of ampersands (&) in the URL.
NumHash	Number of hash (#) symbols in the URL.
NumNumericChars	Number of numeric characters in the URL.
NoHttps	Indicates if the URL does not use HTTPS, potentially less secure.
RandomString	Detects if the URL has random strings, a common tactic in phishing URLs.
IpAddress	Checks if an IP address is used instead of a domain name.
DomainInSubdomains	Checks if the domain name appears in subdomains, which may indicate spoofing.
DomainInPaths	Indicates if the domain name appears in the path, which may be a suspicious sign.
HttpsInHostname	Checks if "https" appears in the hostname, possibly to confuse users.
HostnameLength	Length of the hostname part of the URL.
PathLength	Length of the path section of the URL.
QueryLength	Length of the query string in the URL.
DoubleSlashInPath	Checks for double slashes (//) in the path, which can indicate suspicious behavior.
NumSensitiveWords	Number of sensitive words (e.g., "login," "secure") in the URL.
EmbeddedBrandName	Indicates if the URL includes a well-known brand name, which can be used for phishing.
PctExtHyperlinks	Percentage of external hyperlinks within the web page.
PctExtResourceUrls	Percentage of external resources linked in the page.

ExtFavicon	Indicates if an external favicon is used, which can be a phishing indicator.
InsecureForms	Checks if forms on the page are insecure (i.e., not HTTPS).
RelativeFormAction	Indicates if forms use a relative path for the action attribute.
ExtFormAction	Checks if forms point to an external URL.
AbnormalFormAction	Indicates if the form action is unusual (e.g., points to an unrelated domain).
PctNullSelfRedirectHyperlinks	Percentage of hyperlinks that redirect to the same page or null.
FrequentDomainNameMismatch	Checks for frequent domain mismatches within links on the page.
FakeLinkInStatusBar	Checks if the page manipulates the status bar link display.
RightClickDisabled	Indicates if right-click functionality is disabled on the page.
PopUpWindow	Indicates the presence of pop-up windows, which can be used for phishing.
SubmitInfoToEmail	Checks if forms submit information to an email address, which is often suspicious.
IframeOrFrame	Indicates if the page uses iframes or frames, which may conceal content.
MissingTitle	Indicates if the page has no title.
ImagesOnlyInForm	Checks if forms contain only images, which can be a tactic to avoid text-based detection
SubdomainLevelRT	Relative measure of subdomain depth.
UrlLengthRT	Relative measure of URL length.
PctExtResourceUrlsRT	Relative measure of the percentage of external resource URLs.
AbnormalExtFormActionR	Relative measure of abnormal external form actions.
ExtMetaScriptLinkRT	Relative measure of external meta/script links on the page.
PctExtNullSelfRedirectHyperlinksRT	Relative measure of external null/self-redirect hyperlinks.

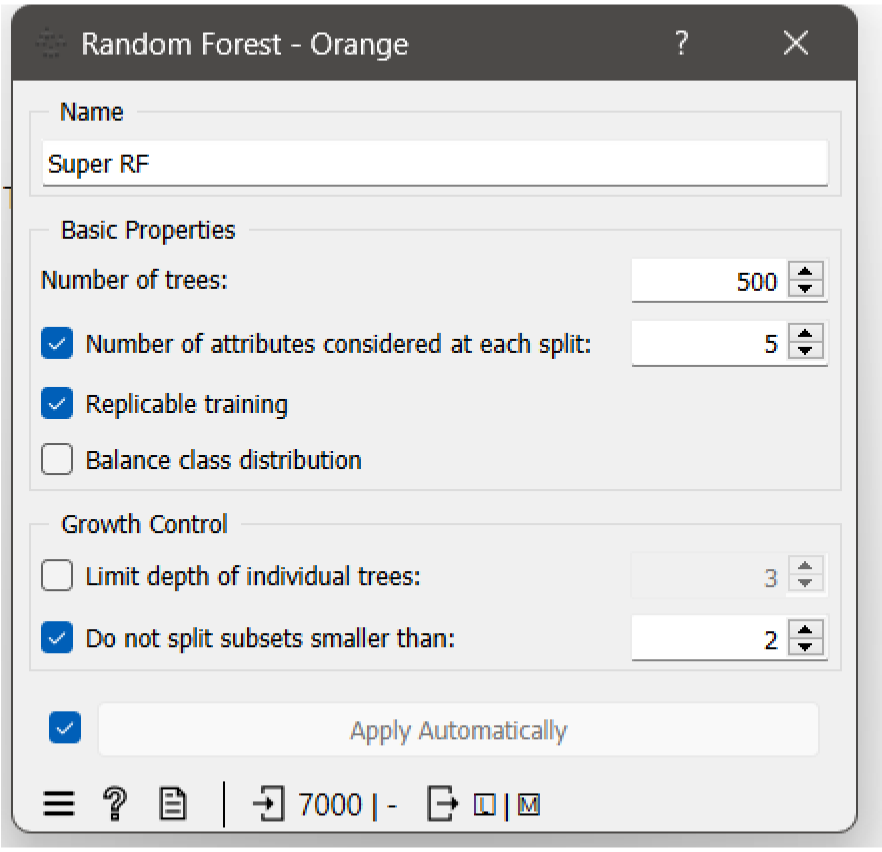
CLASS_LABEL	Target variable indicating if the URL is phishing (1) or legitimate (0).

Latest setup



Hypertune setup

▼ Random Forest


Random Forest - Orange

Name

Super RF

Basic Properties

Number of trees: 500

☑ Number of attributes considered at each split: 5

☑ Replicable training

□ Balance class distribution

Growth Control

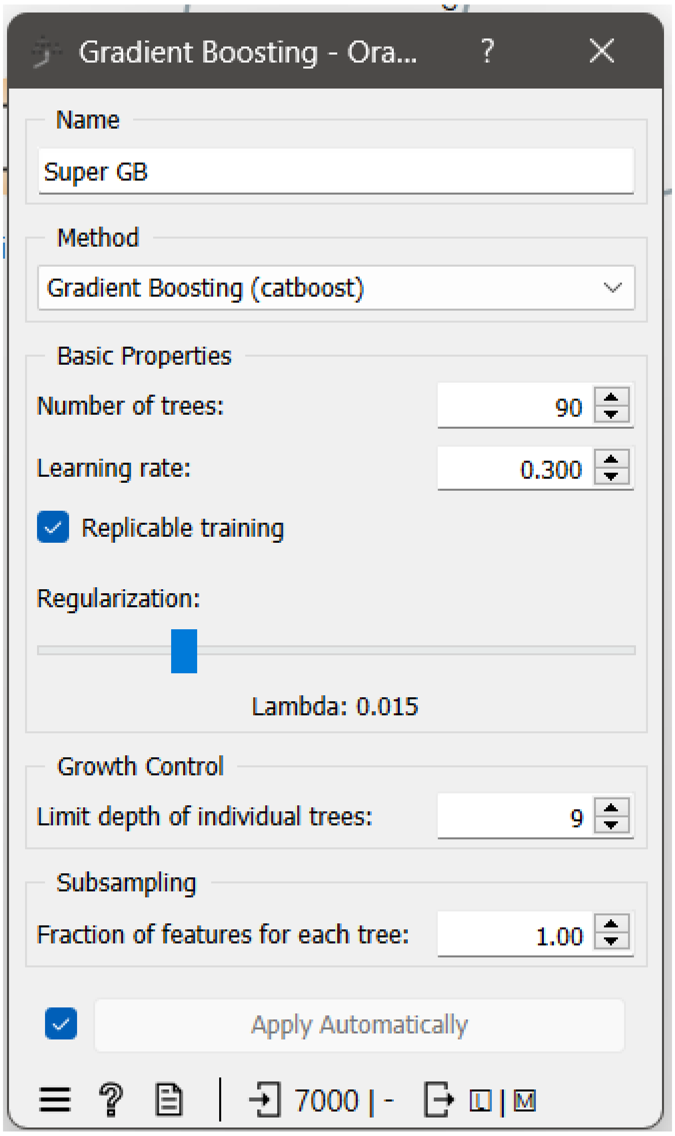
□ Limit depth of individual trees: 3

☑ Do not split subsets smaller than: 2

☑ Apply Automatically

7000 | -

▼ Gradient Boosting



Name

Super GB

Method

Gradient Boosting (catboost)

Basic Properties

Number of trees: 90

Learning rate: 0.300

☑ Replicable training

Regularization:

Lambda: 0.015

Growth Control

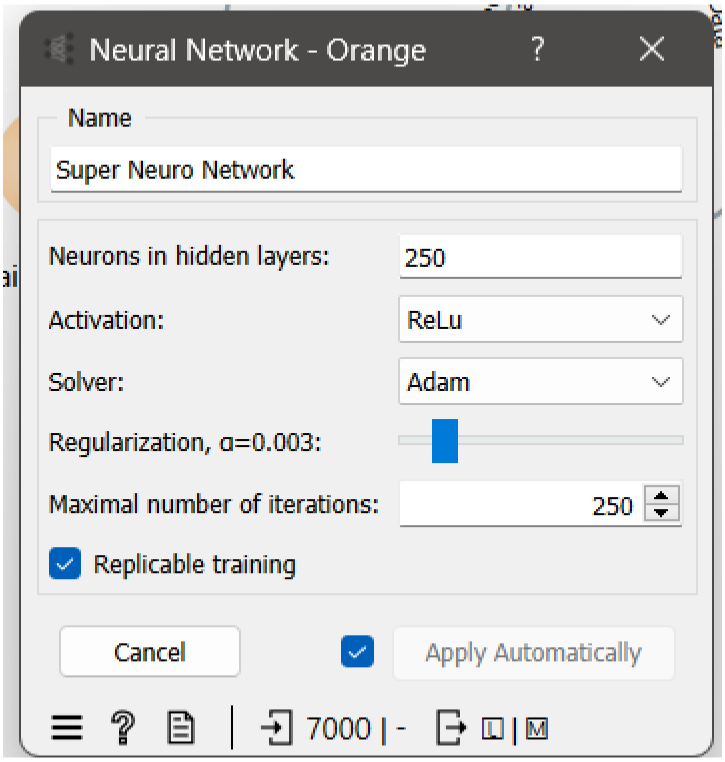
Limit depth of individual trees: 9

Subsampling

Fraction of features for each tree: 1.00

☑ Apply Automatically

▼ Neural Network



Neural Network - Orange

Name

Super Neuro Network

Neurons in hidden layers: 250

Activation: ReLu

Solver: Adam

Regularization, α=0.003:

Maximal number of iterations: 250

☑ Replicable training

New Performance

Model	AUC	CA	F1	Prec	Recall	MCC
Super RF	0.955	0.885	0.885	0.886	0.885	0.771
Super GB	0.953	0.884	0.884	0.884	0.884	0.768
Super Neuro Network	0.954	0.886	0.886	0.887	0.886	0.773