

Advanced Machine Learning: Continuous Learning Lab

This lab is based on <https://medium.com/ibm-data-science-experience/continuous-learning-on-watson-data-platform-cc39f3fd5042> by Adam Massachi

Introduction

We hear from many clients that one of the hardest parts of machine learning is closing the feedback loop. This means that models need to be monitored and updated frequently to incorporate the latest data.

Watson Machine Learning and Watson Studio allow data scientists and analysts to quickly build and prototype models, to monitor deployments, and to learn over time as more data become available. Performance Monitoring and Continuous Learning enable machine learning models to retrain on new data supplied by the user or another data source.

Then, all of your applications and analysis tools which depend on the model are automatically updated as Watson Studio handles selecting and deploying the best model.

Lab description

In this hands-on lab, we will solve a problem for the City of Chicago using the Model Builder to model building violations. We'll predict which buildings are most likely to fail buildings inspections. Then, we can intelligently rank the buildings by their likelihood to fail an inspection, saving time and resources for the City and for our inspectors.

We'll start building a model on publicly available data from 2017, starting in September before we introduce October, November, and December data to simulate learning over time.

You'll need Watson Studio, Watson Machine Learning, and IBM Db2 Warehouse on Cloud connection which will be provided by the instructor.

Train a model on building inspection dataset

1. Add `buildings_data_17.csv` as a Data Asset in your project.

Have a look at what columns are defined using the preview:

My Projects / WatStud_Workshop / buildings_data_17.csv

Preview

Profile

Lineage

Schema: 8 Columns

Preview: 1000 rows | Last refresh: 59 seconds ago | Refresh

INSPECTION_STATUS <small>Type: String</small>	VIOLATION_CODE <small>Type: String</small>	VIOLATION_STATUS <small>Type: String</small>	INSPECTION_CATEGORY <small>Type: String</small>	PROPERTY_GROUP <small>Type: String</small>	LONGITUDE <small>Type: String</small>	LATITUDE <small>Type: String</small>	DEPARTMENT_BUREAU <small>Type: String</small>
FAILED	EV1111	OPEN	PERMIT	large	-87.72750391	41.91299036	ELEVATOR
FAILED	CN067024	COMPLIED	COMPLAINT	large	-87.7224774	41.90204779	CONSERVATION
FAILED	CN198019	COMPLIED	COMPLAINT	large	-87.70537255	41.85088357	CONSERVATION
FAILED	CN134016	COMPLIED	COMPLAINT	large	-87.70537255	41.85088357	CONSERVATION

We will want to predict the value of the `INSPECTION_STATUS` class based on the other attributes.

2. Create a new Watson Machine Learning Model `BuildingViolationsChicago`:

New model

Define model details

Name

BuildingViolationsChicago

Description

Model description

Machine Learning Service

pm-20-sk

Select model type

☒ Model builder

☐ From file

☐ From sample

Spark Service or Environment

Only Spark environments supporting Scala kernels can be used for model builder creation.

spark-yp

Automatic

Prepare my data and create a model automatically

Manual

Let me prepare my data and select which models to train

Need something more flexible? Create a notebook or design a Modeler flow

, using Manual mode.

3. Select the `buildings_data_17.csv` as Input file:

Close
Next

4. Train a model:

- Select `INSPECTION_STATUS` as the value to predict:

Column value to predict (Label Col)

INSPECTION_STATUS (String)

INSPECTION_STATUS (String)

- Keep the default for the features column (all):

Feature columns


All (default)

- Chose Binary Classification technique


- Add Logistic Regression as Estimator:

Select estimator(s)


What type of estimator are you looking for?


Logistic Regression

Analyzes a data set in which there are one or more independent variables that determine one of two outcomes. Only binary l...


Decision Tree Classifier

Maps observations about an item (represented in the branches) to conclusions about the item's target value (represented in...


Random Forest Classifier


Constructs multiple decision trees to produce the label that is a mode of each decision tree. It supports both binary and ...

Cancel
Add
Next

- Run the training by clicking on `[Next]` button

- Once trained, Save the Logistic Regression model

Select model

	ESTIMATOR TYPE	STATUS	PERFORMANCE	AREA UNDER ROC CURVE	AREA UNDER PR CURVE	LAST EVALUATION
	LogisticRegression	Trained & Evaluated	Good	0.84088	0.78352	25 Sep 2018, 11:11 PM

Close
Previous
Save

5. Configure the retraining feedback database

- Switch to the Evaluation tab and select the Configure Performance Monitoring

BuildingViolationsChicago

Overview Evaluation Deployments Lineage

Last Evaluation Result

Version	dea5c9af-5c0d-47a9-876a-3726a7ffed05
Phase	setup
AreaUnderPR	0.784
AreaUnderROC	0.841

Performance Monitoring

Configure performance monitoring to evaluate and retrain the model periodically to ensure the model performance is associated with your project to be used as your feedback data connection.

[Configure Performance Monitoring](#)

- ii. Select `Area under PR` as the metric with a value of 0.8

Metric details (type / optional threshold)

`areaUnderPR`  `0.8` 

- iii. Configure the feedback database, click on Create New Connection:

Feedback data connection (IBM Db2 Warehouse on Cloud - [Create new connection](#) )

Select feedback data reference

 **Db2 Warehouse**

IBM Db2 warehouse database on Cloud

- iv. Select `DB2 Warehouse` as the database type:

- v. Enter the DB name `BLUDB`, and the userid, password and hostname provided by the instructor, a name such as `DB2` then click to `Create` the connection:

New connection (DB2 - Db2 Warehouse)

Connection overview

Name

DB2

Description

IBM Db2 warehouse database on Cloud

Connection details

Database *

BLUDB

Password *

.....

Secure Gateway

☐ Use a secure gateway

Hostname or IP Address *

dashdb-txn-sbox-yp-dal09-03.services.dz

Username *

rkq11138

Enter information for the selected data source

Cancel

Create

Select feedback data reference

- vi. Back to the Model Evaluation, click
- vii. Select any existing schema, for example `ST_INFORMTN_SCHEMA`

Select feedback data reference

WatStud_Workshop	DB2	ST_INFORMTN_SCHEMA
Connections (1)	Schemas (6)	
DB2 >	<div>ERRORSCHEMA ></div> <div>IBM_RTMON_DATA ></div> <div>RKQ11138 ></div> <div>SQL28627 ></div> <div>SQL32811 ></div> <div>ST_INFORMTN_SCHEMA ></div>	No drilldowns currently exist.

Cancel

Select

- viii. Enter a table name, **which must be unique among the participants since we're all sharing the same database**, use e.g. `New2017Table_XYZ` where `xyz` are your initials:

Feedback data connection (IBM Db2 Warehouse on Cloud - [Create new connection](#))

dashdb: BLUDB [Change feedback data reference](#)

New2017Table

- ix. Keep the defaults for Auto retrain and deploy, and finally click Save:

Configure performance monitoring

learning.

spark-yp

Prediction type

binary

Metric details (type / optional threshold)

areaUnderPR

0.8

Feedback data connection (IBM Db2 Warehouse on Cloud - [Create new connection](#))

dashdb: BLUDB [Change feedback data reference](#)

New2017Table

Record count required for re-evaluation

1000

Auto retrain

when model performance is below threshold

Auto deploy

when model performance is better than previous version

Cancel

Save

6. Adding new feedback data:

- From the model Evaluation tab, click the (+) Add feedback data button

Overview

Evaluation

Deployments

Lineage

Evaluation Events

(+) [Add feedback data](#)

(+) [New evaluation](#)

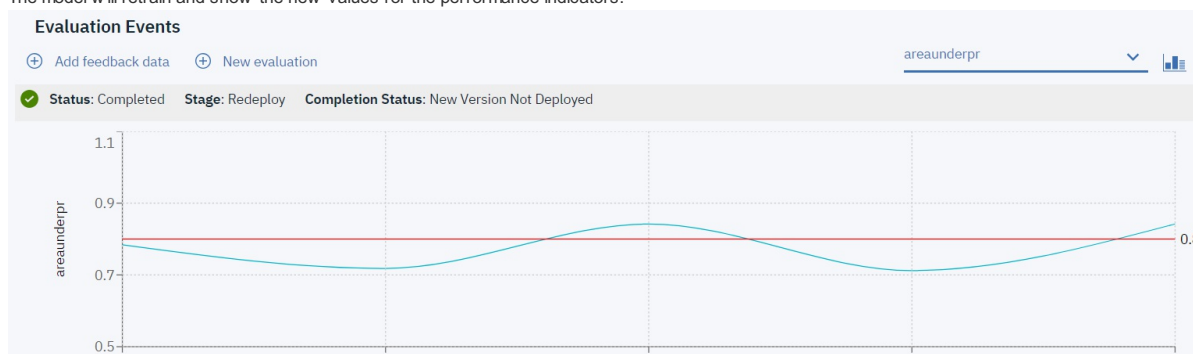
- Select one of the files from the months.zip file that you will have unzipped, starting in October, e.g. buildings_violations_October.csv
- This will prompt for a new evaluation:

New evaluation?

Feedback data was successfully uploaded. If you want to re-evaluate your model immediately, click the "New evaluation" button below. Otherwise, click "Cancel". You can initiate a new evaluation at any time by clicking the "New evaluation" link above the evaluation results table.



iv. The model will retrain and show the new values for the performance indicators:



Conclusion

This short hands-on lab's purpose is to give a taste for the tasks involved in retraining and continuous learning of a model. We have seen how injecting fresh up-to-date data can be used to re-evaluate a model against its quality indicators.