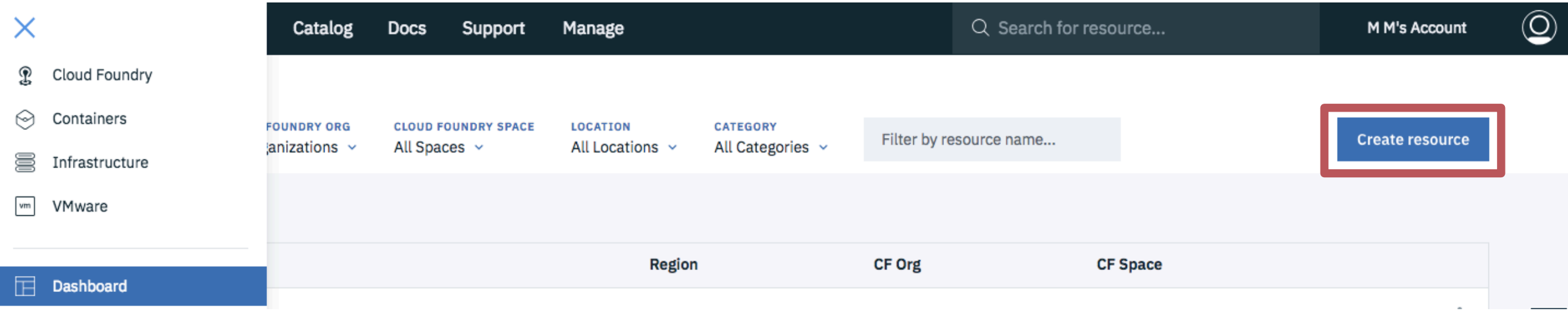
# ibm.biz/localcart


Analyse
Customer Data
and Recommend
Products with
PixieDust
IBM Developer

Arlemi Turpault Yamini Rao Margriet Groenendijk Mofe Salami

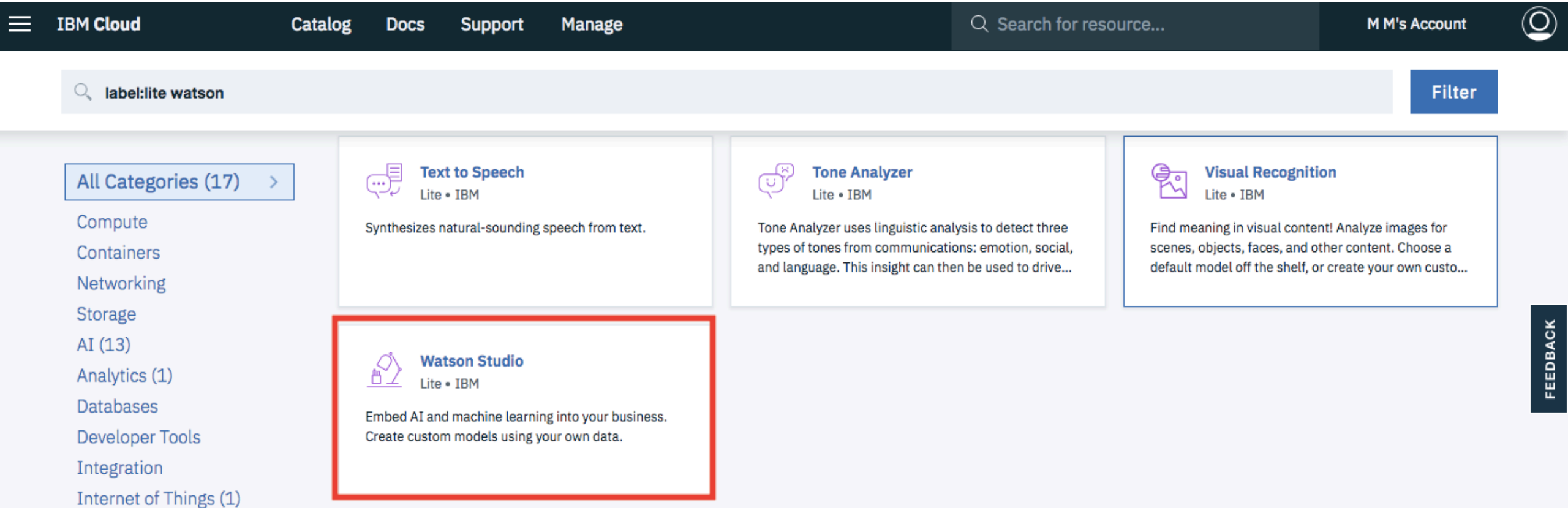
# Access Watson Studio

## At the top of your IBM Cloud dashboard click Create Resource

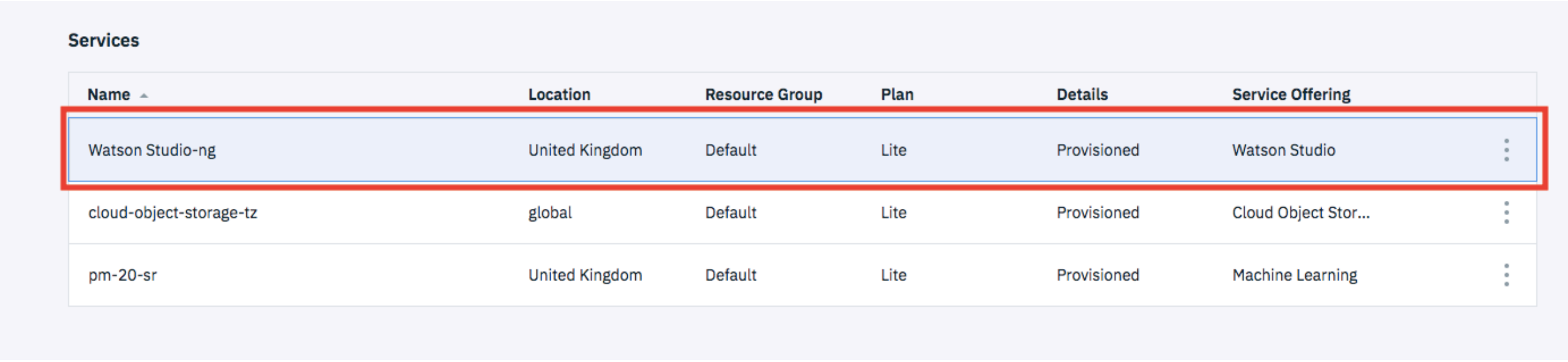


# Search for Watson Studio and click on the tile

# Select the Lite plan and click Create



# Go back to the dashboard and click on your Watson Studio service and then click Get Started



# Part 0: Getting Started

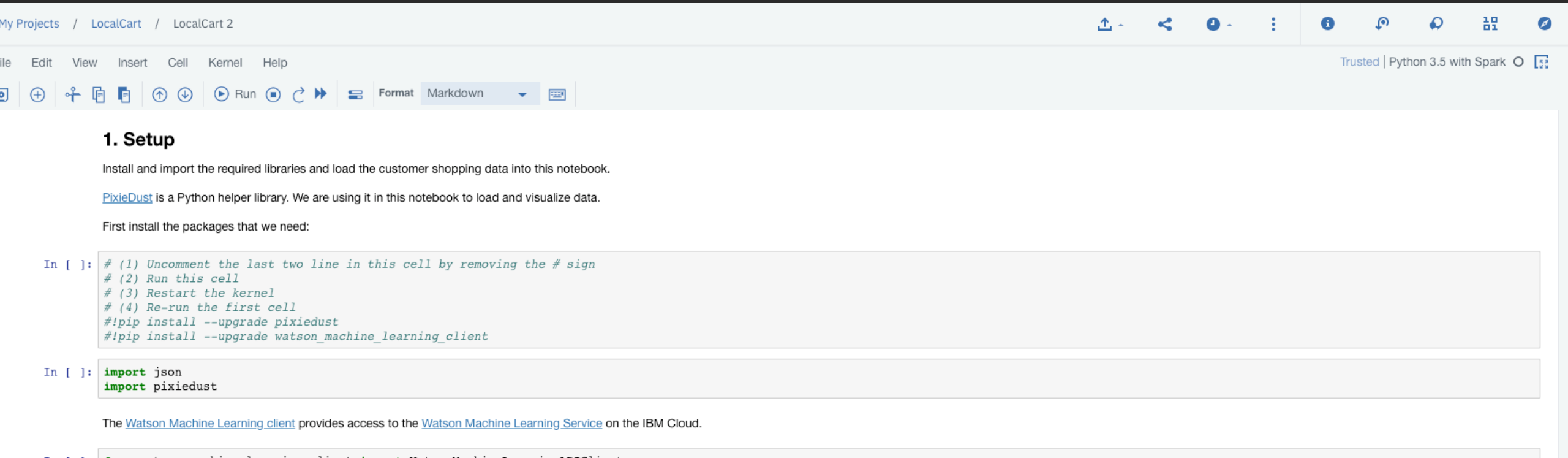
**ibm.biz/localcart**

# Part 1: Static data analysis using Python, Apache Spark, and PixieDust

IBM **Developer**

# Jupyter Notebooks - http://jupyter.org/

- Notes
- Ideas
- Python code
- Charts and tables



# Why Spark?

Cluster computing platform designed to be fast and general purpose

APIs in Python, Java, Scala and SQL

Data Science: data wrangling, machine learning

Data Processing: data pipelines

# Why Spark?

```
df = df.select()

df.filter()

df.join()
```

```
df.groupby().sum()

df.printSchema()

df.show()

df.cache()
```

# Why Spark?

```
df = df.select()

df.filter()

df.join()
```

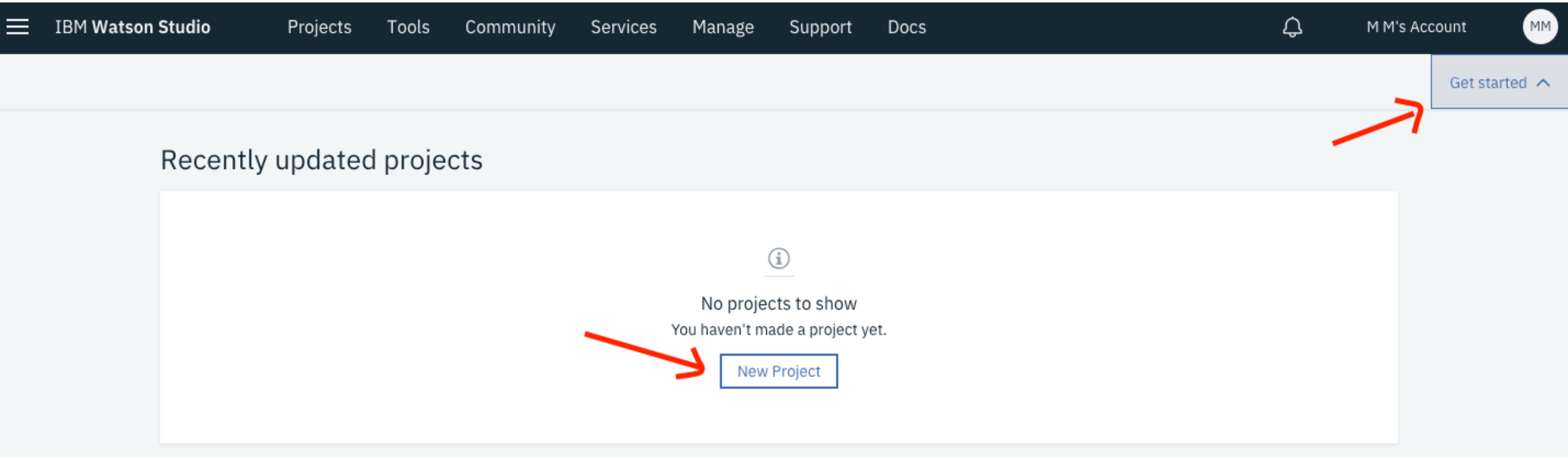
```
df.groupby().sum()

df.printSchema()

df.show()

df.cache()
```

Lazy!

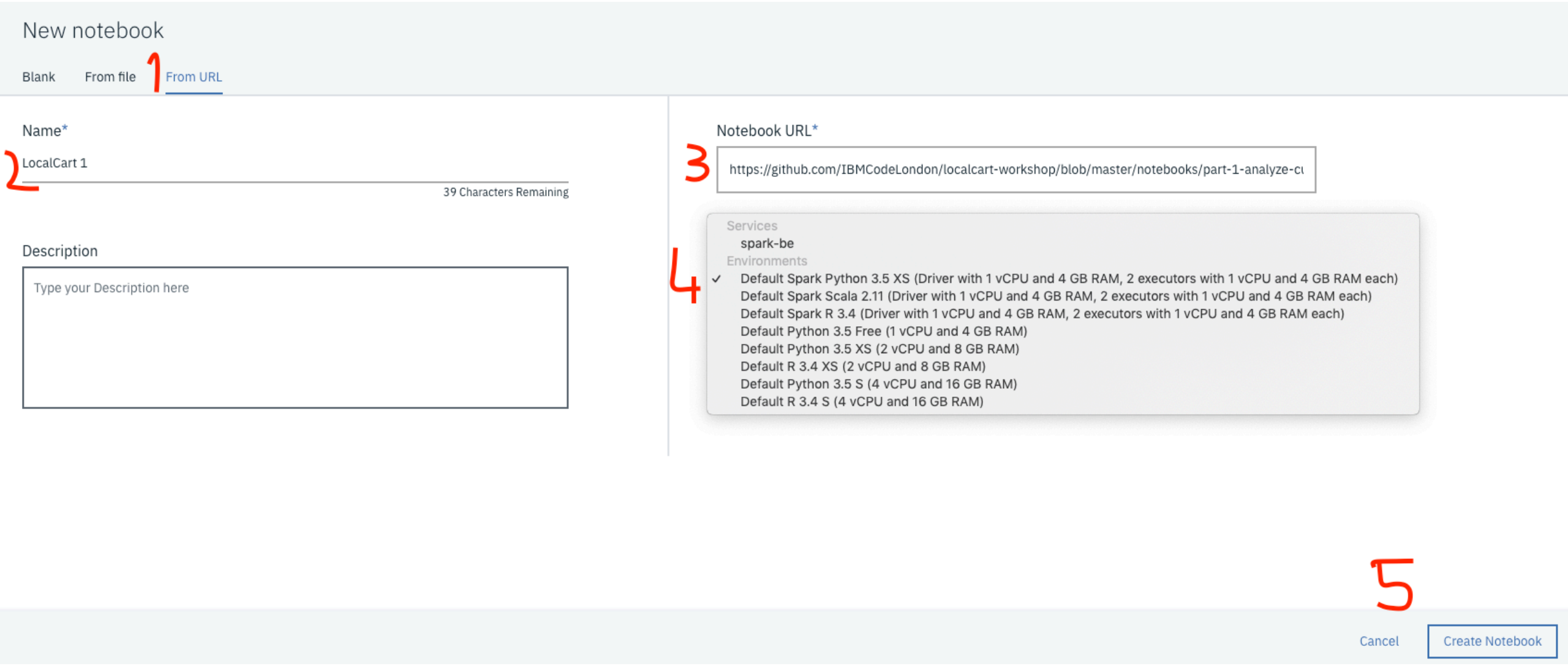
# PixieDust

```
import pixiedust
display(df)
```

# Watson Studio - create a new project



# Watson Studio - create a notebook



# Part 1: Static data analysis using Python, Apache Spark, and PixieDust

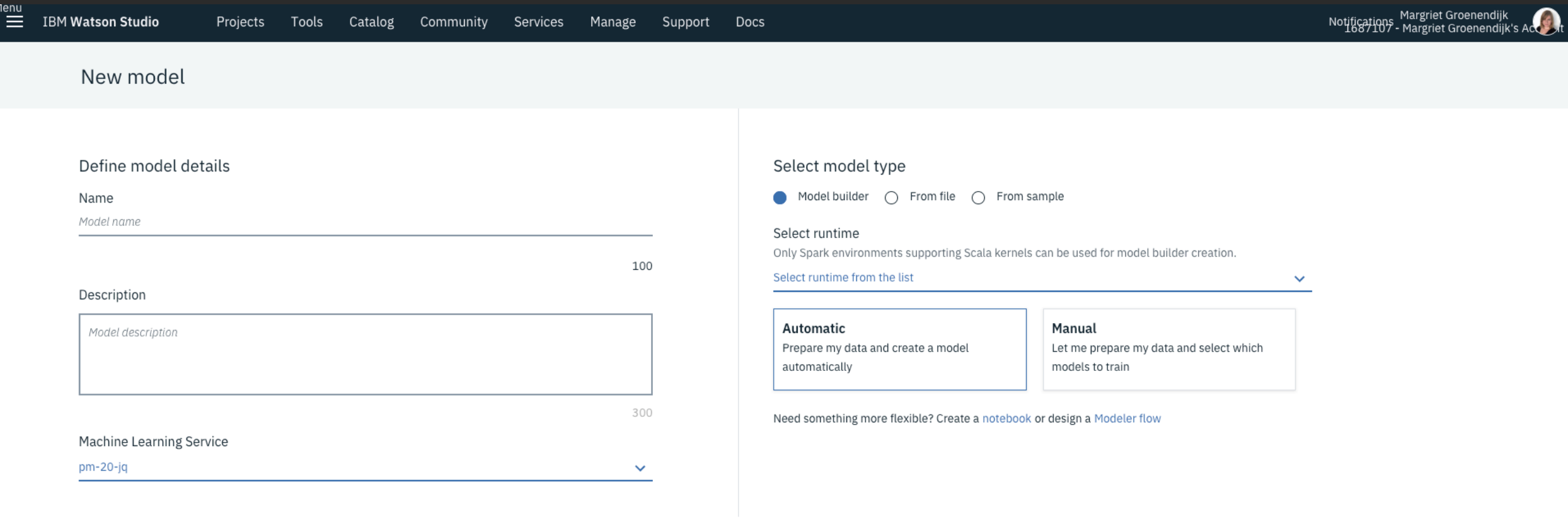
**ibm.biz/localcart**

# Part 2: Build a product recommendation engine

IBM **Developer**

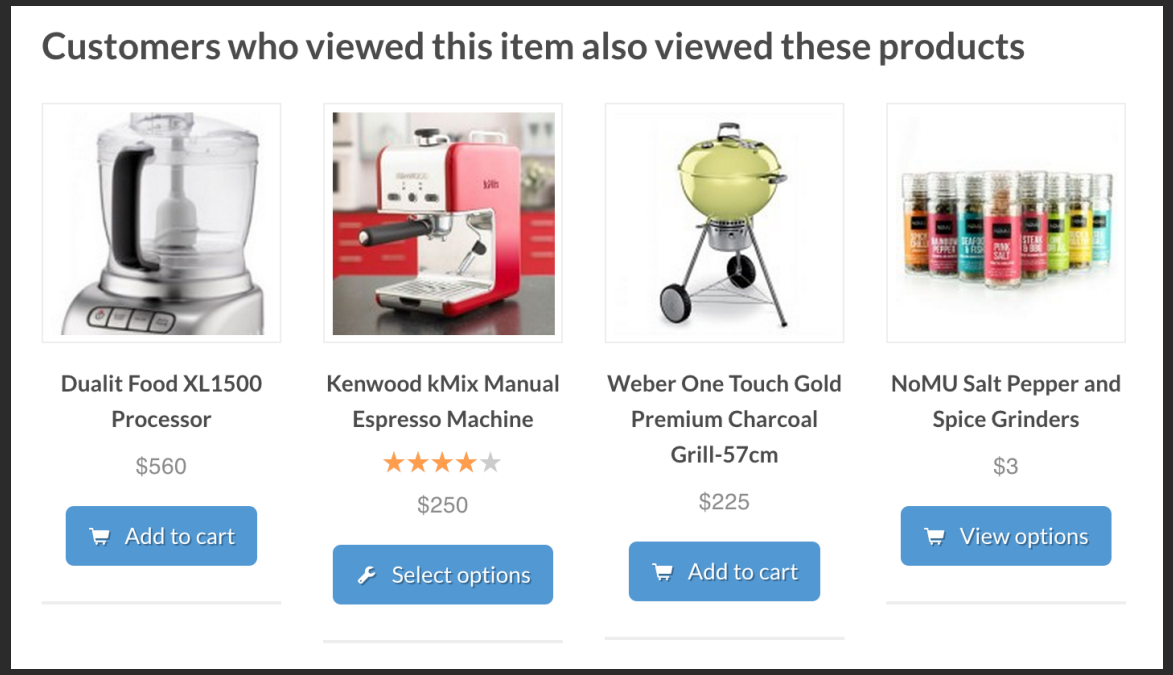
# Watson Machine Learning

# Last meetup: train model in Watson Studio



# Watson Machine Learning

# Today: train model in a notebook



# Spark MLlib

- ML Algorithms: common learning algorithms such as classification, regression, clustering, and collaborative filtering
- Featurization: feature extraction, transformation, dimensionality reduction, and selection
- Pipelines: tools for constructing, evaluating, and tuning ML Pipelines
- Persistence: saving and load algorithms, models, and Pipelines
- Utilities: linear algebra, statistics, data handling, etc.

https://spark.apache.org/docs/2.2.2/ml-guide.html

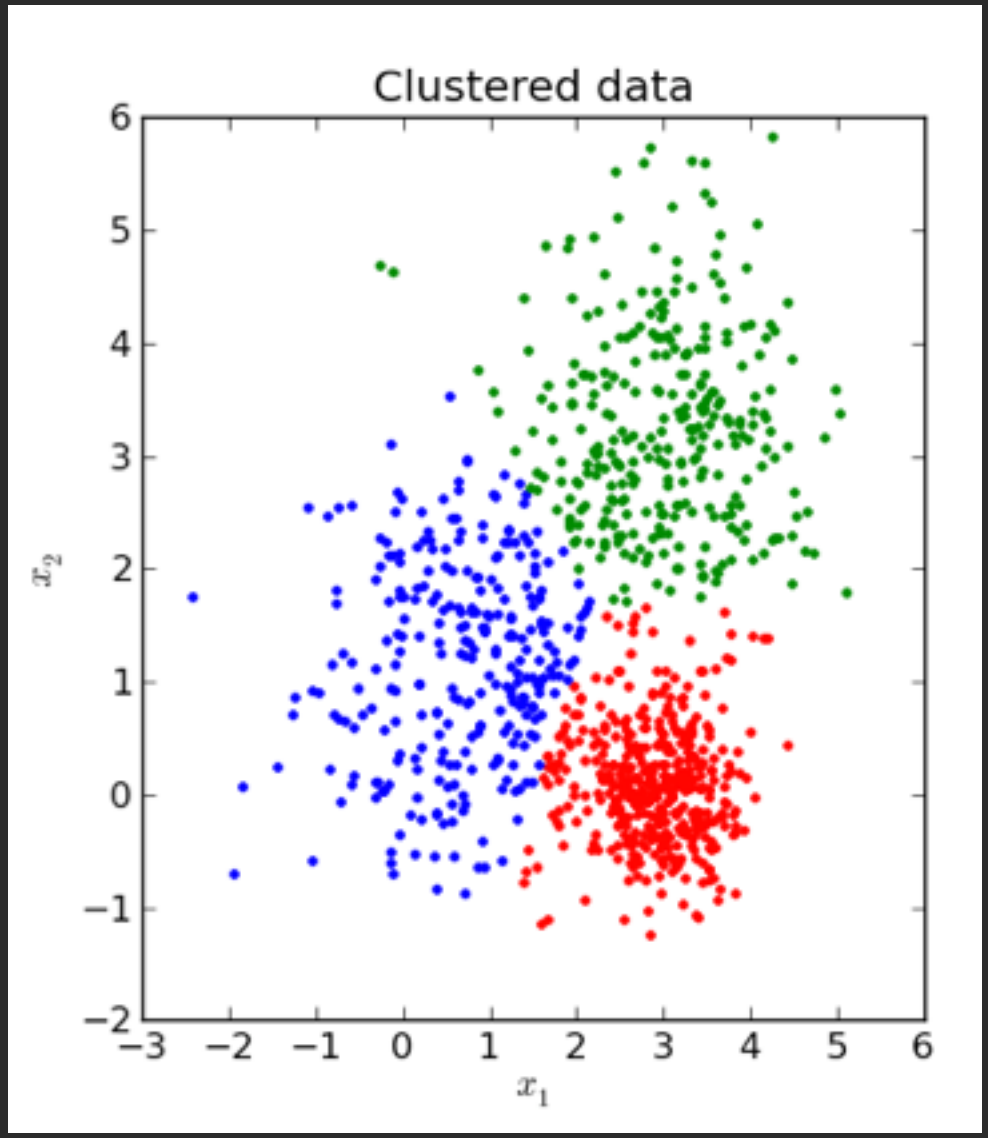
# Recommender Engine

Every dot is a customer
The axes are the number of products A and B bought

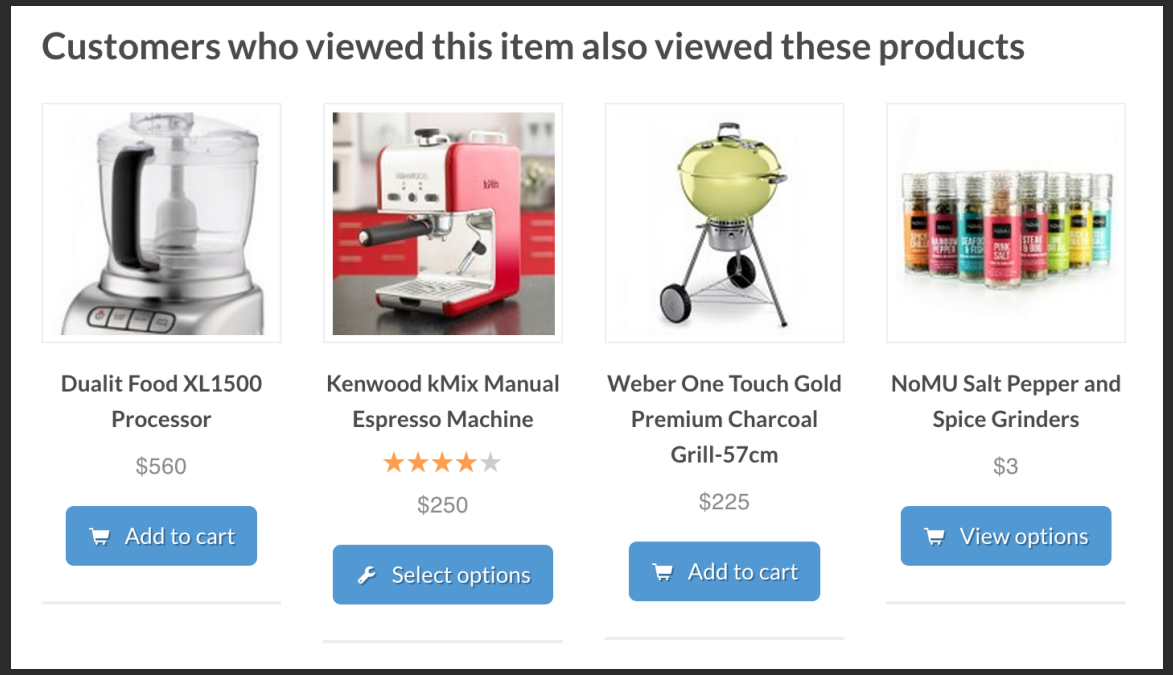
You can now use clustering algorithms to group these customers

Imagine not 2 but thousands of products in a thousands dimensional space

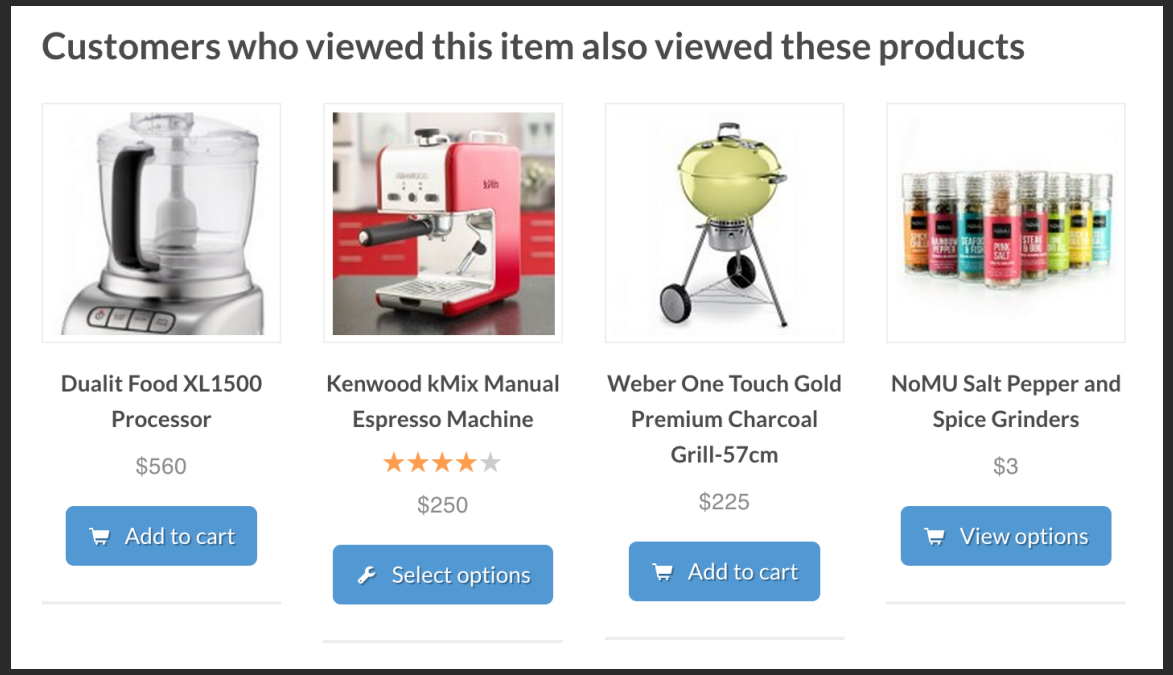
With machine learning algorithms you can still find clusters of customers who are similar



## But how do you get from the clusters to a recommendation?



## But how do you get from the clusters to a recommendation?

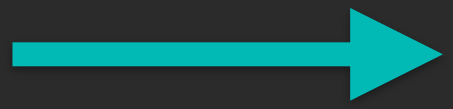



Recommend the most popular items in the cluster

But filter out the products already bought

# Watson Machine Learning

- Spark MLlib
- scikit-learn
- XGBoost
- TensorFlow
- Keras
- Caffe
- PyTorch



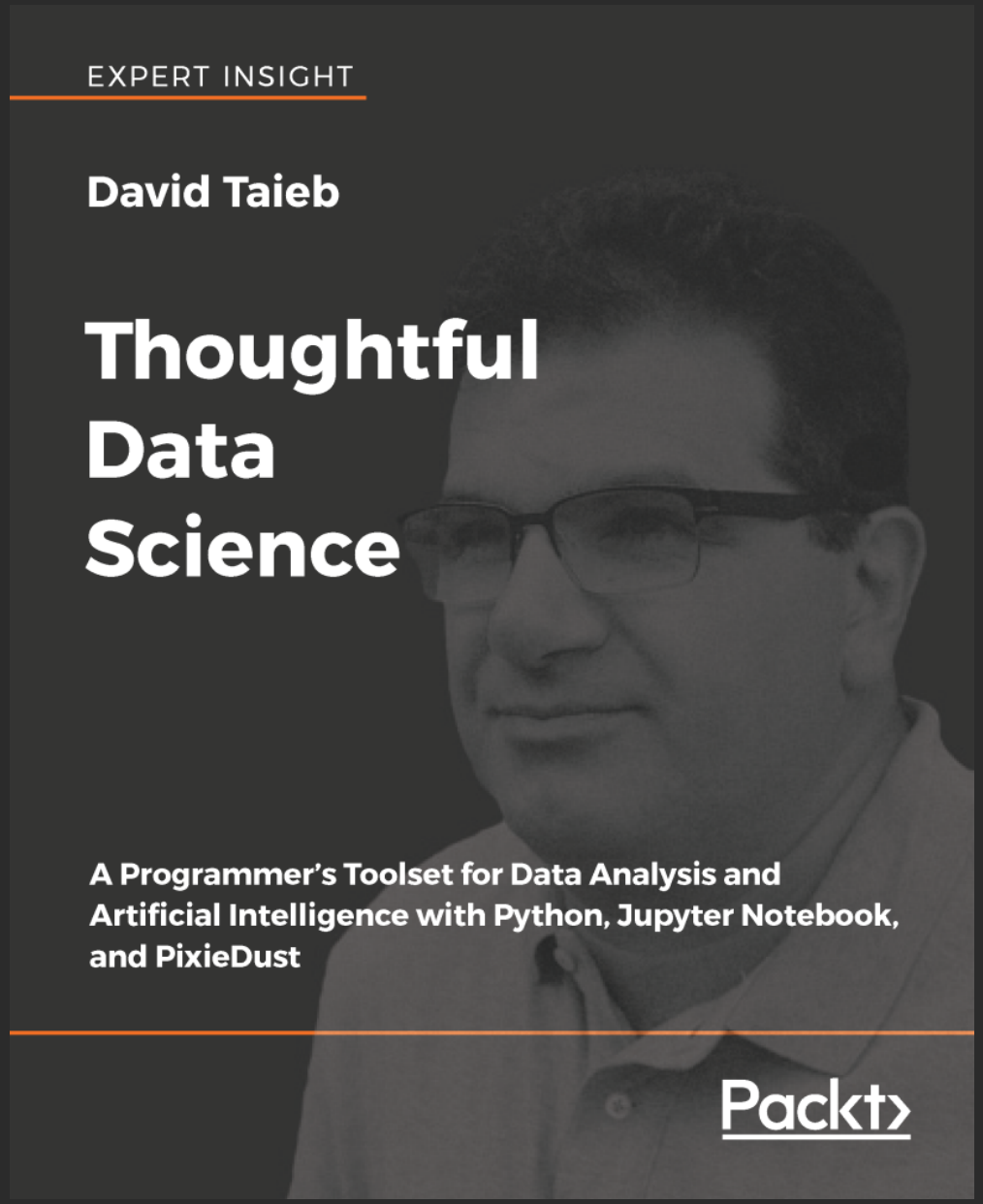
API

# PixieApps

# Part 2: Build a product recommendation engine

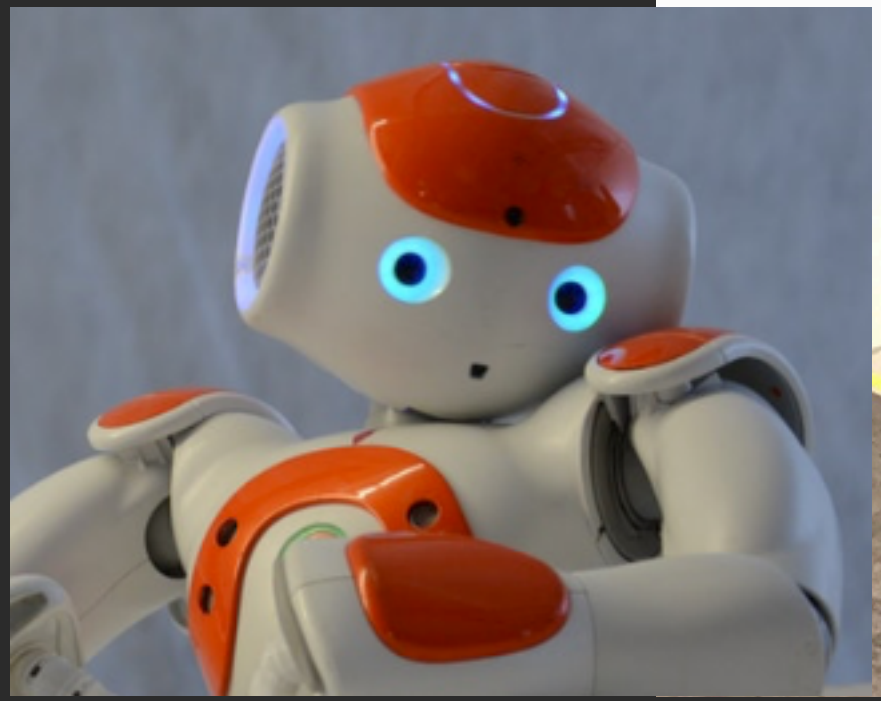
**ibm.biz/localcart**

# PixieDust and PixieApps book!



# Build your own AI applications

## 27 November



Ed Shee

Serverless Workshop

**5 December**