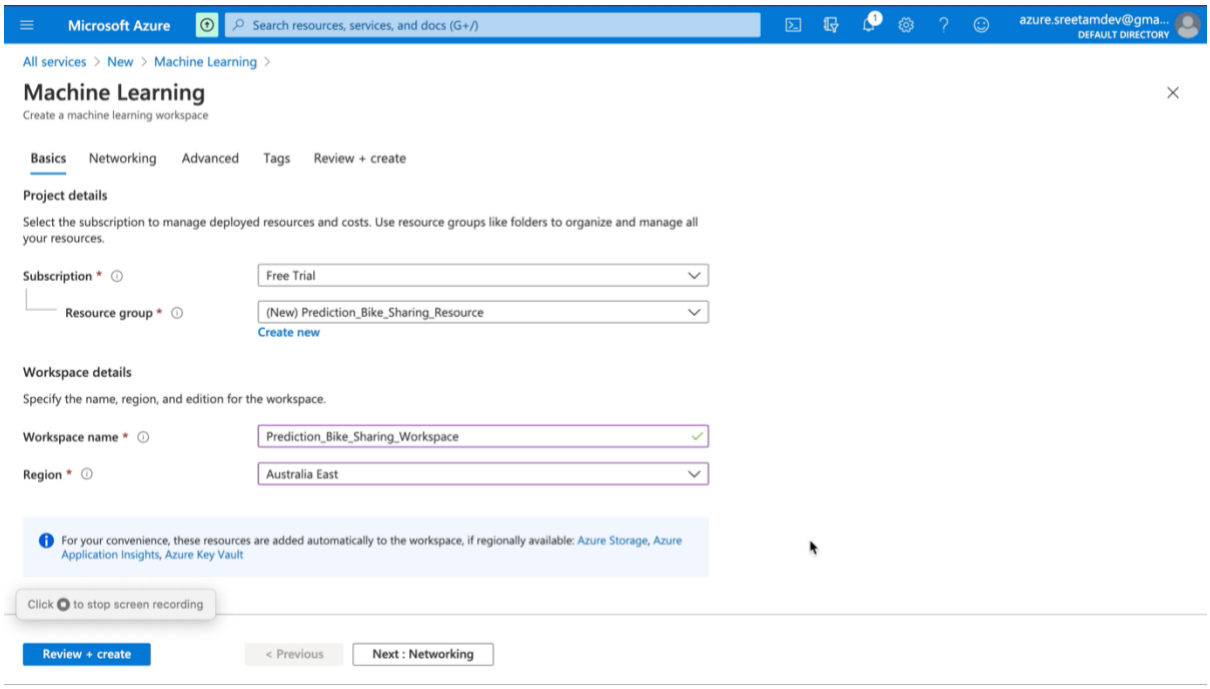
[AZURE ML PERFROMED ON BIKE SHRAING DATA]

By Sreetam Dev

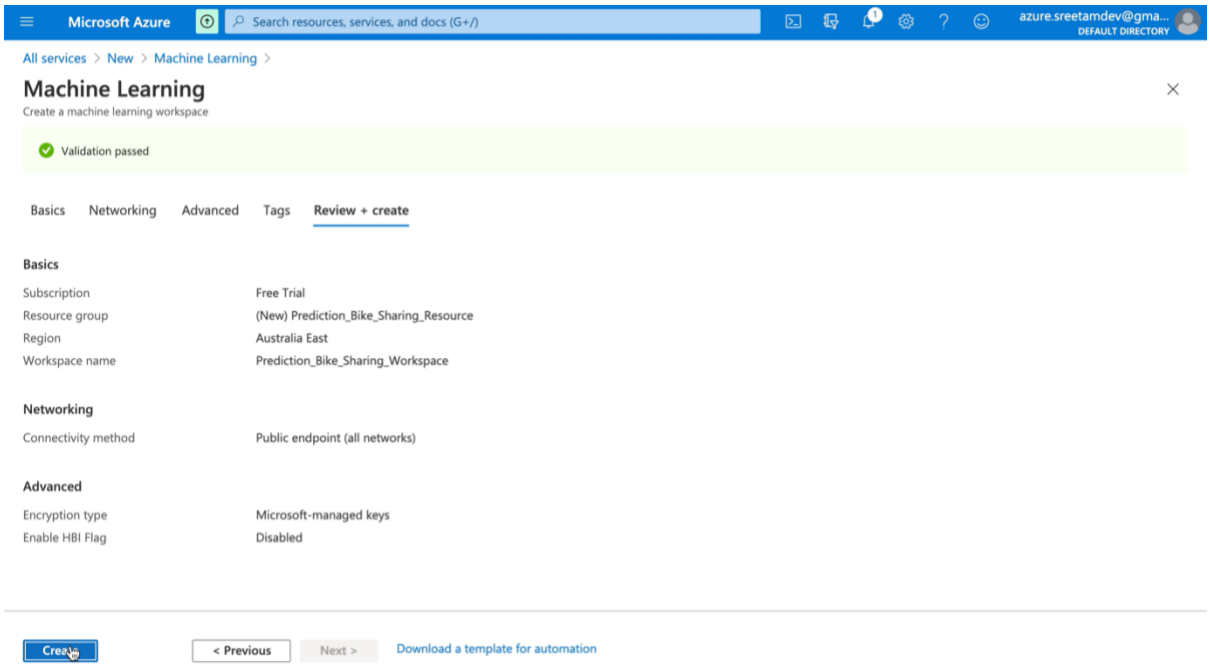
[sreetamkumardev@gmail.com]

Table of Contents

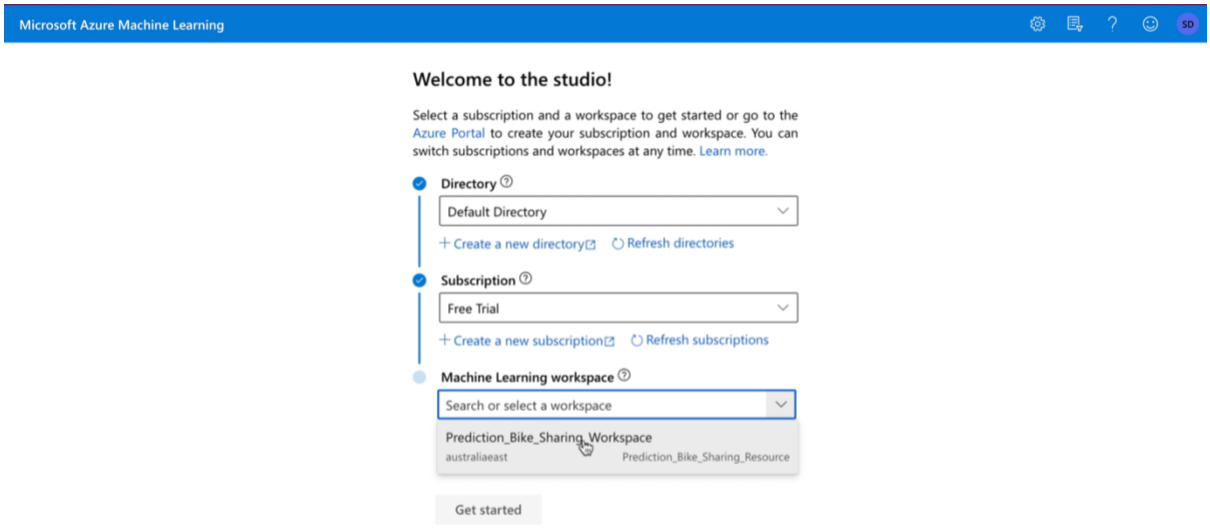
1.Setting up the resource group, workspace for the bike share data.



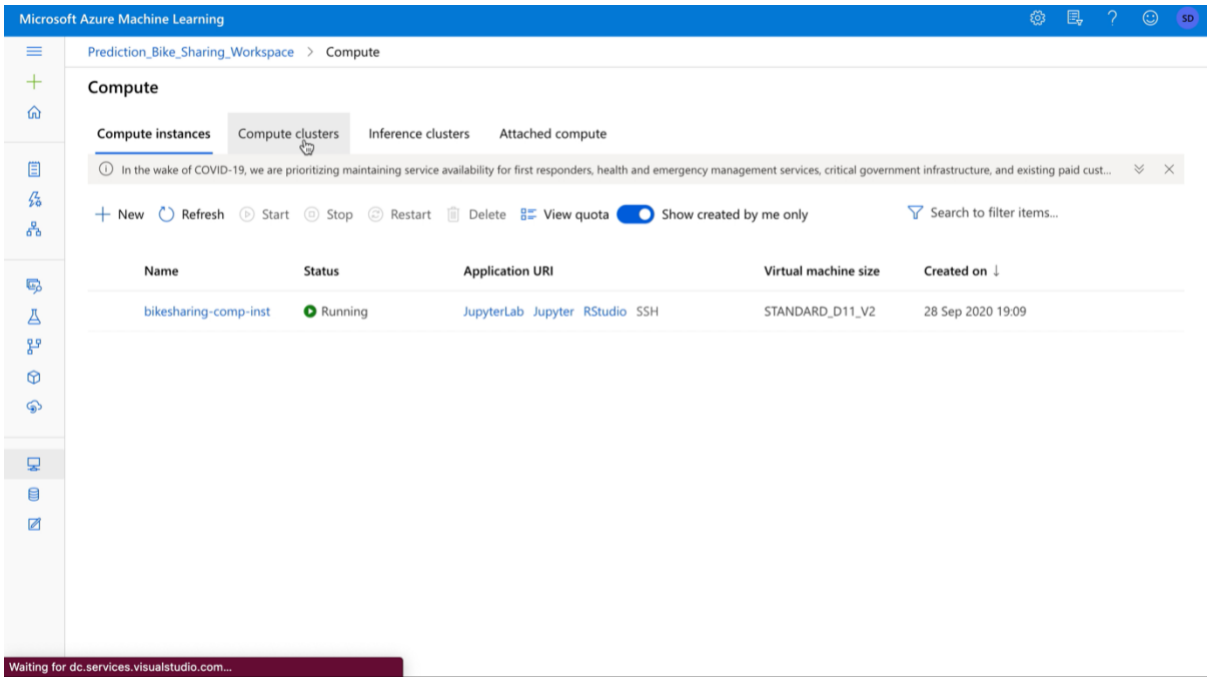
2.Reviewing and creating the workspace for the data.



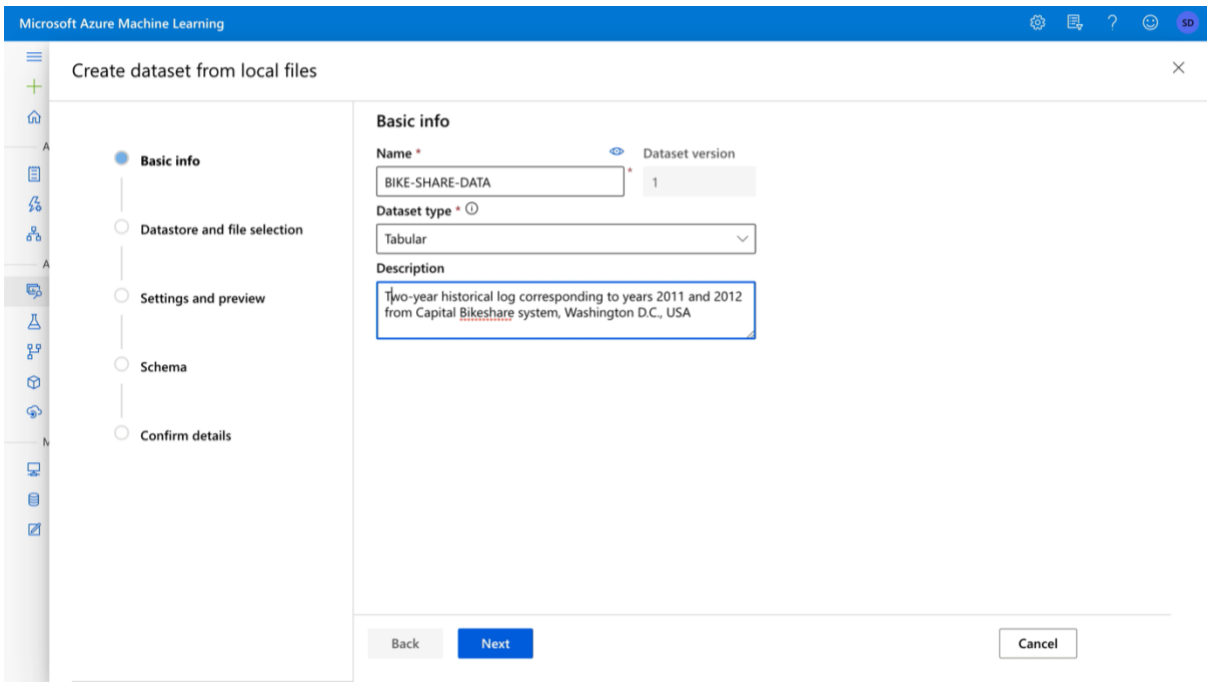
3.Setting up the Azure ML studio.



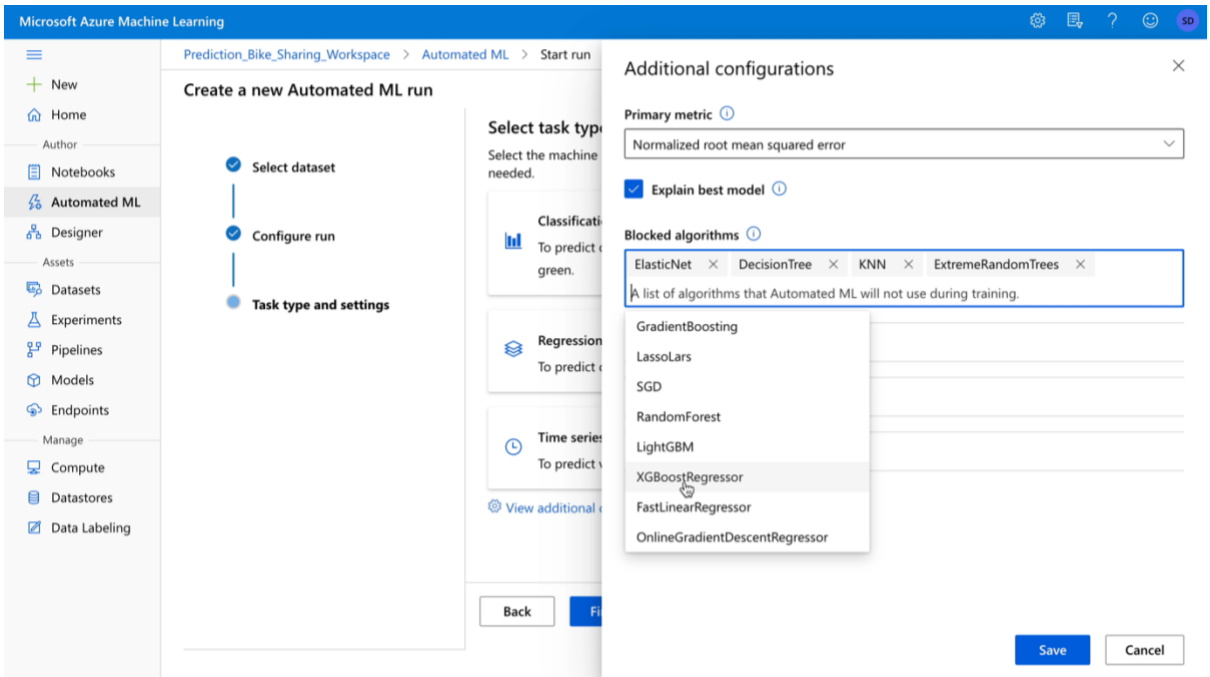
4.Setting up the compute instances, compute clusters, inference clusters before proceeding to data exploration, modelling and deployment.



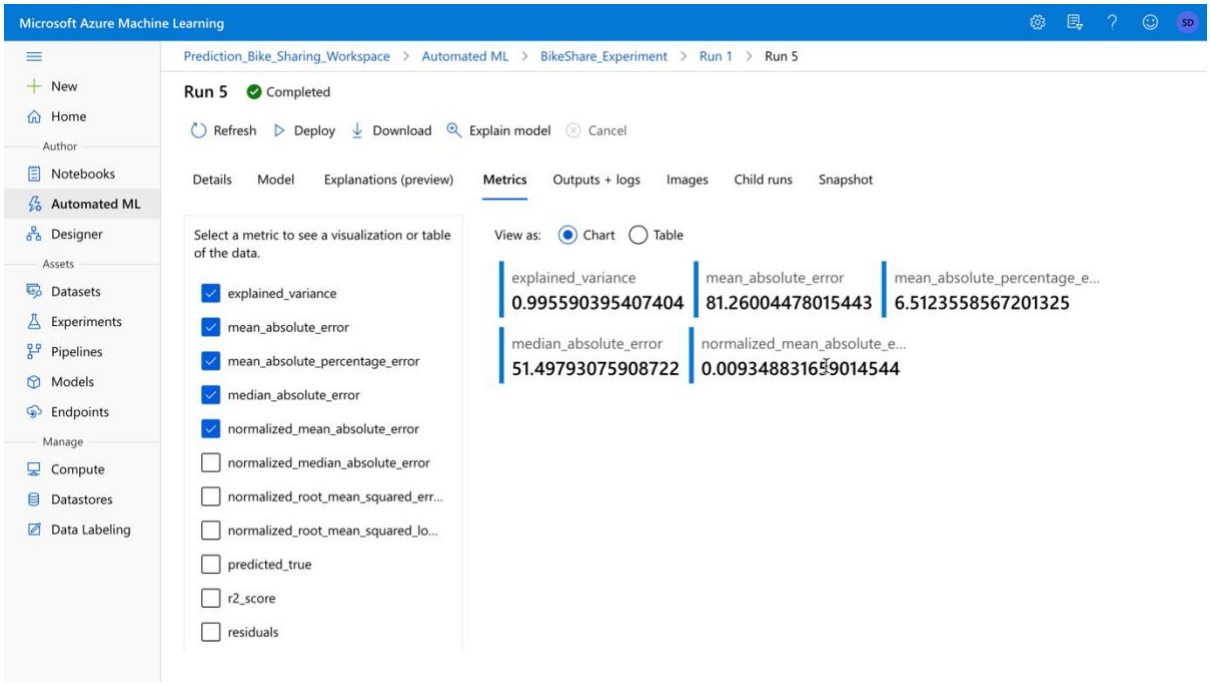
5.Setting up the dataset information, data storage requirements and creating the BIKE-SHARE-DATA dataset.



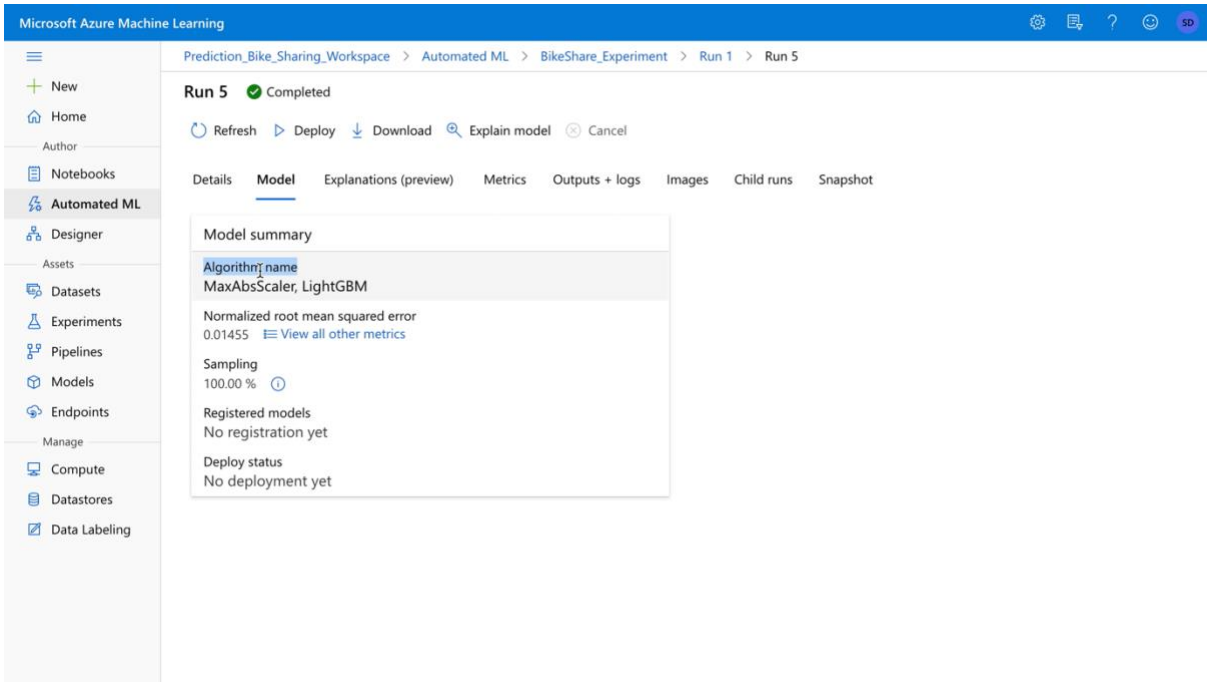
6.Creating Automated ML environment for finding the best model and validation metrics for the dataset . Here, the metric for validation is chosen to normalized root mean squared error. And also the exit strategy for the modelling can be fine-tuned to save computation resource usage and time.



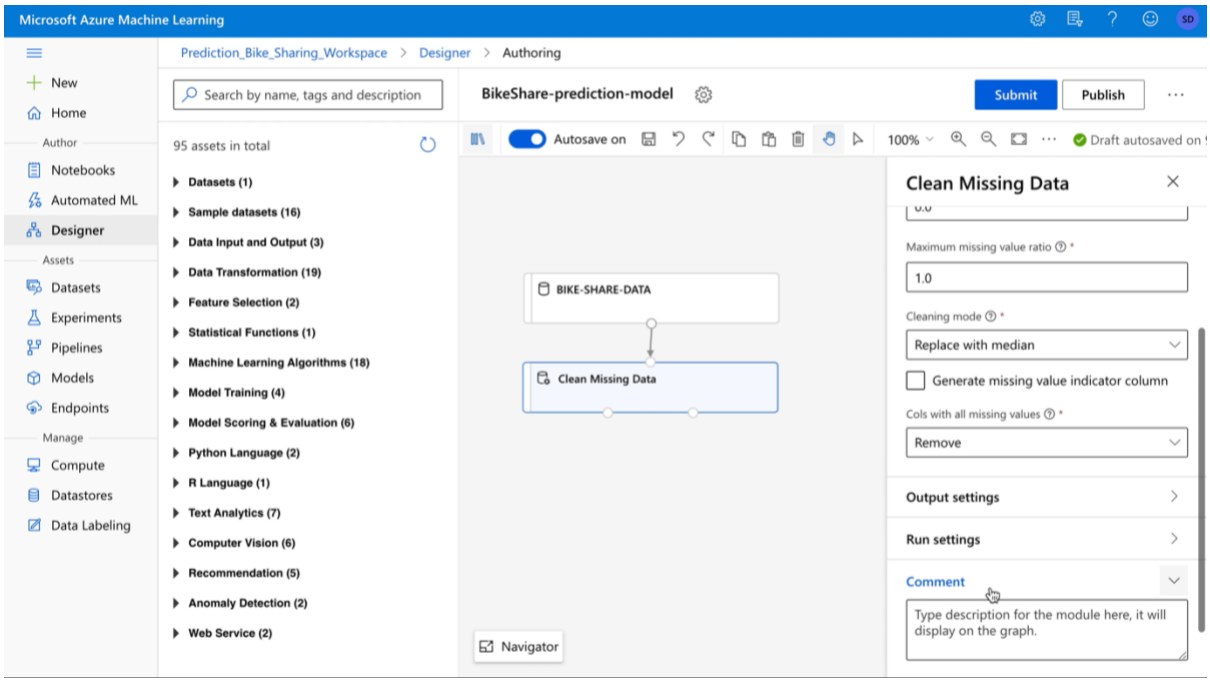
7.Examining the end results in terms of metrics, of the processed Auto ML environment.



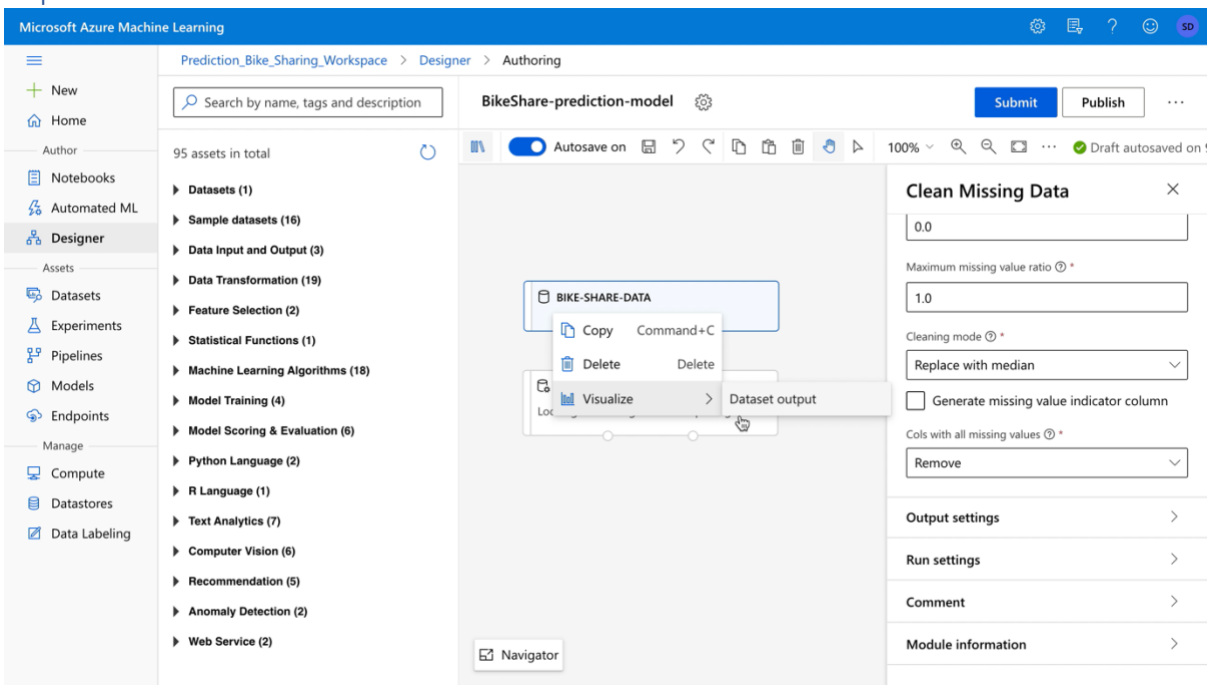
8.The best metric returned, was performed by the following model. Now, this differs from the result which we achieved on the Jupyter notebook as the modelling performed over here has not undergone feature selection.



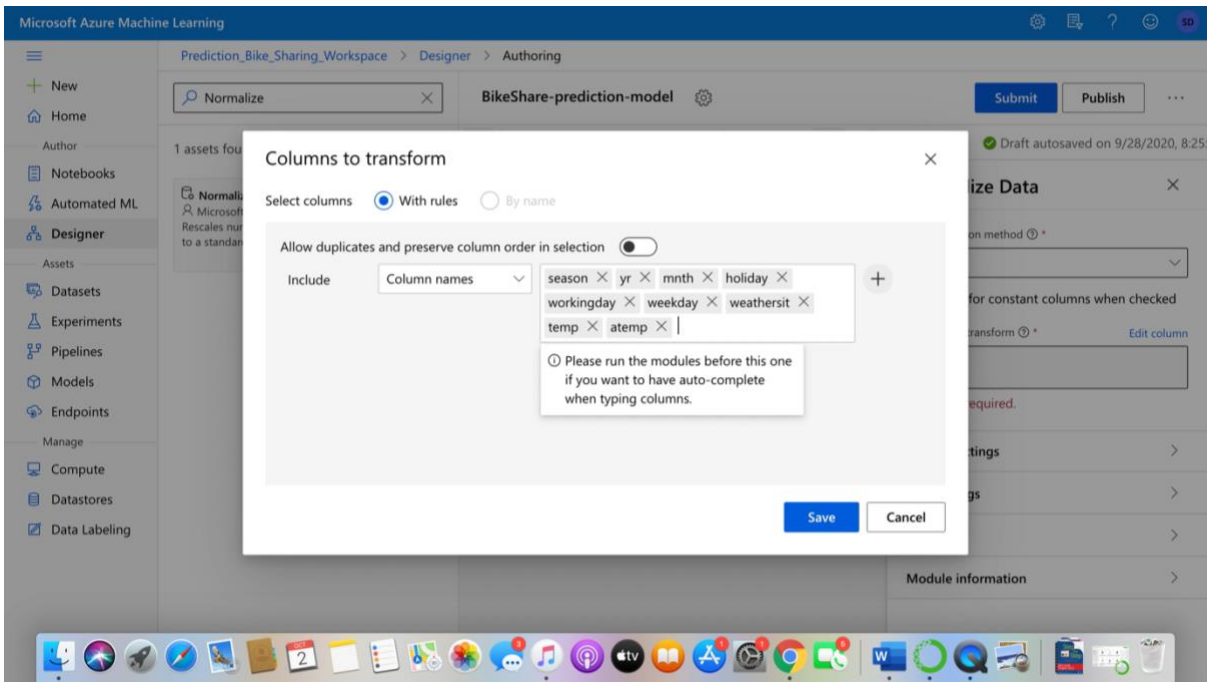
9.Now, we are exploring the designer module of the Azure, where can create the data pipeline based on our earlier findings from the Jupyter notebook data script for the “Bike Share Data”. We need to drag and drop methods defined on the left hand side into the workspace for each implementation that we need to execute. For e.g. Here we have “Cleaning Missing Data method” for handling missing instances.



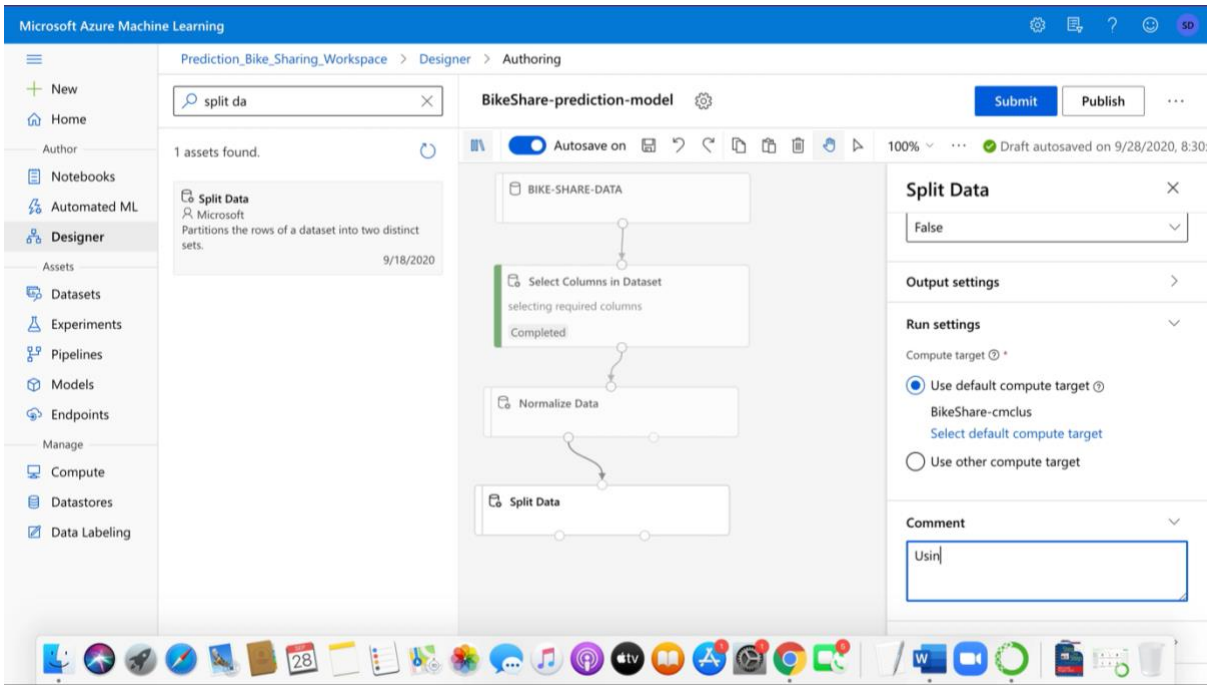
10.Visualization on the data file is performed to explore if the changes have been implemented or not.



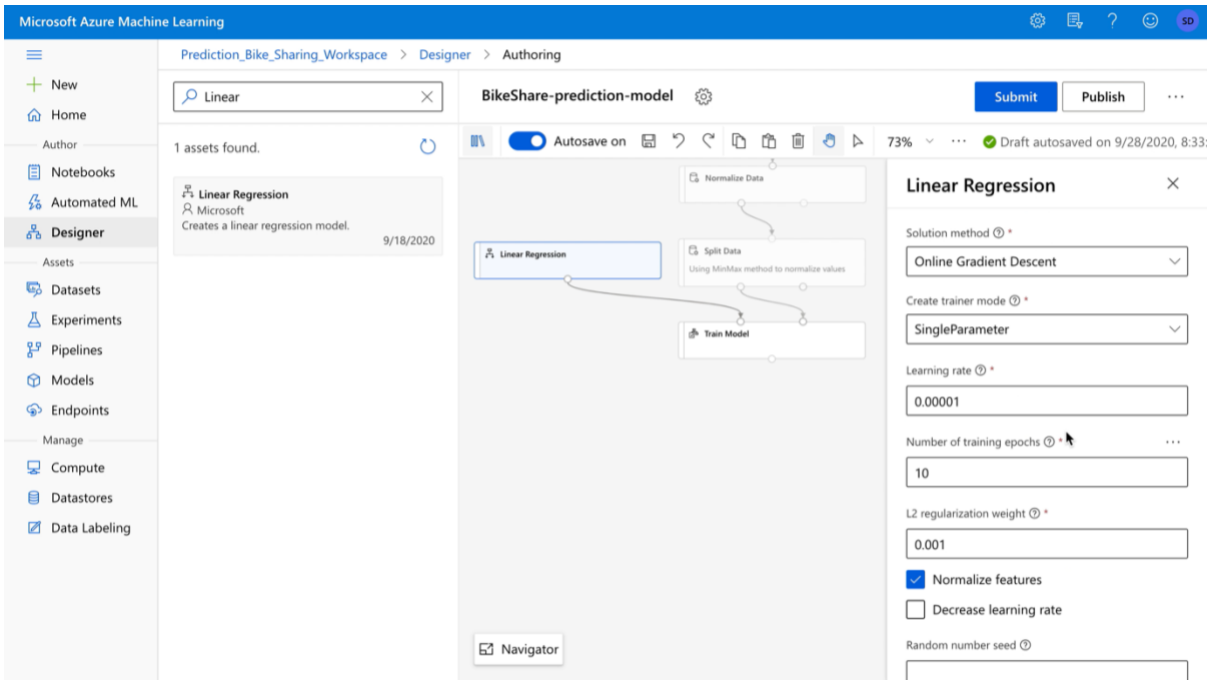
11. Columns are being selected to handle instances of features that are important, as discovered from the Jupyter data script for our predictive modelling .



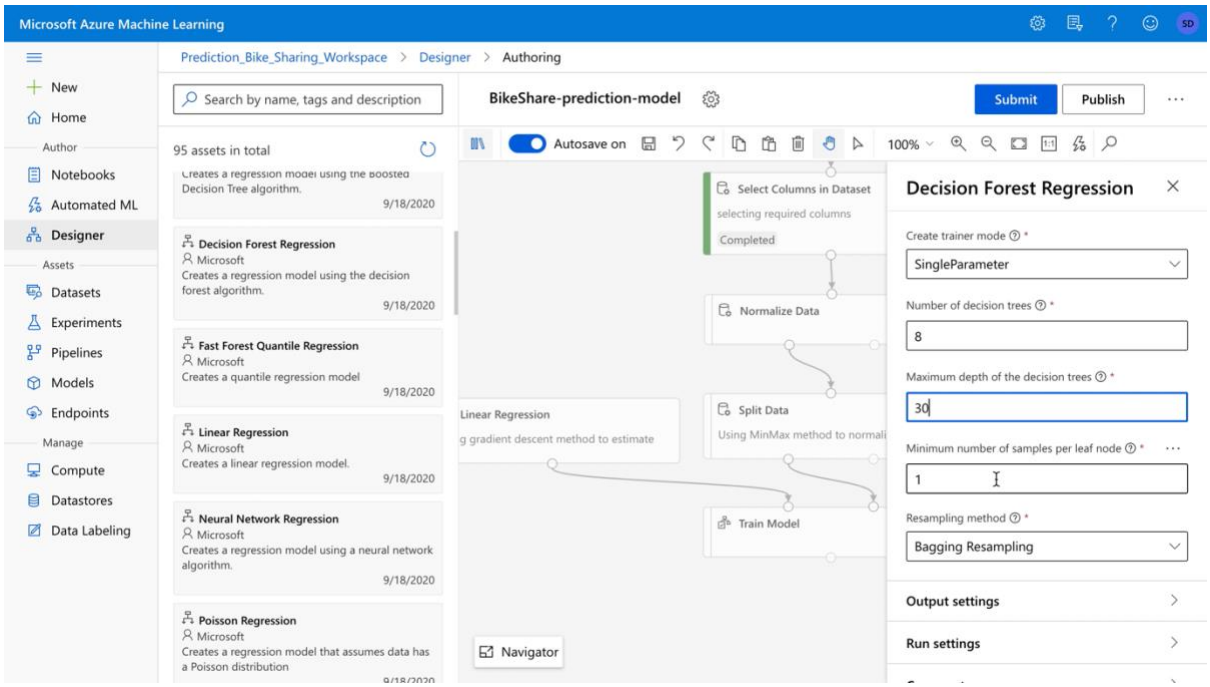
12.Data normalization is performed, to reduce weightage of any influencing feature that might bias the predictive modelling.



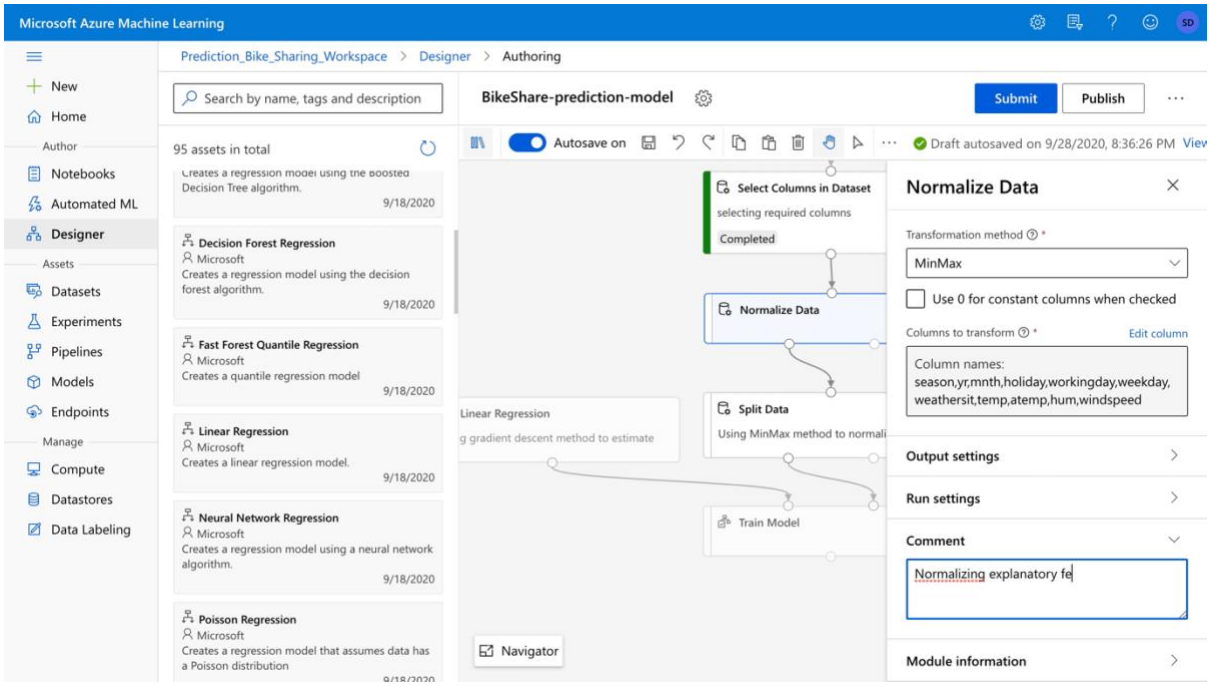
13.Linear Regression along with Train method are drawn in order to train and test the data file.



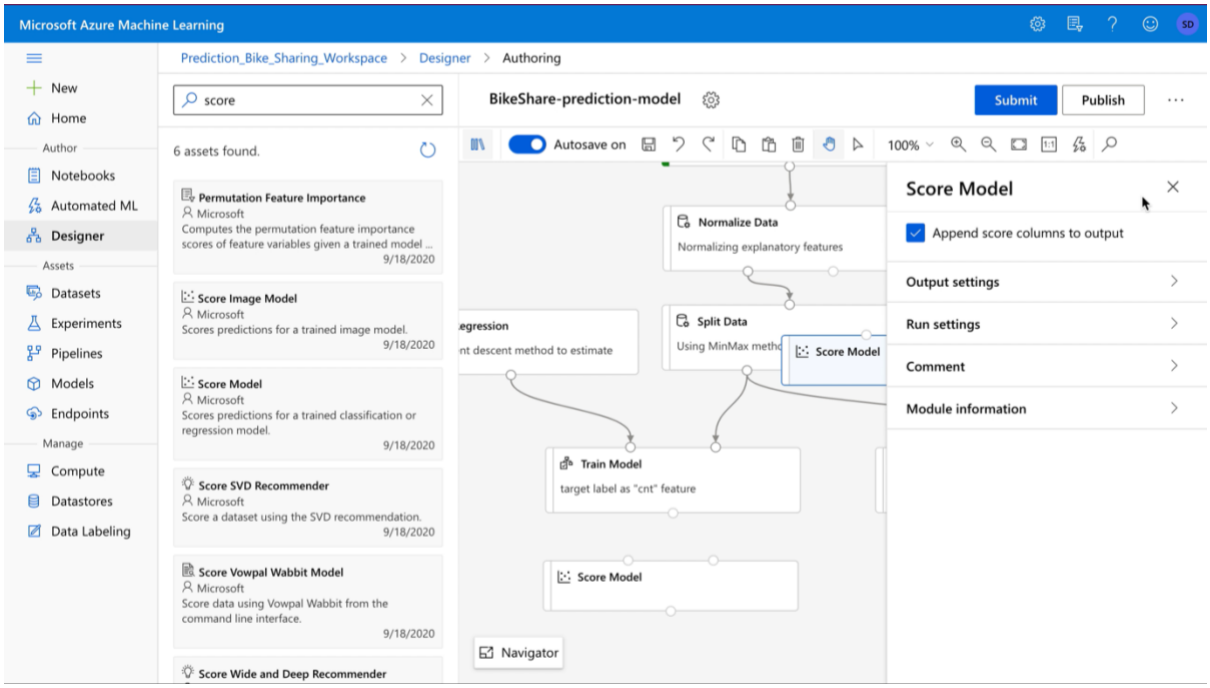
14.Fine tuning the hyper parameters of the Decision Forest Regression model to achieve better prediction results.



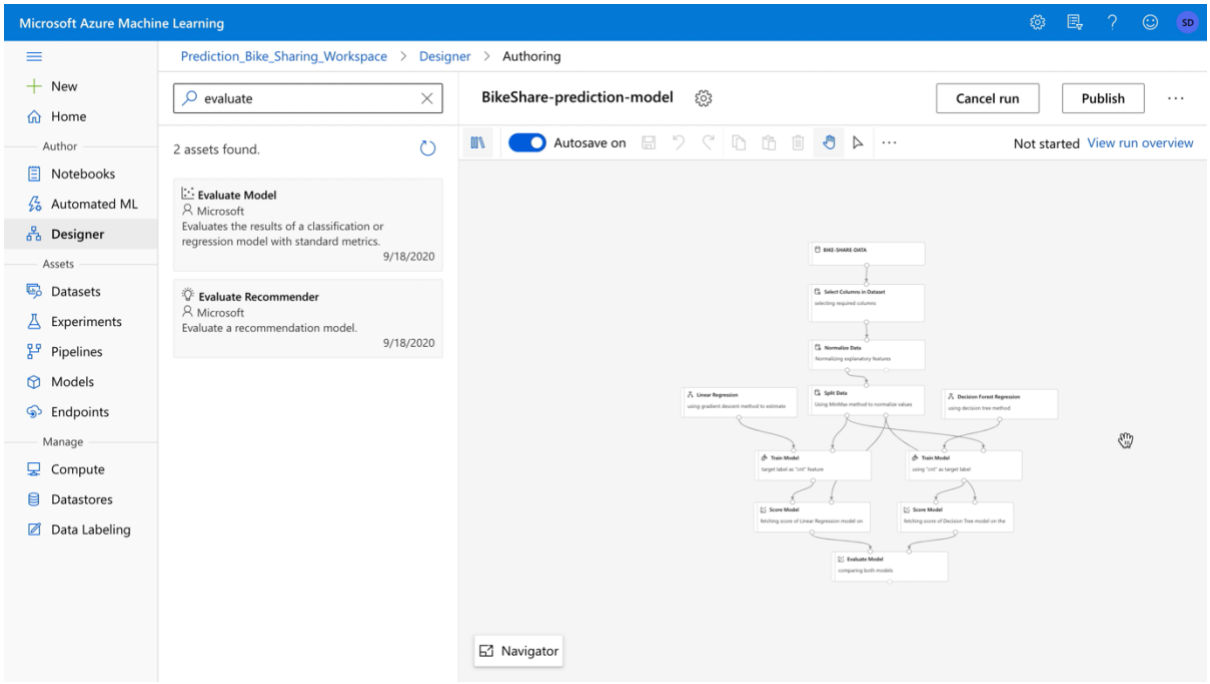
15.Altering the previous Normalizing method to MinMax for data scaling.



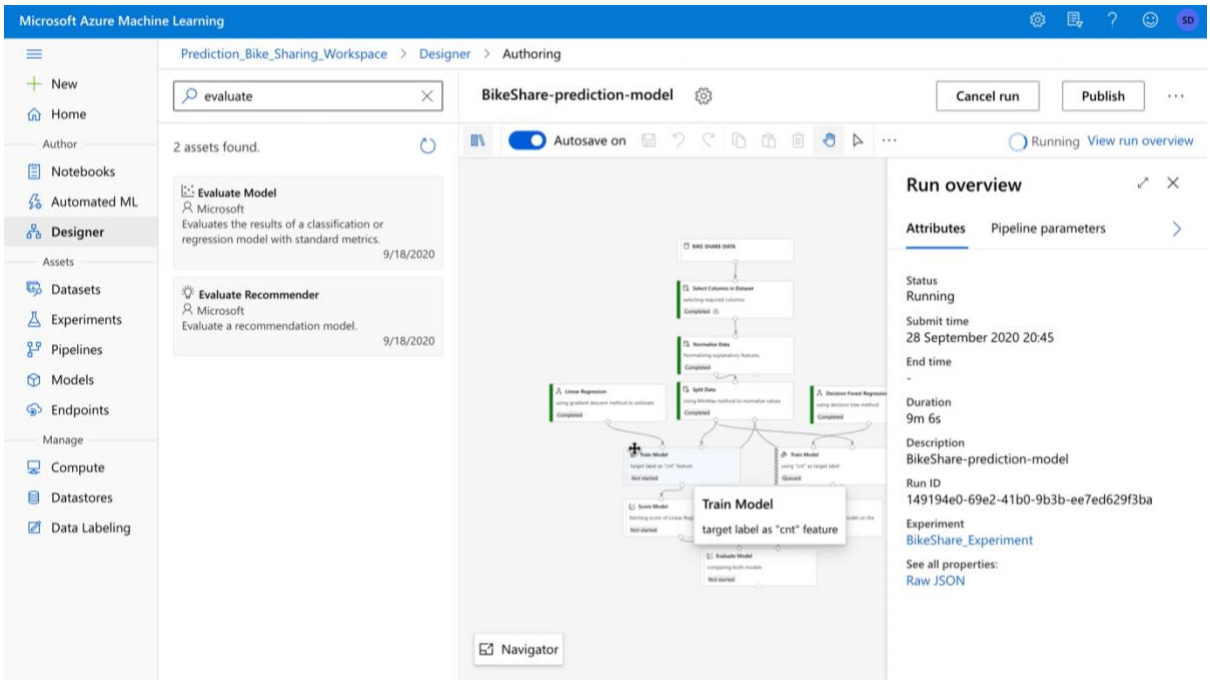
16.Score Model is being fined tuned in terms of setting.



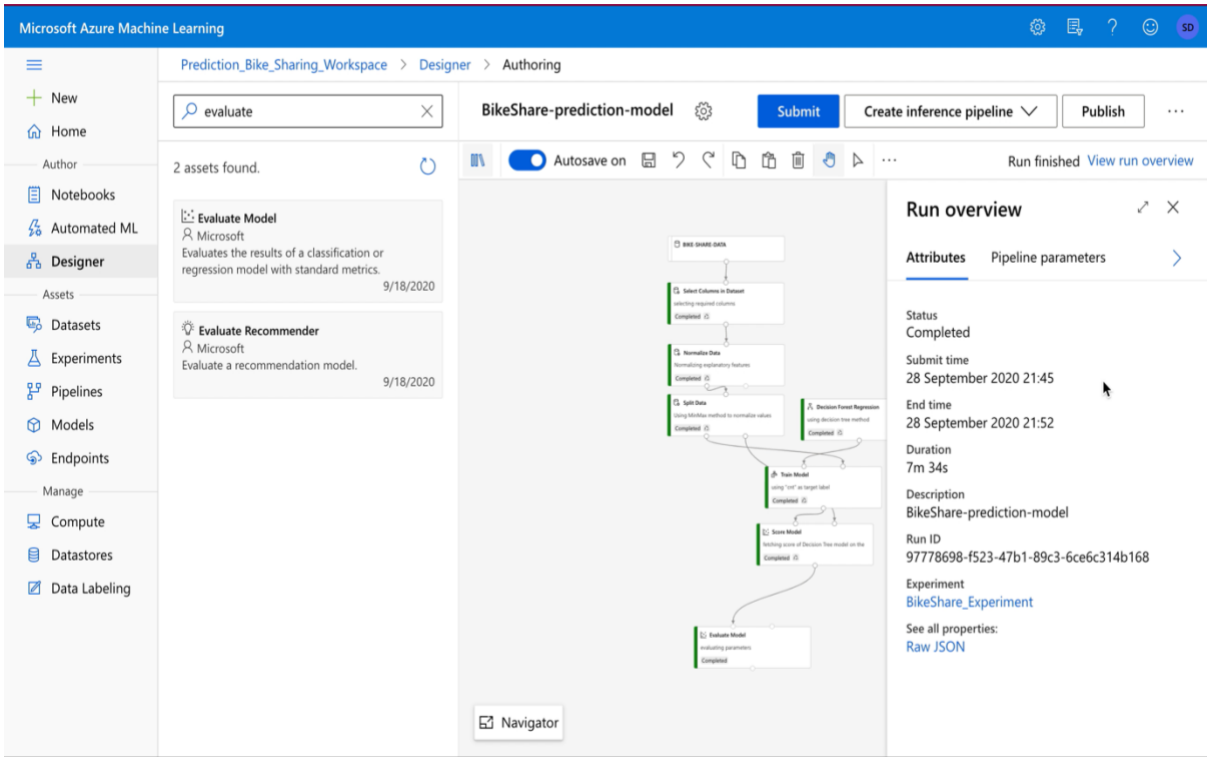
17.The similar process is repeated with Decision Forest Regression to compare both the outcomes in terms of metrics to fine tune our modelling process. At the end nodes we are using Score Model and Evaluate model methods to draw outputs for both modelling techniques.



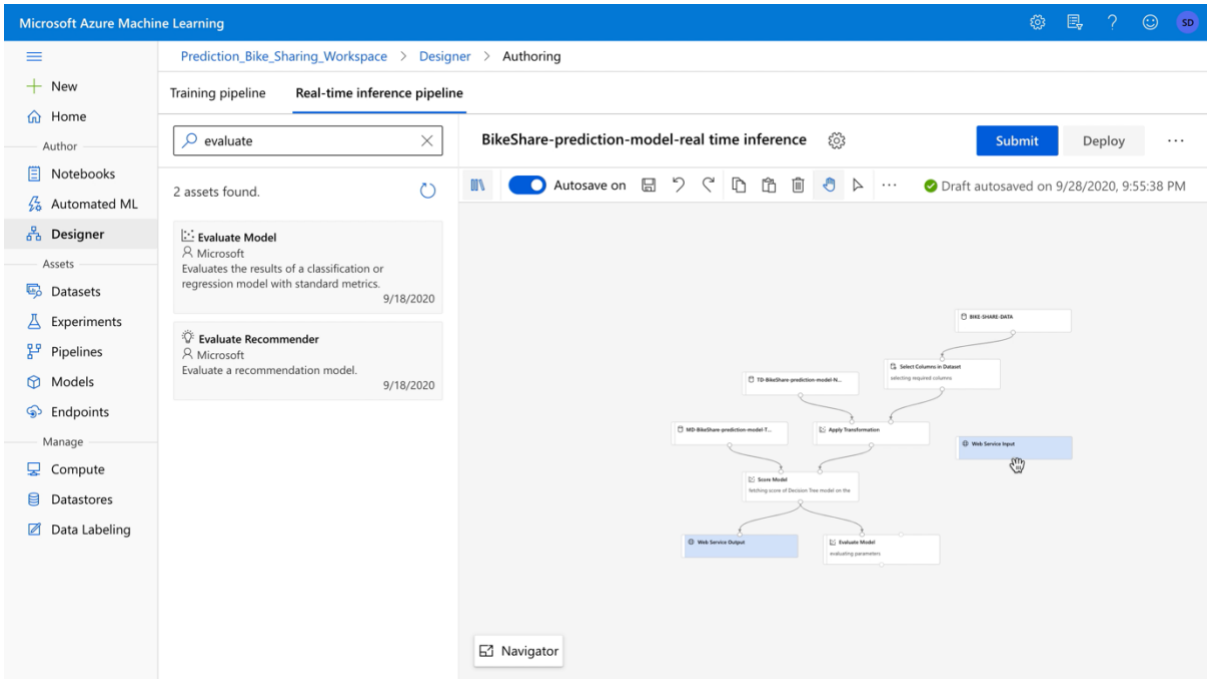
18.The model is being processed to fetch results in terms of prediction and metrics to evaluate the model.



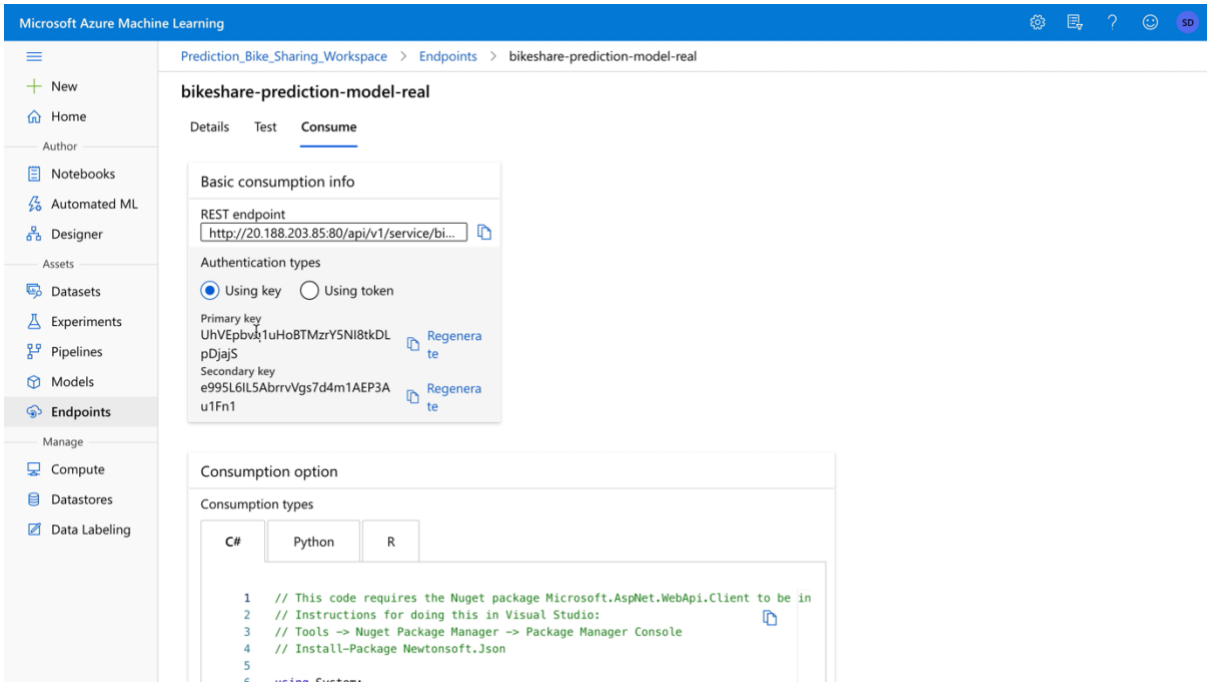
19.Since Decision Forest Regression model performed better, the inference pipeline is being created as the model we be prepared for deployment.



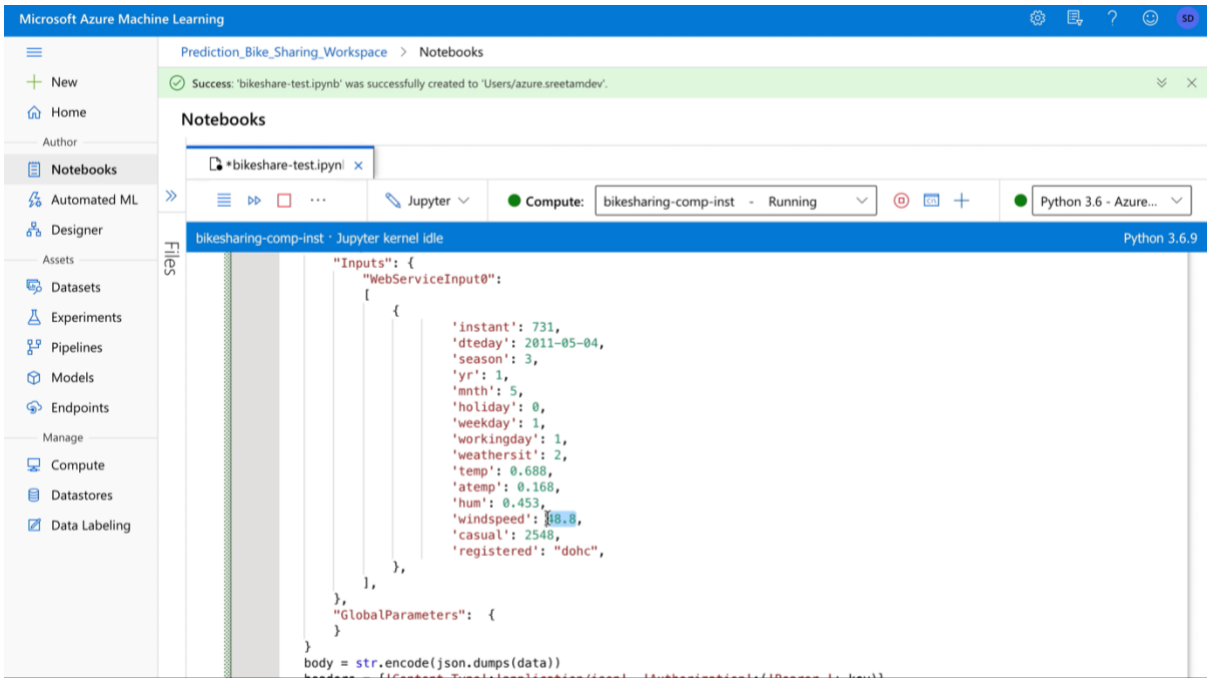
20.The inference pipeline has been created and the web service input node is being allocated to the pipeline so that user can provide input after deployment and model will fetch the predictive results for the input instance provided.



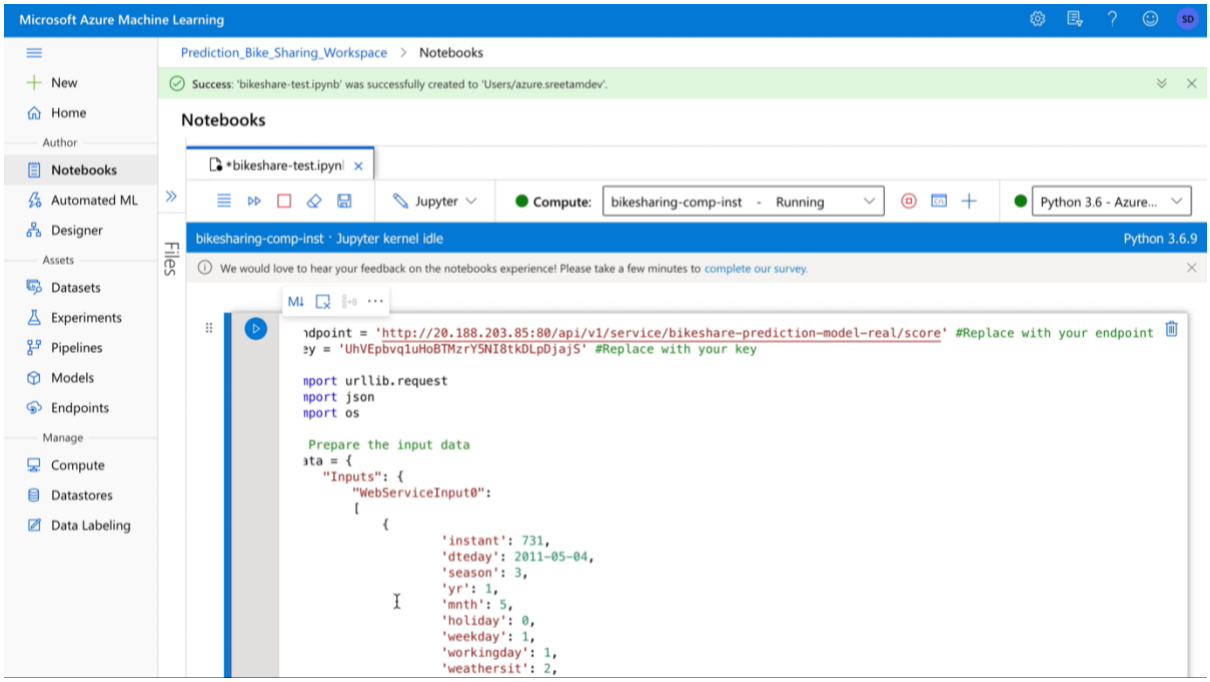
21.The endpoints are being defined for the web api for the prepared data model.



22.A separate data output instance is being designed for the model stored within the notebook to be displayed by the web api.



23.The endpoint and key of the default practise script of Azure ML is being replaced with our Endpoint's.



24.Once the environment is ready, the output python script is connected and submitted. Once processed it is deployed.