⟵ BACK TO BLOG

Building advanced analytics applications with TabPy

SHARE

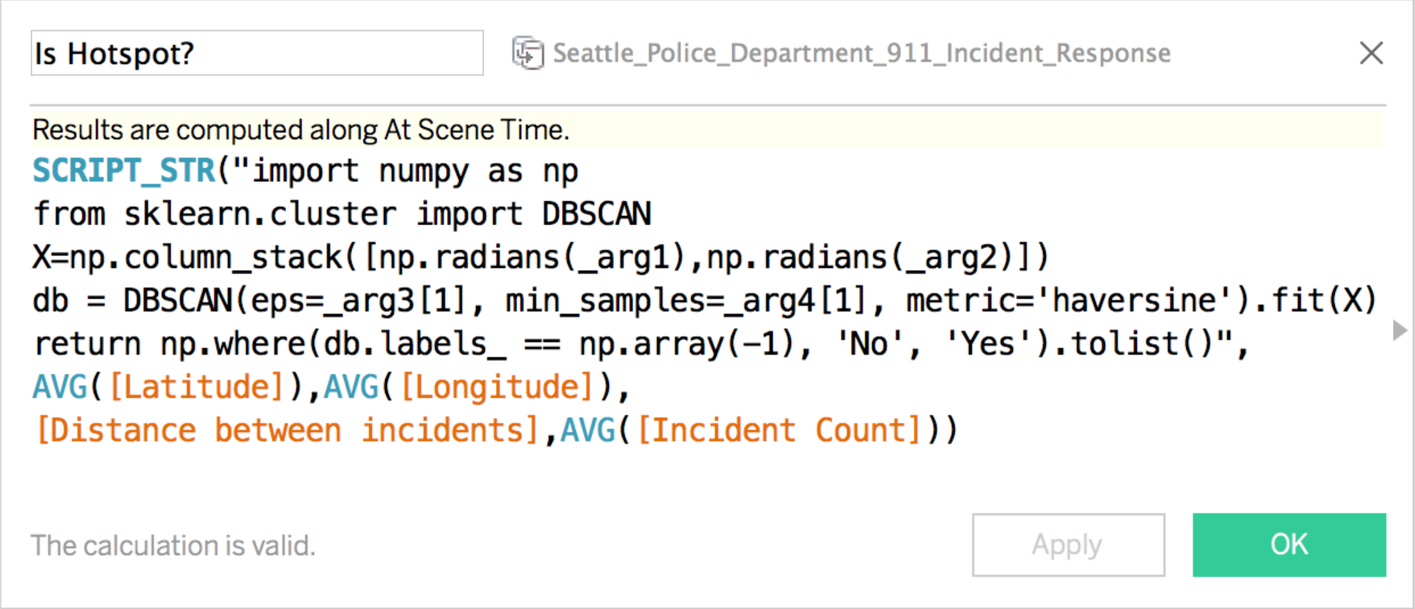
BORA BERAN
PRODUCT MANAGER, TABLEAU SOFTWARE
JANUARY 24, 2017

Back in November, we introduced TabPy (currently in beta), making it possible to use Python scripts in Tableau calculated fields. When you pair Python's machine-learning capabilities with the power of Tableau, you can rapidly develop advanced-analytics applications that can aid in various business tasks.

Let me show you what I mean with an example. Let's say I'm trying to identify criminal hotspots in Seattle, my hometown. I'll use data from the Seattle Police Department showing 911 calls for various type of criminal activities in the past few years.

With this data, it is really hard to visually identify patterns given the density of activity and noise in GPS readings. Let's see what we can find out by applying some unsupervised machine learning.

Density-based spatial clustering of applications with noise (DBSCAN) is a well-suited algorithm for this job. It is also installed conveniently by default as part of TabPy. It takes two parameters: one to specify the maximum allowed distance between points for them to be considered part of the same cluster and one more to specify the minimum number of nearby points to constitute a cluster.

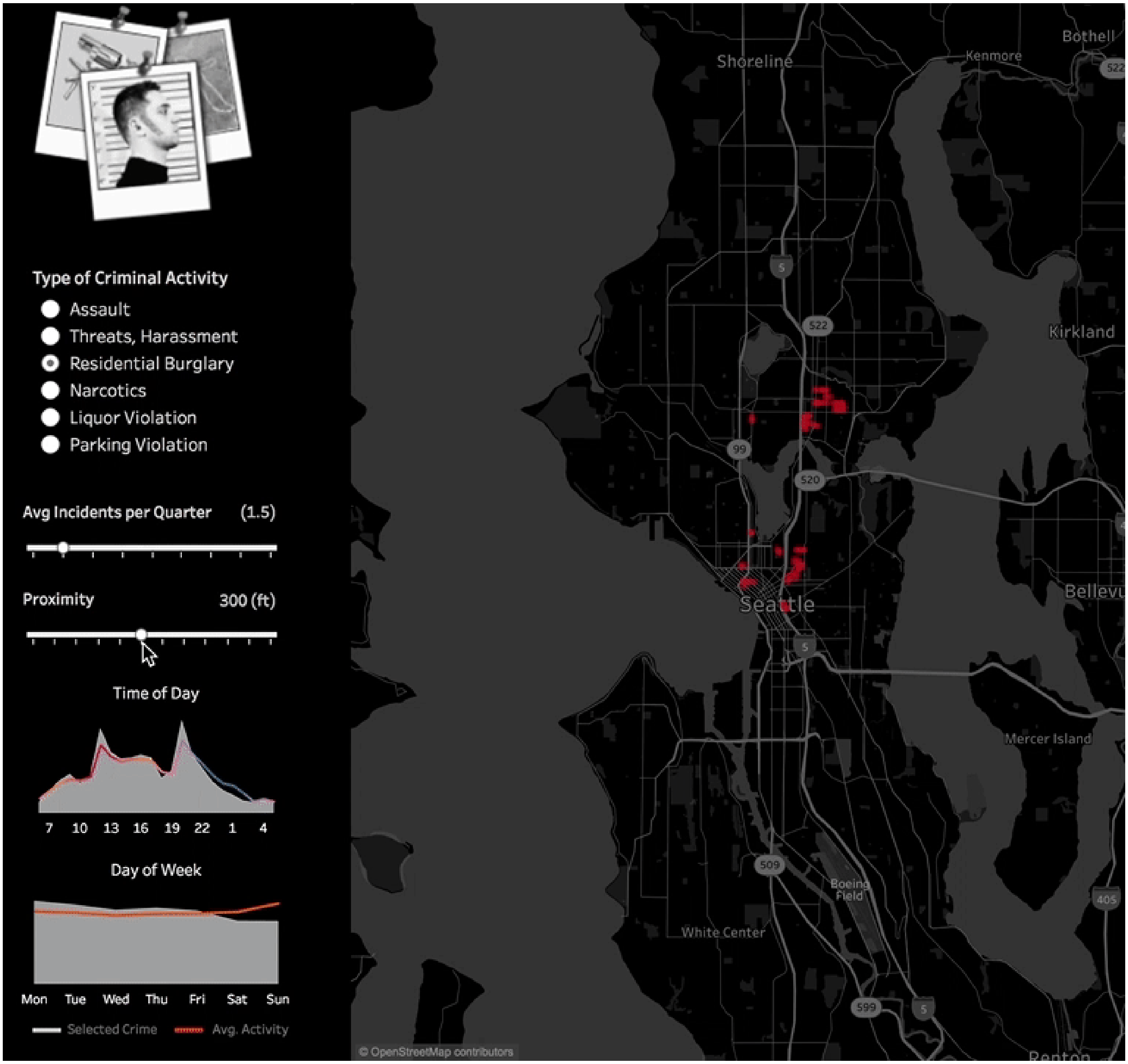


Results are computed along At Scene Time.

```
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([Latitude]),AVG([Longitude]),
[Distance between incidents],AVG([Incident Count]))
```

The calculation is valid.

This allows for experimenting with different values of distance and event frequencies criteria. Different options can be more appropriate for downtown Seattle versus the suburbs, a police officer looking for hotspots versus a tourist looking for places to avoid, or a tenant looking for houses to rent or buy. You can download this example Tableau workbook here.



Embedding the Python code into Tableau worked great in this example. But in some cases, you may want to host your Python scripts outside Tableau workbooks so they are centralized and easier to manage or because the models themselves require upfront training.

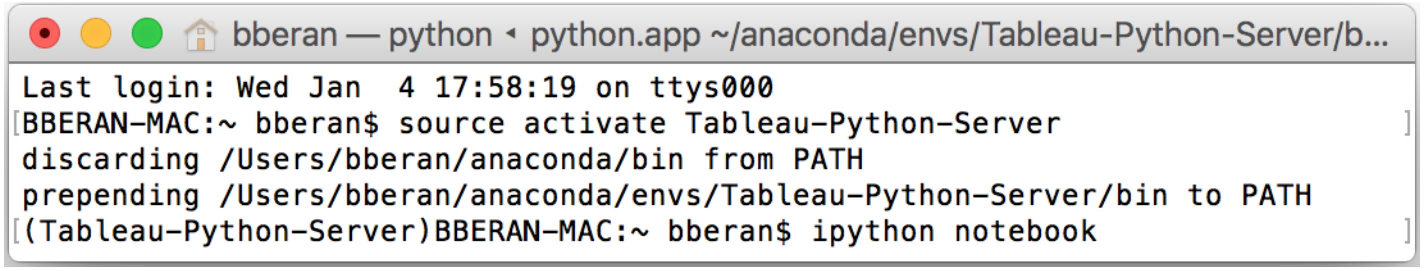
To demonstrate, let's use a data set on breast cancer cases in Wisconsin. And let's see if we can train a model that can provide the correct diagnosis given a patient's test results.

Let's start with the model that is most easily accessible as part of our exploration: clustering, which we introduced in Tableau 10. When I simply double-click on clustering, I see that Tableau automatically finds two clusters corresponding to malignant and benign tumors, and identifies the cases with 92.2% accuracy. That's pretty impressive considering I gave Tableau no hint whatsoever as to what the correct diagnosis were or even that there had to be two categories.

But since we have this information in the data set, could we use a different algorithm that can learn from the actual diagnosis for these patients? Let's try a variety of supervised machine-learning algorithms in Python and see how they will perform.

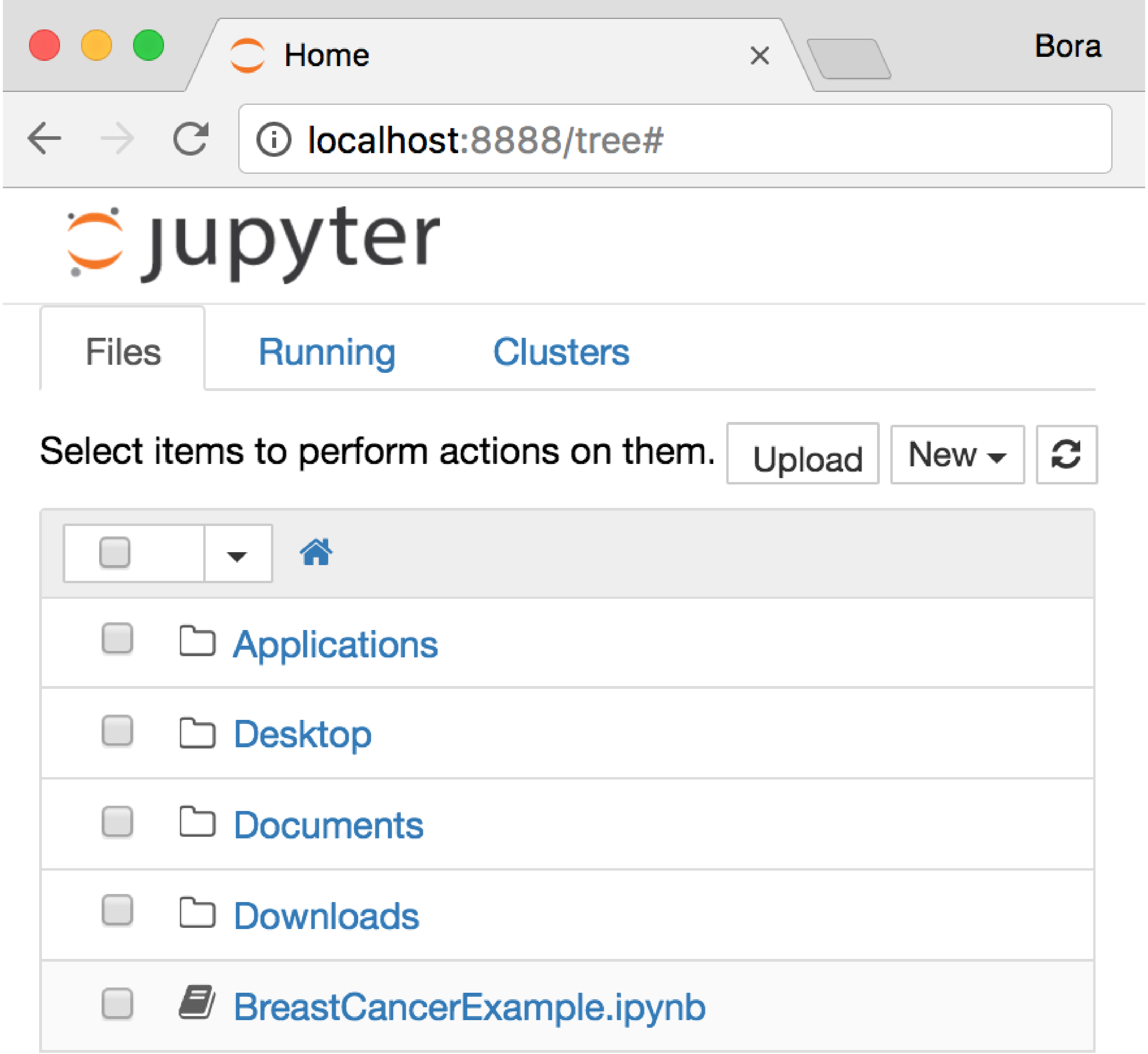
You can download the Jupyter work containing all the Python code used for model training and evaluation here.

Before you can use it, you need to start Jupyter.


bberan — python ◂ python.app ~/anaconda/envs/Tableau-Python-Server/b...

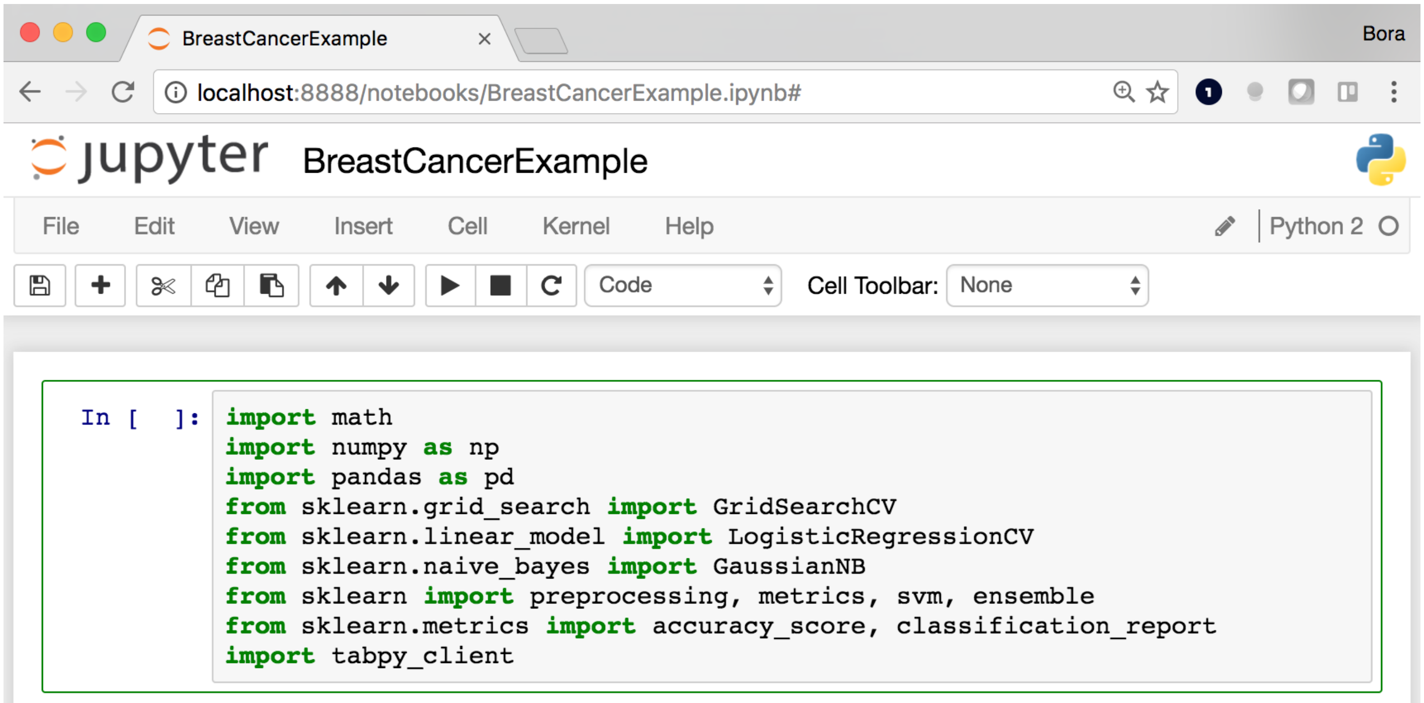
```
Last login: Wed Jan  4 17:58:19 on ttys000
[BBERAN-MAC:~ bberan$ source activate Tableau-Python-Server
discarding /Users/bberan/anaconda/bin from PATH
prepending /Users/bberan/anaconda/envs/Tableau-Python-Server/bin to PATH
[(Tableau-Python-Server)BBERAN-MAC:~ bberan$ ipython notebook
```

This will open Jupyter in your browser. If you downloaded the example notebook, you can navigate to the directory on this screen and click on BreastCancerExample.ipynb to open it.



Select items to perform actions on them.

Once the notebook loads, your browser window should look like this. Note that this workbook relies on many packages in scikit-learn 16.0.1, the version which ships with TabPy by default. Earlier or newer versions may lack some of these methods or have different names for them.



```
import math
import numpy as np
import pandas as pd
from sklearn.grid_search import GridSearchCV
from sklearn.linear_model import LogisticRegressionCV
from sklearn.naive_bayes import GaussianNB
from sklearn import preprocessing, metrics, svm, ensemble
from sklearn.metrics import accuracy_score, classification_report
import tabpy_client
```

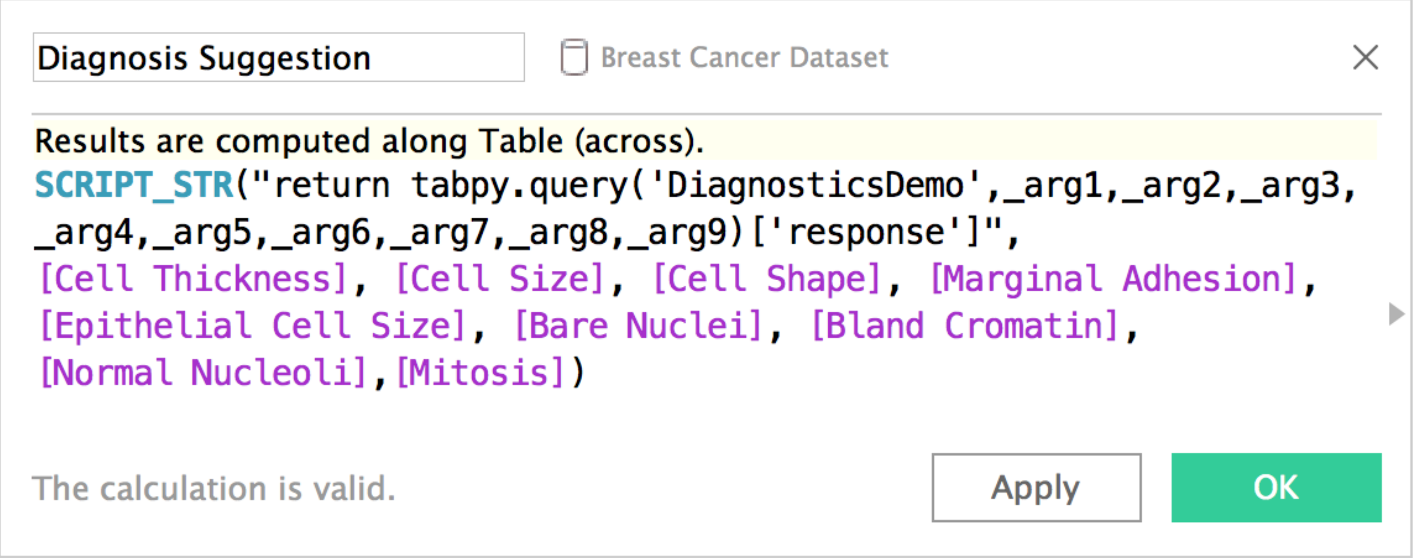
The notebook is extensively documented so I won't get into the details in this post. (In a few words, what it does is to fit Naïve Bayes, Logistic Regression, Support Vector Machine and Gradient Boosted Tree models to the breast cancer data set by doing a grid search with k-fold cross-validation to find the best model.) This sample is also meant to be a template you can swap in different models easily, for example to use a neural network instead.

```
# Connect to TabPy server using the client library
connection = tabpy_client.Client('http://localhost:9004/')
```

```
# The scoring function that will use the Gradient Boosting Classifier to classify new data points
def SuggestDiagnosis(Cl_thickness, Cell_size, Cell_shape, Marg_adhesion, Epith_c_size,
                     Bare_nuclei, Bl_cromatin, Normal_nucleoli, Mitoses):
    X = np.column_stack([Cl_thickness, Cell_size, Cell_shape, Marg_adhesion, Epith_c_size,
                         Bare_nuclei, Bl_cromatin, Normal_nucleoli, Mitoses])
    X = scaler.transform(X)
    return encoder.inverse_transform(gbclf.predict(X)).tolist()
```

```
# Publish the SuggestDiagnosis function to TabPy server so it can be used from Tableau
# Using the name DiagnosticsDemo and a short description of what it does
connection.deploy('DiagnosticsDemo',SuggestDiagnosis,
'Returns diagnosis suggestion based on ensemble model trained using Wisconsin Breast Cancer dataset')
```

Then it deploys the best model (Gradient Boosting in this case) as a function to the TabPy server so it can be used to classify new data from Tableau dashboards.

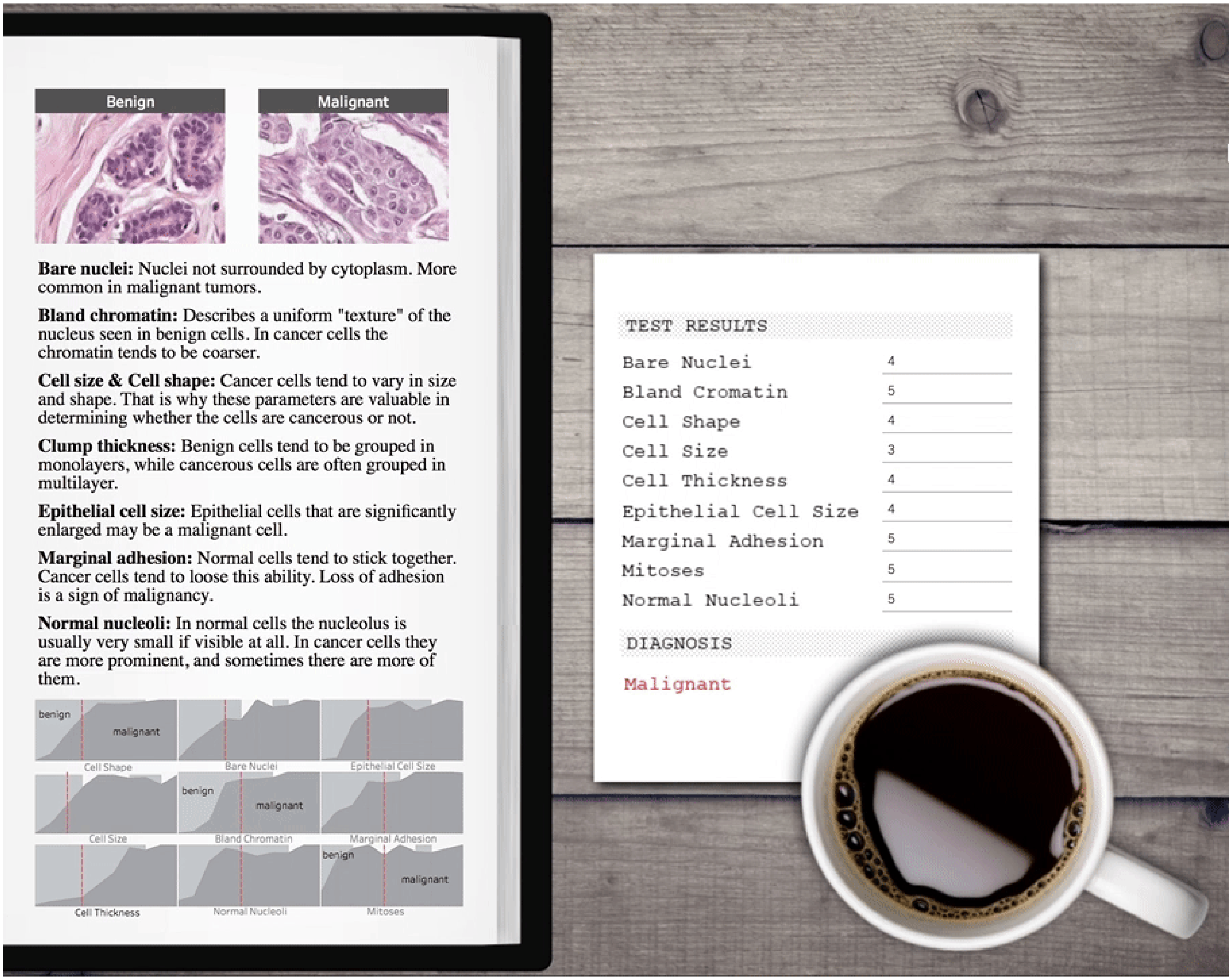


Results are computed along Table (across).

```
SCRIPT_STR("return tabpy.query('DiagnosticsDemo',_arg1,_arg2,_arg3,
_arg4,_arg5,_arg6,_arg7,_arg8,_arg9)['response']",
[Cell Thickness], [Cell Size], [Cell Shape], [Marginal Adhesion],
[Epithelial Cell Size], [Bare Nuclei], [Bland Cromatin],
[Normal Nucleoli],[Mitosis])
```

The calculation is valid.

Now we can call the published function from Tableau with configurable parameters so one can enter values to get a prediction and embed it in a nice dashboard.



Bare nuclei: Nuclei not surrounded by cytoplasm. More common in malignant tumors.

Bland chromatin: Describes a uniform "texture" of the nucleus seen in benign cells. In cancer cells the chromatin tends to be coarser.

Cell size & Cell shape: Cancer cells tend to vary in size and shape. That is why these parameters are valuable in determining whether the cells are cancerous or not.

Clump thickness: Benign cells tend to be grouped in monolayers, while cancerous cells are often grouped in multilayer.

Epithelial cell size: Epithelial cells that are significantly enlarged may be a malignant cell.

Marginal adhesion: Normal cells tend to stick together. Cancer cells tend to loose this ability. Loss of adhesion is a sign of malignancy.

Normal nucleoli: In normal cells the nucleolus is usually very small if visible at all. In cancer cells they are more prominent, and sometimes there are more of them.

TEST RESULTS	
Bare Nuclei	4
Bland Cromatin	5
Cell Shape	4
Cell Size	3
Cell Thickness	4
Epithelial Cell Size	4
Marginal Adhesion	5
Mitoses	5
Normal Nucleoli	5

DIAGNOSIS

Malignant

You can download this example Tableau workbook here.

There are many more use cases for TabPy for data scientists. You might use it to build models for your HR department to predict attrition. You might help your sales department score leads. Or you might be an ISV creating vertical-specific advanced-analytics applications using Tableau to help renters make more educated decisions when picking their next home. Whatever your use case, TabPy can help take your analytics to 11.

To learn more about TabPy and to install it, visit our GitHub page. How are you using TabPy for advanced analytics? Tell us about your use case in the comments below.

Try Tableau 10.2

Try out all the new features in this post, and many more coming to Tableau Desktop, by signing up for Tableau's beta program here. And visit our Coming Soon page to learn about all the features we're planning for Tableau 10.2.

Learn more about Tableau 10.2

Tableau 10.2 beta is here
Leverage the power of Python in Tableau with TabPy
Cut data-prep time with these enhancements in Tableau 10.2
Do more with your data on the web in Tableau Online 10.2

You might also be interested in...

Comments

Submitted by saranya (not verified) on March 6, 2017 - 1:05am

Hii..Your posting is really much more informative and helpful to all people...Thanks for sharing these types of informative updates...
Reply

Submitted by AM (not verified) on March 7, 2017 - 1:04pm

The link to the workbook
http://54.186.231.117/#/site/Python/views/TabPyPublishedModel/BreastCancer
requires a username and password.
Reply

Submitted by Ashwin Rai on March 24, 2017 - 9:52am

Hi Bora,

First of all amazing article. I loved it. I used TabPy and I tried to publish the workbook. The workbook got published but I when I open it I get the below error

An unexpected error occurred. If you continue to receive this error please contact your Tableau Server Administrator.

TableauException: An error occurred while communicating with the external service. Tableau is unable to connect to the service. Verify that the service is running and that you have access privileges.
2017-03-24 16:45:41.355 (WNVNMA0fA8QAABFABIYAAAPl,0,0)

TabPy is running on linux server. Any suggestions would be really helpful
Reply

Submitted by Bora Beran on March 24, 2017 - 10:43pm

Hi Ashwin,
Is TabPy configured on Tableau Server?

```
tabadmin stop
tabadmin set vizqlserver.extsvc.host hostnamegoeshere
tabadmin set vizqlserver.extsvc.port portnumbergoeshere
tabadmin config
tabadmin start
```

If it is still not working it could be a firewall issue or did you have Rserve configured on the same server before?

Thanks,

Bora

http://onlinehelp.tableau.com/current/server/en-us/tabadmin.htm
Reply

Submitted by Brit Cava (not verified) on April 30, 2017 - 5:45pm

Hi Bora,

Thanks for the great write up. I'm walking through your code and am attempting to reproduce it. So far I ran the code in your notebook, resolved some errors, and deployed the model. Next, I set-up the calculation per your post. However, when I attempt to use it I get this error: The endpoint you're trying to query did not respond. Please make sure the endpoint exists and the correct set of arguments are provided.

I thought that perhaps I didn't run all of the code so I reran it and got this error: RuntimeError: An endpoint with that name ('DiagnosticsDemo') already exists. Use 'override = True' to force update an existing endpoint.

I'm confused that it exist but it doesn't respond. What are possible reasons it wouldn't respond that I can work to resolve?

Thanks,
Brit
Reply

> Submitted by Bora Beran on October 2, 2017 - 4:18pm
>
> Most likely answer is that the model wasn't successfully trained. When this happens, you can still publish a function to TabPy server but the error will surface when you try to run the model. On some Windows machines, this could happen due to a bug in Python when you try to run model fitting by using all the CPU cores you have. This is controlled by the "n_jobs=-1" setting in the Jupyter workbook. I suspect this might be the issue.
>
> The solution is to set n_jobs=1 which will run the training using only a single core. After making this, if you add the argument override=True to your deploy function then run the entire Jupyter workbook (Cell > Run All) I suspect it might fix it.
> Reply

Submitted by Edward (not verified) on June 1, 2017 - 12:03pm

Hi Bora,

Firstly, thank you for the amazing tutorial. I'm getting an error when i open the breast cancer workbook. I can see the tabpy server is running perfectly. Tableau shows "There is an error connecting to the predictive service" - Endpoint dosen't exist etc..

I've tried running individual pieces of the jupyter notebook, fixed few errors and changed the csv file path to my local directory. Nothing worked..

The seattle criminal hotspots workbook is working perfectly, but there seems to be some problem with the breast cancer workbook. Please advise on how to approach.

Thank you in advance!

Best,
Edward
Reply

> Submitted by Bora Beran on July 17, 2017 - 5:12pm

> Most likely answer is that the model wasn't successfully trained. When this happens, you can still publish a function to TabPy server but the error will surface when you try to run the model. On some Windows machines, this could happen due to a bug in Python when you try to run model fitting by using all the CPU cores you have. This is controlled by the "n_jobs=-1" setting in the Jupyter workbook. I suspect this might be the issue. The solution is to set n_jobs=1 which will run the training using only a single core. After making this, if you add the argument override=True to your deploy function then run the entire Jupyter workbook (Cell > Run All) I suspect it might fix it.
> Reply

Submitted by jackjuzhang (not verified) on September 21, 2017 - 10:01pm

Hi Bora,

Really great tutorial! Thanks for your sharing.

I don't know if anyone met the same error message 'RuntimeError: LoadFailed: u'Load failed: range() integer end argument expected, got list.'' when conduct the deploy method in the end. Under both windows and Mac environment it just doesn't work.

```
RuntimeError Traceback (most recent call last)
in ()
3 connection.deploy('DiagnosticsDemo',
4 SuggestDiagnosis,
----> 5 'Returns diagnosis suggestion based on ensemble model trained using Wisconsin Breast Cancer dataset',override = True)

C:\Users\jack\anaconda\lib\site-packages\tabpy_client\client.pyc in deploy(self, name, obj, description, schema, override)
330 self._service.set_endpoint(_Endpoint(**obj))
331
--> 332 self._wait_for_endpoint_deployment(obj['name'], obj['version'])
```

```
333
334 def _gen_endpoint(self, name, obj, description, version=1, schema=[]):

C:\Users\jack\anaconda\lib\site-packages\tabpy_client\client.pyc in
_wait_for_endpoint_deployment(self, endpoint_name, version, interval)
451 if ep['status'] == 'LoadFailed':
452 raise RuntimeError("LoadFailed: %r" % (
--> 453 ep['last_error'],
454 ))
455

RuntimeError: LoadFailed: u'Load failed: range() integer end argument expected, got list.'
```

Could you please give some clue on how to fix this issue up?

Thank you very much!
Reply

> Submitted by bile.xu (not verified) on January 30, 2018 - 1:20am
>
> 我也碰到这个问题，后来无意中是这样解决的——我使用的windouw7环境、anaconda3（python=2.7）
> 直接在tabpy_server目录下点击startup.bat启动9004端口是没用的，正确路径：在开始菜单里点击“Anaconda Prompt”，这样会激活你anaconda环境，然后cd到startup.bat所在目录（比如cd D:\Anaconda2\Lib\site-packages\tabpy_server），接着再输入d:,按Enter，终端显示的路径就会是D:\Anaconda2\Lib\site-packages\tabpy_server>，在>后面输入startup.bat接着按Enter，当前窗口就会显示初始化9004端口等信息，这样就会正确启动了tabpy_server服务。
> Reply

Submitted by Andrea Pérez (not verified) on October 17, 2017 - 9:05am

I'm trying to download the breastcancer workbook, but I'm getting a communication error with the external server and I'd like to get the workbook

Reply

Submitted by Scott deVillers on October 17, 2017 - 9:59am

Nice! Haversine is supposed to take into account imperfections in the curvature of the Earth.

Reply

Submitted by Raj (not verified) on October 27, 2017 - 8:12am

Hi Bora,

I am new to Tabpy and Tableau. I want to read two columns from Tableau let say "detailed description" and "Description" and search the keywords 'password', 'high' and 'low' in detailed description and description columns and if the keywords match in either detailed description column or Description column or both columns then it should print the the outcomes what i define . I would be appreciated if you help me out with the running code which can do the calculation and gives result.

Thank you

Reply

Submitted by Sree Gudur on October 28, 2017 - 7:08am

@BORA BERAN, Hello sir,

I'm unable to download the workbook which is based on the seattle criminal activity. It would be a great help if you can share it with me.

Reply

> Submitted by galen.simmons (not verified) on November 22, 2017 - 3:08pm
>
> I'm having the same problem
>
> Reply

Submitted by Paola (not verified) on October 30, 2017 - 6:34am

Hello Bora,

when I run connection.deploy(...) I get "RuntimeError: LoadFailed: "Load failed: No module named 'sklearn'" and then even when trying to use another python function in tableau that has nothing to do with sklearn I get the same error.

I am running Anaconda with python 3.6.2 on Windows and I can use sklearn independently from Tableau (i.e. sklearn is installed on my machine). Any help is very much appreciated. Thanks
Reply

Submitted by Huzefa Barwaniwala (not verified) on November 17, 2017 - 9:48am

Hi Bora,

Thank you for explaining the Python use case. I want to go through your examples but I am unable to download the tableau Workbooks, I think it is a server error. Can you please help me to set it up? I am new to Tableau.

Thanks
Huzefa
Reply

Submitted by Mike Rose (not verified) on December 7, 2017 - 7:53am

Hi Bora,

Thank you for explaining the Python use case. I want to go through your examples but I am unable to download the Tableau Workbooks, I keep getting a server error.

Thanks, Mike.
Reply

Submitted by Juan Ibarra on December 21, 2017 - 3:13am

Hi Bora,

Thank you for explaining the Python use case. I want to go through your examples but I am unable to download the Tableau Workbooks, there is a server error.

Thanks, Juan.
Reply

Submitted by Mohamed Farhat on December 28, 2017 - 3:47am

Very informative Bora, thank you!!
it would be nice if you can share with us the example workbooks. links are not working.
Regards
Reply

Submitted by Jimmy (not verified) on January 15, 2018 - 8:23am

That's a great tool to identify where the Hot Spots are, but what about the magnitude of the hot spot? With that question can you run say ArcPy Library within Tableau or is there another Library that returns more that just a string?
Reply

Submitted by Jose Contreras on February 1, 2018 - 9:58am

Hi Bora,

Am I correct in assuming that you cannot the actual script from Tableau, rather you have to write out the python code of whatever you'd like to do within the calculation itself?

Thanks,

Jose
Reply

Add new comment

Your name Kristy M.

Comment *

□ Notify me when new comments are posted

SAVE

Subscribe to our blog

Business E-mail *

Kristy.McGee@tn.gov

SUBSCRIBE

WE RESPECT YOUR PRIVACY

English

System Status Blog Academic Careers Contact Us

LEGAL PRIVACY UNINSTALL