A Mariner White Paper

MARINER

The Allure of Machine Learning, now within Reach in Microsoft Azure

Or Why AzureML is Better for Data Mining than Excel

By Colby Ford, Associate Data Analytics Consultant

2719 Coltsgate Road • Charlotte, NC 28211
tel. 704.540-9500 • fax. 704.540-9501
www.mariner-usa.com

Contents

Abstract

Coming from the recent buzz about topics like "Big Data" and "Predictive Analytics", machine learning is an interest of many companies today. In this whitepaper, the discussion is about Azure Machine Learning (AzureML) and a comparison of its capabilities with a tried-and-true bike customer data example that is normally used in SQL Server Analysis Services with the Excel Data Mining add-in.

The Allure of Machine Learning

When I think of Machine Learning and how it relates to Big Data analytics, I think of the iconic episode of *I Love Lucy* where Lucy and Ethel take a job at a candy factory. The machinery spits out chocolates faster than the two ladies can place wrappers on them. Their inept skills for handling such an immense amount of chocolates forces them to take drastic measures and stuff the candies into their blouses, aprons, and hats. Doesn't this sound like Big Data? We are collecting more data than we know what to do with. We stuff it into data warehouses and run it through analytics software as fast as we can, but it is humanly impossible to handle it all.

Machine Learning is a discipline, a methodology if you will, that allows us to leverage computing power to combat the volume and velocity of the incoming flood of information. The topic is something that many companies feel they *need* to do, but they have a misconception of what it actually does. Machine Learning is not a tangible piece of software or equipment, though we seem to want to touch it, feel it, click it. It is a process. Machine Learning allows us to build data models as data comes in and then change them as even more data comes in. Plus, it can try many types of models and pick the best one based on criteria that you select. Try doing that with a human! We would end up like Lucy and Ethel and surely miss a few million rows of data by stuffing it into our metaphorical apron. Even the best data scientists cannot outperform an automated computing machine nor handle large chunks of data coming at us at an incredibly fast rate.

The Many Faces of Microsoft Azure

Microsoft has an amazing platform called Microsoft Azure. This platform is a cloud-based solution that allows the user to create virtual servers and networks, Hadoop clusters, Active Directory systems, and development services on the fly with a high degree of scalability. No more buying expensive hardware. With Azure, only pay for what you use! Ready for the best part? Microsoft now has a Machine Learning platform built directly into the Azure collection of integrated services. Now more than ever, predictive analytics is within reach for anyone.

AzureML, the name for the service, is easy for anyone to get started. The interface is drag-and-drop and, with a small amount of statistics knowledge, you can add in your data and run the learning system in minutes.

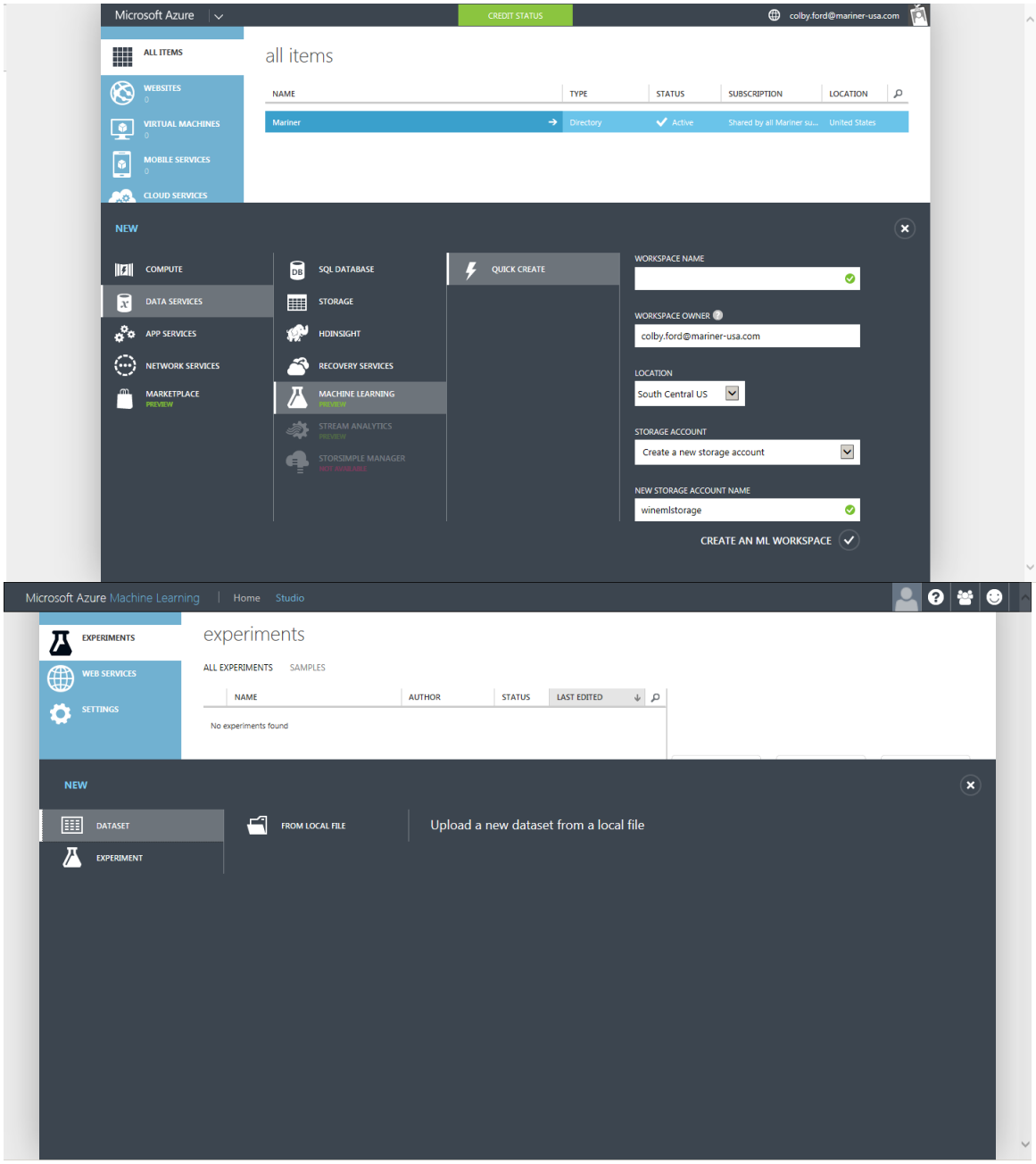
Comparison of Excel Data Mining & Azure Machine Learning

To illustrate the differences in the two systems' results, we first need to create a machine learning service in Microsoft AzureML and test data about bicycle buyers. This data set has been around since 2005 and has been used time and time again to illustrate the capabilities of the Excel Data Mining add-in as well as SQL Server Analysis Services. This time, I wanted to see how it works in AzureML. The data input is as follows:

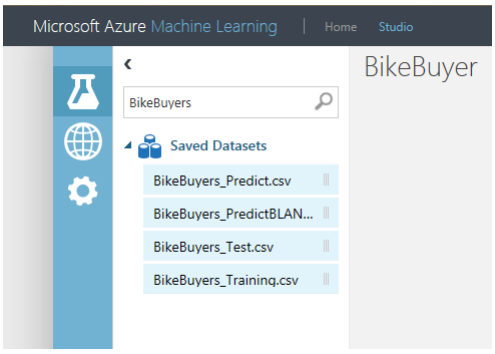
Marital Status	Gender	Yearly Income	Children	Education	Occupation	Home Owner	Cars	Commute Distance	Region	Age	BikeBuyer
Married	Female	40000	0	Graduate Degree	Clerical	Yes	0	0-1 Miles	Europe	37	Yes
Married	Male	160000	5	Partial College	Professional	No	3	10+ Miles	Europe	55	No
⋮					⋮						⋮
Single	Female	70000	0	High School	Professional	Yes	2	5-10 Miles	Pacific	49	Yes

We are trying to create a model that accurately determines whether a person is a potential bike buyer or not. That is, is the BikeBuyer field for a particular customer "Yes" or "No". This example data is prepackaged with the Excel Data Mining add-in for SQL Server Analysis Services. I've run the classification analysis (which is using the decision tree algorithm built into the add-in.) This yielded the prediction dataset that is used later in AzureML.

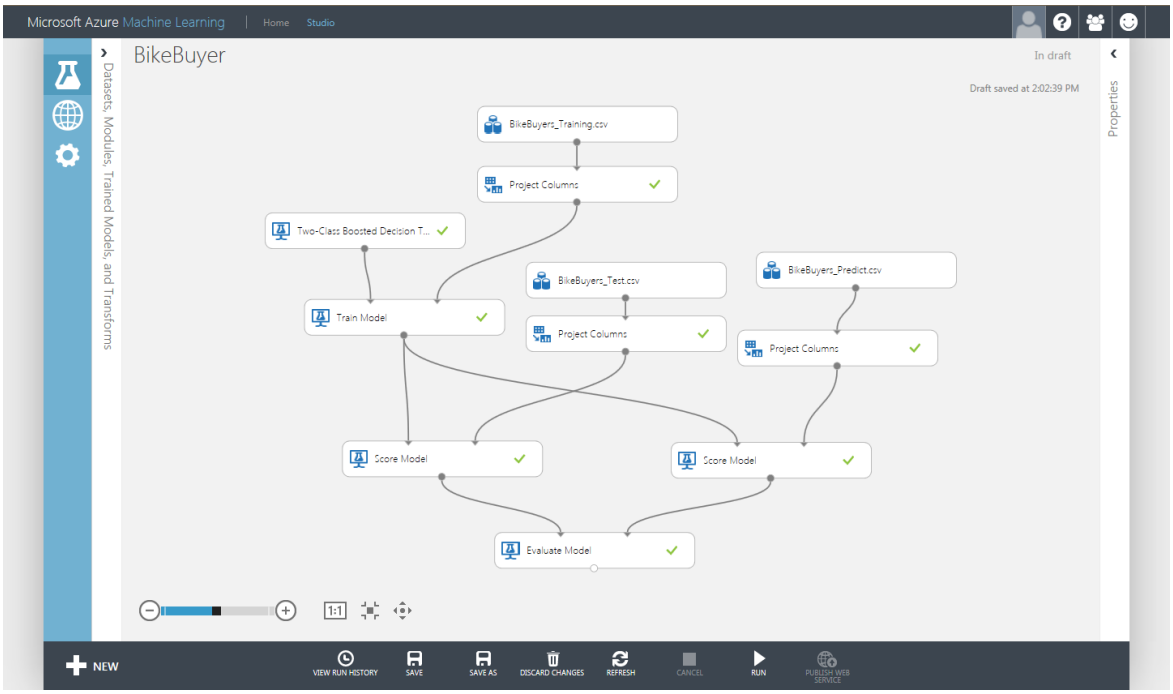
The first step in AzureML is to create your workspace where you'll have your data and publish your services you create and then go into the ML Studio and add your data.



We will have to create a new experiment to house the different tests and models that the ML will build and drag the newly uploaded data into the window. Notice that my datasets are called *BikeBuyers_Training*, *BikeBuyers_Test*, and *BikeBuyers_Predict*. This helps to keep them separate later.



For anyone who has ever used statistical analysis software like R, SAS, or SPSS, you know that after you pull in your data, the next step is to select your desired test to run on it. The same is true in AzureML except we want to run different tests at the same time and evaluate the different results.



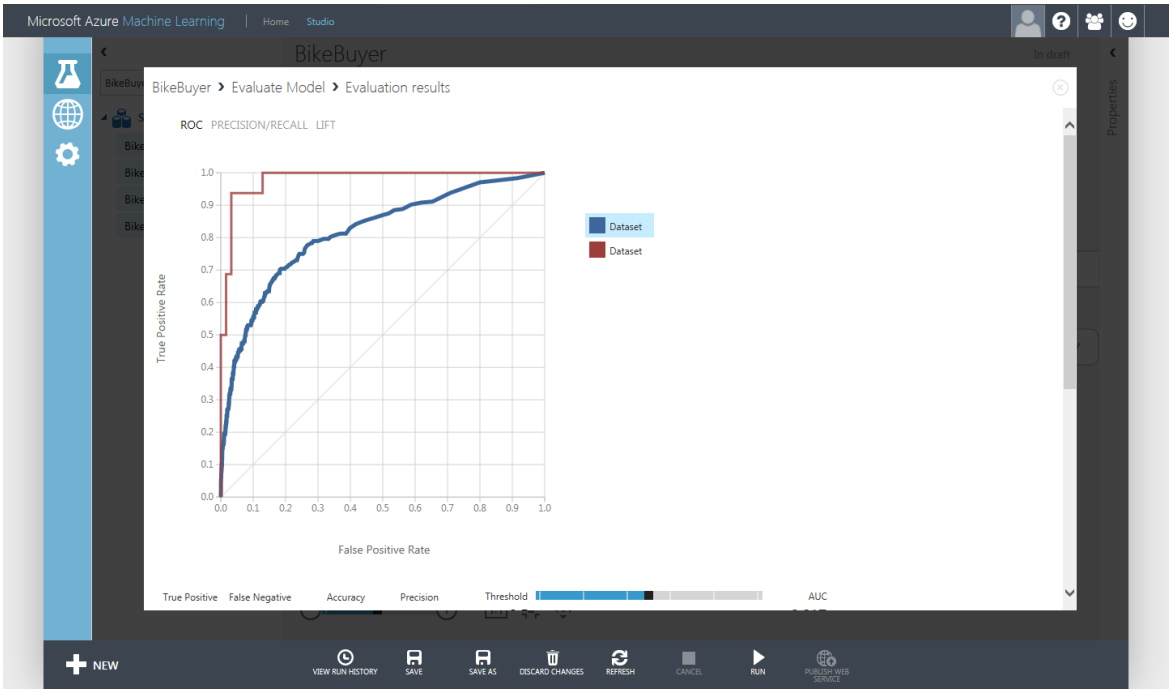
Here, I've selected all the datasets that will be necessary to run the analysis. Note I've pulled the training set, test set, and the prediction set that was uploaded earlier (*BikeBuyers_Training*, *BikeBuyers_Test*, and *BikeBuyers_Predict*, respectively). Then, the Two-Class Boosted Decision Tree is selected along with the nodes to select the desired columns, score the models' performance, and evaluate them against each other.

The training set will be used to build the model and the test set will be used to test its performance. Then, the prediction set is used to test AzureML's model prediction with the prediction given by the Excel Data Mining add-in.

The "Evaluate Model" node allows the user to see the ROC curves, Precision/Recall, and Lift for both of the models as well as statistics like accuracy, F1 scores, AUC, etc*. This is compared to the results from the Excel Data Mining tool below.

*Note: ROC curves, AUC, Precision/Recall, etc. are all comparative metrics used to test the fit of the models.

- True Positive, True Negative, False Positive, and False Negative come measuring in the confusion matrix. To learn more about the confusion matrix, visit: http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html
- ROC curves measure sensitivity and 1-specificity. The "good" spot to be in is with a high sensitivity and a low 1-specificity. Sensitivity is the hit rate and is the number of true positives divided by the total number of positives (true positives and false negatives). Specificity is the number of false positives divided by the total number of negatives (false positives and true negatives).
- AUC is the area under the ROC curve. So, the ideal AUC would be 1.
- Precision is the percentage of positives that were detected that are actually true. Recall is the percentage of true positives that were detected out of all positives.
- The F1 score is given by the function $F_1 = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$ and a "good" score would be anything close to 1.



The smoother, blue line is the comparison of the training and test datasets.

*

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
57	247	0.909	0.679	0.87	0.817
False Positive	True Negative	Recall	F1 Score		
27	2669	0.188	0.294		

Score bin	# Pos	# Neg	Pop.above thresh.	Accuracy	F1	+ve Prec.	+ve Rec.(= TPR)	-ve Prec.	-ve Rec.(= 1 - FPR)	Cumulative AUC
(0.900,1.000]	48	18	0.022	0.909	0.259	0.727	0.158	0.913	0.993	0.001
(0.800,0.900]	15	21	0.034	0.907	0.310	0.618	0.207	0.917	0.986	0.002
(0.700,0.800]	13	12	0.042	0.907	0.353	0.598	0.250	0.921	0.981	0.003
(0.600,0.700]	20	21	0.056	0.907	0.407	0.571	0.316	0.927	0.973	0.005
(0.500,0.600]	12	18	0.066	0.905	0.430	0.545	0.355	0.930	0.967	0.008
(0.400,0.500]	20	29	0.082	0.902	0.465	0.518	0.421	0.936	0.956	0.012
(0.300,0.400]	10	39	0.099	0.892	0.460	0.466	0.454	0.939	0.941	0.018
(0.200,0.300]	22	62	0.127	0.879	0.468	0.421	0.526	0.945	0.918	0.029
(0.100,0.200]	24	128	0.177	0.844	0.440	0.346	0.605	0.951	0.871	0.056
(0.000,0.100]	120	2348	1.000	0.101	0.184	0.101	1.000	1.000	0.000	0.817

The accuracy for this model is at ~91% and the F1 Score is 0.294. Both are only marginally higher than the Data Mining model's finding. Below is the Excel classification matrix output generated from testing the trained model with the test data.

Model name:	**Bike Buyer Decision Tree**	
Total correct:	**89.13 %**	**2674**
Total misclassified:	**10.87 %**	**326**

Results as Percentages for the Bike Buyer Decision Tree Model

	No(Actual)	**Yes(Actual)**
No	**97.37 %**	83.88 %
Yes	2.63 %	**16.12 %**
Correct	**97.37 %**	**16.12 %**
Misclassified	**2.63 %**	**83.88 %**

Results as Counts for the Bike Buyer Decision Tree Model

	No(Actual)	**Yes(Actual)**
No	**2625**	255
Yes	71	**49**
Correct	**2625**	**49**
Misclassified	**71**	**255**

Sensitivity	16.12%
Specificity	97.37%
Precision	40.83%
Accuracy	89.13%
F1 Score	0.231132075

Although 91% is decent for an accuracy measure, once might expect for the accuracy to be much higher from AzureML. AzureML used a boosted, two-class tree algorithm, which should yield better results since the algorithm takes weights into account when looking at each factor in the decision. The decision tree algorithm used in Excel does not. (Note: AUC is a measure of the area under the ROC curve. In short, the closer to 1, the better.)

Now, focusing on the result from the prediction dataset, we should look at accuracy as a measure of how similar the results are from AzureML and Excel, not how "right" or "wrong" it is. If the Excel Data Mining predictions are exactly the same as the AzureML predictions, the accuracy should be 1. Otherwise, AzureML has predicted certain customers to be bike buyers that Excel did not and vice versa.

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
15	1	0.962	0.882	0.61	0.981

False Positive	True Negative	Recall	F1 Score
2	60	0.938	0.909

Score bin	# Pos	# Neg	Pop.above thresh.	Accuracy	F1	+ve Prec.	+ve Rec.(= TPR)	-ve Prec.	-ve Rec.(= 1 - FPR)	Cumulative AUC
(0.900,1.000]	10	1	0.141	0.910	0.741	0.909	0.625	0.910	0.984	0.008
(0.800,0.900]	3	1	0.192	0.936	0.839	0.867	0.813	0.952	0.968	0.019
(0.700,0.800]	1	0	0.205	0.949	0.875	0.875	0.875	0.968	0.968	0.019
(0.600,0.700]	1	1	0.231	0.949	0.882	0.833	0.938	0.983	0.952	0.034
(0.500,0.600]	0	2	0.256	0.923	0.833	0.750	0.938	0.983	0.919	0.065
(0.400,0.500]	0	2	0.282	0.897	0.789	0.682	0.938	0.982	0.887	0.095
(0.300,0.400]	0	1	0.295	0.885	0.769	0.652	0.938	0.982	0.871	0.110
(0.200,0.300]	1	0	0.308	0.897	0.800	0.667	1.000	1.000	0.871	0.110
(0.100,0.200]	0	7	0.397	0.808	0.681	0.516	1.000	1.000	0.758	0.223
(0.000,0.100]	0	47	1.000	0.205	0.340	0.205	1.000	1.000	0.000	0.981

Out of the 78 points to predict, AzureML only differed on 3 points (about a 4% change) and thus the accuracy is 96.2%. Again, this does not mean the AzureML prediction is highly accurate at predicting the correct outcome. Here, this means that the resulting decision of the boosted decision tree in AzureML is very similar to that of the Excel tool.

What does this imply about AzureML? AzureML, although in a developmental state, was able to build and test a model from data and then apply that to a new dataset to make a decision off of. It does this quickly and with a few drag-and-drop motions. Plus, Excel does not automatically compare models. Although SQL Server Analysis Services can be used by someone with T-SQL skills to compare outputs of the analyses, this is generally a hands-on process left up to the analyst. In AzureML, the evaluation of two models is easy to see with one click. Plus, AzureML has R capabilities built-in to run more advanced calculations and predictions. Once the analysis is complete, AzureML allows the user to publish a web service for automatic analysis. For this example, a store could continually collect customer information as new customers fill out loyalty card forms or make purchases online, for example. Then, the store could have an Azure web service that automatically classifies the customer into "Bike Buyer" or "Not a Bike Buyer" for better targeted marketing and insight.

Ideas for Improvement of AzureML

Although Microsoft has done a great job of getting this system to the shape it is currently in, there are still a few things I would like to see.

- Limits as to what you can see from your data.
  - You have to know what your data looks like before you put it into Azure. Otherwise, the *Visualize* feature doesn't show you everything. Even if you are using the *Evaluate Model* node, you can tell which model is which in the comparison. In short, there's no "Excel table" view of your data.
- Outputs from Regression models are invisible.

  - If you are used to seeing the $\hat{y} = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$ format for a regression output, you will be disappointed as AzureML does not show you this output.
  - Clusters and decision trees are also mildly cryptic. In other mining and analysis tools, the user can create graphs and attempt to understand what the clusters or trees are actually representing. In AzureML, you're left with a bland cluster 1, cluster 2, etc. or a simple output where the data went through the tree, but no knowledge of the levels that it had to progress through to get to that point.
    - Take a decision tree, for example. In the Excel Data Mining add-in as well as SAS and SPSS, the user will get a visual output that shows the actual tree the data is put through. In order to do this in AzureML, the user must write an R script. To get an output that is up to the quality that you would get from other tools, you might need up to 7 basic R packages (rpart, rpart.plot, rattle, RColorBrewer, party, partykit, and caret). Only 4 of these 7 are in AzureML. So, the user is stuck with a text-based output only. Also, more sophisticated packages (like prp, rx.tree, and fancyRpartPlot) that can generate nice-looking trees in a single line of code are not included either.
- R implementation is not as easy as it's made out to be
  - R in itself is not an easy feat. AzureML allows you to add R scripts into your data analysis process. However, it is often difficult to reference the data that you have imported and produce a viewable output without practice. Plus, you cannot add other R packages other than what is pre-installed. If you have something custom, you're out of luck. Currently, there are over 6100 packages available for R. AzureML only includes about 410 of them; thus certain scripts are not doable given that the user needs a package that isn't available (as in my previous point about regression outputs). It is my understanding that users of AzureML are expected to use R to fill in the gaps in the system or to make custom analyses. This is sure to improve as more R packages are available in AzureML.
    - *Note: In attempting to run the analysis on the Bike Buyer data, I attempted to use R to generate a decision tree and classify the customers as AzureML does on its own with the proper, pre-built node. I failed to get the R script to run properly. However, I plan to write more about using R in AzureML later as it unlocks tons of features that are not built in to AzureML "out of the box" despite its shortcomings.*

About the Author

Colby Ford, Associate Data Analytics Consultant

Colby Ford is a data scientist at Mariner focusing on statistical modeling and data visualization. His educational background is in mathematics and statistics and now onto data science and business analytics. Prior to joining Mariner, Colby worked in education and in healthcare, holding positions over database systems, technology, and in instruction. His independent research interests are in bioinformatics and genomic data analysis as it relates to disease prevalence and other concepts in biology.

Connect with Colby

LinkedIn: www.linkedin.com/in/colbytylerford
Twitter: @colbytylerford
Website: www.colbyford.com

About Mariner

If you're looking for a business partner that can harness all types of information to enhance customer experiences, optimize internal processes or create new business models, then look no further. Leveraging big data, cloud, social, predictive analytics, Internet of Things, decision management and mobile technology, Mariner quickly draws a target on the results you want and then delivers your digital business solution.

Microsoft Partner
Gold Business Intelligence
Silver Data Platform

Geared toward helping the leaders of mid-market companies, we help you drive automated decision making and empower decision makers with actionable, trusted information. for your company, your department and you. Mariner is a Microsoft Certified Gold Partner with competencies in Data Analytics, Data Platform and Intelligent Systems Service. Mariner-Insight to Achieve. For more information about our Digital Business solutions, visit http://www.mariner-usa.com/digital-business. For more information about Microsoft Azure Machine Learning, visit http://azure.microsoft.com/en-us/services/machine-learning/.

Copyright Information