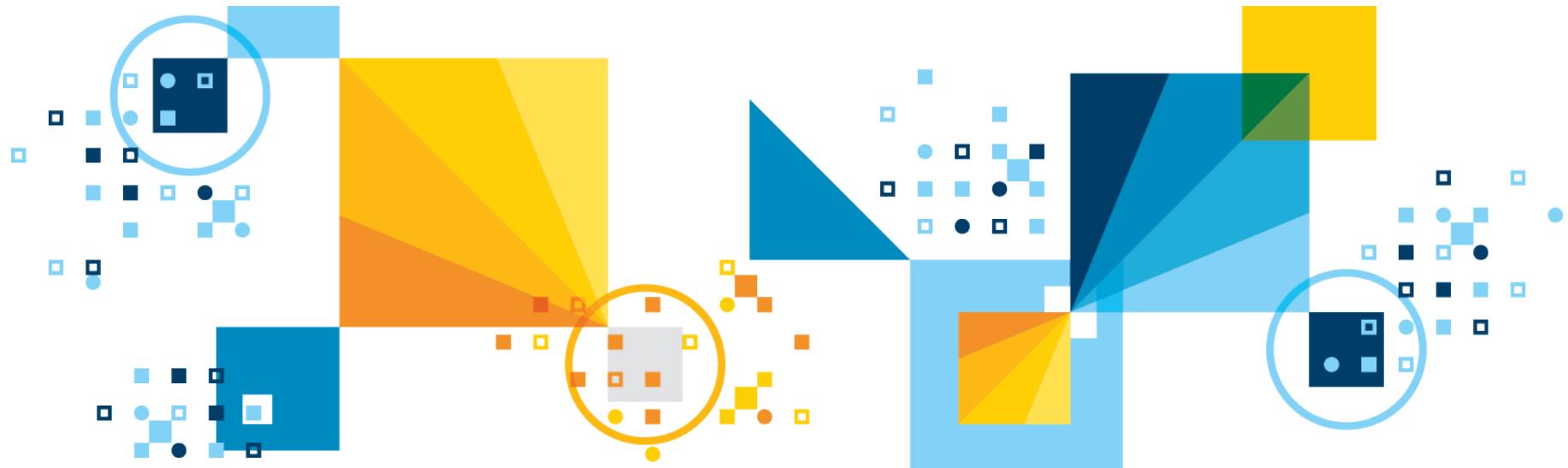
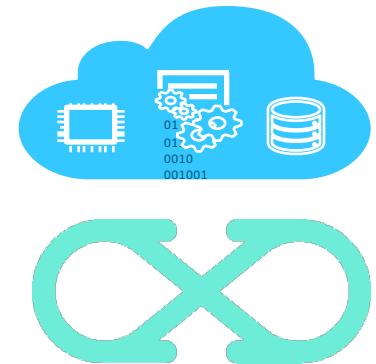


IBM IoT Analytics in the Cloud

Steve Geringer, Data Science Solutions Architect

Steve.Geringer@IBM.com

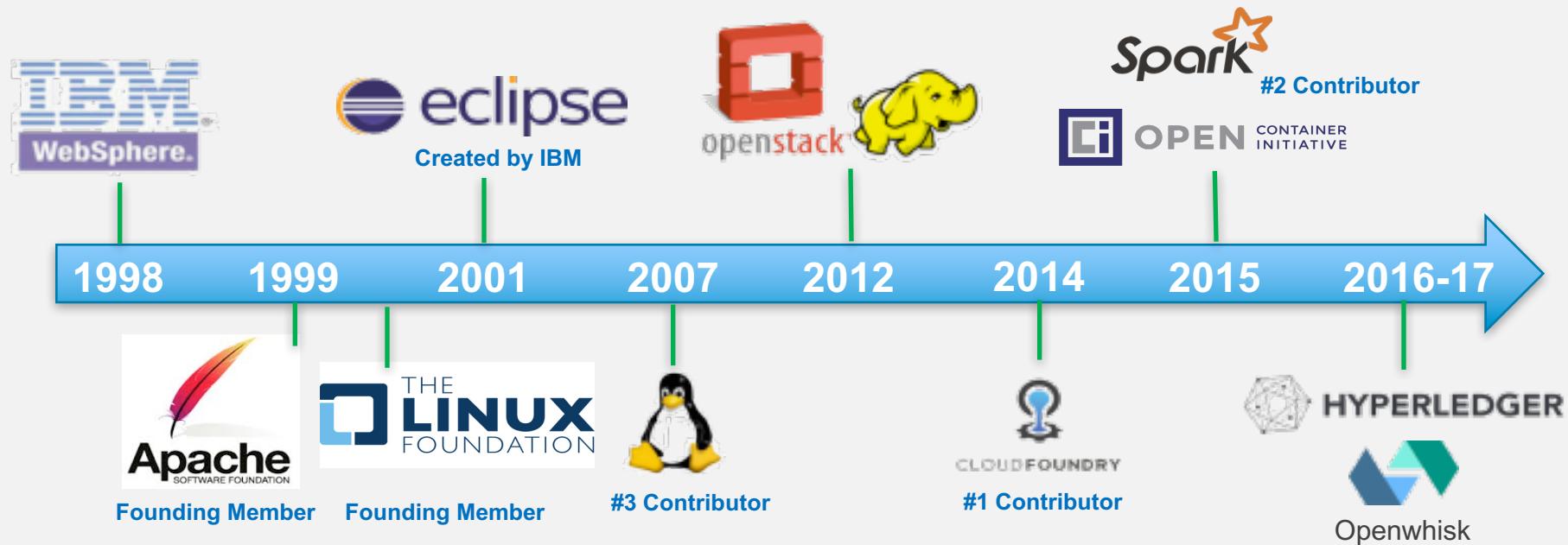


Background

■ About Me:

- Data Science Solutions Architect
- IBM Analytics team covering
 - Healthcare, Life Sciences and State/Local Government.
- Achieved the "Kaggle Competitions Expert" Designation
 - Earned two Silver Medals in Kaggle Machine Learning Competitions.
- Steve.Geringer@ibm.com

IBM Has a Rich History with Open Source



IBM and Open Source:

Today companies have lots of options

Open Source Only



**Significant
Assembly**

Open Plus



**Enterprise Ready
Innovation**

Proprietary



**Vendor
Lock-in**



BlueMix is a Platform as a Service (PaaS)

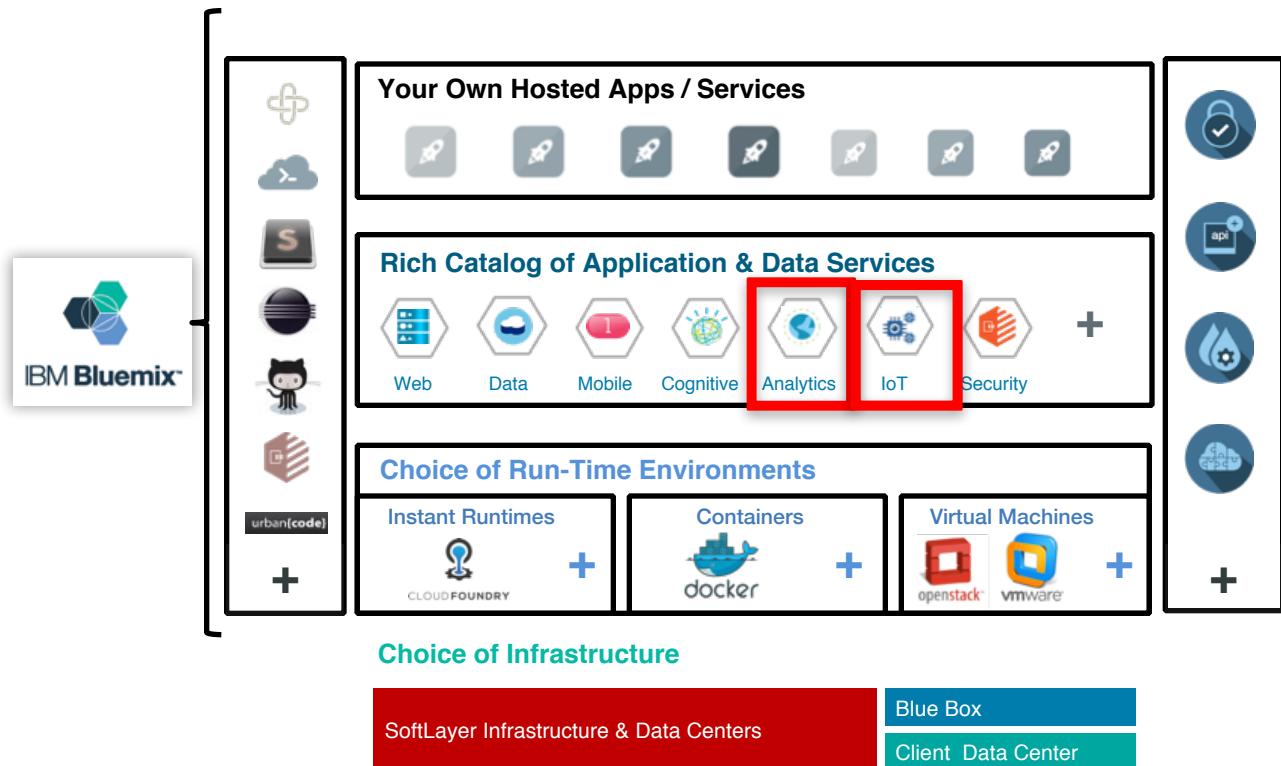
Traditional IT	IaaS	PaaS	SaaS
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S management	O/S management	O/S management	O/S management
O/S installation	O/S installation	O/S installation	O/S installation
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking
Datadatacenter	Datadatacenter	Datadatacenter	Datadatacenter

Managed by you
 Managed by IBM

IBM Softlayer
Amazon EC2
Azure Infrastructure

IBM BlueMix
Google App Engine
Amazon Elastic Beanstalk

Guiding Principles Define the IBM Cloud Platform



Comprehensive

- Broadest selection of data and analytics services
- Seamless integrations
- Open-source leadership

Trusted

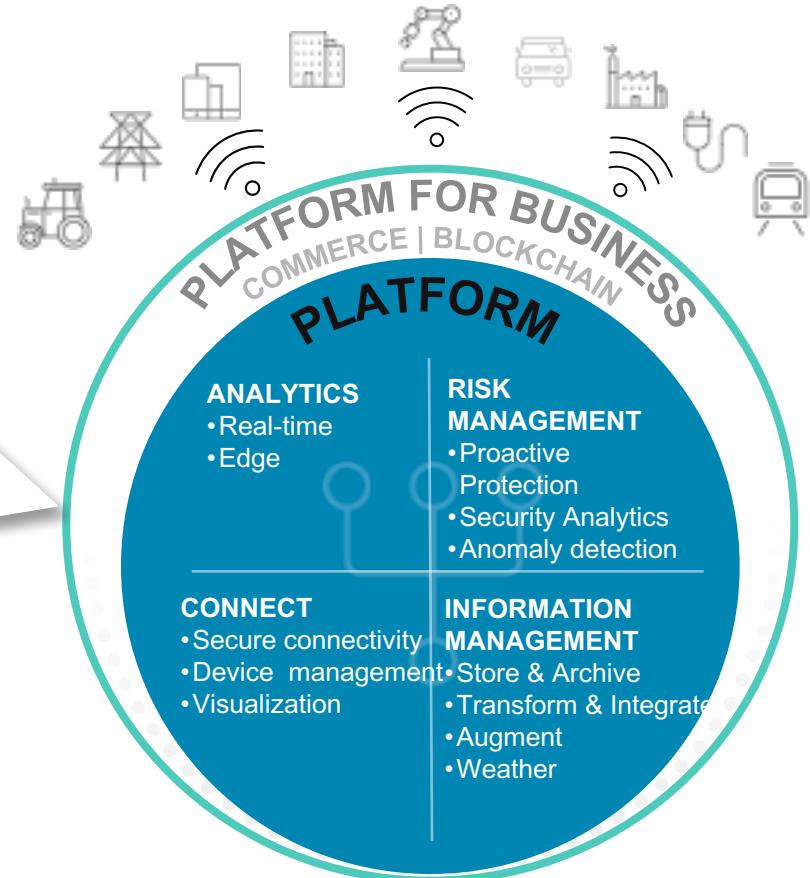
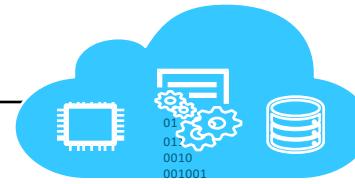
- Fully managed environment 24 x 7
- Secure infrastructure
- No installation, configuration, maintenance required

Flexible

- Cloud, on-premises & hybrid support
- No vendor lock-in
- Subscription pricing

IBM Watson IoT Platform

Make sense of data to optimize operations, manage assets, rethink products and services, and transform customer experience.



IBM Watson IoT Platform

Devices and Sensors



Cloud Platform

Watson IoT Platform

ANALYTICS

- Real-time
- Edge

CONNECT

- Secure connectivity
- Device management
- Visualization

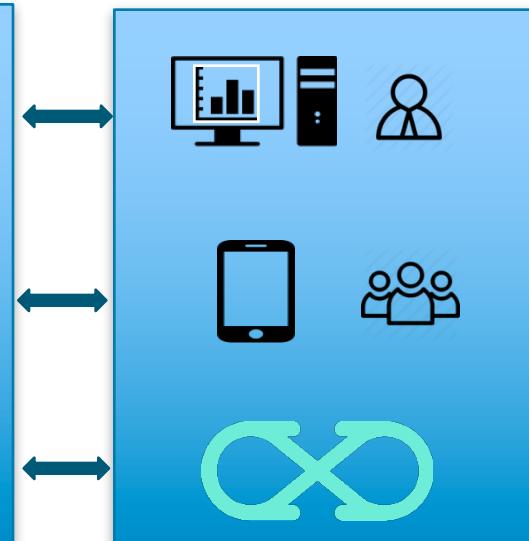
RISK MANAGEMENT

- Proactive Protection
- Security Analytics
- Anomaly detection

INFORMATION MANAGEMENT

- Store & Archive
- Transform & Integrate
- Augment
- Weather

Analytics and Applications



Other Data Sources



Weather



Map

MQTT – The Protocol for IoT Messaging

OASIS



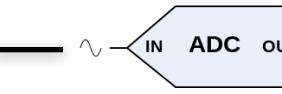
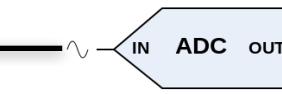
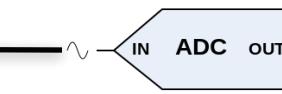
Analogue electrical power meter



Analogue temperature gauge



Analogue vibration sensor (accelerometer)



IBM Watson IoT Platform



MQTT is a publish and subscribe, extremely simple and lightweight messaging protocol, designed for low-bandwidth, high-latency or unreliable networks.



Platform

Everything you need to innovate with IoT



Connect

Attach: MQTT, HTTPS
Visualize
Collect & Organize
Device Management
Users

101
010
101

Information Management

Parsing Unstructured Data
Storage & Archiving
Metadata Management
Reporting
Streaming Information



Risk Management

Proactive Protection



Analytics

Predictive
Cognitive
Realtime Performance
Contextual

Product Family

Watson IoT Platform
Continuous Engineering

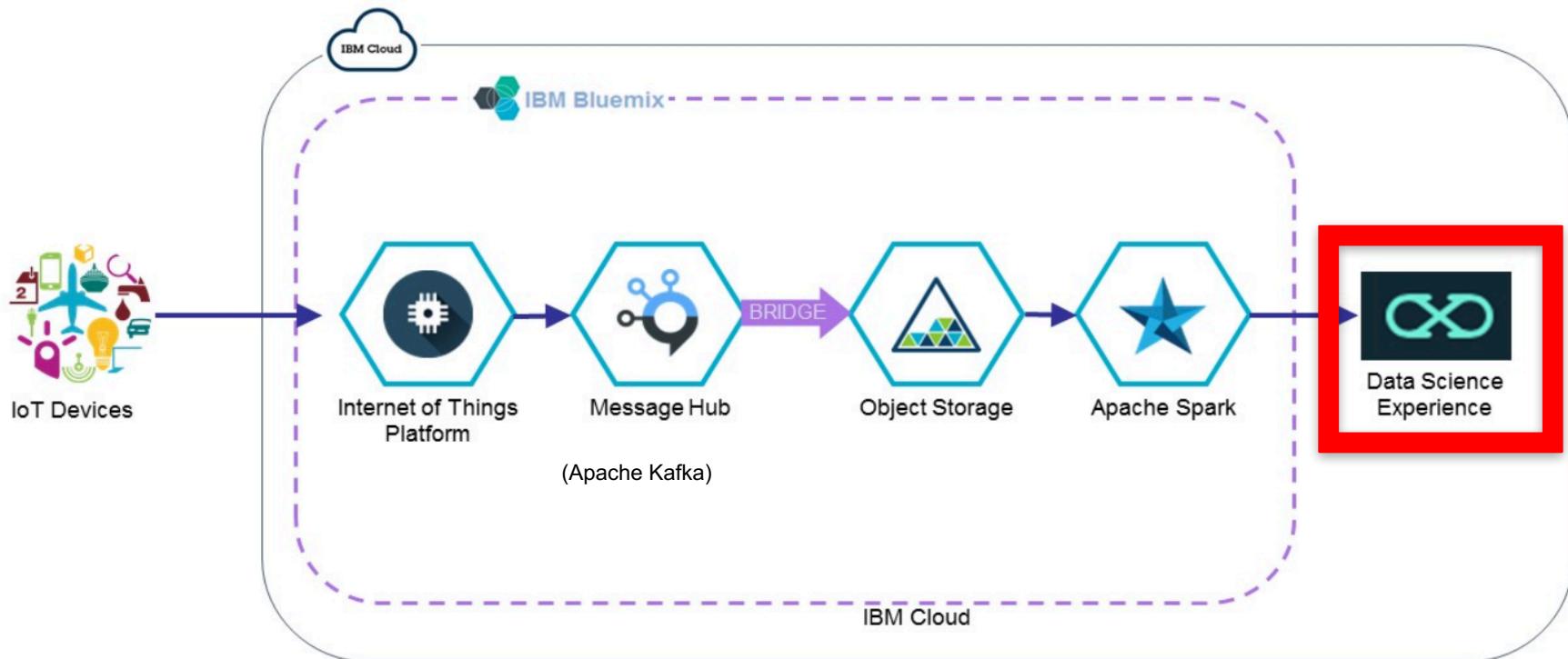


CLOUD FOUNDRY

Bluemix Open Standards Based Services

Full Development Lifecycle
DevOps Services
IBM Security

End-to-End IoT Data Pipeline

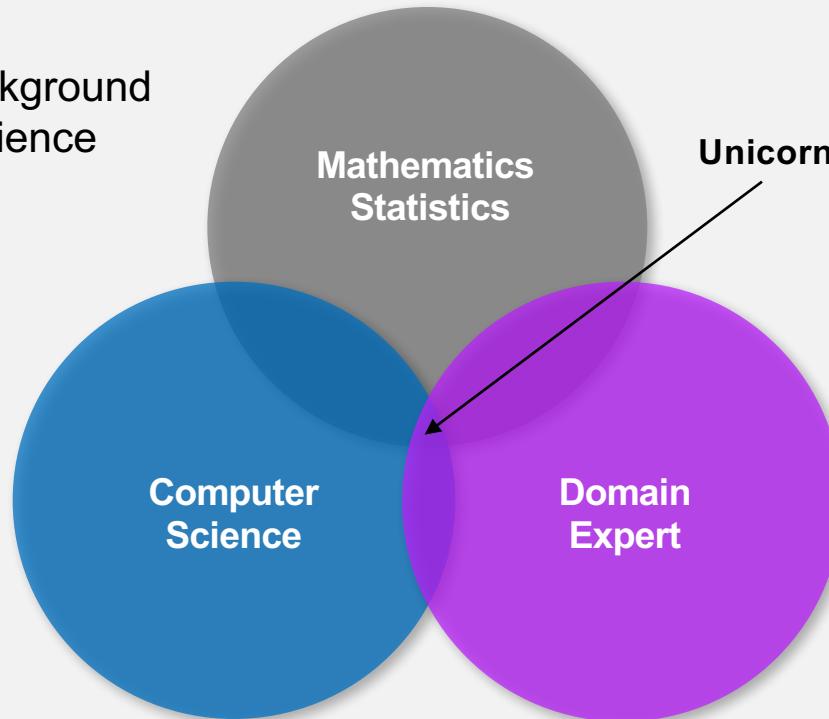


Data Science Professional

Data Scientists combine skills across areas of Expertise

Mathematical Background
Computational Science

Scripting, SQL
Python, R Scala
Data Pipelines
Big Data/
Apache Spark



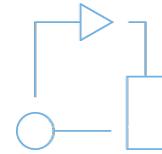
Business/Industry
Expertise
Domain Knowledge
Supply Chain
CRM
Financials
Networking

Data Science Is a Team Sport



Introducing IBM Watson Data Platform

An Integrated Data Science Environment in the Cloud



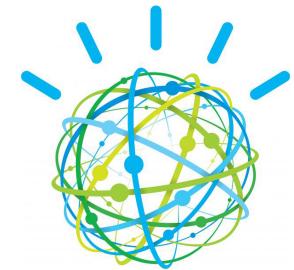
Platform.



Method.



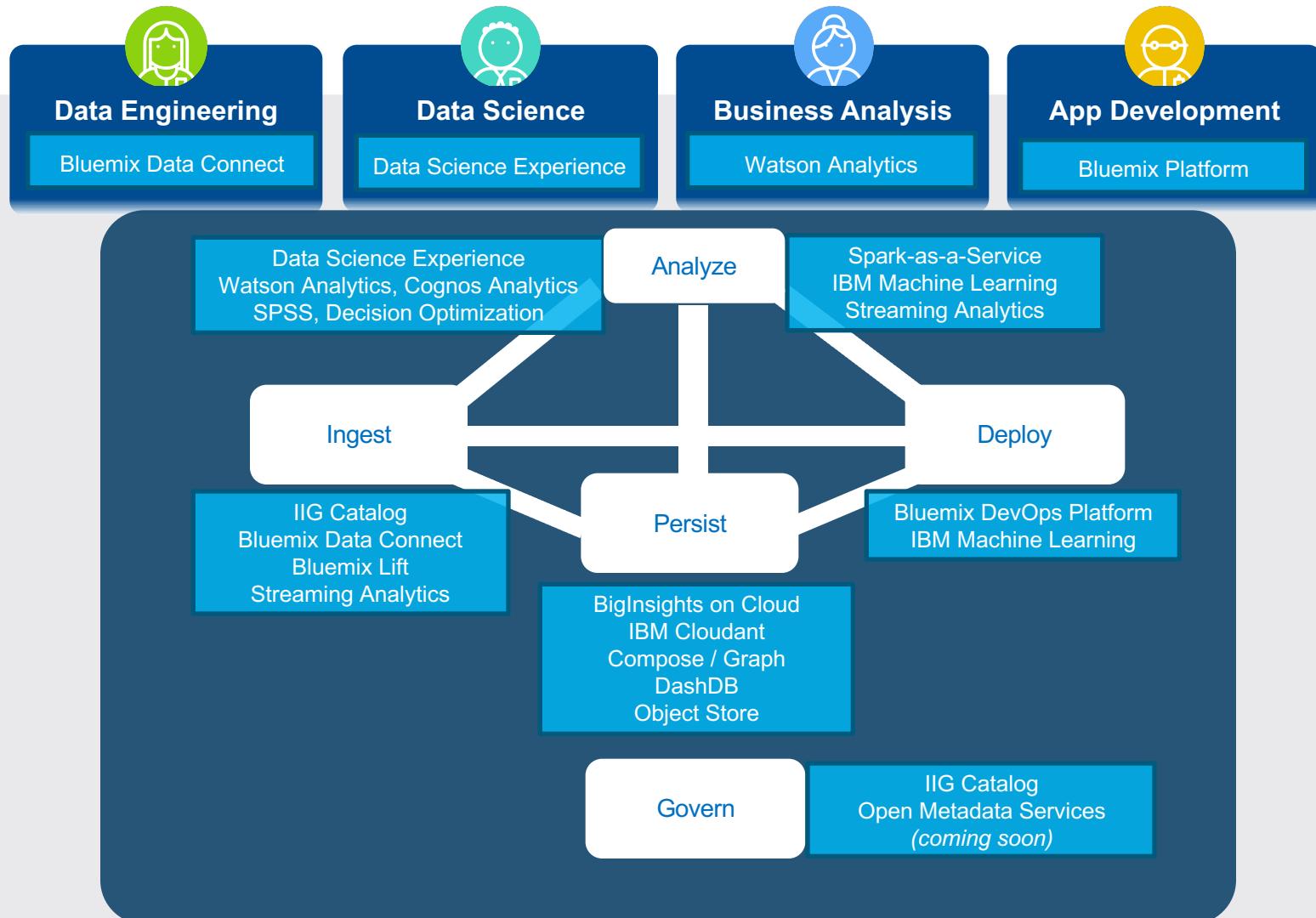
Ecosystem.



<http://ibm.co/makedatasimple>

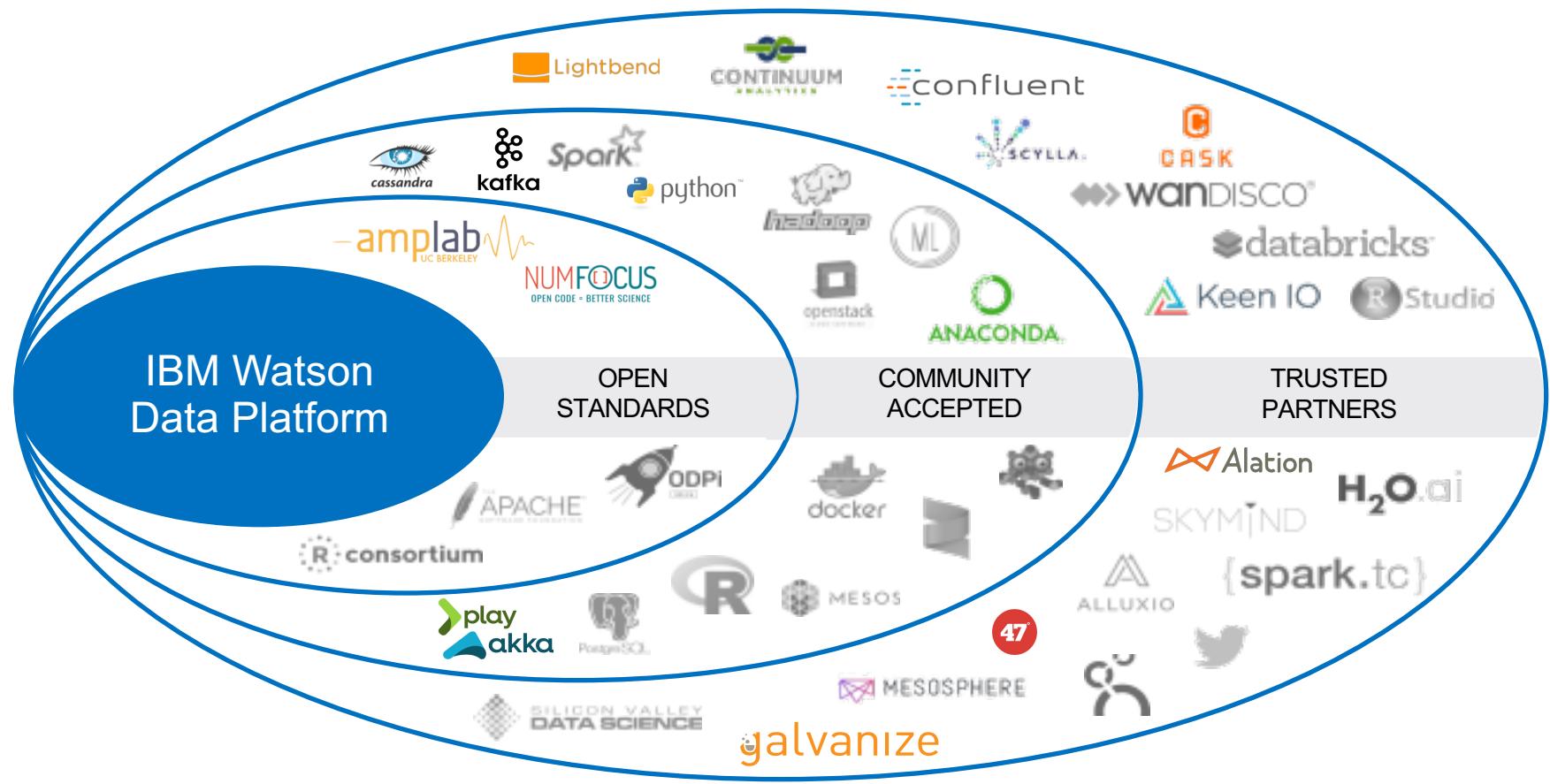
IBM Watson Data Platform

A closer look at what makes up the platform



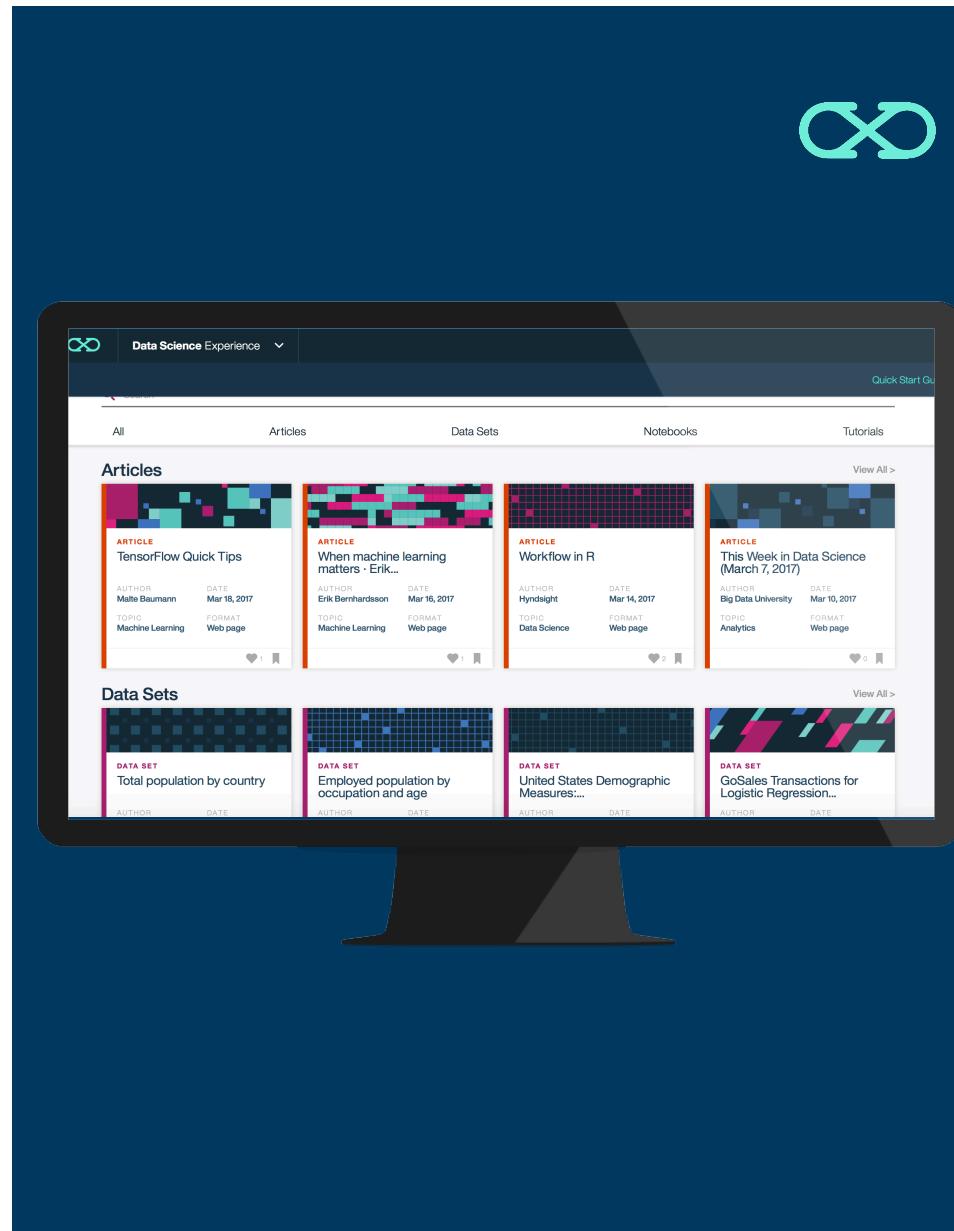
IBM Watson Data Platform Partner Ecosystem

The Open Community To Innovate Faster With Data



Data Science Experience

IBM Data Science Experience is an environment that brings together everything that a Data Scientist needs. It includes the most popular **Open Source tools** such as Code in Scala/Python/R/SQL, Jupyter Notebooks, RStudio IDE and Shiny apps, Apache Spark and IBM unique value-add functionalities with **community** and **social features**, integrated as a first class citizen to make Data Scientists more successful.



Data Science Experience: Centralizes Open Source

The screenshot shows the Data Science Experience interface. At the top, there are logos for R, Scala, NumPy, Python, Jupyter, and Spark. The main area displays a Jupyter notebook titled "Deep Learning for Image Classification". The notebook contains text about building a classification model for handwritten digits, adapted from a Python tutorial. Below the text are three examples of digit images with their corresponding numerical labels: 7 → 7, 5 → 5; 8 → 8, 3 → 3; and 2 → 2, 4 → 4. At the bottom of the notebook, there is a code cell for installing the nolearn library:

```
In [1]: #pip install --user nolearn
```

```
In [ ]: import warnings
warnings.filterwarnings("ignore")
from sklearn.cross_validation import train_test_split
```

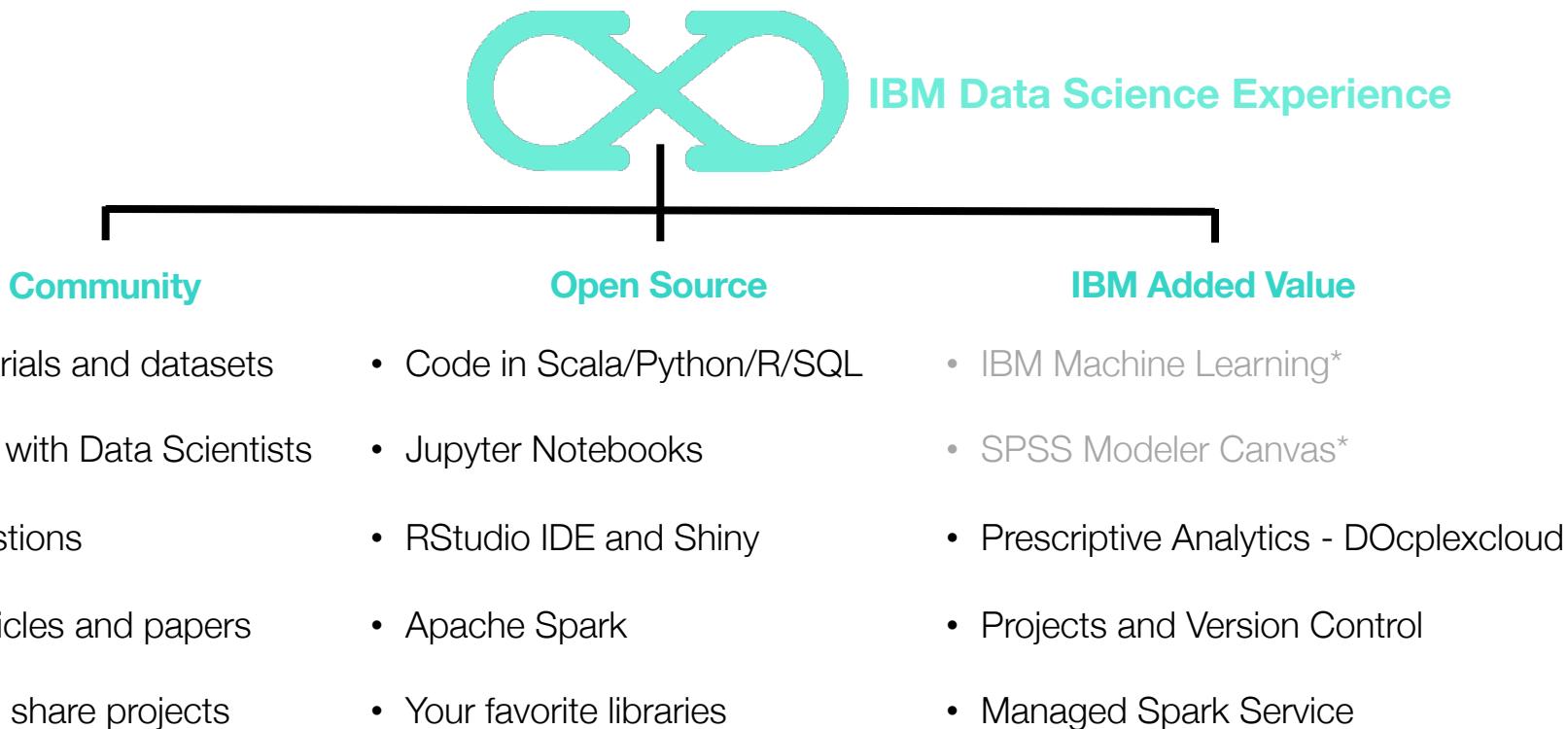


pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Core Attributes of the Data Science Experience



Powered by IBM **Watson Data Platform**

* Closed beta

IBM Data Science Experience Brings Together Visual and Programmatic Worlds

Collaborate Using Projects

This screenshot shows the main interface of the IBM Data Science Experience. At the top, there's a navigation bar with 'Data Science Experience' and a dropdown for 'My Projects'. Below it, a 'New Sales campaign' project is selected. The main area is divided into several sections: 'Notebooks' (listing 'Retail Sales Analysis v2' and 'Machine Learning using R'), 'Data Assets' (listing 'Great Outdoors Orders for BBBBT RTite' and 'Presence Data (Cloudant NoSQL)'), and 'Bookmarks' (listing 'From Machine Learning to Learning M...' and others). A bottom section shows recent activity.

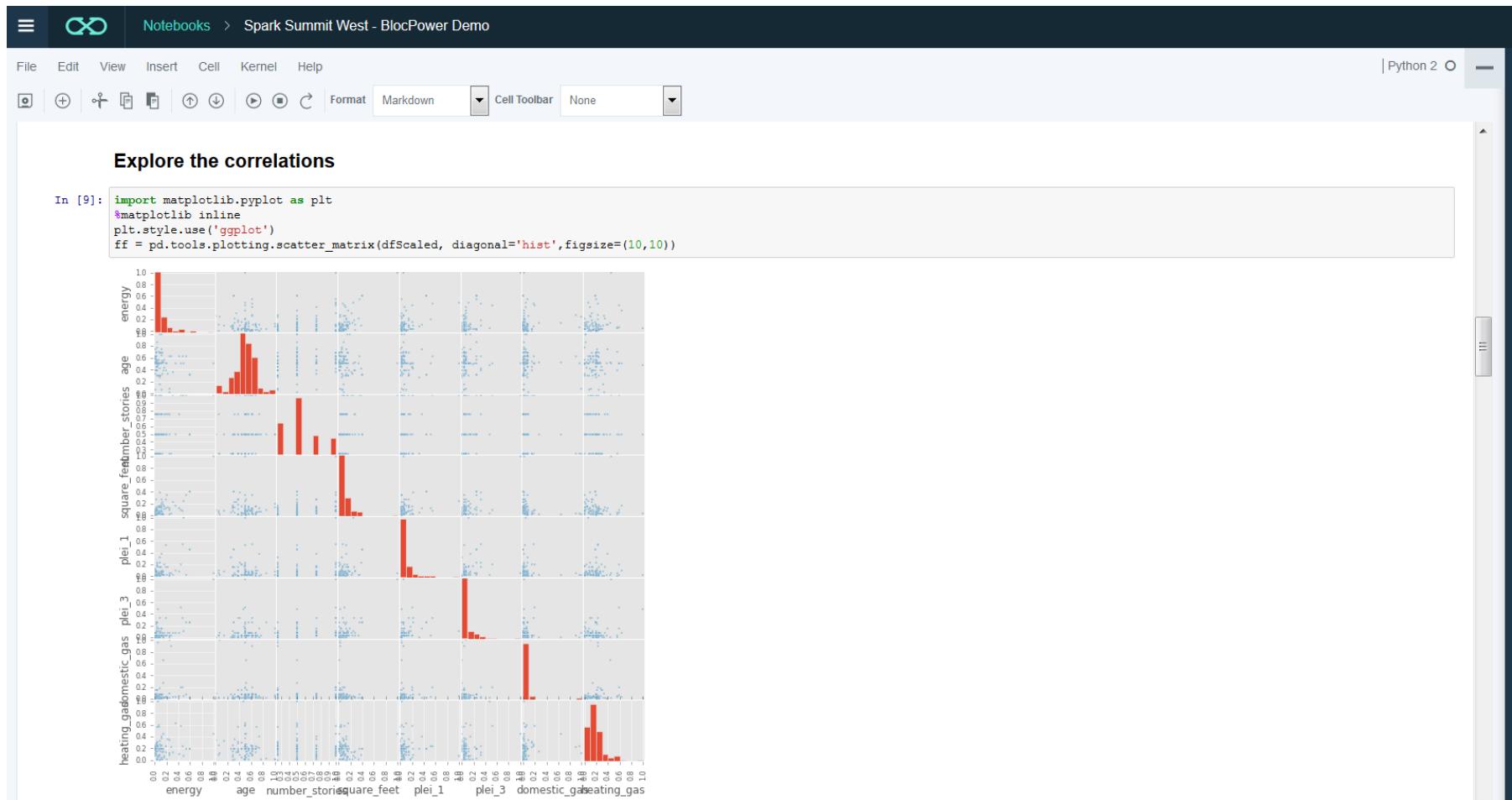
Model Deployment (batch, streaming and real-time)

This screenshot shows the 'Monitor' tab of the IBM Data Science Experience. It displays two sections: 'Scheduled Jobs' and 'Deployments'. The 'Scheduled Jobs' section lists five entries: 'Petran Flight Delay Model', 'Weekly Executive KPIs', 'Business Impact by Weather', 'Evaluate Business Impact', and 'SF Traffic Delay Pattern'. The 'Deployments' section lists three entries: 'Awesome Deployment', 'Weekly Executive KPIs', and 'Business Impact by Weather'. Each entry includes details like pipeline name, endpoint URL, and status.

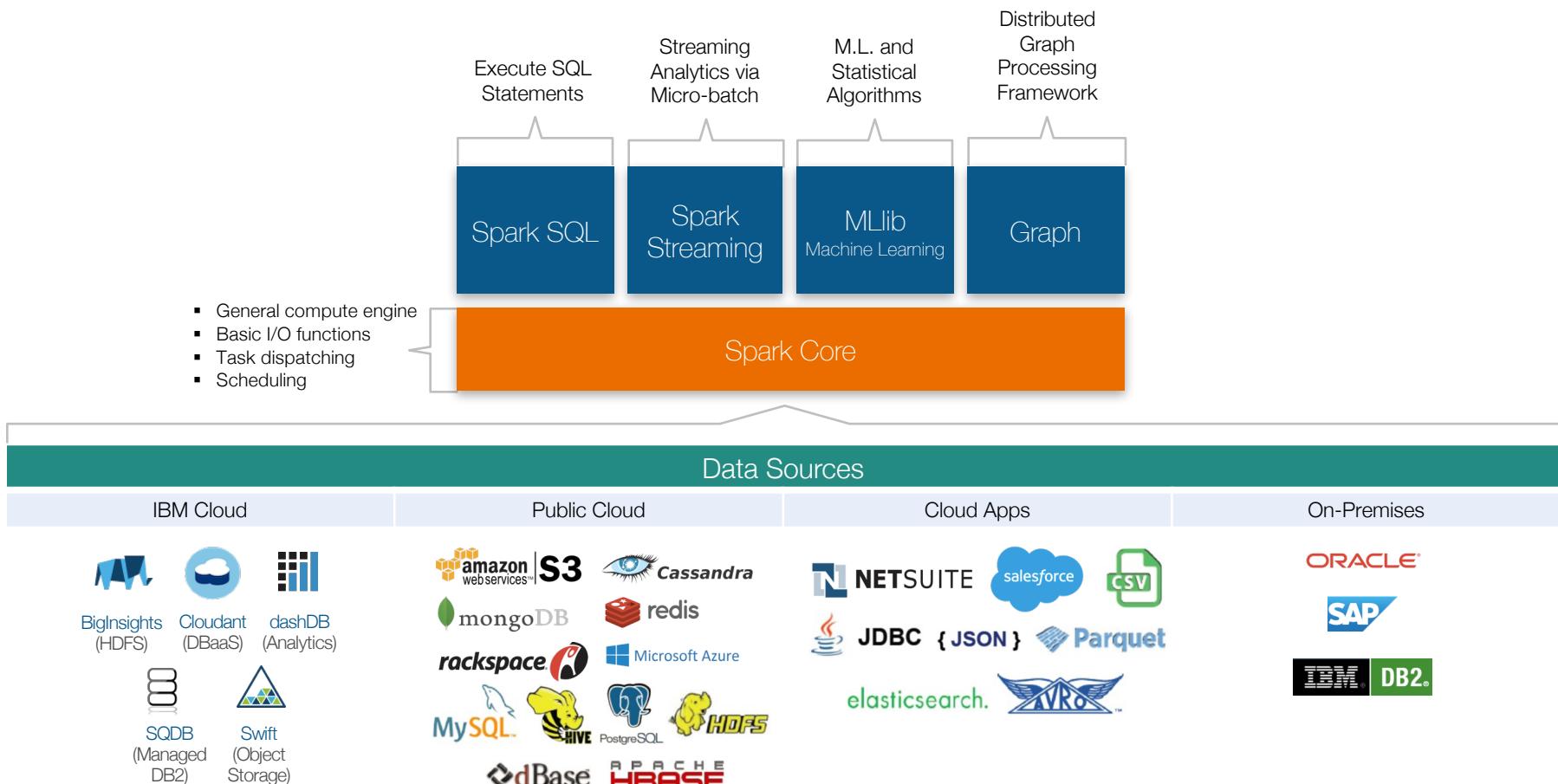
Code Algorithms in Python, R and Scala: Automatic Model Visualization

This screenshot shows a notebook titled 'Brad's Test' in the IBM Data Science Experience. The code cell contains Python code for setting up an SQLContext and reading data from Swift. The right panel shows a 'Predictor Importance' visualization for a model named 'Variation C-2234'. The visualization includes a bar chart for 'TARGET: SALARY' with 'Importance 100' and a table of asset statistics. Other tabs in the visualization include 'Model Visualizations', 'Test of Model Effects', 'Observed by Predicted', 'Title', 'Salary-gender: Female', 'Salbegin-gender: Male', 'Residuals by Predicted', 'Target Histogram', 'Residual Histogram', and 'Parameter Estimates'.

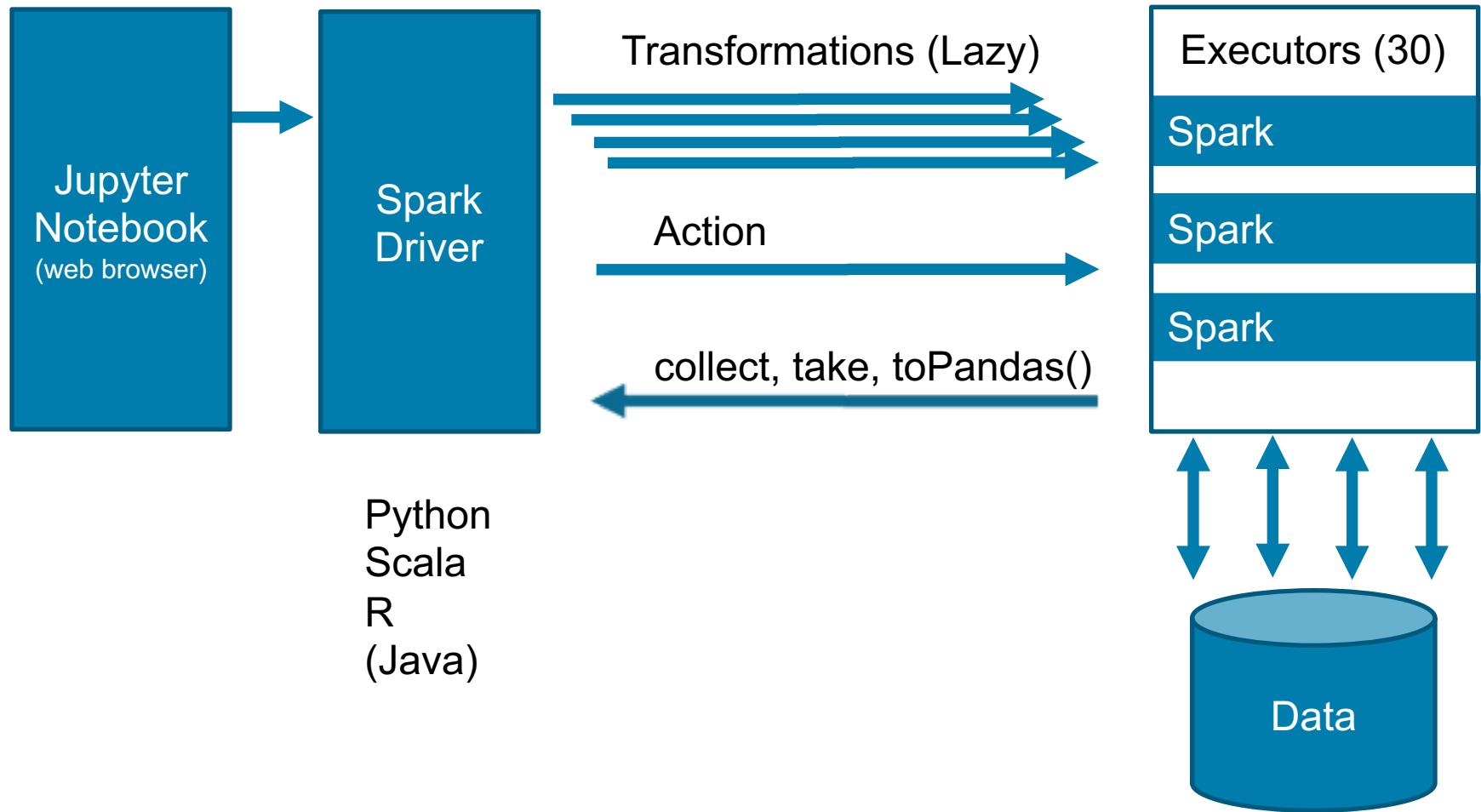
Integrated Jupyter Notebooks for interactive and collaborative development - seamless execution on Spark



IBM's Managed Apache Spark Service in DSX



DSX - Spark Architecture



Collaborate Using Projects

☰ Data Science Experience ▾

My Projects > New Sales campaign

Overview Analytics Assets Data Assets Bookmarks Collaborators Settings

Notebooks [view all \(2\)](#)

NAME	SHARED	STATUS	LANGUAGE
Retail Sales Analysis v2			Python 2.7
Machine Learning using R			R 3.3.0

Data Assets [view all \(23\)](#)

NAME	TYPE
Great Outdoors Orders for BBBT Ritika	Catalog File
Great Outdoors Orders for BBBT Ritika	Catalog File
gchn-daily-by_year-format.rtf	RTF
Presence Data (Cloudant NoSQL)	Connection
Sales Data (dashDB)	Connection

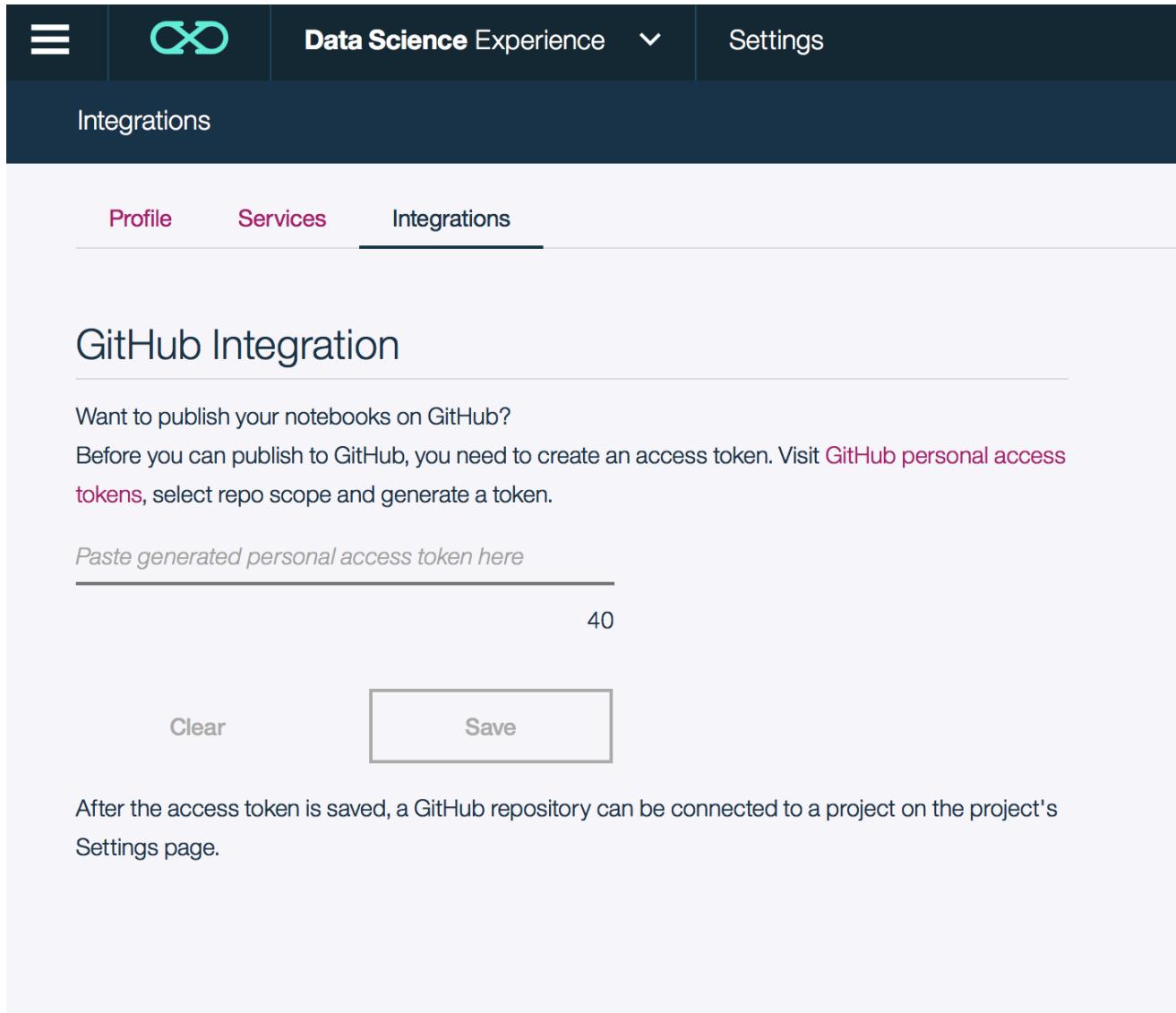
Bookmarks [view all \(3\)](#)

ARTICLE
From Machine Learning to Learning M...
Nov 10, 2016

NOTEBOOK
Use deep learning for image classifica...
Oct 4, 2016

TUTORIAL
Analyze open data sets using pandas ...
Oct 19, 2016

GitHub Integration



The screenshot shows the IBM Data Science Experience interface with the "Integrations" tab selected. The main section is titled "GitHub Integration" and contains instructions for publishing notebooks to GitHub, mentioning the need for a personal access token. A text input field is provided for pasting the token, with a character count indicator (40). Below the input are "Clear" and "Save" buttons.

Data Science Experience

Integrations

Profile Services Integrations

GitHub Integration

Want to publish your notebooks on GitHub?

Before you can publish to GitHub, you need to create an access token. Visit [GitHub personal access tokens](#), select repo scope and generate a token.

Paste generated personal access token here

40

Clear Save

After the access token is saved, a GitHub repository can be connected to a project on the project's Settings page.

Community Cards provide in-context learning for users

<p>ARTICLE How can data scientists collaborate to build...</p> <p>SOURCE IBM DATE Jun 24, 2016</p>	<p>ARTICLE What is machine learning?</p> <p>SOURCE IBM DATE Jun 24, 2016</p>	<p>NOTEBOOK Insights from Twitter data about car makers</p> <p>SOURCE IBM DATE Jun 22, 2016</p>
<p>NOTEBOOK Insights from New York car accident reports</p> <p>SOURCE IBM DATE Jun 16, 2016</p>	<p>DATA SET Country Surface Area (sq. km)</p> <p>SOURCE IBM DATE Jun 16, 2016</p>	<p>NOTEBOOK Improved Flight delay prediction</p> <p>SOURCE IBM DATE Jun 06, 2016</p>
<p>NOTEBOOK Load data from different sources</p> <p>SOURCE IBM DATE Jun 02, 2016</p>	<p>NOTEBOOK Learn basics about notebooks and Apache Spark</p> <p>SOURCE IBM DATE Jun 02, 2016</p>	<p>NOTEBOOK Analyze precipitation data</p> <p>SOURCE IBM DATE Jun 02, 2016</p>

Supported Data Sources for DSX via on-premises and cloud Connections



Cloud Sources	On-Premises Sources	Cloud Targets	On-Premises Targets
Amazon Redshift	Apache Hive	Amazon S3	IBM DB2® LUW
Amazon S3	Cloudera Impala	Bluemix Object Storage	IBM Pure Data for Analytics®
Apache Hive	IBM DB2® LUW	IBM Cloudant™	Teradata
Bluemix Object Storage	IBM Informix®	IBM dashDB	
IBM BigInsights™ on Cloud	IBM Pure Data for Analytics®	IBM BigInsights™ on Cloud	
IBM Cloudant™	Microsoft SQL Server	IBM DB2® on Cloud	
IBM dashDB	MySQL Enterprise Edition	IBM SQL Database	
IBM DB2® on Cloud	Oracle	IBM Watson™ Analytics	
IBM SQL Database	Pivotal Greenplum	PostgreSQL on Compose	
Microsoft Azure	PostgreSQL	SoftLayer Object Storage	
PostgreSQL on Compose	Sybase		
Salesforce	Sybase IQ		
SoftLayer Object Storage	Teradata		

DSX has RStudio built into the experience thanks to our strategic partnership

The screenshot shows the RStudio IDE interface. On the left, the code editor displays R code for data import and analysis. A tooltip is shown over the 'annotate' argument in the 'calendarPlot' function call, explaining its purpose: "This option controls what appears on each day of the calendar. Can be: 'date' - shows day of the month; 'wd' - shows vector-averaged wind direction, or 'ws' - shows vector-averaged wind direction scaled by wind speed." Below the code editor is the RStudio console window, which shows the same R code being run. To the right of the code editor is the RStudio workspace, which lists the dataset 'bloomsbury' with 50804 observations and 18 variables. The main area of the interface features a large, colorful calendar heatmap titled 'O3 in 2006'. The heatmap uses a color scale from light yellow (low values) to dark red (high values) to represent ozone levels across the months of January through December. A vertical color bar on the right side of the heatmap indicates the scale, ranging from 20 to 100.

```

File Edit View Workspace Plots Help
RStudioTest.R* | Run Line(s) Run All
Source on Save
library(openair)
## import some example data
bloomsbury <- importKCL(site = "bl0", year = 2005:2010, met = TRUE)
## have a look at the data
summary(plot(bloomsbury))
## trend in o3 by wd
smoothTrend(bloomsbury, pollutant = "o3", deseason = TRUE, type = "ad")
## polarPlot of nox
polarPlot(bloomsbury, pollutant = "o3", type = "daylight")
## calendar plot
calendarPlot(bloomsbury, pollutant = "o3", )
mydata<
  pollutant<-
  years<
  type<
  annotate<
  statistic<
  cols<
  limits<
  Press F1 for additional help
Data
bloomsbury      50804 obs. of 18 variables
  
```

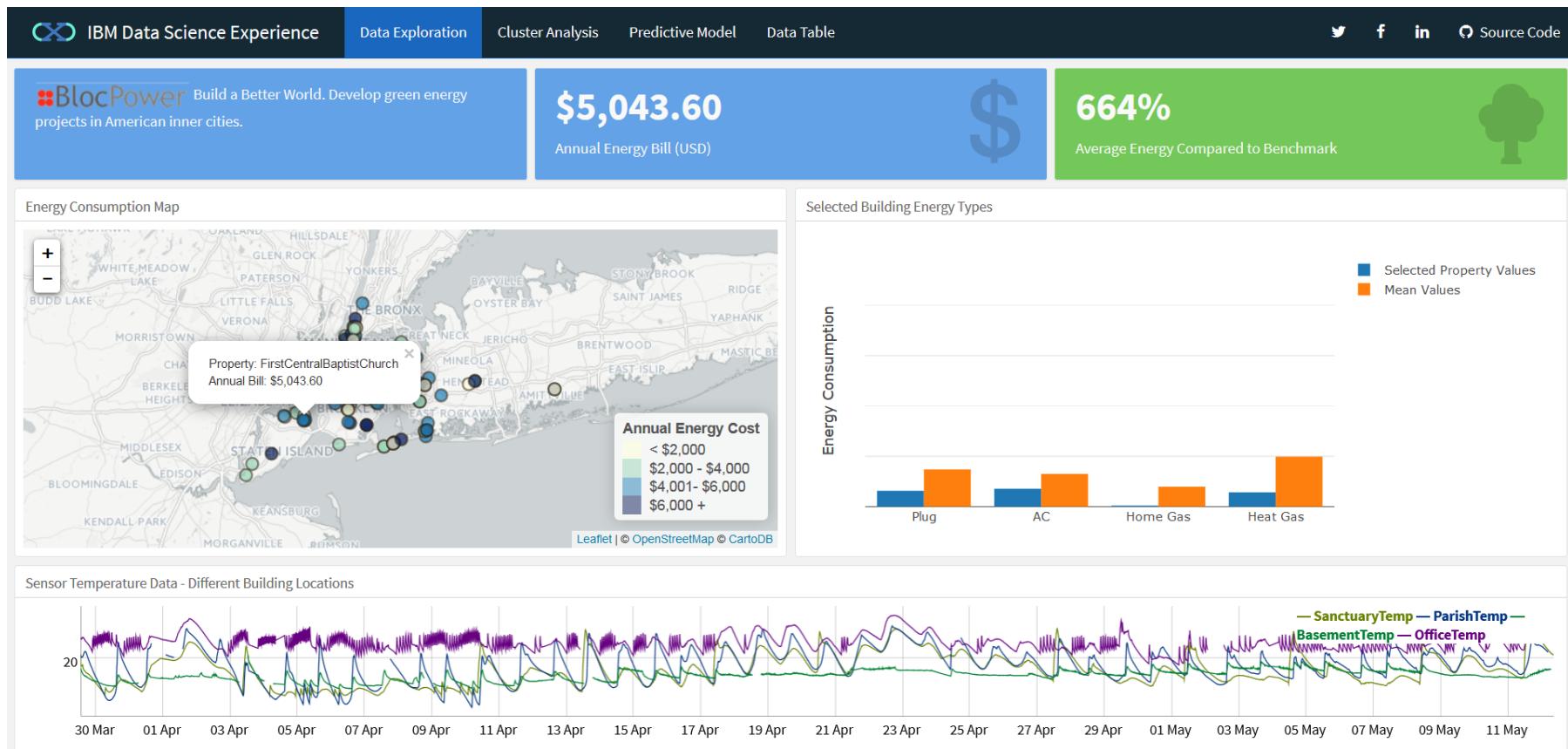
Console ~/

```

>
>
>
>
> library(openair)
## import some example data
bloomsbury <- importKCL(site = "bl0", year = 2005:2010, met = TRUE)
## have a look at the data
summary(plot(bloomsbury))
## trend in o3 by wd
smoothTrend(bloomsbury, pollutant = "o3", deseason = TRUE, type = "ad")
## polarPlot of nox
polarPlot(bloomsbury, pollutant = "o3", type = "daylight")

NOTE - mass units are used
ug/m3 for NOx, NO2, SO2; mg/m3 for CO
PM10_raw is raw data multiplied by 1.3
Warning message:
In importKCL(site = "bl0", year = 2005:2010, met = TRUE) :
  Some of the more recent data may not be ratified.
  date1   date2    nox     no2     o3     so2     co    pm10_raw    pm10    pm25    site
  "POSIXct" "POSIXct" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "factor"
  "code"    "ws"      "wd"      "solar"   "rain"   "temp"   "bp"      "rhum"   "numeric" "character"
  "character" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
> calendarPlot(bloomsbury, pollutant = "o3", year = 2006)
  
```

With RStudio you can create Shiny web applications to make your analysis accessible to the business



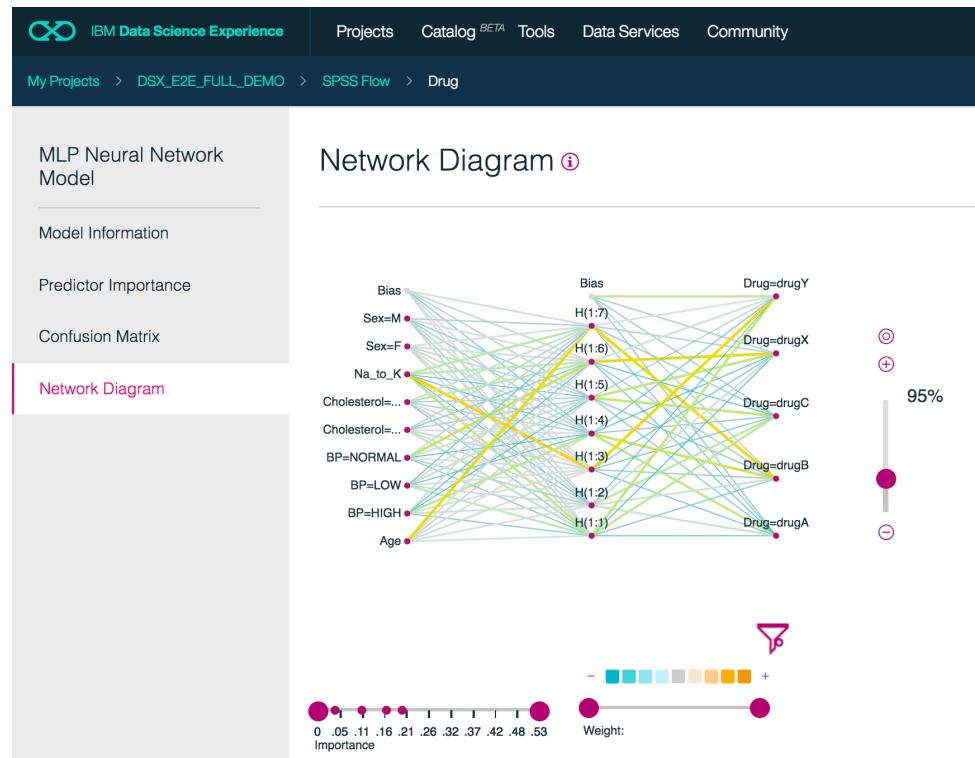
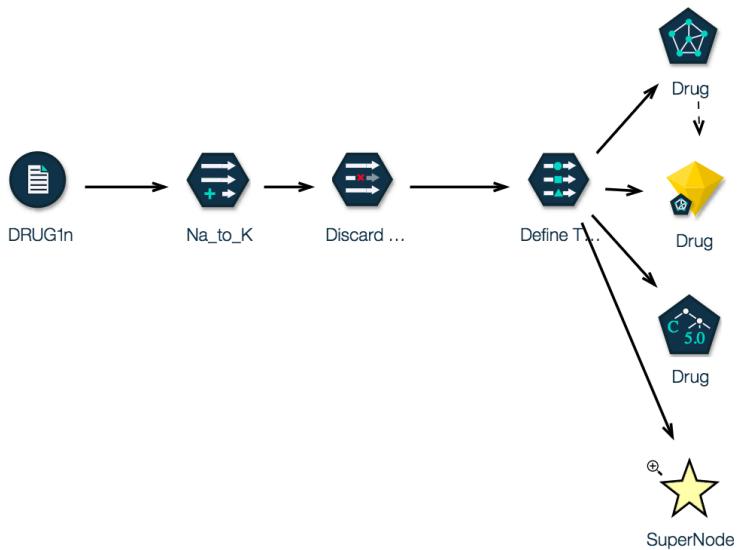
IBM Watson Machine Learning in Data Science Experience



Data Scientist

Flows

*Open
Beta*



Create Advanced Models without coding

Advanced Model Visualization: Easy to understand their performance

Three Paths to Operational Machine Learning on DSX

Model Development

```
Almost ready to train. Let's define two common functions used in the training process
Here we define two functions we'll reuse in our ML iteration.

In [21]:
# run_and_test_classifier does what is sounds like
# Parameters are the instantiated classifier object and the 4 dataframes mentioned just above.
def run_and_test_classifier(classifier, X_train, y_train, X_test, y_test):
    print("Training classifier")
    print(classifier)
    print("Fitting classifier to the training set")
    classifier.fit(X_train, y_train)
    print("Predicting class for X_test")
    y_pred = classifier.predict(X_test)

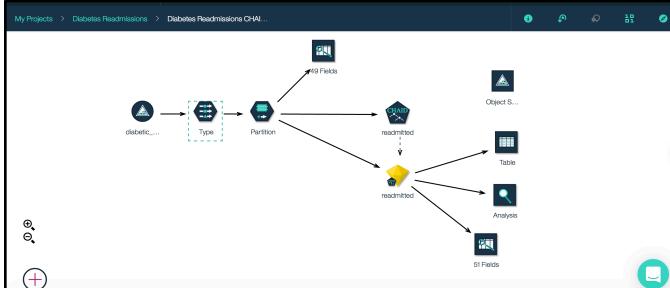
    # fit is the method to perform the training part of the process,
    # so provide both the input matrix (X_train) and the target answers (y_train)
    # classifier.fit(X_train, y_train)

    # the predict method returns predictions as 1 or 0
    # the predict_proba returns predictions as a probabilities matrix, that is, a probabilities vector for each row in the X-test matrix
    # y_prob = classifier.predict_proba(X_test)
    # Extract just the probability for 'Survived'
    y_prob = classifier.predict_proba(X_test)[:,1]
    # drop the first column, but keep only second column, that is, the probability of survived = 1
    # Call the calc_classifier_stats function to print basic statistics on the quality of our classifiers for comparison
    calc_classifier_stats(y_train, y_pred, y_prob)

    # Some classifier algorithms, allow retrieval of "Feature Importances". Which columns are most useful in prediction.
    # Go through the columns printing the name of the column and the importance value for that column
    print("Feature Importances:")
    print(classifier.feature_importances_)
    print("*****")
    print("Feature Importances:")
    print(classifier.feature_importances_)

    for col, imp in zip(X_train.columns, imp):
        print("col: %s, imp: %s" % (col, imp))
```

Custom Coding



Visual Builder

The screenshot shows the 'Hospital Readmissions Automatic Model' configuration page. It lists the following details:

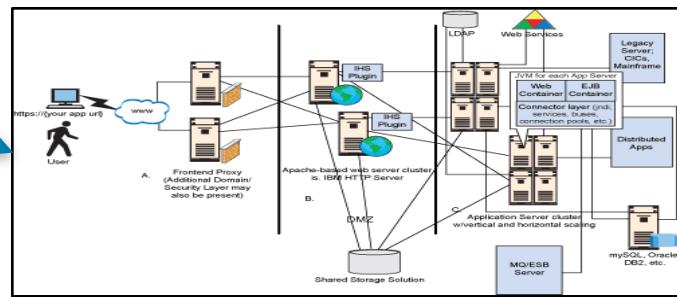
- Machine learning service: WML
- Label column: Readmitt_30days
- Training data schema: View
- Input data schema: View
- Runtime environment: spark-2.0
- Training date: 28 Jul 2017, 2:37 PM

At the bottom, there is a 'Deployments' section.

Automatic Model Builder

Model Deployment

Custom built deployment



Deploy WML Bluemix Service

The screenshot shows the 'ServiceBluemix' catalog page for the 'wml service'. It includes sections for 'Manage', 'Plan', 'Connections', and 'Dashboards'. The 'Dashboards' section provides an overview of the Watson Machine Learning service, mentioning its REST API integration capabilities. The 'SPSS Streams Service' and 'Watson Machine Learning' sections also contain descriptive text and 'Launch Dashboard' buttons.

DSX is *The Power Tool for Today's Data Scientist*

- **Fully Configured and Managed Data Science Environment**
 - Running on Powerful IBM Cloud Servers
 - Integrated Apache Spark – 30 Executors
- **Built on Open Source Tools**
 - Maximum Innovation
 - Choice of Languages, Libraries and Visualization Tools
- **Easily Deploy Machine Learning Models to Production**
- **Part of IBM's Comprehensive Big Data Cloud Story**

BlocPower Demo

Analyze energy consumption in NYC buildings

Modelling Energy Usage in NYC – BlocPower



— Tooraji Arvajeh,
Chief Engineering Officer,
BlocPower

"BlocPower operation is diverse from outreach and targeting, origination of investment-grade clean energy projects to financing projects through our crowdfunding marketplace. Data is the underlying tool of our operation and IBM's Data Science Experience will facilitate a closer integration across it and help our business scale up faster."



Blog Link: <http://ibm.co/29KLbvU>

Use Shiny apps to share your analysis with business users

BlocPower Demo v3 - Blue X

<https://apsx-dev.stage1.ng.bluemix.net/analytics/>

BlocPower Demo v3

```
credentials_1['filename'] = 'CDD-MDU-Features.csv'
dfCH = pd.read_csv(get_file_content(credentials_1['filename']))
```

1. Energy Usage (kWh) Metrics

Clean and Prepare Data

```
In [8]: sc.addPyFile("https://raw.githubusercontent.com/blocpower/blocpower-dash/master/utility.py")
import utilities as ut
```

```
In [9]: # energy usage in (kwh)
energy, age, num_stories, sq_feet, plenty_of_sunlight, heating_gas, domestic_gas = np.asarray(dfHDD[['energy_usage', 'age', 'number_stories', 'sq_feet', 'plenty_of_sunlight', 'heating_gas', 'domestic_gas']])
```

Build Feature Matrix, Fill Missing Values

```
In [10]: matrix = np.transpose(np.matrix([energy, domestic_gas, heating_gas]))
cols = ['energy_usage', 'age', 'number_stories', 'sq_feet', 'plenty_of_sunlight', 'heating_gas', 'domestic_gas']
feat = pd.DataFrame(data=matrix)
# fill missing values with average
feat = feat.fillna(feat.mean())
# scale data
scaler = preprocessing.MaxAbsScaler()
feat = scaler.fit_transform(feat)
dfScaled = pd.DataFrame(feat, columns=cols)
```

Explore Correlations

```
In [11]: plt.style.use('ggplot')
ff = pd.tools.plotting.scatter_matrix(dfScaled)
```

IBM Data Science Experience

Data Exploration

BlocPower Build a Better World.
Develop green energy projects in American inner cities.

23643
Annual Enery Use (kwh)

\$4,255.74
Annual Energy Bill (USD)

Energy Consumption Map

Location Values

Sensor Temperature Data - Different Building Locations

Interactively explore the analysis of your data science team

The screenshot displays the IBM Data Science Experience interface, which integrates a Jupyter notebook, a map, and a scatter plot.

Jupyter Notebook (Left):

- In [19]:**

```
# binary variable to identify inefficient buildings
label_binary = []
for v in labels:
    label_binary.append(0 if (v == 0) else 1)
label_binary = np.asarray(label_binary)
```
- Classification Model Identify Inefficient Buildings:**
- In [20]:**

```
# train classifier
log = linear_model.LogisticRegression(tol = 0.001)
log.fit(featReduced, label_binary)
accuracy = log.score(featReduced, label_binary)
y_pred = log.predict(featReduced)
```
- In [21]:**

```
Model Accuracy: 0.893203883495
```
- In [22]:**

```
def plot_confusion_matrix(cm, title='Confusion matrix'):
    plt.imshow(cm, interpolation='nearest', cmap=plt.cm.Greens)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(2)
    plt.xticks(tick_marks, ['Efficient', 'Inefficient'])
    plt.yticks(tick_marks, ['Efficient', 'Inefficient'])
    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
```
- To [23]:**

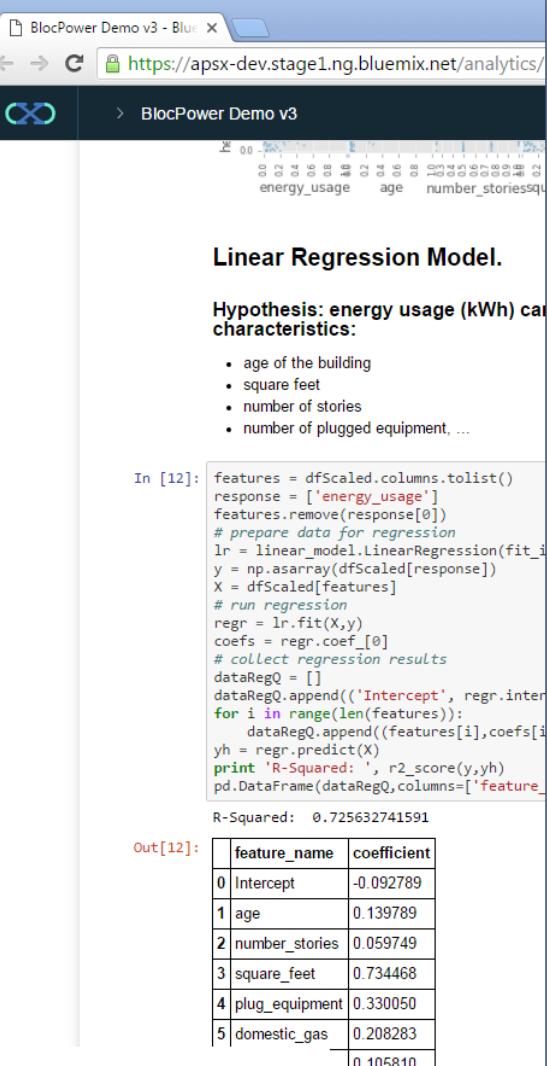
```
from sklearn.metrics import confusion_matrix
```

Data & Analytics Portal (Center):

Energy Consumption Map: A map of New York City showing building clusters. Most buildings are green (Efficient), while some are red (Inefficient). A callout for "CommunityBaptistChurch" shows its cluster label.

Clusters by Heating and Plug Consumption (Right): A scatter plot showing the relationship between heating_gas consumption (x-axis, 0 to 0.8) and plug_load_consumption (y-axis, 0 to 1.0). Red dots represent inefficient buildings, and green dots represent efficient buildings.

Adjust parameters on-the-fly and visualize model predictions



Linear Regression Model.

Hypothesis: energy usage (kWh) can be predicted by characteristics:

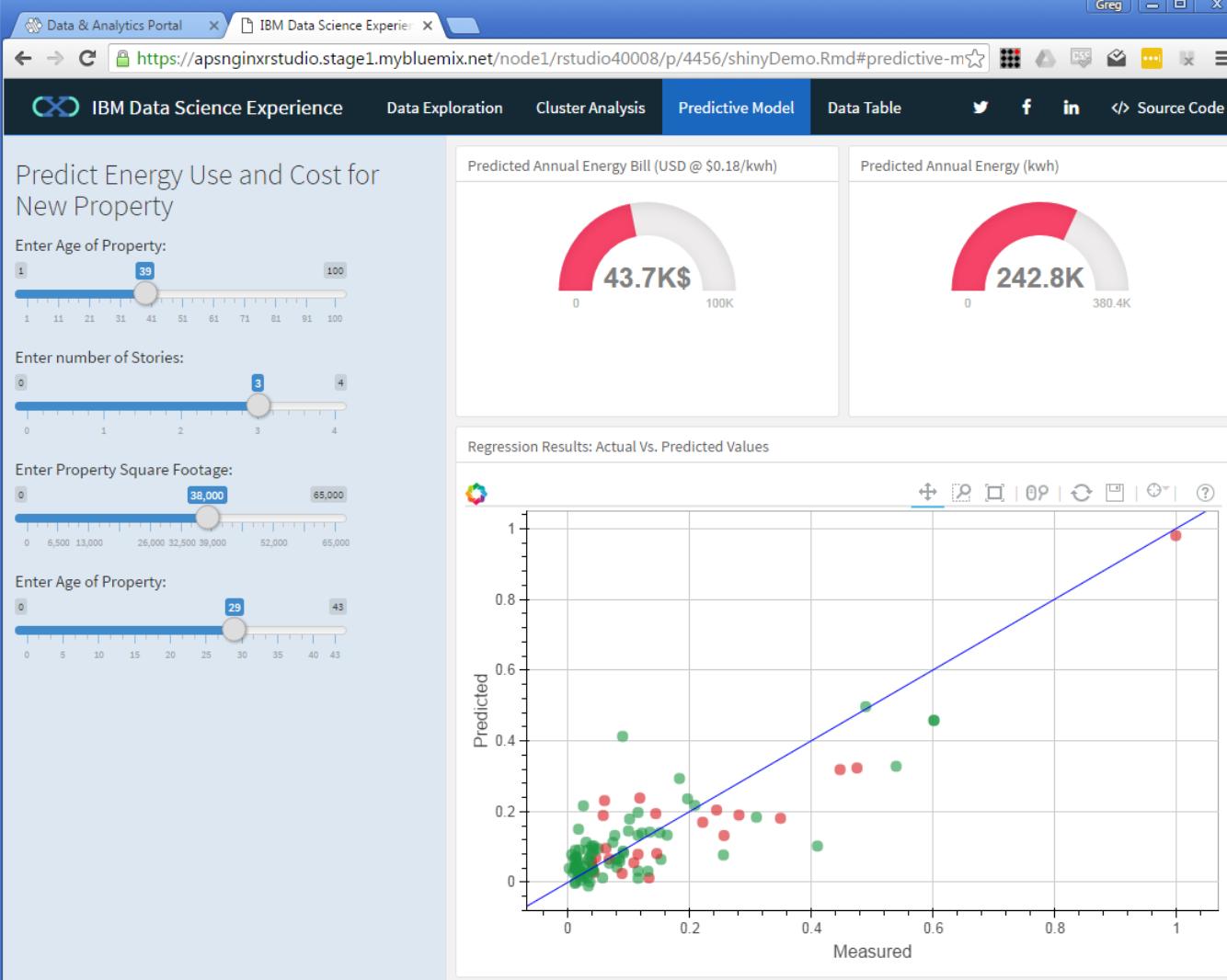
- age of the building
- square feet
- number of stories
- number of plugged equipment, ...

```
In [12]: features = dfScaled.columns.tolist()
response = ['energy_usage']
features.remove(response[0])
# prepare data for regression
lr = linear_model.LinearRegression(fit_intercept=True)
y = np.asarray(dfScaled[response])
X = dfScaled[features]
# run regression
regr = lr.fit(X,y)
coefs = regr.coef_[0]
# collect regression results
dataRegQ = []
dataRegQ.append(['Intercept', regr.intercept_])
for i in range(len(features)):
    dataRegQ.append(([features[i]],coefs[i]))
yh = regr.predict(X)
print 'R-Squared: ', r2_score(y,yh)
pd.DataFrame(dataRegQ,columns=['feature_name', 'coefficient'])
```

R-Squared: 0.725632741591

Out[12]:

feature_name	coefficient
0 Intercept	-0.092789
1 age	0.139789
2 number_stories	0.059749
3 square_feet	0.734468
4 plug_equipment	0.330050
5 domestic_gas	0.208283
	0.105810



Predict Energy Use and Cost for New Property

Enter Age of Property: 39

Enter number of Stories: 3

Enter Property Square Footage: 38,000

Enter Age of Property: 29

Predicted Annual Energy Bill (USD @ \$0.18/kwh)

Predicted Annual Energy (kwh)

Regression Results: Actual Vs. Predicted Values

The scatter plot shows the relationship between Measured values (x-axis) and Predicted values (y-axis). A blue diagonal line represents the line of perfect prediction. Most data points are clustered around this line, indicating good model performance.

▪Sign Up For Your Free Trial Today

- <http://datascience.ibm.com>



IBM Data Science Experience

Plans

Community

Blog

Docs

Sign In

Sign Up

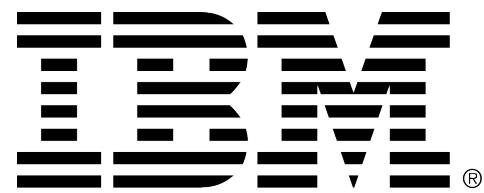
Master the art of data science

Solve your toughest data challenges with the best tools and the latest expertise in a social environment built by data scientists.



Watch Now (2:31)

[Sign Up For a Free Trial](#)



Legal Disclaimer

- © IBM Corporation 2015. All Rights Reserved.
- The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
- If the text contains performance statistics or references to benchmarks, insert the following language; otherwise delete:
Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- If the text includes any customer examples, please confirm we have prior written approval from such customer and insert the following language; otherwise delete:
All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.
- Please review text for proper trademark attribution of IBM products. At first use, each product name must be the full name and include appropriate trademark symbols (e.g., IBM Lotus® Sametime® Unyte™). Subsequent references can drop "IBM" but should include the proper branding (e.g., Lotus Sametime Gateway, or WebSphere Application Server). Please refer to <http://www.ibm.com/legal/copytrade.shtml> for guidance on which trademarks require the ® or ™ symbol. Do not use abbreviations for IBM product names in your presentation. All product names must be used as adjectives rather than nouns. Please list all of the trademarks that you use in your presentation as follows; delete any not included in your presentation. IBM, the IBM logo, Lotus, Lotus Notes, Notes, Domino, Quickr, Sametime, WebSphere, UC2, PartnerWorld and Lotusphere are trademarks of International Business Machines Corporation in the United States, other countries, or both. Unyte is a trademark of WebDialogs, Inc., in the United States, other countries, or both.
- If you reference Adobe® in the text, please mark the first use and include the following; otherwise delete:
Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
- If you reference Java™ in the text, please mark the first use and include the following; otherwise delete:
Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- If you reference Microsoft® and/or Windows® in the text, please mark the first use and include the following, as applicable; otherwise delete:
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- If you reference Intel® and/or any of the following Intel products in the text, please mark the first use and include those that you use as follows; otherwise delete:
Intel, Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- If you reference UNIX® in the text, please mark the first use and include the following; otherwise delete:
UNIX is a registered trademark of The Open Group in the United States and other countries.
- If you reference Linux® in your presentation, please mark the first use and include the following; otherwise delete:
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Other company, product, or service names may be trademarks or service marks of others.
- If the text/graphics include screenshots, no actual IBM employee names may be used (even your own), if your screenshots include fictitious company names (e.g., Renovations, Zeta Bank, Acme) please update and insert the following; otherwise delete: All references to [insert fictitious company name] refer to a fictitious company and are used for illustration purposes only.