

Forward-Looking Statements



This presentation may contain forward-looking statements regarding future events, plans or the expected financial performance of our company, including our expectations regarding our products, technology, strategy, customers, markets, acquisitions and investments. These statements reflect management's current expectations, estimates and assumptions based on the information currently available to us. These forward-looking statements are not guarantees of future performance and involve significant risks, uncertainties and other factors that may cause our actual results, performance or achievements to be materially different from results, performance or achievements expressed or implied by the forward-looking statements contained in this presentation.

For additional information about factors that could cause actual results to differ materially from those described in the forward-looking statements made in this presentation, please refer to our periodic reports and other filings with the SEC, including the risk factors identified in our most recent quarterly reports on Form 10-Q and annual reports on Form 10-K, copies of which may be obtained by visiting the Splunk Investor Relations website at www.investors.splunk.com or the SEC's website at www.sec.gov. The forward-looking statements made in this presentation are made as of the time and date of this presentation. If reviewed after the initial presentation, even if made available by us, on our website or otherwise, it may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statement based on new information, future events or otherwise, except as required by applicable law.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. We undertake no obligation either to develop the features or functionalities described, in beta or in preview (used interchangeably), or to include any such feature or functionality in a future release.

Splunk, Splunk> and Turn Data Into Doing are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names or trademarks belong to their respective owners. © 2023 Splunk Inc. All rights reserved.

Easily Extend Splunk® Enterprise Security

With Custom Machine Learning
Models and Algorithms for the
Highest Fidelity Incident Detection
SEC2001C

Ann-Drea Small

Principal Splunk Development Engineer | Kinney Group

Michael Simko

Senior Staff Technical Trainer, Enablement | Splunk





Ann-Drea Small

Principal Splunk Development Engineer
Kinney Group

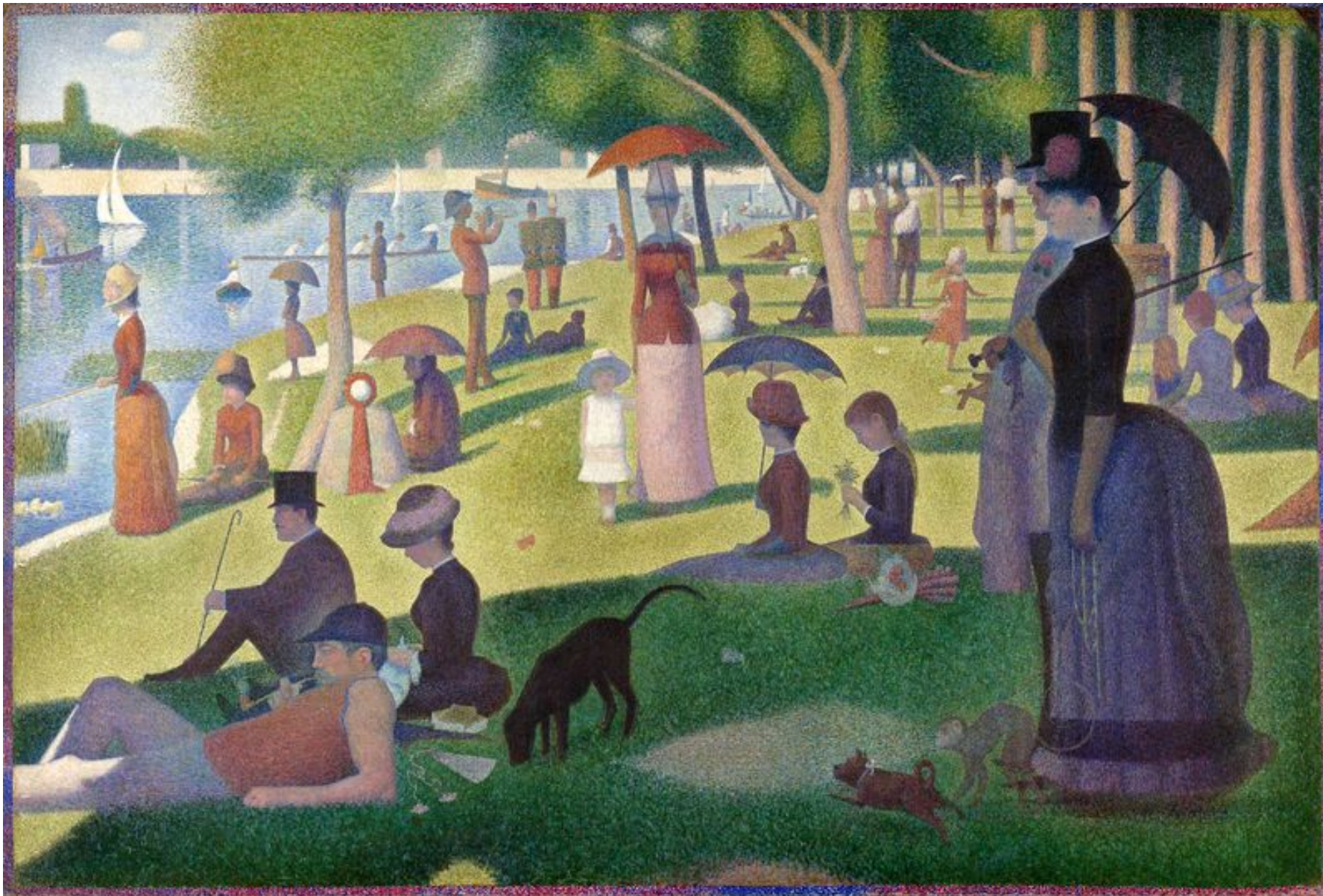


Michael Simko

Senior Staff Technical Trainer, Enablement
Splunk



* **Public Domain Painting:** Close up of the paint on the canvas of A Sunday Afternoon on the Island on La Grande Jatte by Georges Seurat, 1884



* **Public Domain Painting:** A Sunday Afternoon on the Island on La Grande Jatte by Georges Seurat, 1884

What Are We Talking About?

When to use Machine Learning

Why use Machine Learning in Splunk® Enterprise Security

How using ML in ES can help with Incident Review

How to import algorithms and train models

How to create a custom algorithm

splunk> .conf23



Why Should You Care?

In our defense, if you don't use Enterprise Security, you shouldn't care.

Adding Machine Learning to ES uses cases will:

- 1) Decrease time to remediation
 - Higher Fidelity means more value for a SOC
- 2) Move team from being reactionary to proactive
 - detect anomalies and predict events before the occur
- 3) Help address more advanced use cases
 - Import and create algorithms that fit your needs



ML? Why Not Just Use Stats?

Spoiler: You can do a lot with stats – but not everything



Stats

Uses finite data points with limited parameters

Makes inferences by quantifying uncertainty

Correlates data based on a known hypothesis.

Must understand the relationships between the variables

* dependent upon your skills in statistics



Machine Learning

Uses vast amounts of data with endless features and labels.

Predicts future events or classifies data

Tries to find patterns hidden in the data

Is not reliant on rule-based programming






How Do We Get Started?

splunk> App: DGA App for Splunk ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾ Find


DGA App for Splunk Dashboards ▾ Search More ▾


DGA Analysis 


DGA App for Splunk philipp@splunk.com


Edit Export ▾ ...


Content overview

1. Exploratory Data Analysis


2. Feature Engineering and Selection


3. Create Machine Learning Models


4. Operationalize Machine Learning


5. Test and Benchmark


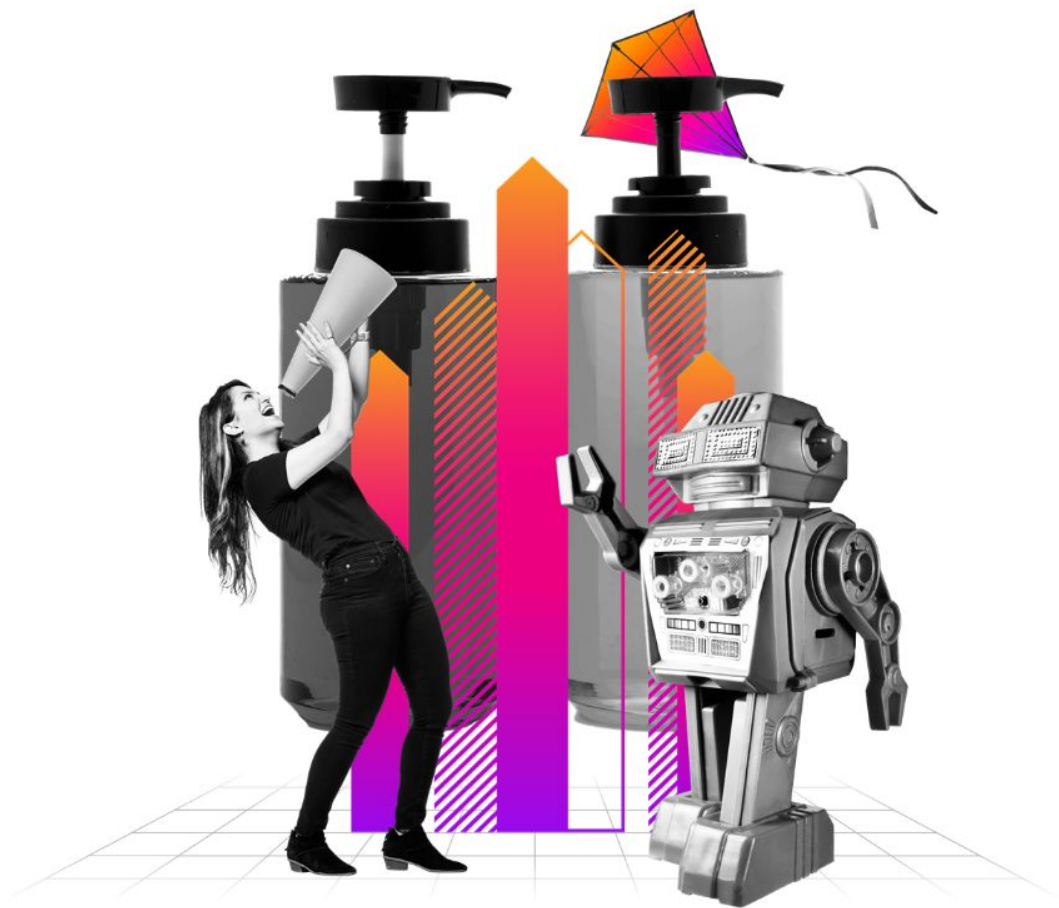
Setup

For full functionality of the app please check and review the [setup dashboard page](#) and make sure that all setup steps are completed.

About Support File a Bug Documentation Privacy Policy

© 2005-2017 Splunk Inc. All rights reserved.

Domain Generation Algorithms app on Splunkbase gives us a good step by step plan to follow.
<https://splunkbase.splunk.com/app/3559>



Select the Data

Netflow is a protocol used to collect and analyze metadata from active network traffic traversing a Cisco router or switch.

Sample Netflow Data

Standard netflow data contains source port and IP address, destination port and IP address, TCP flags, type of service, and protocol number among other values.

Sample data obtained from:

<https://open.scayle.es/dataset/netflow-data-with-sampling-for-test>

splunk>enterprise App: Splunk Machine Learning Toolkit

Administrator Messages Settings Activity Help Find

Showcase Experiments **Search** Models Classic Settings Docs Video Tutorials

Splunk Machine Learning Toolkit

New Search

Save As New Table Close

| inputlookup netflow_sampling_1000_50-50_test.csv Last 24 hours

✓ 2,646 results (5/18/23 11:00:00.000 AM to 5/19/23 11:25:26.000 AM) No Event Sampling

Job

Events Patterns **Statistics (2,646)** Visualization

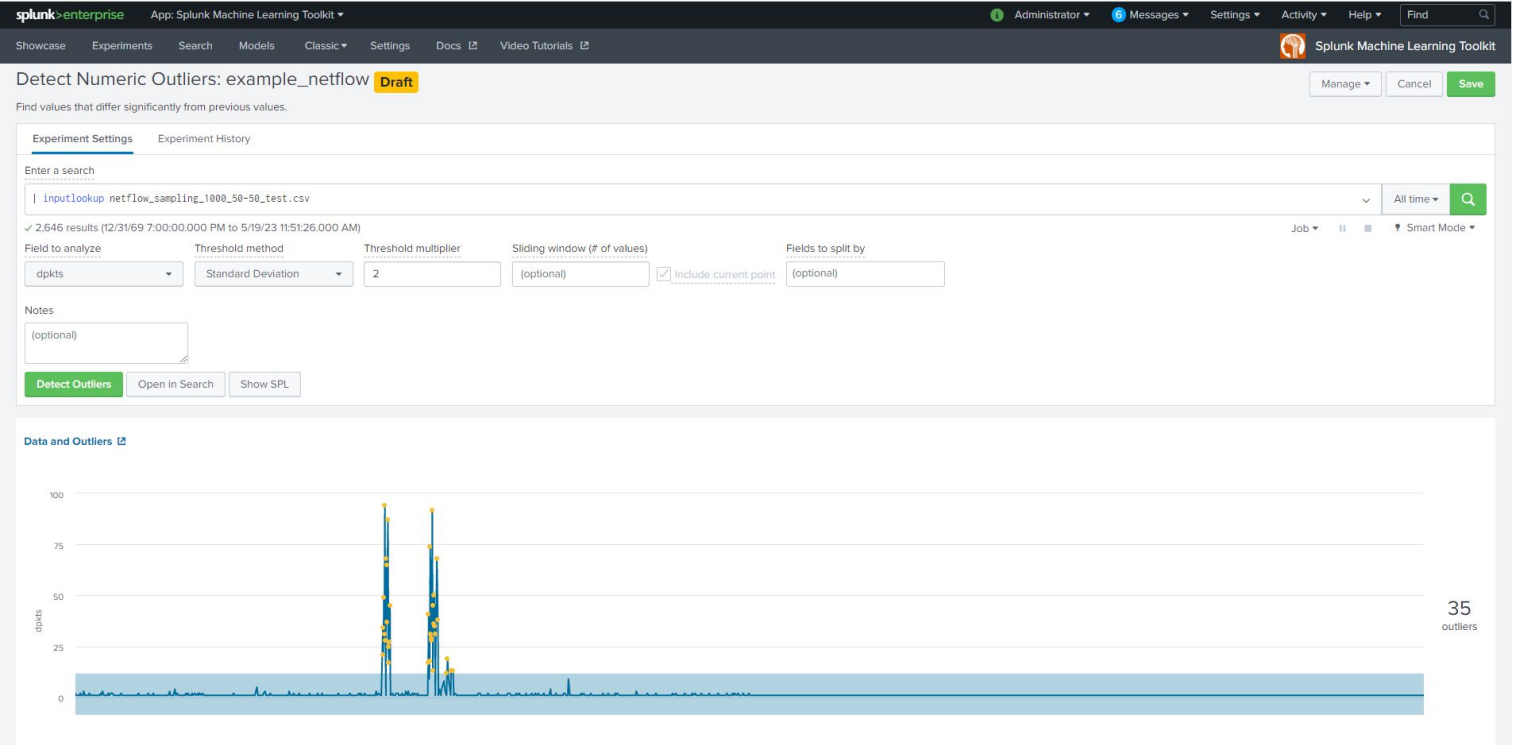
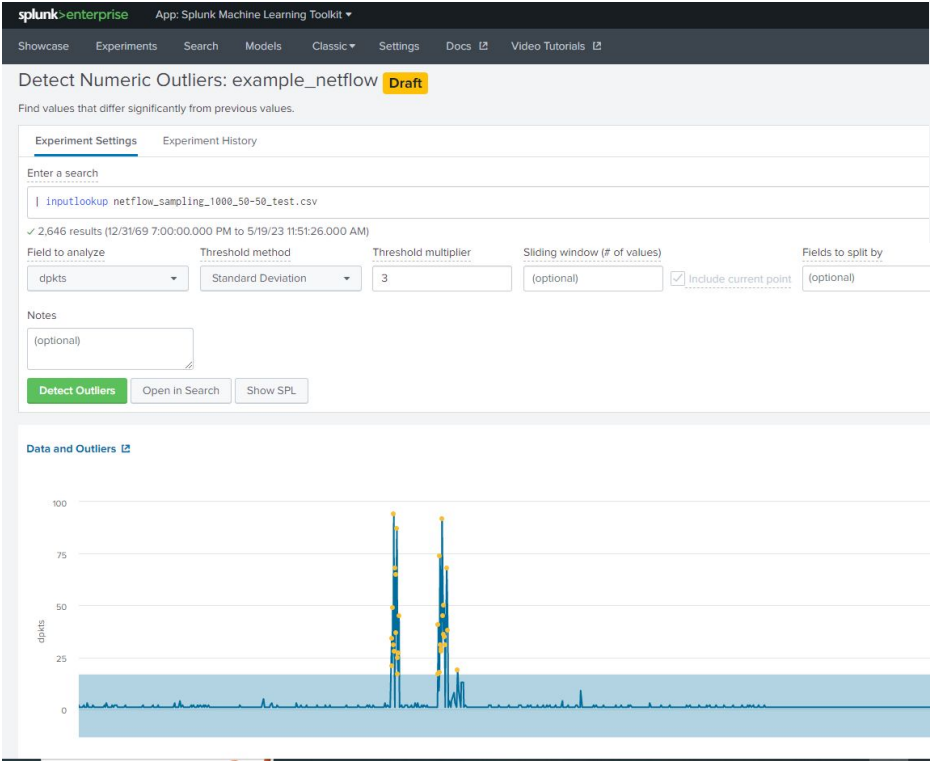
20 Per Page Format Preview

#unix_secs	Label	doctets	dpkts	dst_as	dst_mask	dstaddr	dstport	engine_id	engine_type	exaddr	first	input	last	nexthop	output	prot	src_as	src_mask	srcaddr	srcport	sysuptime	tcp_flags
1592206897	0	2984	2	0	0	140.30.20.2	40924	0	0	152.148.48.1	34429852	232	34429868	152.148.48.2	232	6	0	0	13.33.235.17	494	34489240	16
1592206897	0	4372	1	0	0	140.30.20.2	38350	0	0	152.148.48.1	34431104	232	34431104	152.148.48.2	232	6	0	0	13.33.235.49	222	34489240	16
1592206897	0	2984	2	0	0	140.30.20.2	42706	0	0	152.148.48.1	34431592	232	34431632	152.148.48.2	232	6	0	0	13.33.235.46	163	34489240	16
1592206897	0	1492	1	0	0	140.30.20.2	58850	0	0	152.148.48.1	34431824	232	34431824	152.148.48.2	232	6	0	0	13.33.235.46	273	34489240	16
1592206897	0	52	1	0	0	13.33.235.46	443	0	0	152.148.48.1	34431792	232	34431792	152.148.48.2	232	6	0	0	152.148.48.6	42738	34489240	16
1592206897	0	91	1	0	0	140.30.20.2	40174	0	0	152.148.48.1	34432068	232	34432068	152.148.48.2	232	6	0	0	216.58.211.227	179	34489240	24
1592206897	0	52	1	0	0	216.58.201.131	443	0	0	152.148.48.1	34432376	232	34432376	152.148.48.2	232	6	0	0	152.148.48.5	51932	34489240	16
1592206897	0	52	1	0	0	216.58.211.227	443	0	0	152.148.48.1	34433380	232	34433380	152.148.48.2	232	6	0	0	152.148.48.6	40174	34489240	16
1592206897	0	1470	1	0	0	140.30.20.2	34316	0	0	152.148.48.1	34434064	232	34434064	152.148.48.2	232	6	0	0	216.58.211.35	75	34489240	16
1592206897	0	40	1	0	0	212.170.159.195	443	0	0	152.148.48.1	34441960	232	34441960	152.148.48.2	232	6	0	0	152.148.48.5	40226	34489240	16
1592206897	0	4436	2	0	0	140.30.20.2	40226	0	0	152.148.48.1	34441888	232	34442004	152.148.48.2	232	6	0	0	212.170.159.195	147	34489240	16
1592206897	0	52	1	0	0	195.235.205.198	443	0	0	152.148.48.1	34442460	232	34442460	152.148.48.2	232	6	0	0	152.148.48.5	33292	34489240	16
1592206897	0	1470	1	0	0	140.30.20.2	41982	0	0	152.148.48.1	34443820	232	34443820	152.148.48.2	232	6	0	0	172.217.17.13	11	34489240	24
1592206897	0	52	1	0	0	13.33.235.2	443	0	0	152.148.48.1	34443940	232	34443940	152.148.48.2	232	6	0	0	152.148.48.7	49036	34489240	16
1592206897	0	4306	1	0	0	140.30.20.2	39082	0	0	152.148.48.1	34444352	232	34444352	152.148.48.2	232	6	0	0	216.58.201.174	145	34489240	24
1592206897	0	1492	1	0	0	140.30.20.2	33296	0	0	152.148.48.1	34458368	232	34458368	152.148.48.2	232	6	0	0	195.235.205.198	162	34489240	16
1592206897	0	3074	3	0	0	140.30.20.2	49036	0	0	152.148.48.1	34444908	232	34459168	152.148.48.2	232	6	0	0	13.33.235.2	446	34489240	24

Visualize the Data



Use MLTK to detect Numeric Outliers using packets within 2 or 3 standard deviations from the mean.



splunk>enterpriseApp: Splunk Machine Learning Toolkit

ShowcaseExperimentsSearchModelsClassicSettingsDocsVideo Tutorials

Detect Categorical Outliers: sample_netflow_categoriesDraft

Find events that contain unusual combinations of values.

Experiment SettingsExperiment History

Enter a search

inputlookup netflow_sampling_1000_50-50_test.csv

2,646 results (12/31/69 7:00:00.000 PM to 5/19/23 12:10:46.000 PM)

Field(s) to analyze

dpkts, dstport, srcpor... (8)

Notes

(optional)

Detect Outliers

Open in Search

Show SPL

Outlier(s)

1

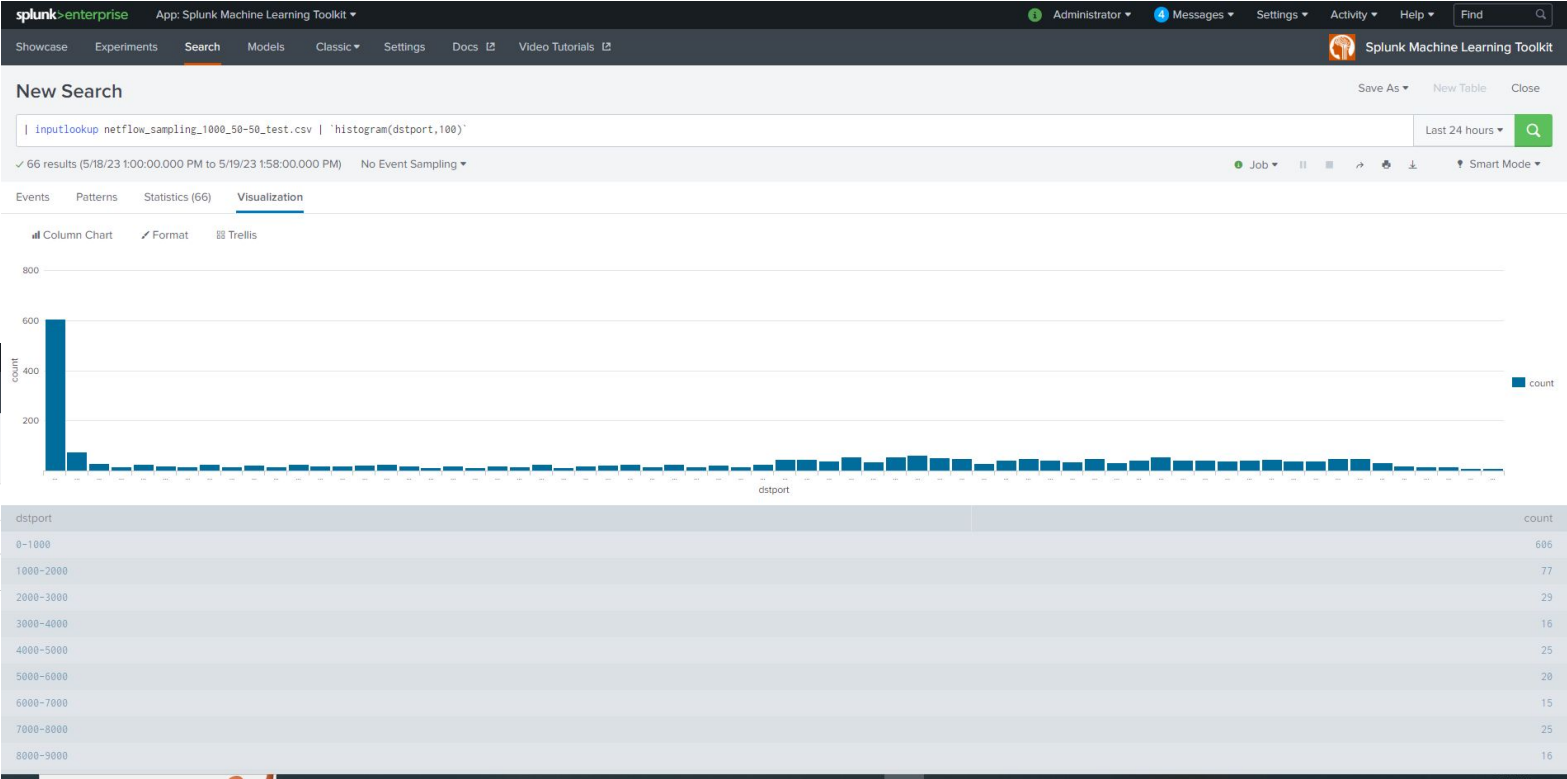
Outlier(s)

Open in Search

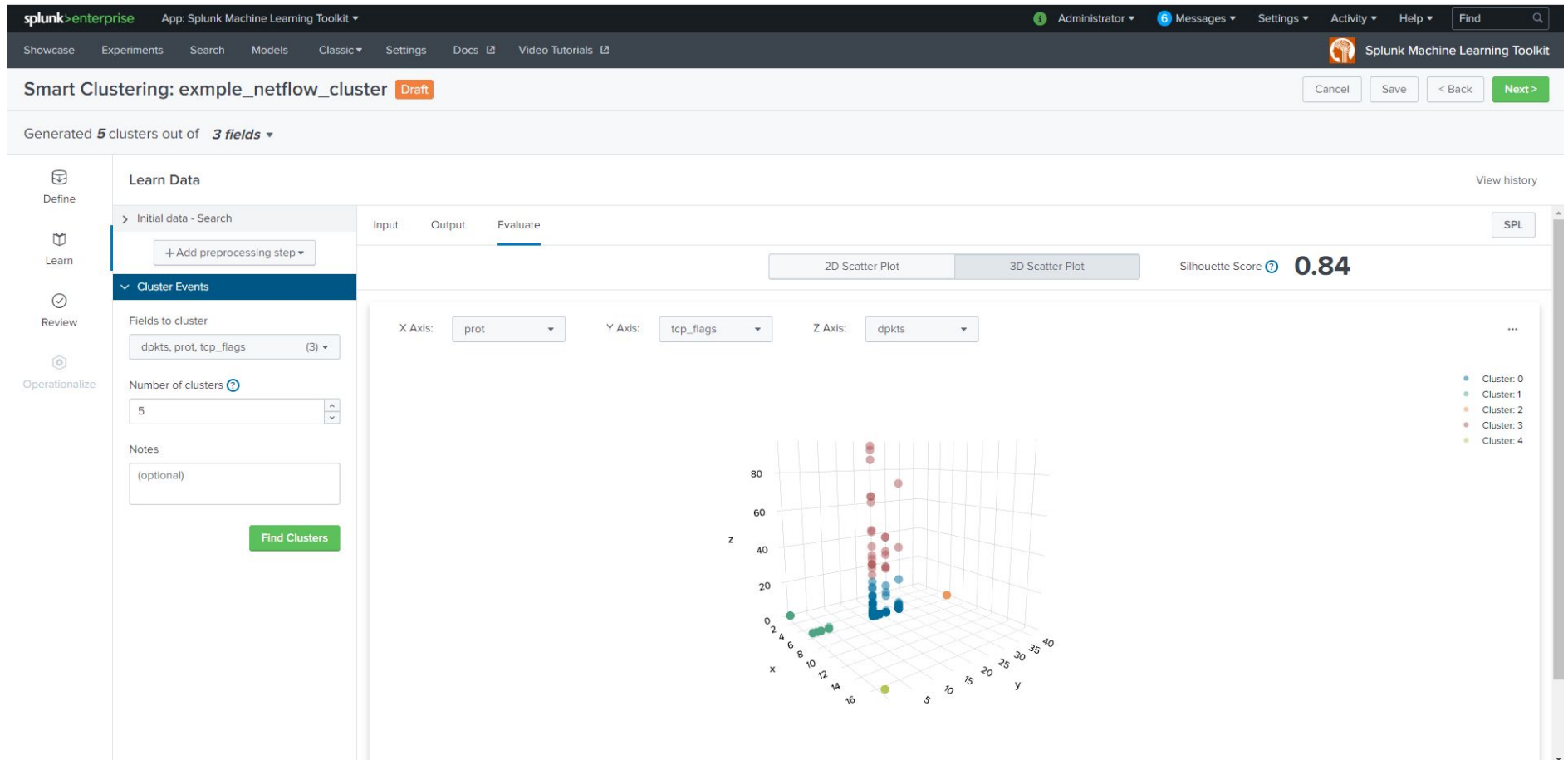
Show SPL

Data and Outliers

dpkts	dstport	srcport	prot	tos	tcp_flags	sysuptime	unix_nsecs	probable_cause	isOutlier
74	39980	85	6	32	24	40482288	969541	dpkts	1
2	40924	494	6	0	16	34489240	921732		0
1	38350	222	6	0	16	34489240	921732		0
2	42706	163	6	0	16	34489240	921732		0



Categorize the Data



Apply the Machine Learning Model - SVM

There are several models that would be a good fit to look at Netflow data. Some of them are:

- Stochastic Gradient Descent (SGD)
- **Support Vector Machines (SVM)**
- **K-Nearest Neighbor (K-NN)**
- Gaussian Naive Bayes (GNB)
- Decision Tree (DT)
- Random Forest (RF)
- **AdaBoost (AB)**

Note: The ones in **RED** are currently not in the MLTK. The one in **BOLD** is currently in MLTK and can be found at the link below.

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

```
#!/usr/bin/env python

from sklearn.svm import SVC

from codec import codecs_manager
from base import BaseAlgo, ClassifierMixin
from util.param_util import convert_params

class SVM(ClassifierMixin, BaseAlgo):
    def __init__(self, options):
        self.handle_options(options)

        out_params = convert_params(options.get('params', {}), floats=['gamma', 'C'])

        self.estimator = SVC(class_weight='balanced', **out_params)

    @staticmethod
    def register_codecs():
        from codec.codecs import SimpleObjectCodec

        codecs_manager.add_codec('algos.SVM', 'SVM', SimpleObjectCodec)
        codecs_manager.add_codec('sklearn.svm.classes', 'SVC', SimpleObjectCodec)
```




How Do We Determine the Best Model?

Try Them All!

Use the Score Command

Compare the accuracy scores of multiple models to determine which of the below MLTK algorithms are a good fit.

- Support Vector Machines (SVM),
- Gaussian Naive Bayes (GNB),
- Decision Tree (DT),
- Random Forest (RF)

Note: It is generally accepted that an accuracy score of between 70% - 90% is a good fit.

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html

<https://docs.splunk.com/Documentation/MLApp/latest/User/Scorecommand>

```
#!/usr/bin/env python

from sklearn.svm import SVC

from codec import codecs_manager
from base import BaseAlgo, ClassifierMixin
from util.param_util import convert_params

class SVM(ClassifierMixin, BaseAlgo):
    def __init__(self, options):
        self.handle_options(options)

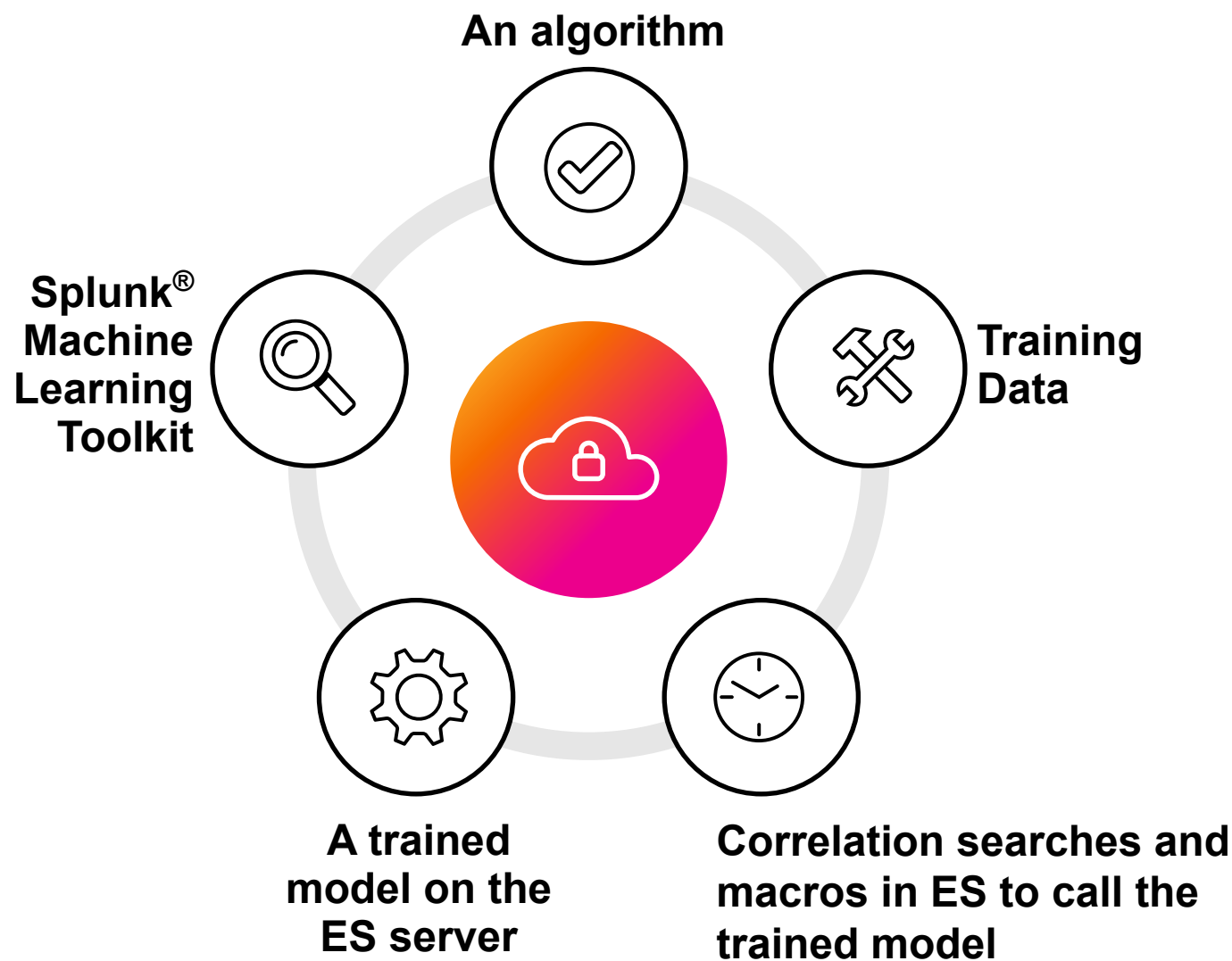
        out_params = convert_params(options.get('params', {}), floats=['gamma', 'C'])

        self.estimator = SVC(class_weight='balanced', **out_params)

    @staticmethod
    def register_codecs():
        from codec.codecs import SimpleObjectCodec

        codecs_manager.add_codec('algos.SVM', 'SVM', SimpleObjectCodec)
        codecs_manager.add_codec('sklearn.svm.classes', 'SVC', SimpleObjectCodec)
```

What We Need for ML To Work in Splunk® ES



The Easy Method: Mimic What Splunk® ES Already Does



Review Current ML Models in ES

- In ES, Navigate to Configure > Content Management
- Examine sample Correlation Searches in ES that incorporates ML
- Take note of:
 - REST API calls
 - macros
 - lookups
 - data models
 - ML model
 - output that are used in the Correlation search
- With large amounts of data, please use tstats

```
| rest splunk_server=local count=0 /services/saved/searches
| where match('action.correlationsearch.enabled', "1|[Tt][Tt][Rr][Uu][Ee]")
| rename eai:acl.app as app, title as csearch_name, action.correlationsearch.label as
csearch_label, action.notable.param.security_domain as security_domain,
action.notable.param.severity as severity, action.correlationsearch.annotations as
annotations
| eval AAAseverity =
case(severity=="critical","5",severity=="high","4",severity=="medium","3",severity=="low","2",
severity=="informational","1",1==1,"0")
| sort - AAAseverity
| eval status = case(disabled=="0","Enabled",1==1,"Disabled")
| table severity, status, search, csearch_label, description, security_domain, app,
action.notable.param.rule_title, annotations
| search search=*mltk*m,n
```

high	Enabled	from datamodel:"Authentication"."Authentication" stats values(tag) as tag,values(app) as app,count(eval('action'=="failure")) as failure,count(eval('action'=="success")) as success by src search success>0 'mltk_apply_upper("app:failures_by_src_count_1h", "high", "failure")'	Brute Force Access Behavior Detected
high	Enabled	tstats `summariesonly` values(Authentication.app) as app,count from datamodel=Authentication.Authentication by Authentication.action,Authentication.src 'drop_dm_object_name("Authentication")' eval success=if(action="success",count,0),failure=if(action="failure",count,0) stats values(app) as app,sum(failure) as failure,sum(success) as success by src where success > 0 'mltk_apply_upper("app:failures_by_src_count_1d", "medium", "failure")'	Brute Force Access Behavior Detected Over One Day
high	Disabled	tstats `summariesonly` sum(All_Email.size) as bytes, values(All_Email.recipient) as recipient from datamodel=Email.All_Email where NOT 'cim_corporate_email_domain_search("All_Email.recipient")' by All_Email.src_user, All_Email.src_user_bunit 'drop_dm_object_name("All_Email")' 'mltk_apply_upper("app:email_activity_to_non_corporate_by_user_1h", "medium", "bytes")'	High Volume Email Activity to Non- corporate Domains by User

Our Correlation Search in ES

Creating a Notable Event

Once it is configured for Notable Events and enabled, triggered events should display in Incident Review.

Correlation Search

Search Name

App

UI Dispatch

Context
Set an app to use for links such as the drill-down search in a notable event or links in an email adaptive response action. If None, uses the Application Context.

Description

Mode ☐ Guided


☐ Manual

Search

```
| inputlookup netflow_sampling_1000_50-50_test.csv
| fit SVM "nexthop" from "dstaddr" "dstport" into "netflow_predicted_nexthop"
| apply netflow_predicted_nexthop | rename predicted(nexthop) as predicted_nexthop | eval isPredicted = if(nexthop
=predicted_nexthop, "true", "false") | search isPredicted="false" | stats count
```

Adaptive Response Actions

+ Add New Response Action ▼

 Notable

Title	<input type="text" value="NetFlow Unexpected Hop in 24 Hours"/> Notable events created by this search will have this title. Supports variable substitution.
Description	<input type="text"/> Notable events created by this search will have this description. Supports variable substitution.
Security Domain	<input type="text" value="Network"/>
Severity	<input type="text" value="Medium"/> Used to calculate urgency for notable events. Learn more
Default Owner	<input type="text" value="asmall"/>
Default Status	<input type="text" value="(leave as system default)"/>
Drill-down Name	<input type="text"/> Supports variable substitution with fields from the matching event.

Importing ML Algorithms



Import a Machine Learning Model

Remember, there are several models not currently in the MLTK that would be a good fit to look at Netflow data. Some of them are:

- K-Nearest Neighbor (K-NN)
- **AdaBoost (AB)**

Note: The one in **RED** is declared as the regressor not classifier in the Splunk GitHub for Machine learning app at:

<https://github.com/splunk/mltk-algo-contrib> OR

<https://splunkbase.splunk.com/app/4403>

It is not complete.

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html>

```
#!/usr/bin/env python

from pandas import DataFrame
from sklearn.ensemble import AdaBoostClassifier as _AdaBoostClassifier

from cexc import get_messages_logger
from base import ClassifierMixin, BaseAlgo
from util.param_util import convert_params
from util.algo_util import handle_max_features
from codec import codecs_manager
from codec.codecs import SimpleObjectCodec

messages = get_messages_logger()

class AdaBoostClassifier(ClassifierMixin, BaseAlgo):
    def __init__(self, options):

        self.handle_options(options)

        out_params = convert_params(
            options.get('params', {}),
            strs=['estimator', 'max_features'],
            floats=['learning_rate'],
            ints=[
                'n_estimators',
                'random_state',
            ],
        )
        self.estimator = _AdaBoostClassifier(**out_params)

        if 'max_features' in out_params:
            out_params['max_features'] = handle_max_features(out_params['max_features'])
```

Importing Algorithms From Other Sources

Where to find algorithms, and how to import them

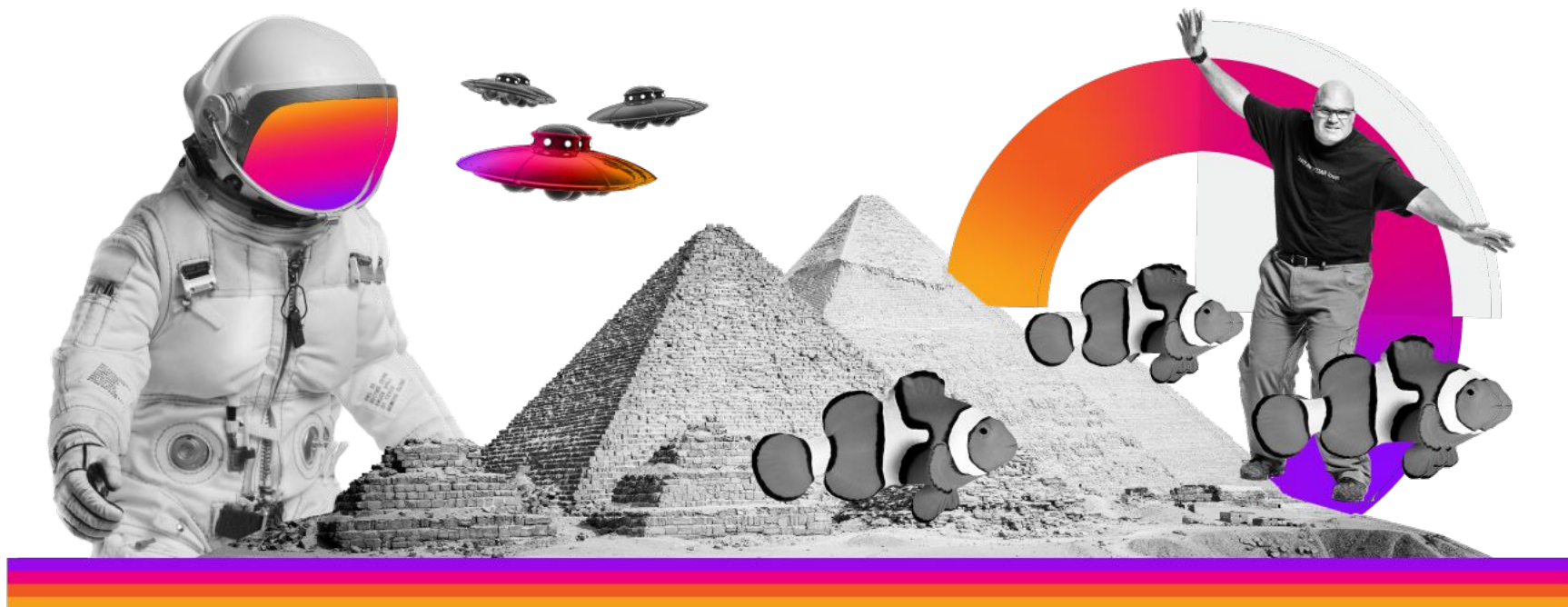
Splunkbase		
	Splunk ES Content Update	https://splunkbase.splunk.com/app/3449
	Splunk MLTK Algorithms on GitHub	https://splunkbase.splunk.com/app/4403
		https://github.com/splunk/mltk-algo-contrib
Scikit-Learn		https://scikit-learn.org/stable/modules/classes.html#
Make your own	Use Python, numpy, pandas and grit!	

Please read the MLTK Docs to find out current algorithms:

<https://docs.splunk.com/Documentation/MLEApp/5.4.0/API/Overview>

Creating Your Own Machine Learning Algorithm

Go forth and explore!



Create a Machine Learning Algorithm

Erlang C Distribution

```
#!/usr/bin/env python
# coding=utf-8

import sys, math
from splunklib.searchcommands import dispatch, StreamingCommand, Configuration, Option, validators

@Configuration(local=True)
class ErlangCCommand(StreamingCommand):

    calls = Option(require = True, validate = validators.Float())
    svc = Option(require = True, validate = validators.Float())
    staff = Option(require = False, validate = validators.Integer(0), default = 5)
    tat = Option(require = False, validate = validators.Integer(0), default = 20)
    shrinkage = Option(require = False, validate = validators.Float(), default = 0.3)
    max_occupancy = Option(require = False, validate = validators.Float(), default = 0.85)
    result = Option(require = False, validate = validators.Fieldname(), default = "support_staff")
    result0 = Option(require = False, validate = validators.Fieldname(), default = "service_level")
    result1 = Option(require = False, validate = validators.Fieldname(), default = "wait_probability")
    result2 = Option(require = False, validate = validators.Fieldname(), default = "avg_answer_speed")
    result3 = Option(require = False, validate = validators.Fieldname(), default = "actual_staff_need")

    def stream(self, events):
        self.logger.debug("ErlangCCommand: %s", self)
        intensity = 0
        numerator = 0
        summation = 0
        staff = 0
        sl = 0
        pw = 0
        asa = 0
        ia = 0
        o = 0
        rr = 0
        for event in events:
            if self.staff < 5:
                start = 1
            else:
                start = self.staff - 4
            for i in range(start, self.staff + 5):
                staff = long(i)
                intensity = self.traffic(float(self.calls), float(self.svc))
                numerator = self.xtop(float(intensity), long(i))
                summation = self.ybottom(float(intensity), long(i))
                pw = self.erlangc(float(numerator), float(summation))
                sl = self.svcvl(float(pw), float(intensity), long(i), long(self.tat), float(self.svc))
                asa = self.asa(float(pw), float(intensity), long(i), float(self.svc))
                ia = self.ia(float(pw))
                o = self.occupy(float(intensity), long(i))
                rr = self.shrink(float(self.shrinkage), long(i))
                if o <= float(self.max_occupancy):
```


Erlang Distribution

Queueing Model based on Poisson distribution

The probability density function of the Erlang distribution is:

$$f(x; k, \lambda) = \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!} \quad \text{for } x, \lambda \geq 0,$$

The parameter k is called the shape parameter, and the parameter λ is called the rate parameter.

Kansas City, Missouri.
Private branch
exchange (PBX)
operator at her
switchboard in the
freight depot. Jack
Delano. March 1943.
Library of Congress.





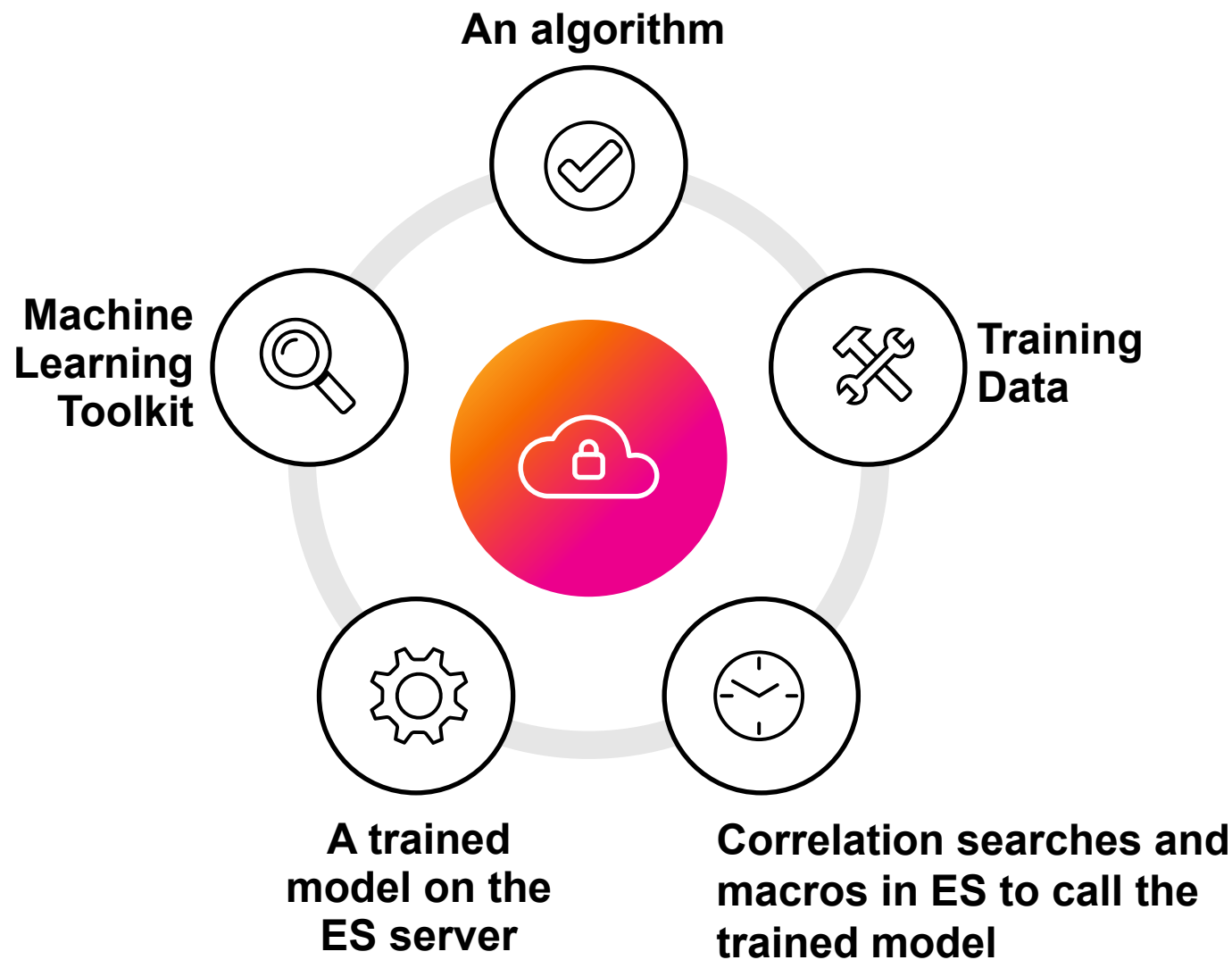
N. Y. Post Office --
assorting [i.e., sorting]
mail. Bain News
Service, publisher.
Between ca. 1914 and
1915. Library of
Congress.



Image by https://www.freepik.com/free-photo/colleagues-working-together-call-center-with-headphones_22196567.htm#query=call%20center%20office&position=16&from_view=keyword&track=ais
>Freepik

How Do We Get the New Algorithm Into ES?

SAME PROCESS!



Our New Correlation Search in ES

From conception to implementation

Correlation Search

Search Name

Network Traffic Where Next Hop Isn't Expected (AdaBoost)

App

DA-ESS-NetworkProtection

UI Dispatch

Context

Enterprise Security

Set an app to use for links such as the drill-down search in a notable event or links in an email adaptive response action. If None, uses the Application Context.

Description

Mode

Guided

Manual

Search

```
| inputlookup netflow_sampling_1000_50-50_test.csv
| fit AdaBoostClassifier "nexthop" from "dstaddr" "dstport" into "ada_netflow_predicted_nexthop"
| apply ada_netflow_predicted_nexthop | rename predicted(nexthop) as predicted_nexthop | eval isPredicted = if(nexthop
=predicted_nexthop, "true", "false") | search isPredicted="false" | stats count
```

What Happens if the Models Stop Working?



Machine Learning is Iterative

Start Again!

- Determine why the model no longer works
 - Did something change with the data?
- Determine if a better model or multiple models give you a better fit?
- Visualize the data
- Test and benchmark against new data samples
- Explore the features used

Your Call to Action





Add Machine Learning to new ES uses cases by:

- 1) Assessing your new use case
 - What are the important fields?
- 2) Finding a data sample to test against
 - How well do you know your data?
- 3) “Borrowing” from other correlation searches
- 4) Creating and/or applying the MLTK algorithms
- 5) Creating the correlation search
 - Do you want notables?
- 6) Testing and benchmarking
 - How well is it performing?

Suggestions For Additional Use Cases

New use cases means new opportunities

- Time Series Analysis:
 - Failed Logins over time: More failed logins than normal
 - Downloading more data over time: More downloads than normal
 - Web traffic over time: Reaching out to websites at unusual hours
- Natural Language Processing:
 - Phishing: Flag or block suspicious emails
- Behavior Analytics:
 - Abnormal user or system behavior such as devices communicating with more other systems than typical for that system
- Risk:
 - Watch for risk scores rising at a greater rate than the corpus, or falling to an extreme amount
- Detect overall trends across the enterprise. Are we failing more logins than normal? Are we downloading much more data? Are we reaching out to websites at unusual hours?

Zoo Data Set

A classic multivariate dataset containing 17 attribute values and an additional type attribute. See the dataset and the linked papers using the dataset at <https://archive.ics.uci.edu/dataset/111/zoo>.

Beginner data set for the new ML enthusiast.

Forsyth, Richard. (1990). Zoo. UCI Machine Learning Repository. <https://doi.org/10.24432/C5R59V>.

inputlookup zoo.csv																Last 24 hours	🔍
✓ 101 results (7/5/23 2:00:00.000 AM to 7/6/23 2:30:25.000 AM) No Event Sampling																Job	Smart Mode
Events Patterns Statistics (101) Visualization																	
20 Per Page Format Preview																< Prev 1 2 3 4 5 6 Next >	
airborne	animal name	aquatic	backbone	breathes	catsize	domestic	eggs	feathers	fins	hair	legs	milk	predator	tail	toothed	type	venomous
0	aardvark	0	1	1	1	0	0	0	0	1	4	1	1	0	1	1	0
0	antelope	0	1	1	1	0	0	0	0	1	4	1	0	1	1	1	0
0	bass	1	1	0	0	0	1	0	1	0	0	0	1	1	1	4	0
0	bear	0	1	1	1	0	0	0	0	1	4	1	1	0	1	1	0
0	boar	0	1	1	1	0	0	0	0	1	4	1	1	1	1	1	0
0	buffalo	0	1	1	1	0	0	0	0	1	4	1	0	1	1	1	0
0	calf	0	1	1	1	1	0	0	0	1	4	1	0	1	1	1	0
0	carp	1	1	0	0	1	1	0	1	0	0	0	0	1	1	4	0
0	catfish	1	1	0	0	0	1	0	1	0	0	0	1	1	1	4	0
0	cavy	0	1	1	0	1	0	0	0	1	4	1	0	0	1	1	0
0	cheetah	0	1	1	1	0	0	0	0	1	4	1	1	1	1	1	0
1	chicken	0	1	1	0	1	1	1	0	0	2	0	0	1	0	2	0



Question Time

Please remember to ask questions in the form of a question

Thank You

