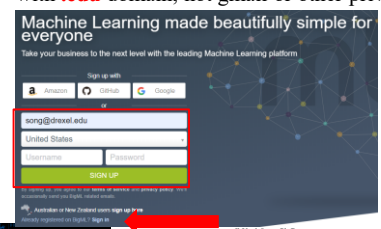
BIGML

Il-Yeol Song, Ph.D.
College of Computing & Informatics
Drexel University
Philadelphia, PA 19104

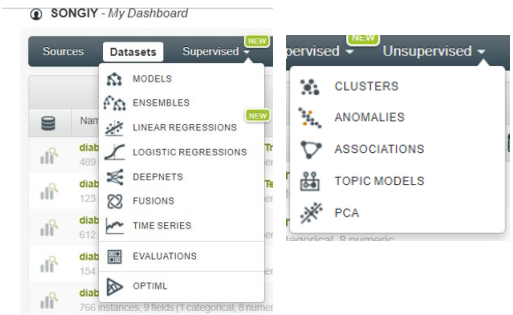
Hands-on Experience with BigML

- Access to BigML: https://bigml.com/

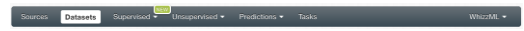
Step 1: Create your own ID and PW. In order to get a free ID for one year, you must use an email ID with **.edu** domain, not gmail or other private IDs.



ML Methods supported in BigML

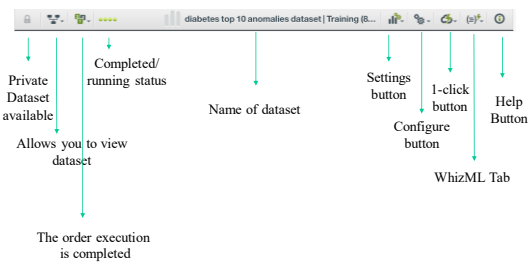


Dashboard Interface (First row)



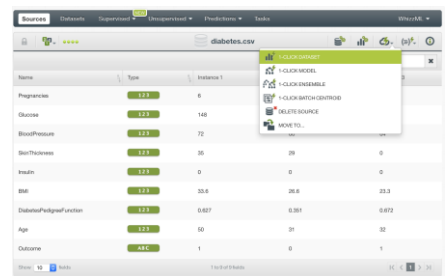
- Sources: Denotes the .csv files loaded into the BigML
- Datasets: Denotes the files stored/saved after processing. While choosing algorithms, you select files under this folder
- Supervised: Consist of algorithms that can be performed under supervised learning
- Unsupervised: Consists of unsupervised learning algorithms
- Predictions: Consists of files that have been saved after performing predictions
- Tasks: Consists of running and killed tasks
- WhizzML: Consists of Scripts, Libraries and executions that have been downloaded externally and imported into the software
  Example: To perform 5-fold validation, you need to externally download the script and import it into BigML

BigML Dashboard

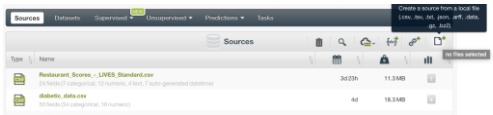


Displaying Data Distribution

- Click on the Cloud icon and then choose '1-click dataset' to create a structured format



Uploading Data File Into BigML



- Click on the above button to upload csv locally from your desktop
- The '{abc} sign' allows you to create an inline source to upload data files
- The 'configure sign' allows you to upload data files by specifying the url link

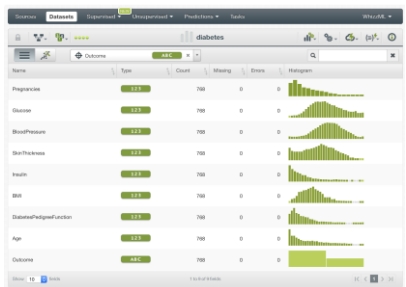
Data Description

- ✓ The Diabetes dataset consists of **9** variables
- ✓ (https://www.kaggle.com/uciml/pima-indians-diabetes-database)
- ✓ There are a total of **768 rows**.

- **Pregnancies**: Number of times pregnant
- **Glucose**: Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- **BloodPressure**: Diastolic blood pressure (mm Hg)
- **SkinThickness**: Triceps skin fold thickness (mm)
- **Insulin**: 2-Hour serum insulin (mu U/ml)
- **BMI**: Body mass index (weight in kg/(height in m)^2)
- **DiabetesPedigreeFunction**: Diabetes pedigree function
- **Age**: Age (years)
- **Outcome**: Class variable (0 or 1) 268 of 768 are 1, the others are 0

Diabetes Data Description

- ✓ The Diabetes dataset consists of **9** variables
- ✓ There are a total of **768 rows**.



Dataset Layout

Figure 5.1: Dataset layout overview

Updating Field Windows

Figure 5.5: Updating field view

Scatter Plot

- Once your dataset, is loaded click on the scatterplot option as shown below:

- The visualization on the right will be displayed.
- The scatterplot uses a sample of 500 instances to generated the scatterplot
- The data inspector shows how each field affects the data set
- They are ranked in the order of how they affect other properties
- You can change the x and y axis to see how the scatter plot varies.

Anomaly Detection

- Uses the method of isolation forest to separate anomalies in a dataset.
- Ranks the list of anomalies detected in a dataset by a score called as *anomaly score*.
- If the given instance has an **anomaly score > 60**, it is considered to be anomalous.
- To create an anomaly, you can use the 1-click option, or change the parameters using the configuration option.

Diabetes Dataset- Anomaly Detection

- The image on the right shows the TOP 10 anomalies for the given dataset.
- The field importance is indicated by histogram indicating the contribution of input filed to the anomaly.
- When you move your mouse over an instance in the Top Anomalies, you can see the values change in the data inspector.

Field importance

Field histogram

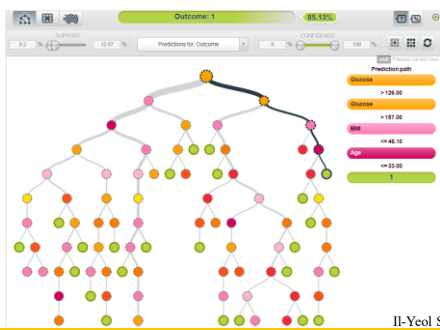
Allows you to select the anomalies and create a dataset without them

Splitting the dataset (Step A)

- Click the Training| Test Split before starting to work on any model
- This will generate 80% Training set and 20% data set to work on

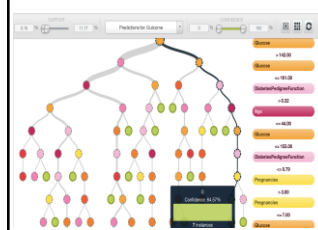
MODEL - DECISION TREE

Decision Tree Model



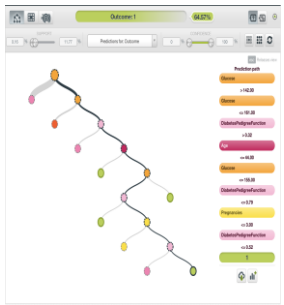
18

Visualization



- Model was generated from 80% Training dataset.
- The model varies as it depends on random sampling of 80% dataset.
- The decision tree model works using the probability values of each branch
- If we click on a node, it will show the values that were chosen and the path with the probability values.

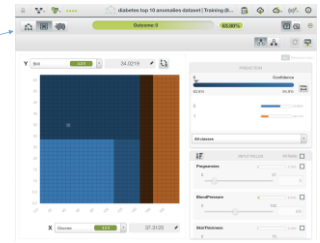
Interpretation



- This branch of the tree shows how each instance value is branched out.
- Each branched node is assigned a probability and based on that, the next node is decided to arrive at the answer.
- Click on the top node to display the complete tree.
- Press 'shift' to freeze the screen (Press ESC to unfreeze).

Partial Dependency Plot

PDP ICON



You can select the input fields here. Unchecked fields are considered missing fields.

Model Summary

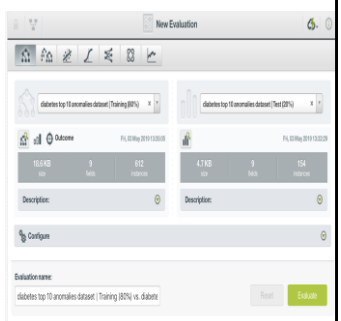
- Field importance provides how important one filed is to another
- Computed by taking an weighted average of how much each field reduces the predicted error of the tree at each split in the end
- Click on the model summary report to view the field importance

Evaluation

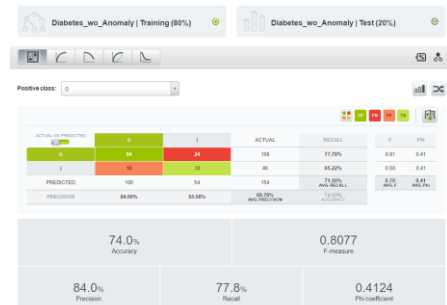
- Accuracy: denotes the correct percentage of correctly classified instances over total instances evaluated
- Precision: Percentage of correctly predicted instances over total instances
- Recall: percentage of correctly classified instances over total actual instances
- F-Measure: Balanced harmonic mean between precision and recall
- Phi Coefficient: correlation coefficient between predicted and tual values
- Icons can be expanded as below:
- General Evaluation, ROC Curve, Precision-Recall Curve, Gain curve and Lift Curve

Evaluation

- To perform evaluation of the Ensemble Decision Tree, choose the Training 80% from the dataset tab
- After performing the algorithm, select evaluate button from the 'cloud button'
- The evaluation is performed between Data Source | Training (80%) vs Test 20%
- You can change the settings by clicking on the configuration button
- Click evaluate button to perform the evaluation



Confusion Matrix



Creating Model Predictions

Three options to make predictions from your models:
• **PREDICT BY QUESTION**: to predict a single instance answering just the relevant questions required by the model.
• **PREDICT:** to predict a single instance using the prediction form.
• **BATCH PREDICTION**: to predict multiple instances simultaneously.

Method:

Under the '**Supervised tab**' click on the algorithm you wish to perform prediction on.

Click on the **drop down button** that appears near the dataset

CLUSTERING

Clustering Technique Provided by BigML

K-means Algorithm

- User already knows the number of clusters(k) to be formed using the dataset.
- If k is not known beforehand, it might yield poor results.
- Maximum number of cluster = 300

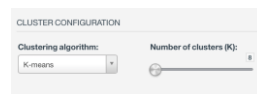
G-Means Algorithm

- If user does not know optimal number of clusters, G-means is used.
- Tries to find the number of clusters by iteratively taking existing clusters and testing whether the cluster's neighborhood appears Gaussian in its distribution.
- Maximum number of cluster = 128

Configuring Clusters

K-Means

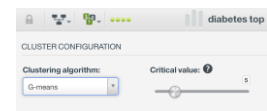
- While selecting K-Means algorithm, you select the **number of clusters**



- If you do not have a clear idea of the no. of clusters, then it is better to use G-Means algorithm.

G-Means

- In this, you are required to choose a **critical value.**



- Sets how 'strict' the value should be
- By default BigML uses **5** as critical value
- Usually range between 1 to 10 is used

CLUSTERS

- Centroid cluster is the center of the visualization
- Computed using mean or each numeric field and mode for categorical ones
- You can create a model or a dataset (containing the instances from the cluster) by pressing **shift** button on the screen
- To unfreeze the screen press the **ESC** button.

NOTE: The create 'model' option will be displayed only if you select the option in the configuration menu.