Leveraging R and Python in Tableau

with RStudio Connect

October 29, 2021

James Blair

Solutions Engineer @ RStudio

RStudio

Use the right tool for the job

Photo by Elena Rouame on Unsplash

FastAPI

jupyter

plotly | Dash

Flask
web development,
one drop at a time

bokeh

Streamlit

RStudio

tableau®

FastAPI
jupyter
plotly | Dash
Flask
web development,
one drop at a time
bokeh
Streamlit
RStudio®

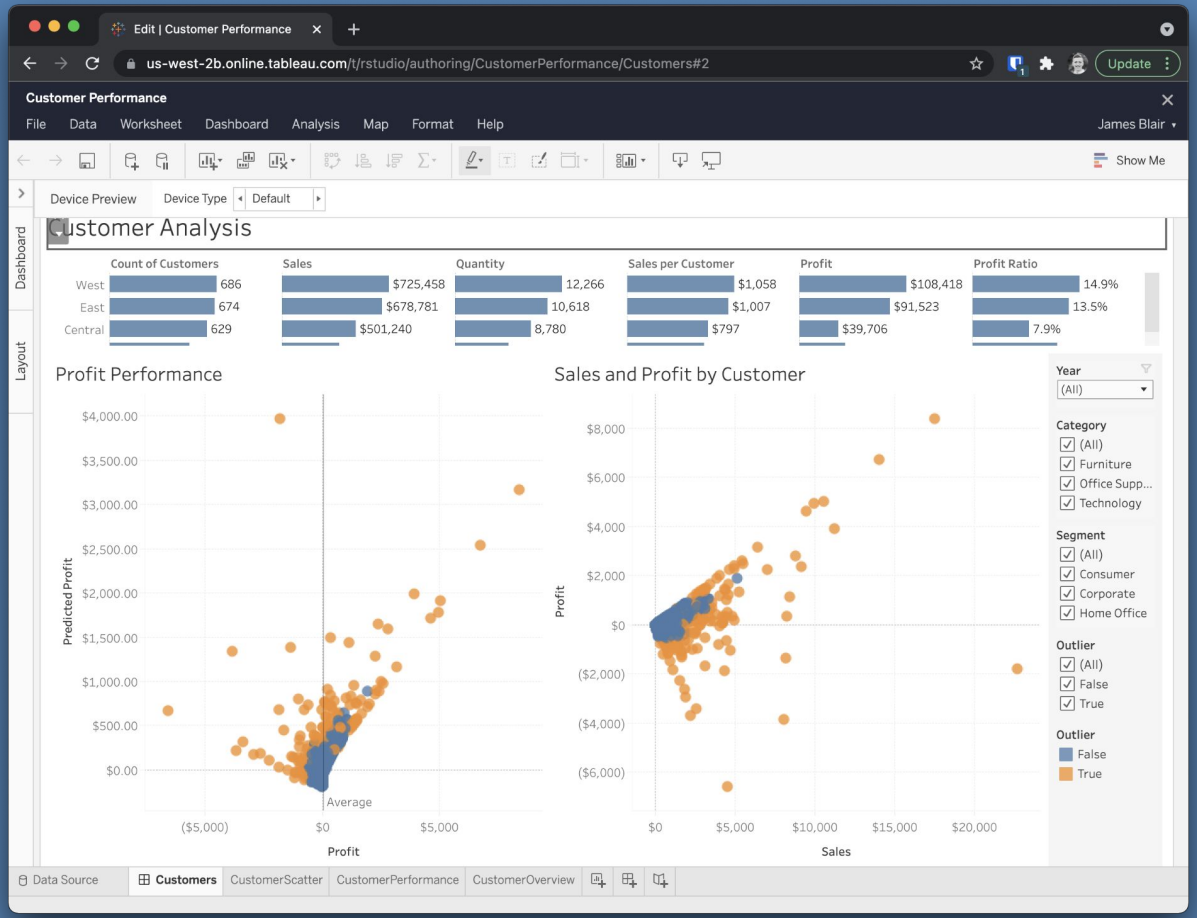
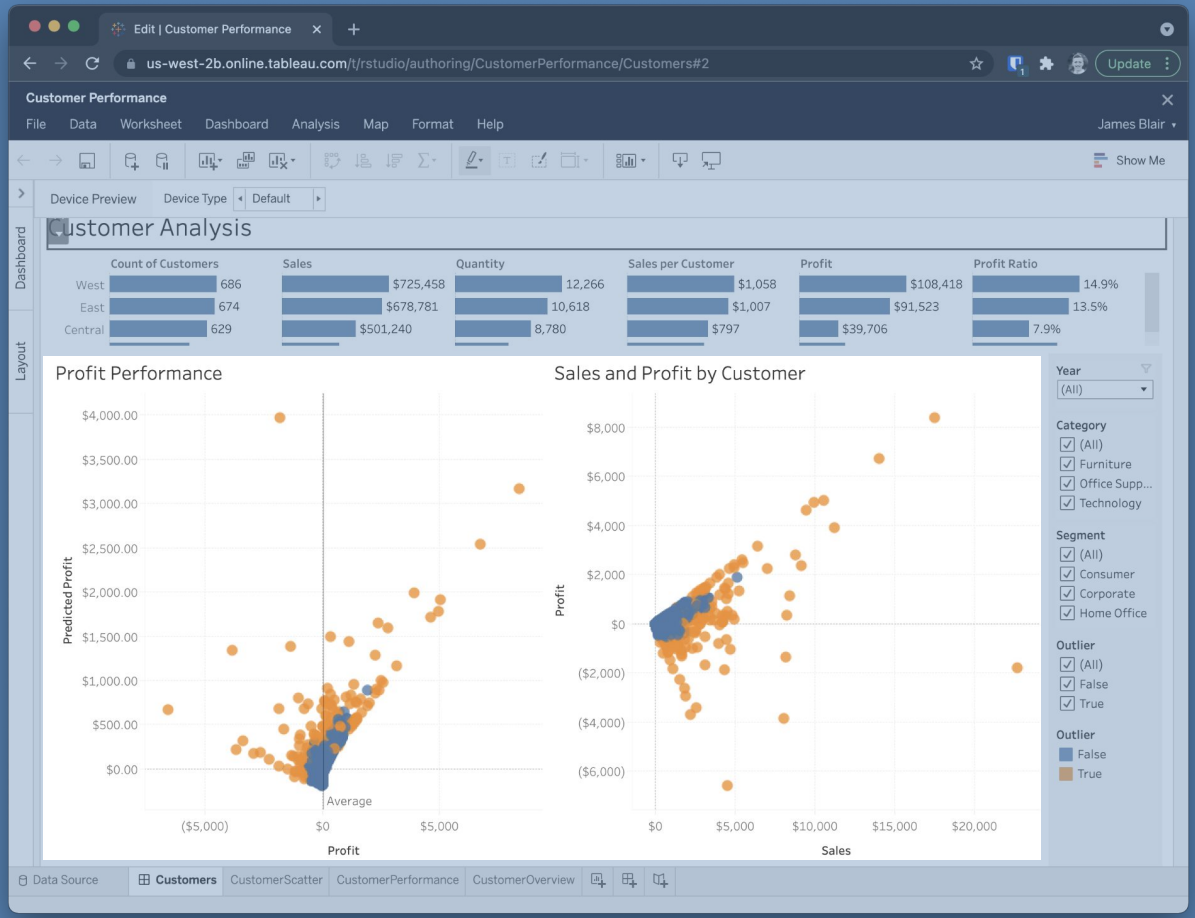


Tableau Dashboard

Customer sales performance evaluation

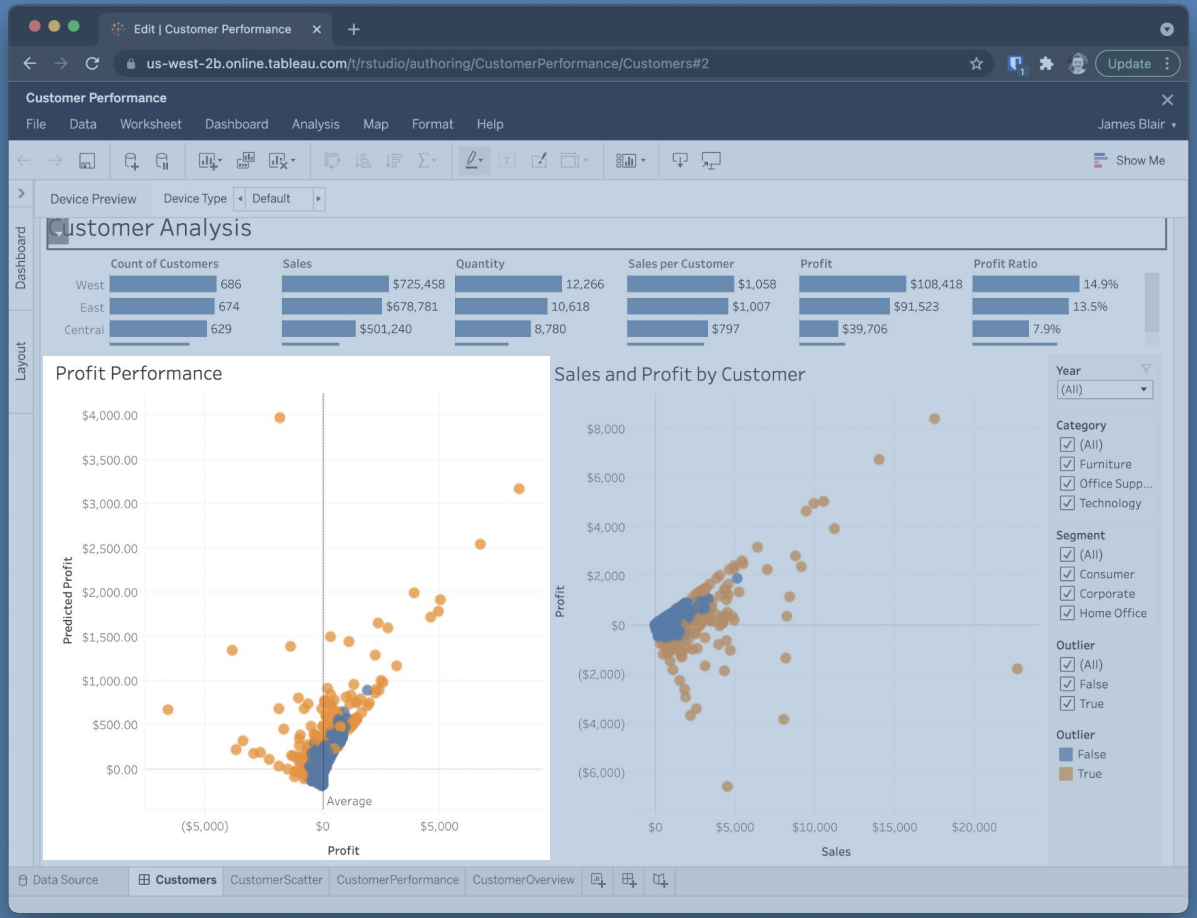
RStudio



R extension

Detect outliers given a set of inputs

RStudio



Python extension

Predict profit using a previously trained ML model and return predicted values to Tableau

RStudio

Back to the beginning

Photo by Jukan Tateisi on Unsplash

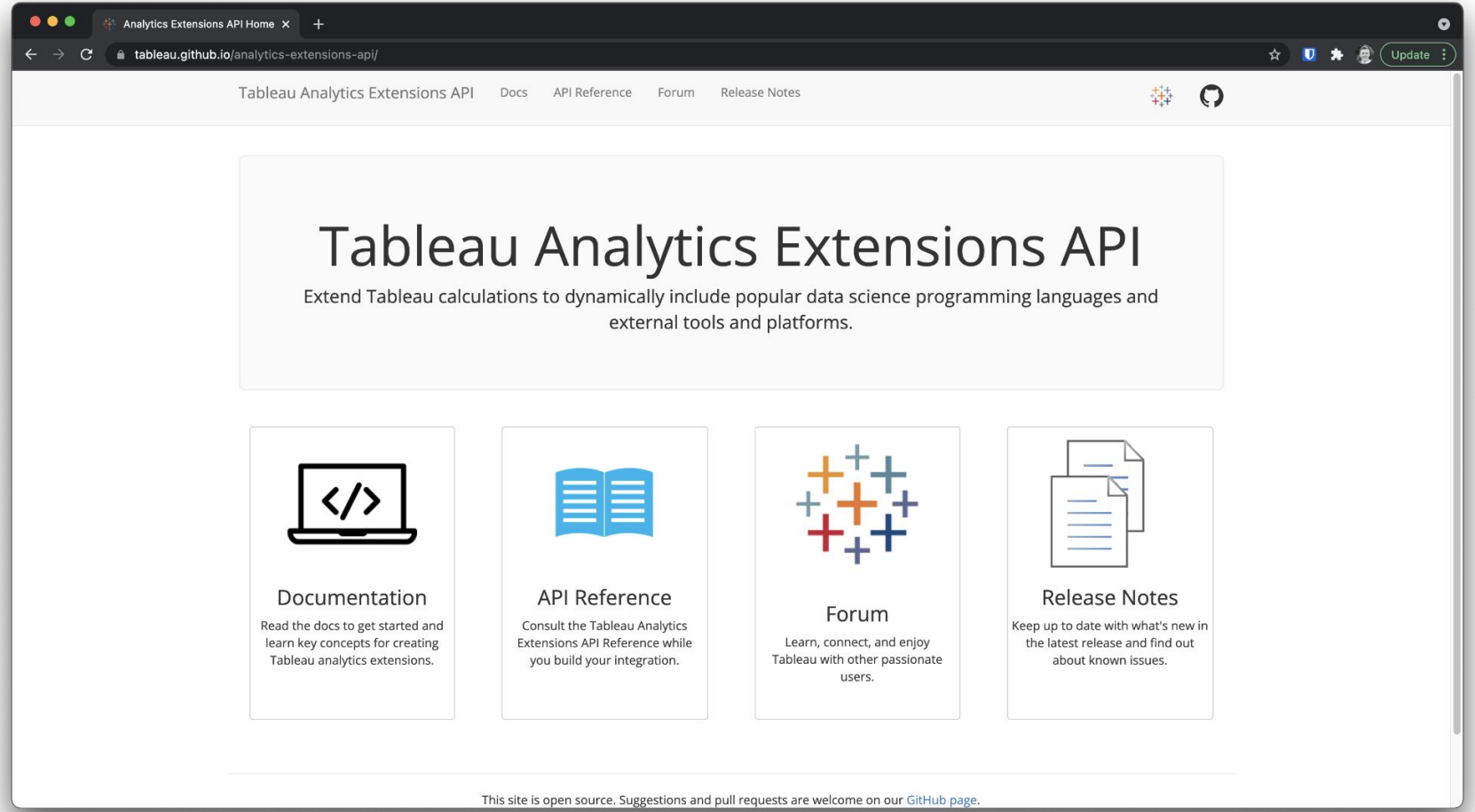
TabPy and Rserve

Groups | US Tornados 1950-2017

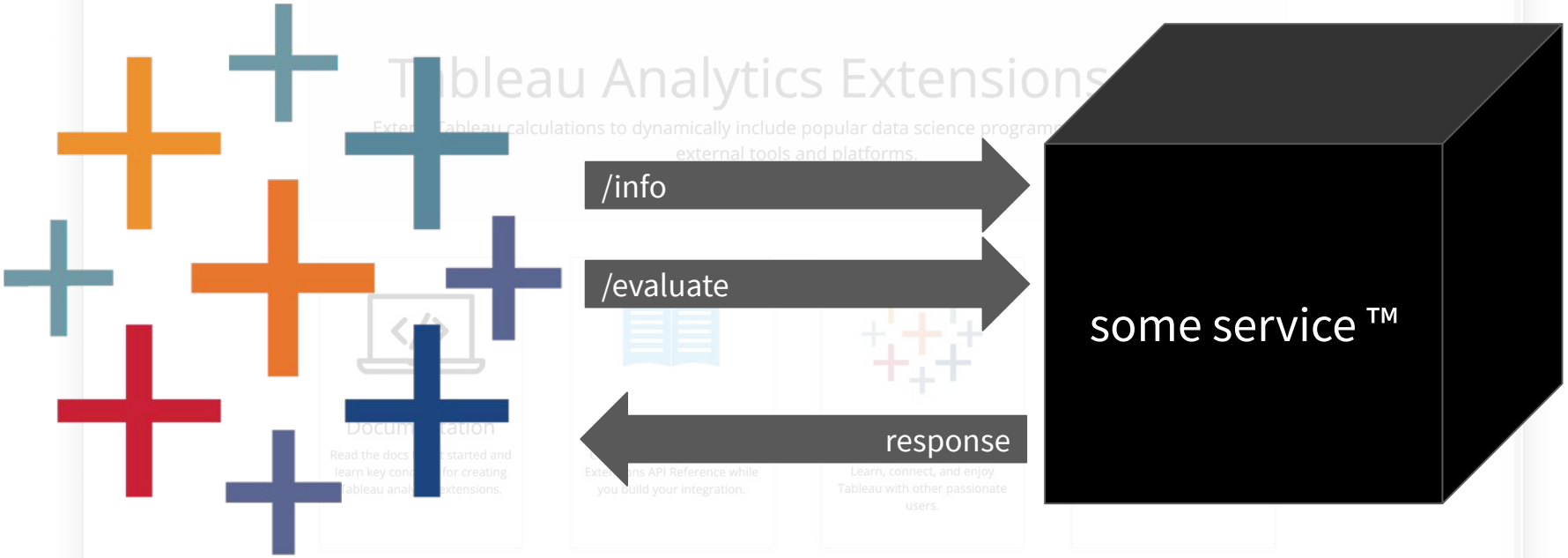
Results are computed along Table (across).

```
SCRIPT_STR("import numpy as np
from sklearn.cluster import DBSCAN
X=np.column_stack([np.radians(_arg1),np.radians(_arg2)])
db = DBSCAN(eps=_arg3[1], min_samples=_arg4[1], metric='haversine').fit(X)
return np.where(db.labels_ == np.array(-1), 'No', 'Yes').tolist()",
AVG([Slat]),AVG([Slon]),
[Distance between incidents],[Number of Incidents per Quarter])
```

RStudio

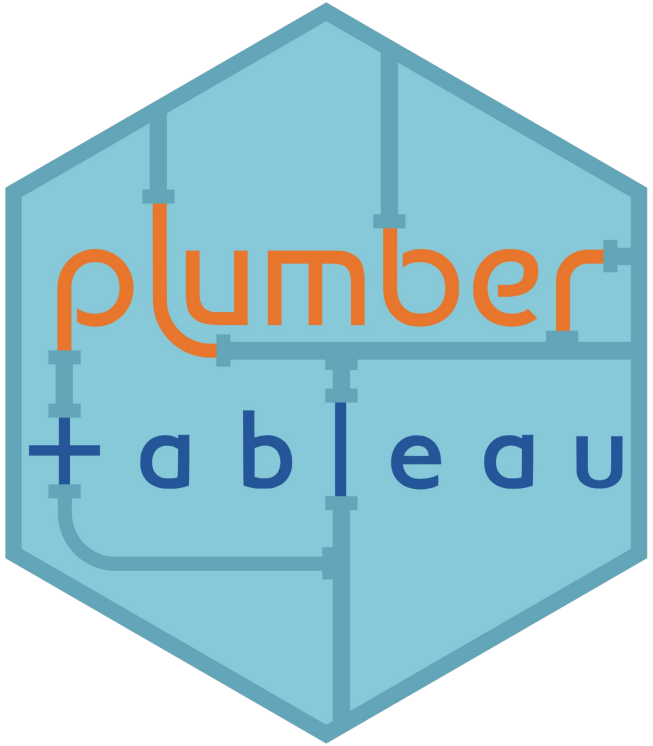

Analytics Extensions API Home
tableau.github.io/analytics-extensions-api/
Update
Tableau Analytics Extensions API
Docs
API Reference
Forum
Release Notes
Tableau Analytics Extensions API
Extend Tableau calculations to dynamically include popular data science programming languages and external tools and platforms.
Documentation
Read the docs to get started and learn key concepts for creating Tableau analytics extensions.
API Reference
Consult the Tableau Analytics Extensions API Reference while you build your integration.
Forum
Learn, connect, and enjoy Tableau with other passionate users.
Release Notes
Keep up to date with what's new in the latest release and find out about known issues.
This site is open source. Suggestions and pull requests are welcome on our GitHub page.

RStudio


/info
/evaluate
response
some service ™

RStudio

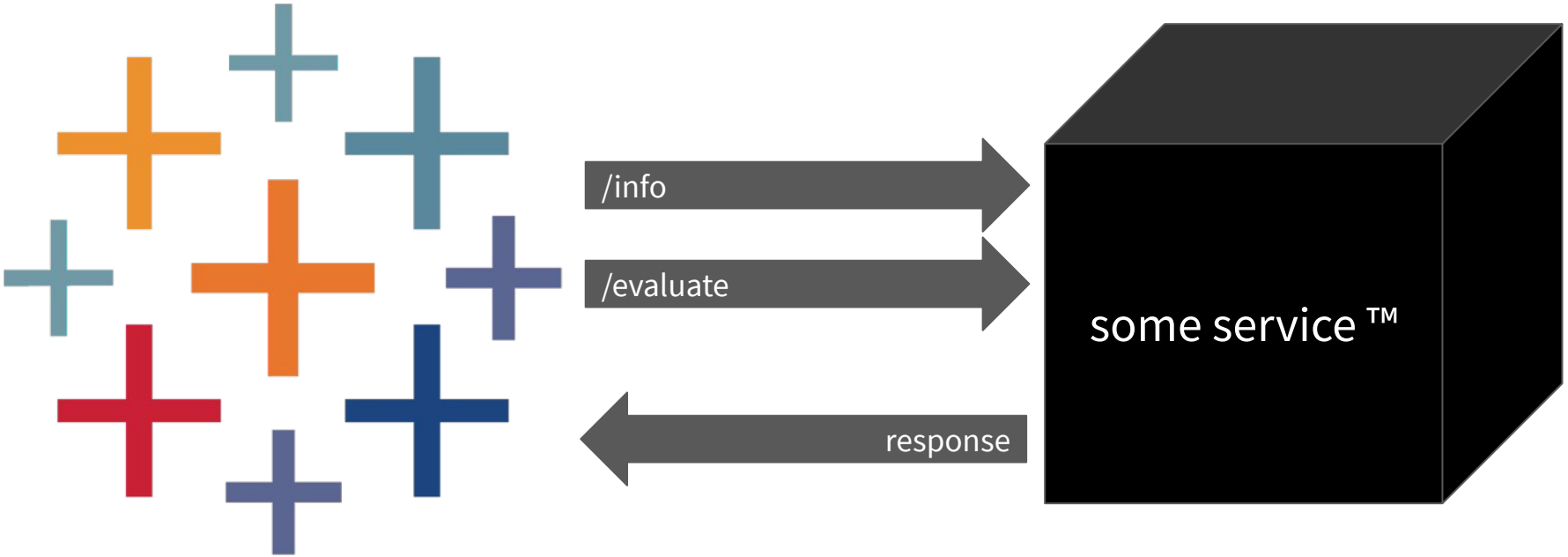
```r
library(plumber)
library(plumbertableau)
library(outForest)
library(dplyr)

#* @apiTitle Outlier Detection for Tableau
#* @apiDescription Detect outliers in real-time on Tableau
data using a Random Forest

#* Calculate outliers on input data
#* @tableauArg sales:numeric Numeric values representing
sales for a given transaction
#* @tableauArg profit:numeric Numeric values representing
profit for a given transaction
#* @tableauReturn logical A vector indicating the outlier
status of each original observation
#* @post /detect-outliers
function(sales, profit) {
  dat <- tibble(sales, profit)
  out <- outForest(dat)
  outlier_rows <- outliers(out) %>%
    select(row) %>%
    distinct()

  dat %>%
```



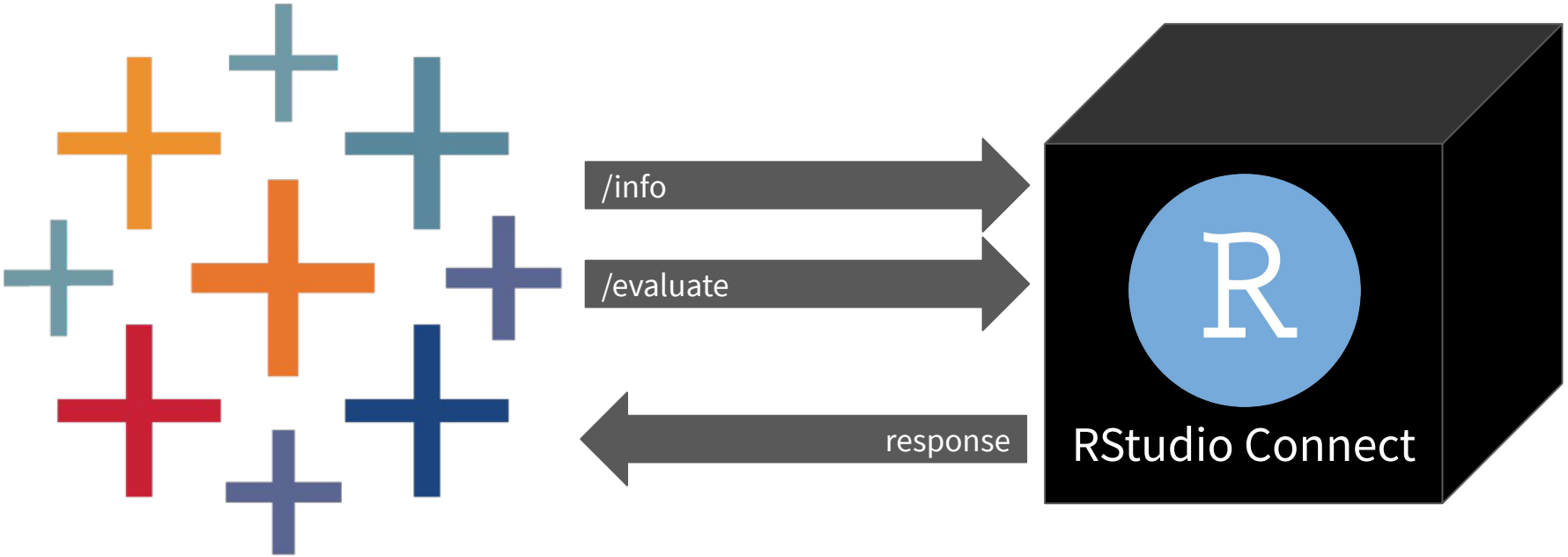
RStudio


```
from fastapitableau import FastAPITableau
from joblib import load
import pandas as pd
from typing import List

# Load model
model = load("model.joblib")

# Define model_pipline
def model_pipeline(dict):
    model_data = pd.DataFrame(dict)
    model_data["ship_diff"] = model_data.days_to_ship_actual - model_data.days_to_ship_scheduled
    pred_columns = [
        "ship_diff",
        "quantity",
        "sales",
        "discount"
    ]
    return model.predict(model_data.loc[:, pred_columns]).tolist()

# Define the extension
app = FastAPITableau(
    title = "Predicted Profit",
    description = "A simple linear prediction of sales profit given new input data"
```

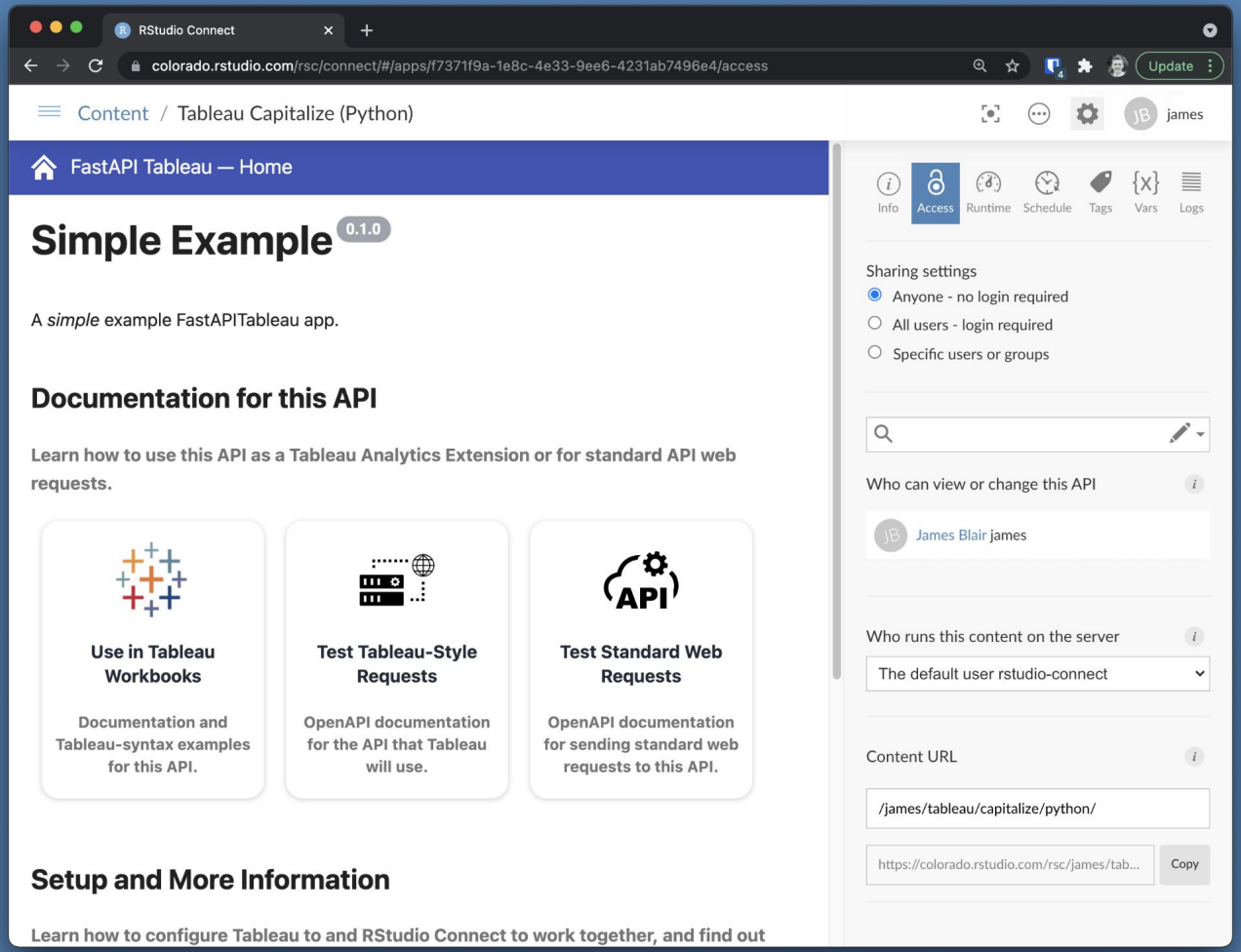
RStudio



RStudio

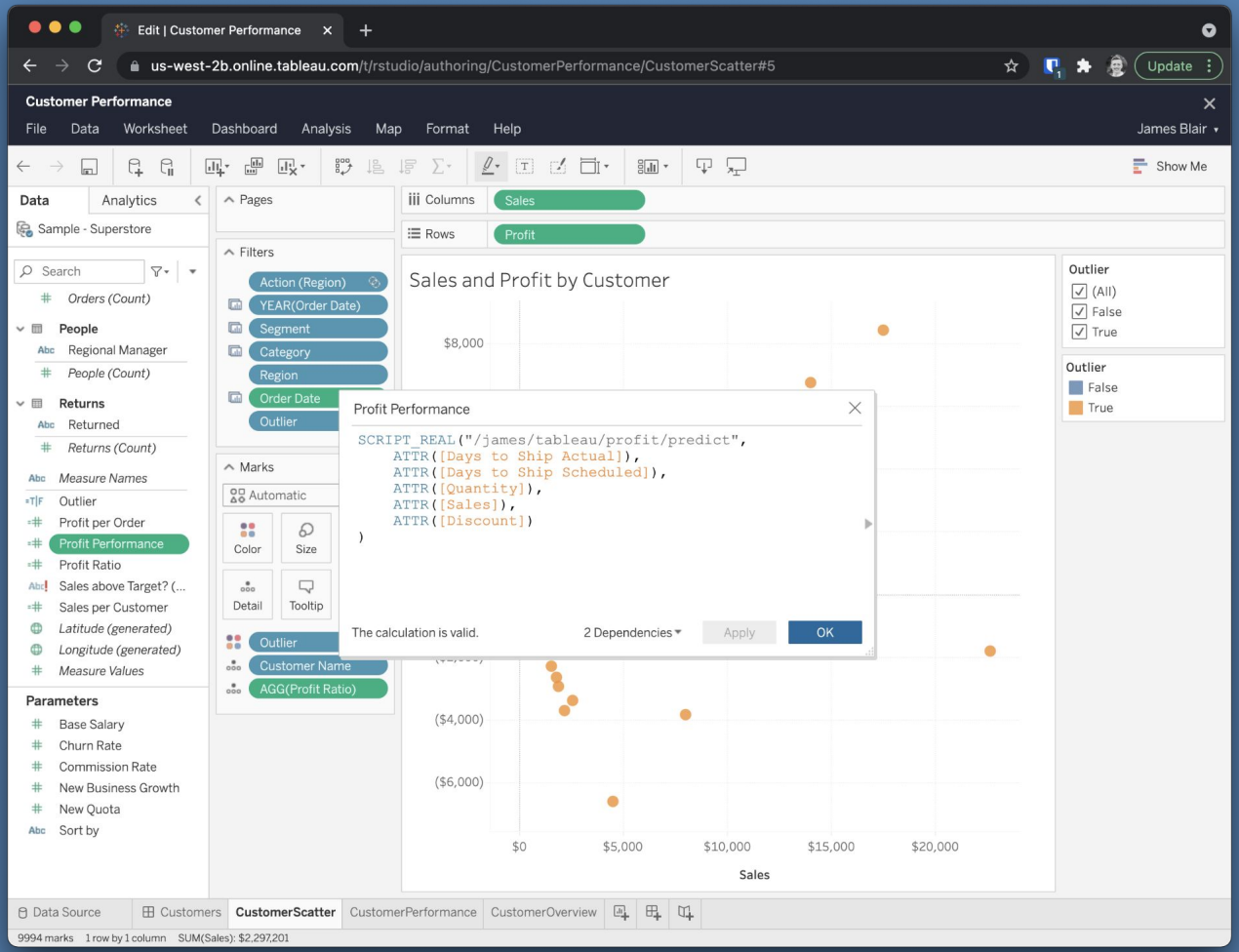


RStudio



RStudio Connect

RStudio



Tableau

RStudio

Groups

US Tornados 1950-2017

Results are computed along Table (across).

```
SCRIPT_STR("import numpy as np
from sklearn.cluster import DBSCAN
X=np.column_stack([np.radians(_arg1),np.radians(_arg2)])
db = DBSCAN(eps=_arg3[1], min_samples=_arg4[1], metric='haversine').fit(X)
return np.where(db.labels_ == np.array(-1), 'No', 'Yes').tolist()",
AVG([Slat]),AVG([Slon]),
[Distance between incidents],[Number of Incidents per Quarter])
```

RStudio

Groups

US Tornados 1950-2017

Results are computed along Table (across).

```
SCRIPT_STR("import numpy as np
from sklearn.cluster import DBSCAN
X=np.column_stack([np.radians(_arg1),np.radians(_arg2)])
db = DBSCAN(eps=_arg3[1], min_samples=_arg4[1], metric='haversine').fit(X)
return np.where(db.labels_ == np.array(-1), 'No', 'Yes').tolist()",
AVG([Slat]),AVG([Slon]),
[Distance between incidents],[Number of Incidents per Quarter])
```

Tornados

```
SCRIPT_STR(
  "/extensions/tornado/cluster",
  AVG([Slat]),
  AVG([Slon]),
  [Distance between incidents],
  [Number of Incidents per Quarter]
)
```

Tableau + RStudio Connect

- Use R and Python extensions from the same Tableau workbook
- Enterprise level security provided by RStudio Connect
- R/Python developers and Tableau developers can be separate
- Environment management provided by RStudio Connect

RStudio

Resources

- Blog post
- plumbertableau
- fastapitableau
- shinytableau
- Collection of examples

Photo by Ed Robertson on Unsplash

bit.ly/rstudio-tableau

RStudio

Photo by Alexas_Fotos on Unsplash