

**National Research University
“Higher School of Economics”**

Graduate School of Business

Report

**Practice 1. Bike Share Demand Forecasting with MS Azure
Machine Learning**

Area of studies: Business Informatics

Master's programme: Business Analytics and Big Data Systems

Completed by student: Akmal Kholikulov

Group number: BASB 212

Instructor: Yulia Morozova

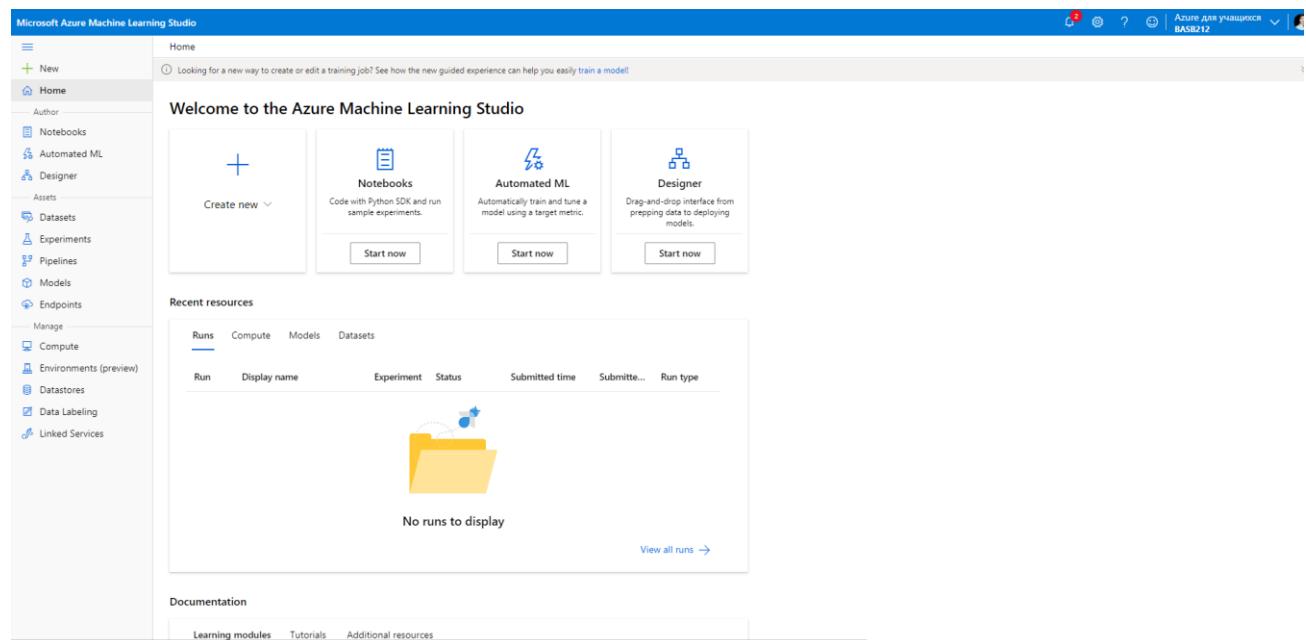
Automated Machine Learning

BikeShare Demand Forecasting

Content

- Create and load a dataset.
- Configure and run an automated ML experiment.
- Specify forecasting settings.
- Explore the experiment results.
- Deploy the best model.

Step 1. We need to register in MS Azure create an account, an Azure Machine Learning workspace and so on.



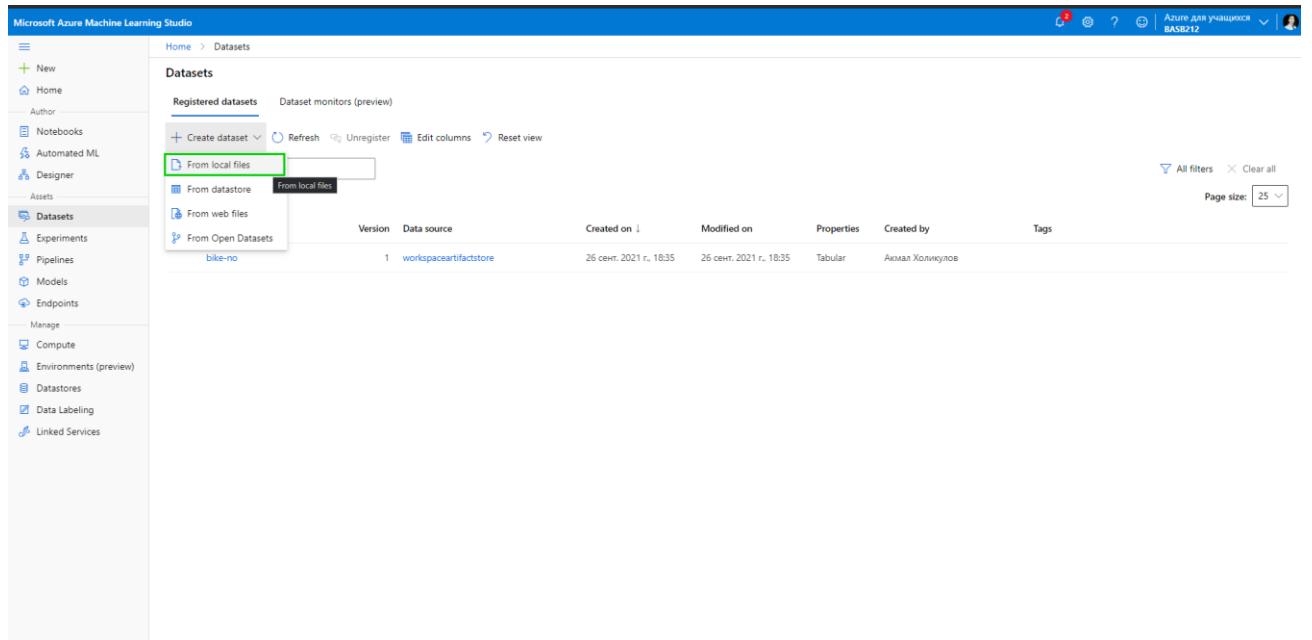
Step 2. Download datasets from [github](#). In order to download dataset press “Alt” + Click on “Raw” simultaneously. (file will be downloaded in [.txt](#) format)

A screenshot of a GitHub repository page for 'MachineLearningNotebooks / how-to-use-azureml / automated-machine-learning / forecasting-bike-share / bike-no.csv'. The page shows the file's content as a table with 15 rows of data. A green box highlights the 'Raw' button in the toolbar above the table. The URL at the bottom of the page is <https://github.com/Azure/MachineLearningNotebooks/raw/master/how-to-use-azureml/automated-machine-learning/forecasting-bike-share/bike-no.csv>.

Step 3. After downloading datasets. Go back to Microsoft Azure Machine Learning Studio and Click on “+ Create dataset”

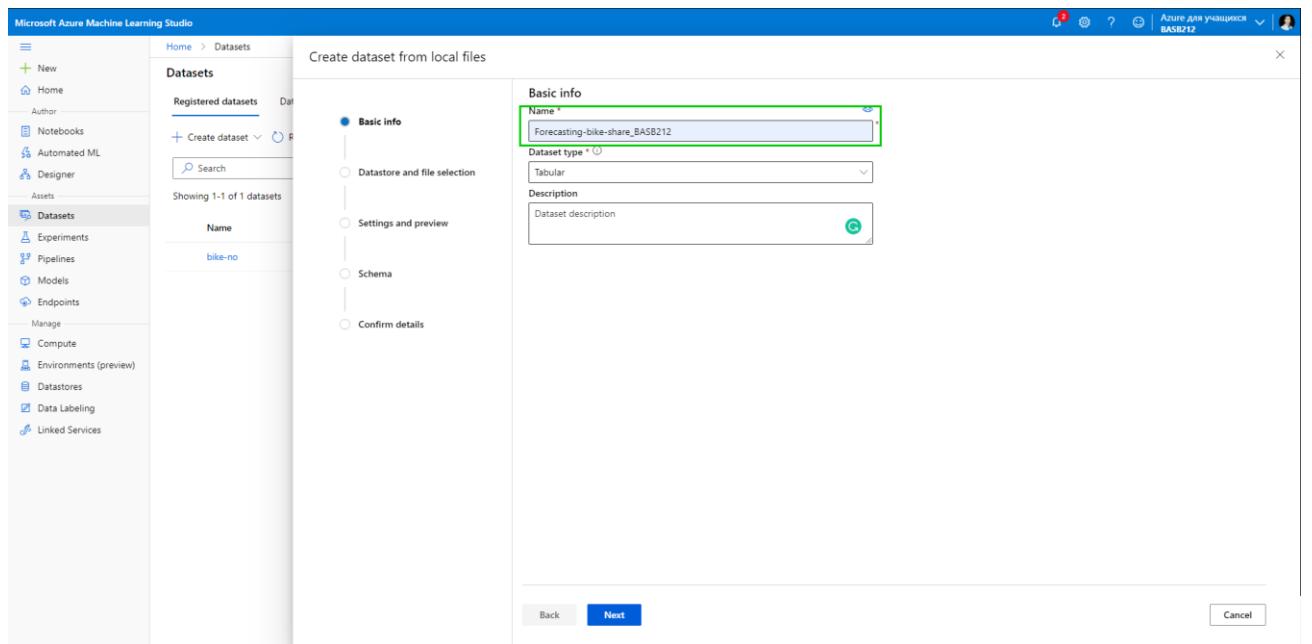
A screenshot of the Microsoft Azure Machine Learning Studio interface, specifically the 'Datasets' section. The left sidebar shows various studio components like Notebooks, Automated ML, Designer, and Assets. The 'Datasets' section is selected and highlighted with a green box around its name. The main area shows a table of registered datasets, with one entry named 'bike-no'. A green box highlights the '+ Create dataset' button in the top navigation bar of the datasets section.

Step 4. After then, click on “From local files”



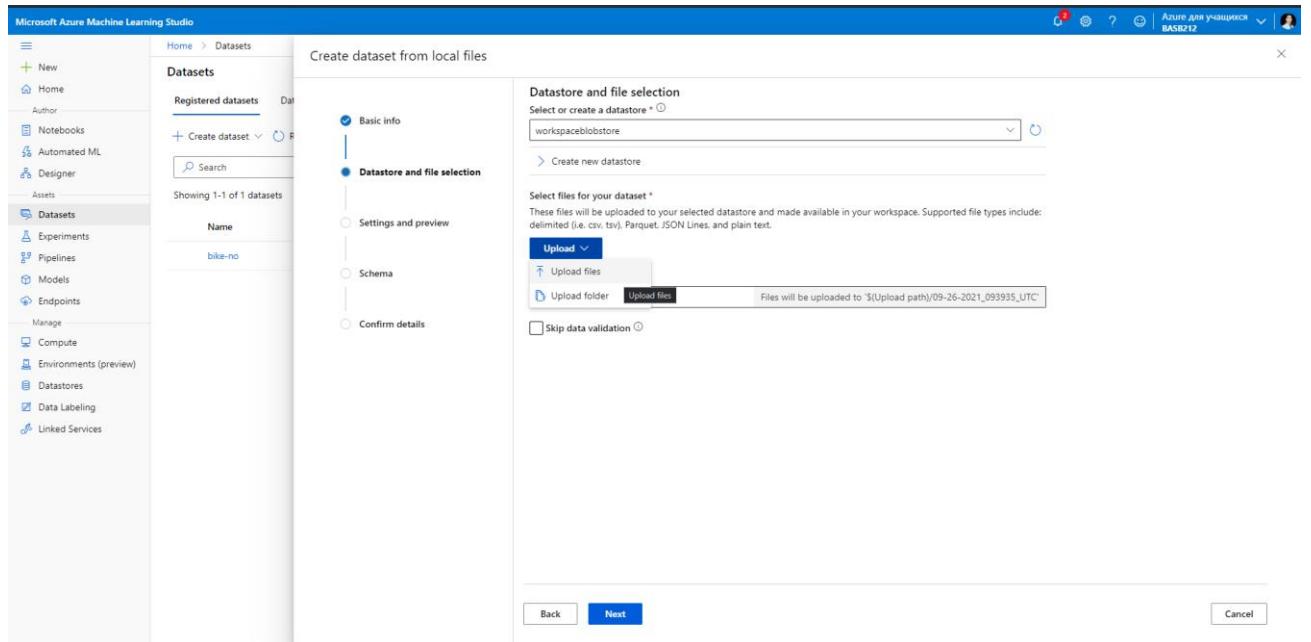
The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with various options like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets' (which is selected and highlighted in blue), 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The main area is titled 'Datasets' and shows a table of registered datasets. One dataset, 'bike-no', is listed with a version of 1, data source 'workspaceartifactstore', created on '26 сент. 2021 г., 18:35', modified on '26 сент. 2021 г., 18:35', properties 'Tabular', and created by 'Ахмад Халиуллов'. At the top of the main area, there are buttons for '+ Create dataset', 'Refresh', 'Unregister', 'Edit columns', and 'Reset view'. A dropdown menu is open, showing options: 'From local files' (highlighted with a green box), 'From datastore', 'From web files', and 'From Open Datasets'. There are also filters and a 'Page size' dropdown at the bottom right of the table.

Step 5. Provide basic information. Text a name in raw “Forecasting-bike-share_BASB212”. Do not change dataset type, remain it “Tabular”. Then click “Next”

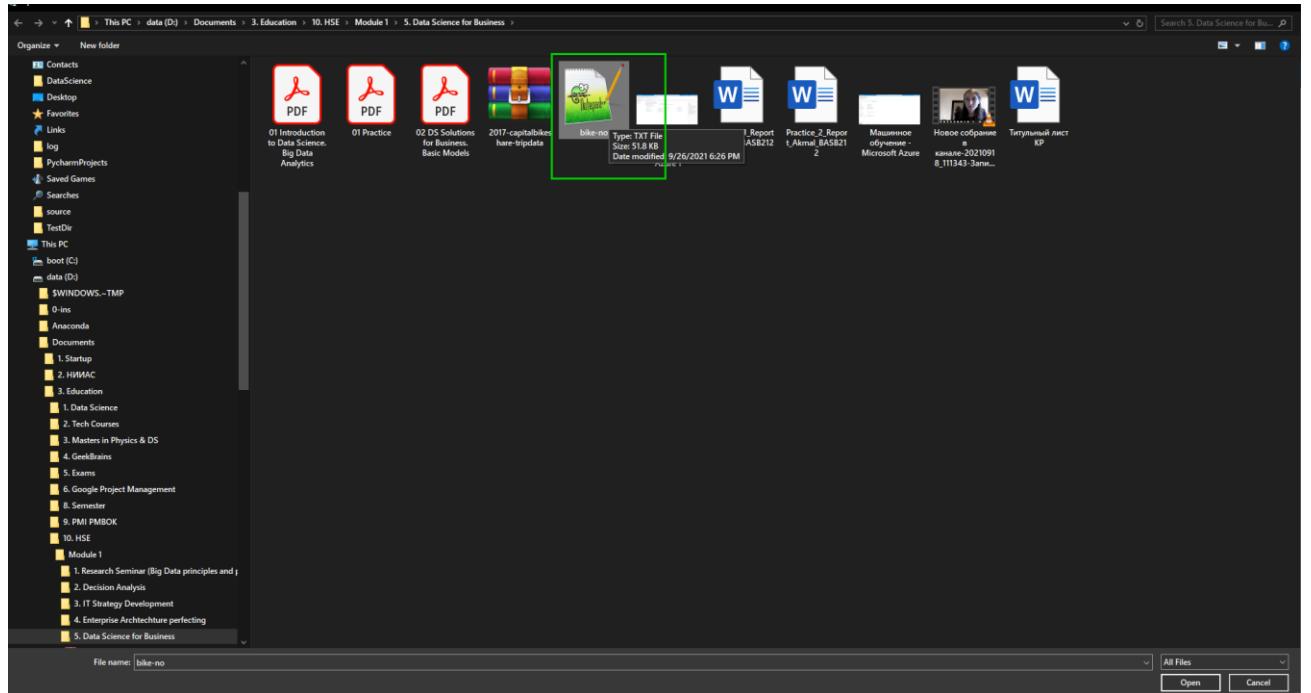


The screenshot shows the 'Create dataset from local files' wizard. On the left, there's a sidebar with 'Datasets' selected. The main area has tabs: 'Basic info' (selected and highlighted with a green box), 'Datastore and file selection', 'Settings and preview', 'Schema', and 'Confirm details'. Under 'Basic info', there's a 'Name' input field containing 'Forecasting-bike-share_BASB212', a 'Dataset type' dropdown set to 'Tabular', and a 'Description' input field. At the bottom, there are 'Back', 'Next', and 'Cancel' buttons.

Step 6. To upload a file, click on “Upload” then “Upload files”. Then click “Next”



Step 7. Choose the downloaded dataset in TXT format “bike-no”. Then click “Open”



Step 8. We see that dataset “bike-no.txt” is uploaded. Then click “Next”

The screenshot shows the 'Create dataset from local files' wizard in Microsoft Azure Machine Learning Studio. The left sidebar shows the 'Datasets' section with various options like 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Experiments', 'Pipelines', 'Models', and 'Endpoints'. The main panel is titled 'Create dataset from local files' and has a left sidebar with steps: 'Basic info', 'Datastore and file selection' (which is selected), 'Settings and preview', 'Schema', and 'Confirm details'. In the 'Datastore and file selection' step, it says 'Select or create a datastore' and shows 'workspaceblobstore'. Below that is a 'Select files for your dataset' section with a table showing 'File name: bike-no.txt, Size (MiB): 0.05067, Upload %: 100, Status: green dot'. A green box highlights this table. At the bottom are 'Back', 'Next', and 'Cancel' buttons.

Step 9. We do not need to change anything here. Just click “Next”

The screenshot shows the 'Settings and preview' step of the 'Create dataset from local files' wizard. The left sidebar is the same as the previous screenshot. The main panel shows 'Basic info' and 'Datastore and file selection' steps completed. The 'Settings and preview' step shows 'File format: Delimited', 'Delimiter: Comma', 'Encoding: UTF-8', 'Column headers: All files have same headers', and 'Skip rows: None'. A note at the bottom says: 'Dataset contains multi-line data' with a note: 'Processing tabular files with multi-line data is slower because multiple CPU cores cannot be used to ingest the data in parallel. Checking this option may result in slower processing times.' Below this is a preview of the dataset data:

ID	instant	date	season	yr	mth	weekday	weathersit
1	1	2011-01-01 00:00:00	1	0	1	6	2
2	2	2011-01-02 00:00:00	1	0	1	0	2
3	3	2011-01-03 00:00:00	1	0	1	1	1
4	4	2011-01-04 00:00:00	1	0	1	2	1
5	5	2011-01-05 00:00:00	1	0	1	3	1
6	6	2011-01-06 00:00:00	1	0	1	4	1

At the bottom are 'Back', 'Next', and 'Cancel' buttons.

Step 10. We do not need to include in Schema “casual” and “registered”.

The screenshot shows the 'Create dataset from local files' interface in Microsoft Azure Machine Learning Studio. The 'Schema' section is displayed, listing columns: mnth, weekday, weathersit, temp, atemp, hum, windspeed, casual, registered, and cnt. The 'casual' and 'registered' columns are highlighted with a green border. The 'Type' column for these two columns is set to 'Integer'. Other columns like mnth, weekday, etc., have their types set to 'Integer' as well. The 'Properties' column indicates that these properties are not applicable to the selected type.

Include	Column name	Properties	Type	Format settings and example
mnth		Not applicable to selected type	Integer	1, 1, 1
weekday		Not applicable to selected type	Integer	6, 0, 1
weathersit		Not applicable to selected type	Integer	2, 2, 1
temp		Not applicable to selected type	Decimal (dot ',')	0.344167, 0.363478, 0.196364
atemp		Not applicable to selected type	Decimal (dot ',')	0.363625, 0.353739, 0.189405
hum		Not applicable to selected type	Decimal (dot ',')	0.805833, 0.696087, 0.437273
windspeed		Not applicable to selected type	Decimal (dot ',')	0.160446, 0.248539, 0.248309
casual		Not applicable to selected type	Integer	331, 131, 120
registered		Not applicable to selected type	Integer	654, 670, 1229
cnt		Not applicable to selected type	Integer	985, 801, 1349

Also do not include “Instant” column. Check the “type”. If all is okay, then click “Next”

The screenshot shows the 'Create dataset from local files' interface in Microsoft Azure Machine Learning Studio. The 'Schema' section is displayed, listing columns: Path, instant, date, season, yr, mnth, weekday, weathersit, temp, and atemp. The 'instant' column is highlighted with a green border. The 'Type' column for this column is set to 'Integer'. Other columns like Path, date, etc., have their types set to 'String' or 'Date'. The 'Properties' column indicates that these properties are not applicable to the selected type.

Include	Column name	Properties	Type	Format settings and example
Path		Not applicable to selected type	String	
instant		Not applicable to selected type	Integer	1, 2, 3
date		None	Date	2011-01-01 00:00:00, 2011-01-02 00:00:00...
season		Not applicable to selected type	Integer	
yr		Not applicable to selected type	Integer	0, 0, 0
mnth		Not applicable to selected type	Integer	1, 1, 1
weekday		Not applicable to selected type	Integer	6, 0, 1
weathersit		Not applicable to selected type	Integer	2, 2, 1
temp		Not applicable to selected type	Decimal (dot ',')	0.344167, 0.363478, 0.196364
atemp		Not applicable to selected type	Decimal (dot ',')	0.363625, 0.353739, 0.189405

Step 11. Once check again and confirm by clicking “Create”.

Microsoft Azure Machine Learning Studio

Home > Datasets

Datasets

Registered datasets

Create dataset

Search

Name: bike-no

Basic info

Datastore and file selection

Settings and preview

Schema

Confirm details

Basic info

Name: Forecasting-bike-share_BASB212_v2

Dataset type: Tabular

Datastore and file selection

Datastore: workspaceblobstore

Selected files (1): bike-no.txt

Path: U/09-26-2021_095241_UTC/bike-no.txt

File settings

File format: Delimited

Delimiter: Comma

Encoding: UTF-8

Column headers: All files have same headers

Skip rows: None

Profile this dataset after creation

Back Create Cancel

Step 12. Go to the “Automated ML” (left side of the screen). Click on “+ New automated ML run”

Microsoft Azure Machine Learning Studio

Home > Automated ML

Automated ML

Let Automated ML train and find the best model based on your data without writing a single line of code. Learn more about Automated ML

+ New Automated ML run Refresh

No recent Automated ML runs to display.

Click "New Automated ML run" to create your first run

Learn more on creating Automated ML runs

Documentation

View all documentation

Concept: What is Automated ML?

Tutorial: Create your first classification model with Automated ML

Blog: Build more accurate forecasts with new capabilities in Automated ML

Step 13. In the meanwhile, we can see our dataset by clicking to “Datasets”. We can see “Details”, “Consume”, “Explore” and so on.

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there is a navigation sidebar with various options like New, Home, Author, Notebooks, Automated ML, Designer, Assets, Datasets (which is highlighted with a green box), Experiments, Pipelines, Models, Endpoints, Manage, Compute, Environments (preview), Datastores, Data Labeling, and Linked Services. The main content area displays the details for a dataset named 'bike-no'. At the top, there are tabs for Details (selected), Consume, Explore, and Models. Below the tabs, there are buttons for New version, Refresh, Generate profile, and Unregister. The 'Attributes' section contains properties such as Tabular, Created by (Акмал Холикулов), Profile (No profile generated), Files in dataset (1), Total size of files in dataset (51.88 KiB), Current version (1), Latest version (1), Created time (Sep 26, 2021 6:35 PM), and Modified time (Sep 26, 2021 6:35 PM). To the right, there are sections for Tags (No data), Description (Click edit icon to add a description), and Data sources (Datstore: workspaceartifactstore UI/09-26-2021_032901.UTC/bike-no.txt).

This screenshot shows the same interface as the previous one, but the 'Consume' tab is selected. The 'Sample usage' section contains the following Python code:

```
# azurerm.core of version 1.0.72 or higher is required
# azurerm-dataprep[pandas] of version 1.1.34 or higher is required
from azurerm.core import Workspace, Dataset

subscription_id = 'c7fb00a7-8927-4acd-b230-723208b5e933'
resource_group = 'HSE_B458_Akmal'
workspace_name = 'BASB212'

workspace = Workspace(subscription_id, resource_group, workspace_name)

dataset = Dataset.get_by_name(workspace, name='bike-no')
dataset.to_pandas_dataframe()
```

The 'Dataset tutorials' section lists several items:

- Create and run machine learning pipelines with Azure Machine Learning SDK
- Run batch inference on large amounts of data by using Azure Machine Learning
- Train models with automated machine learning in the cloud
- Train with datasets in Azure Machine Learning
- [GitHub] Azure Machine Learning datasets notebooks

Microsoft Azure Machine Learning Studio

Home > Datasets > bike-no

bike-no Version 1 (latest)

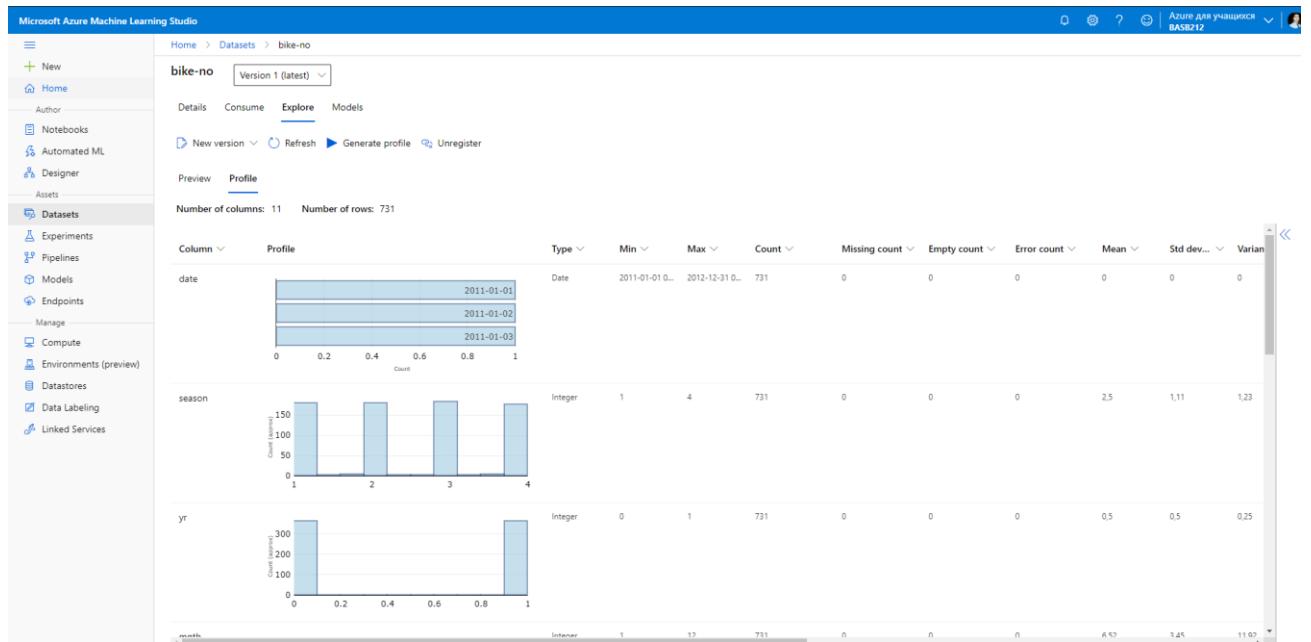
Details Consume Explore Models

New version Refresh Generate profile Unregister

Preview Profile

Number of columns: 11 Number of rows: 50 (of 731)

	Id	date	season	yr	mnth	weekday	weathersit	temp	atemp	hum	windspeed	cnt
1	2011-01-01 00:00:00	1	0	1	6	2	0	0.344	0.364	0.806	0.16	985
2	2011-01-02 00:00:00	1	0	1	0	2	1	0.363	0.354	0.696	0.249	801
3	2011-01-03 00:00:00	1	0	1	1	1	1	0.196	0.189	0.437	0.248	1349
4	2011-01-04 00:00:00	1	0	1	2	1	1	0.2	0.212	0.59	0.16	1562
5	2011-01-05 00:00:00	1	0	1	3	1	1	0.227	0.229	0.437	0.187	1600
6	2011-01-06 00:00:00	1	0	1	4	1	1	0.204	0.233	0.518	0.09	1606
7	2011-01-07 00:00:00	1	0	1	5	2	1	0.197	0.209	0.499	0.169	1510
8	2011-01-08 00:00:00	1	0	1	6	2	1	0.165	0.162	0.536	0.267	959
9	2011-01-09 00:00:00	1	0	1	0	1	1	0.138	0.116	0.434	0.362	822
10	2011-01-10 00:00:00	1	0	1	1	1	1	0.151	0.151	0.483	0.223	1321
11	2011-01-11 00:00:00	1	0	1	2	2	1	0.169	0.191	0.686	0.122	1263
12	2011-01-12 00:00:00	1	0	1	3	1	1	0.173	0.16	0.6	0.305	1162
13	2011-01-13 00:00:00	1	0	1	4	1	1	0.165	0.151	0.47	0.301	1406
14	2011-01-14 00:00:00	1	0	1	5	1	1	0.161	0.188	0.538	0.127	1421
15	2011-01-15 00:00:00	1	0	1	6	2	1	0.233	0.248	0.499	0.158	1248
16	2011-01-16 00:00:00	1	0	1	0	1	1	0.232	0.234	0.484	0.188	1204
17	2011-01-17 00:00:00	1	0	1	1	2	1	0.176	0.177	0.538	0.194	1000
18	2011-01-18 00:00:00	1	0	1	2	2	1	0.217	0.232	0.862	0.147	683
19	2011-01-19 00:00:00	1	0	1	3	2	1	0.292	0.298	0.742	0.208	1650



Step 14. Turn back to “Automated ML”, then click on dataset “bike-no”. Click “Next”.

The screenshot shows the 'Create a new Automated ML run' wizard. Step 1: 'Select dataset'. A table lists datasets: 'Forecasting-bike-share_BASB212_v2' (Tabular, created on 27 сент. 2021 г., 1:05, modified on 27 сент. 2021 г., 1:05) and 'bike-no' (Tabular, created on 26 сент. 2021 г., 18:35, modified on 26 сент. 2021 г., 18:35). Step 2: 'Configure run'. Step 3: 'Select task and settings'. Buttons at the bottom: 'Back', 'Next', and 'Cancel'.

Step 15. New experimental name “ML_bakeshare_prediction_Akmal”. Target column “cnt (Integer)”.

Create a new compute cluster. Click on “Create on new compute”. Find out virtual machine that is free of charge and with minimum technical parameters, since we have not a big dataset.

The screenshot shows the 'Create compute cluster' wizard. Step 1: 'Create new Automated ML run'. Step 2: 'Virtual Machine'. Location: Юго-Восточная Азия. Priority: Dedicated. Type: CPU. Size: Standard_DS1_v2. Quota: 24 cores. Buttons at the bottom: 'Back', 'Next', and 'Cancel'.

Name	Category	Workload types	Available quota	Cost
Standard_DS1_v2 2 cores, 14GB RAM, 28GB storage	Memory optimized	Development on Notebooks (or other IDE) and light weight testing	6 cores	--
Standard_DS2_v2 4 cores, 14GB RAM, 28GB storage	General purpose	Classical ML model training, AutoML runs, pipeline runs (default compute)	6 cores	--
Standard_DS12_v2 4 cores, 2608 RAM, 56GB storage	Memory optimized	Training on large datasets (>1GB) parallel run steps, batch inferencing	6 cores	--
Standard_F4s_v2 4 cores, 8GB RAM, 32GB storage	Compute optimized	Real-time inferencing and other latency-sensitive tasks	6 cores	\$0.20/hr

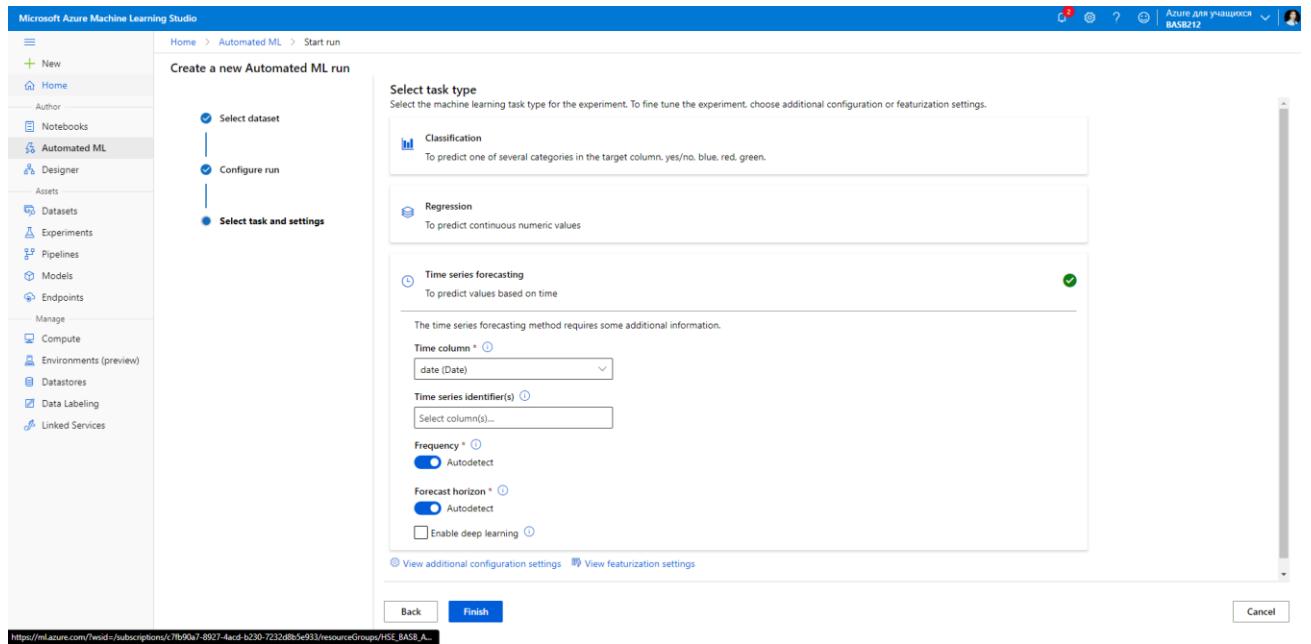
Compute a name “Standard-Vostok”. Then click “Next”

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with various options like 'New', 'Home', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The 'Automated ML' option is selected. The main area is titled 'Create a new Automated ML' and shows a flow: 'Select dataset' → 'Virtual Machine' → 'Configure run' (which is currently active) → 'Advanced Settings' → 'Select task and settings'. On the right, under 'Configure Settings', there's a table for compute cluster configurations. One entry is shown: Name: Standard_DS1_v2, Category: Memory optimized, Cores: 2, Available quota: 6 cores, RAM: 14 GB, Storage: 28 GB, Cost/Node: --. Below this table, there are fields for Compute name (containing 'Standard-Vostok'), Minimum number of nodes (0), Maximum number of nodes (1), Idle seconds before scale down (1800), and a toggle for Enable SSH access (which is off). There are also 'Back', 'Create', and 'Cancel' buttons at the bottom.

After then, we see this screen. Click just “Next”.

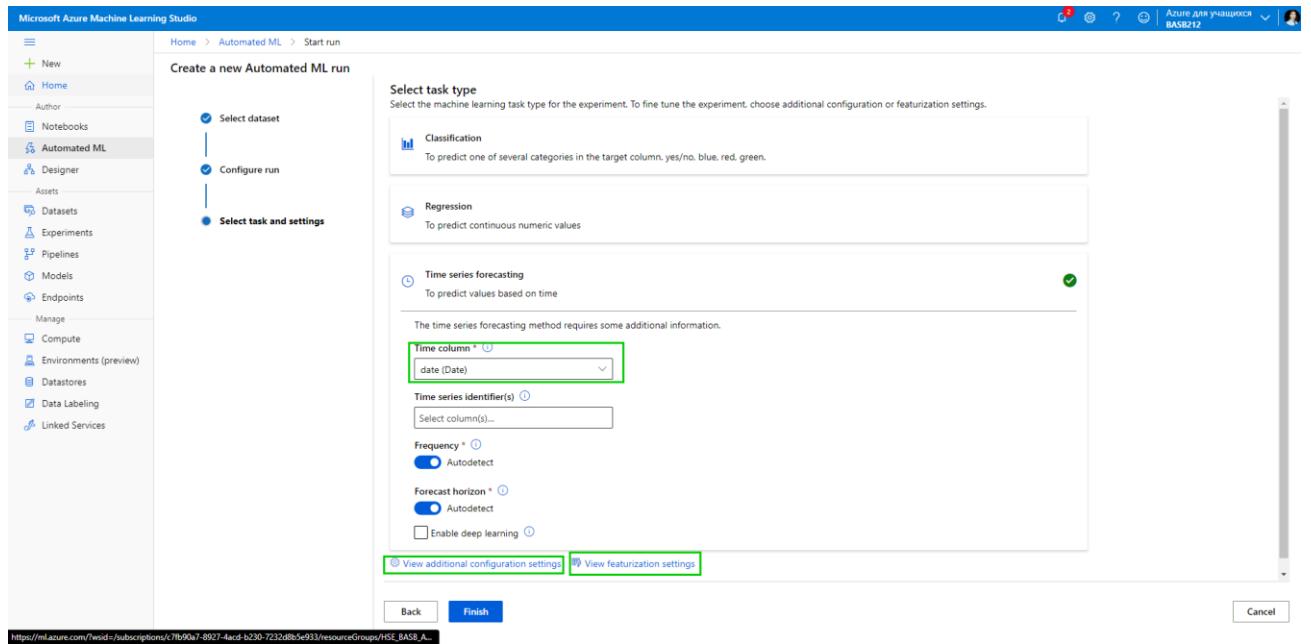
This screenshot shows the 'Create a new Automated ML run' step. The sidebar on the left is identical to the previous one. The main area has a similar flow: 'Select dataset' → 'Configure run' (active) → 'Select task and settings'. The 'Configure run' section displays a summary of the configuration: Dataset: bike-no (View dataset), Experiment name: New experiment name (containing 'ML_bakeshare_prediction_Akmal'), Target column: cnt (Integer), and Select compute cluster: Standard-Vostok. There are also buttons for 'Create a new compute' and 'Refresh compute'. At the bottom, there are 'Back', 'Next', and 'Cancel' buttons. The 'Next' button is highlighted in blue.

Screenshot of Microsoft Azure Machine Learning Studio showing the "Create a new Automated ML run" wizard. The "Select task type" step is selected. Under "Time series forecasting", the "Time column" dropdown is set to "date (Date)". Other settings like "Frequency" (Autodetect), "Forecast horizon" (Autodetect), and "Enable deep learning" are also visible.



Step 16. Select the machine learning task type for the experiment. We choose “time series forecasting”. Time column “date (Date)”

Screenshot of Microsoft Azure Machine Learning Studio showing the "Create a new Automated ML run" wizard. The "Select task type" step is selected. Under "Time series forecasting", the "Time column" dropdown is set to "date (Date)". The "View additional configuration settings" and "View featurization settings" buttons are highlighted with green boxes.



Then we go to “View featurization settings”. If needed we click. In this case we do not change anything. Just we need to persuade that there are not included “date” and “cnt (target column)”. Click “Save”

After then, we click on “View additional configuration settings”. There we choose primary metric “Normalized root mean squared error”. Block “ExtremeRandomTrees” algorithm. Turn on “Season and trend” and choose (Season and trend). Country or region for holidays “United States (US)”. In the Exit criterion we put 1 for training job time (hours).

In the “Validation” and “Concurrency” we do not change anything. Click “Save” button

The screenshot shows the 'Create a new Automated ML run' wizard in Microsoft Azure Machine Learning Studio. The current step is 'Select task and settings'. On the right, there are several configuration sections:

- Additional forecasting settings**: Includes 'Forecast target lags' (Autodetect), 'Target lags' (Target lags), and 'Target rolling window size' (Autodetect).
- Season and trend**: Set to 'On'.
- Country or region for holidays**: Set to 'United States (US)'.
- Exit criterion**: Training job time (hours) is set to 1, and Metric score threshold is empty.
- Validation**: Validation type is 'k-fold cross validation', and Number of cross validations is set to 5.
- Concurrency**: Max concurrent iterations is set to 1.

The 'Validation' and 'Concurrency' sections are highlighted with a green border. At the bottom right are 'Save' and 'Cancel' buttons.

Click “Finish”. There will be creating a new Automated ML run

The screenshot shows the 'Create a new Automated ML run' wizard in Microsoft Azure Machine Learning Studio. The current step is 'Finish'. A progress dialog box is centered on the screen, displaying the message "Creating a new Automated ML run..." and "Starting a new run with run ID AutoML_85be8679-e3aa-4490-862f-30a4c06fce7a...".

We need to wait while machine is running.

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a navigation sidebar with various options like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The main area shows a 'Properties' panel for a run named 'sincere_beard_fs86n28g (Run 1)'. The 'Status' field is highlighted with a green box and shows 'Running'. Other properties listed include 'Created by' (Akmal Xolikulov), 'Input datasets' (None), 'Output datasets' (None), 'Arguments' (None), 'Compute target' (Standard-Vostok), 'Run ID' (AutoML_85be8679-e3aa-4490-862f-30a4c06fce7a), and 'Script name' (..). To the right, there are sections for 'Best model summary' (No data), 'Run summary' (Task type: Forecasting, Featurization: Auto, Primary metric: Normalized root mean squared error, Experiment name: ML_bakeshare_prediction_Akmal), and 'Notifications'.

Status: Completed.

This screenshot shows the same interface after the process has completed. The 'Status' field in the 'Properties' panel is now highlighted with a green box and shows 'Completed'. The rest of the properties remain the same as in the previous screenshot. The 'Run summary' section also remains identical, showing the task type as Forecasting, featurization as Auto, primary metric as Normalized root mean squared error, and experiment name as ML_bakeshare_prediction_Akmal.

Step 17. We have to go to “Models” in order to see which is the best algorithm. VotingEnsemble is the best ML algorithm.

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with various navigation options like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The 'Models' option is selected. The main area is titled 'sincere_beard_fs86n28g (Run 1)'. Below it, there are tabs for 'Details', 'Data guardrails', 'Models' (which is active and highlighted in green), 'Outputs + logs', 'Child runs', and 'Snapshot'. There are also buttons for 'Refresh', 'Deploy', 'Download', 'Explain model', 'Edit columns', and 'Reset view'. A search bar is present, along with filters for 'Submitted time', 'All filters', and 'Clear all'. The results table shows 1-25 of 37 models. The first row, 'VotingEnsemble', is selected and has a green border. The columns include 'Algorithm name', 'Explained', 'Normalized ro...', 'Sampling', 'Submitted time', 'Duration', and 'Hyperparameter'. The 'VotingEnsemble' row shows an explained value of 0.12903 and a duration of 3m 47s. The 'Hyperparameter' column shows 'algorithm : [LightGBM, 'ElasticNet' ...]'. Other models listed include MinMaxScaler, LightGBM, Arimax, StandardScalerWrapper, XGBoostRegressor, StandardScalerWrapper, ElasticNet, StandardScalerWrapper, ElasticNet, Naive, StandardScalerWrapper, ElasticNet, MaxAbsScaler, ElasticNet, StandardScalerWrapper, ElasticNet, and MaxAbsScaler, ElasticNet. At the bottom, there are navigation buttons for 'Page' (1 of 2).

Lets to see more details about algorithm by clicking on “View explanation”

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The left sidebar is identical to the previous screenshot. The main area is titled 'nifty_battery_wt5jqf9 (Run 40)'. Below it, there are tabs for 'Details', 'Model', 'Explanations (preview)', 'Metrics', 'Data transformation (preview)', 'Outputs + logs', 'Images', 'Child runs', 'Snapshot', and 'Monitoring (preview)'. The 'Explanations (preview)' tab is active. The page is divided into sections: 'Properties' (Status: Completed, Script name: automl_driver.py, Created by: Ахмал Холиколов, Started: Sep 27, 2021 3:16 AM, Duration: 3m 47.42s, Compute duration: 3m 47.42s, Compute target: Standard-Vostok, Run ID: AutoML_85be8679-e3aa-4490-862f-30a4c06fce7a_36), 'Metrics' (Explained variance: 0.53344, Mean absolute error: 1003.6, Mean absolute percentage error: 99.933, a link to 'View all other metrics'), and 'Description' (a placeholder 'Click edit icon to add a description'). At the bottom, there's a 'Tags' section with several tags listed: miflow.source.name : automl_driver.py, miflow.source.type : JOB, model_explain_run_id : AutoML_85be8679-e3aa-4490-862f-30a4c06fce7a_ModelExplain, and model_explanation : True.

Microsoft Azure Machine Learning Studio

Home > Automated ML > ML_bakeshare_prediction_Akmal > sincere_beard_fs86n28g (Run 1) > nifty_battery_wt5jqf9 (Run 40)

nifty_battery_wt5jqf9 (Run 40)

Refresh Deploy Download Explain model Cancel Delete

Details Model Explanations (preview) Metrics Data transformation (preview) Outputs + logs Images Child runs Snapshot Monitor

Model summary

Algorithm name VotingEnsemble

Ensemble details

[View ensemble details](#)

Normalized root mean squared error 0.12903 [View all other metrics](#)

Sampling 100.00 %

Registered models No registration yet

Deploy status No deployment yet

Ensemble details

Select an ensemble algorithm to see the ensemble weights and hyperparameters.

MinMaxScaler, LightGBM

StandardScalerWrapper, Elastic...

Naive

Ensemble weight: 0.6666666666666666

Data transformation:

```

1  {
2   "class_name": "MinMaxScaler",
3   "module": "sklearn.preprocessing",
4   "param_args": [],
5   "param_kwargs": {},
6   "prepared_kwargs": {},
7   "spec_class": "preproc"
8 }
```

Training algorithm:

```

1  {
2   "class_name": "LightGBMRegression",
3   "module": "automl_client.core.common.model_wrappers",
4   "param_args": [],
5   "param_kwargs": {
6     "boosting_type": "gbdt",
7     "colsample_bytree": 0.7000000000000001,
8     "learning_rate": 0.15789684210526317,
9     "max_bin": 63,
10    "max_depth": 6,
11    "min_data_in_leaf":
```

Close

Microsoft Azure Machine Learning Studio

Home > Automated ML > ML_bakeshare_prediction_Akmal > sincere_beard_fs86n28g (Run 1) > nifty_battery_wt5jqf9 (Run 40)

nifty_battery_wt5jqf9 (Run 40)

Refresh Deploy Download Explain model Cancel Delete

Details Model Explanations (preview) Metrics Data transformation (preview) Outputs + logs Images Child runs Snapshot Monitor

Model summary

Algorithm name VotingEnsemble

Ensemble details

[View ensemble details](#)

Normalized root mean squared error 0.12903 [View all other metrics](#)

Sampling 100.00 %

Registered models No registration yet

Deploy status No deployment yet

Run Metrics

Explained variance 0.53344

Mean absolute error 100.3.6

Mean absolute percentage error 99.933

Median absolute error 917.08

Normalized mean absolute error 0.11546

Normalized median absolute error 0.10551

Normalized root mean squared error 0.12903

Normalized root mean squared log error 0.12138

R2 score -0.53073

Root mean squared error 1121.6

Root mean squared log error 0.72066

Spearman correlation 0.77857

Close

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a navigation sidebar with various options like Home, Designer, Assets, Datasets, Experiments, Pipelines, Models, Endpoints, Compute, Environments (preview), Datastores, Data Labeling, and Linked Services. The main area displays a run named "nifty_battery_wt5jqf9 (Run 40)". Below the run name are buttons for Refresh, Deploy, Download, Explain model, Cancel, and Delete. A tab bar at the top includes Details, Model, Explanations (preview), Metrics (which is selected), Data transformation (preview), Outputs + logs, Images, Child runs, Snapshot, and Monitoring (preview). Under the Metrics tab, there's a section to "Select a metric to see a visualization or table of the data". A search bar is present, and a list of metrics is shown with checkboxes: explained_variance (checked, value 0.533), mean_absolute_error (checked, value 1003.621), mean_absolute_percentage_error (checked, value 99.933), median_absolute_error (checked, value 917.081), normalized_mean_absolute_error (checked, value 0.115), normalized_mean_squared_error (checked, value 0.106), normalized_root_mean_squared_error (checked, value 0.129), normalized_root_mean_squared_log_error (checked, value 0.121), r2_score (checked, value -0.531), residuals (checked), predicted_true (checked), r2_score (checked), residuals (checked), root_mean_squared_error (checked), root_mean_squared_log_error (checked), and spearman_correlation (unchecked). To the right of this list are three visualizations: a Residuals Histogram showing Bin Count vs. Residuals, a Predicted vs. True plot showing Predicted Value vs. True Value with Average Pred and Ideal lines, and another histogram for True Value.

Choose the needed metrics on left side. Click on the “Deploy”. Then on the right side appears “Deploy a model” window. Put a name “bikeshare-timeseries-v1”. Then choose Compute type “Azure Container Instance”. After that click Deploy button.

This screenshot shows the "Deploy a model" dialog box overlaid on the Azure Machine Learning Studio interface. The dialog has fields for Name (set to "bikeshare-timeseries-v1"), Description (empty), Compute type (set to "Azure Container Instance"), and other deployment settings like Enable authentication (unchecked) and Use custom deployment assets (unchecked). At the bottom are "Advanced" and "Deploy" buttons, with the "Deploy" button highlighted by a green border.

We see that “Model deployment is successfully triggered”. We need to click on Deploy details

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The left sidebar has a tree view with nodes like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The 'Endpoints' node is selected. The main area shows a project structure: Home > Automated ML > ML_bikeshare_prediction_Akmal > sincere_beard_f586n28g (Run 1) > nifty_battery_wt5jzqf9 (Run 40). A success message 'Success: Model deployment is successfully triggered' is displayed. A green box highlights the 'Deploy details' button in the top right corner of a modal window titled 'Endpoint "bikeshare-timeseries-v1" deployment in progress'. Below the message, there are tabs for 'Metrics', 'Data transformation (preview)', 'Outputs + logs', 'Images', 'Child runs', 'Snapshot', and 'Monitoring (preview)'. Under 'Metrics', a chart and table show various performance metrics. A histogram titled 'Residuals Histogram' and a scatter plot titled 'Predicted vs. True' are also visible.

Appears deployment details on the screen

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The left sidebar has a tree view with nodes like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Environments (preview)', 'Datastores', 'Data Labeling', and 'Linked Services'. The 'Endpoints' node is selected. The main area shows an endpoint named 'bikeshare-timeseries-v1'. A green box highlights the 'Deployment state' field, which is currently 'Unhealthy'. Other fields shown include 'Service ID' (bikeshare-timeseries-v1), 'Description' (empty), 'Compute type' (Container instance), 'Created by' (Акмал Холиколов), 'Model ID' (AutoML_85be8679e36:1), 'Created on' (9/28/2021 12:49:22 ночи), 'Last updated on' (9/28/2021 12:49:22 ночи), 'Image ID' (empty), 'REST endpoint' (null), 'Key-based authentication enabled' (false), 'Swagger URI' (empty), 'CPU' (1.8), and 'Memory' (empty). There are tabs for 'Details' and 'Deployment logs'. On the right, there are sections for 'Tags' (No data) and 'Properties' (runId: AutoML_85be8679-e3aa-4490-862f-30a4c06fce7a_36).

Let's test it

Also in the “Consume” there are the codes in 3 programming languages. Lets choose Python.

```
import urllib.request
import json
import os
import ssl
```

```
def allowSelfSignedHttps(allowed):
    # bypass the server certificate verification on client side
```

```

    if allowed and not os.environ.get('PYTHONHTTPSVERIFY', '') and getattr(ssl,
        '_create_unverified_context', None):
        ssl._create_default_https_context = ssl._create_unverified_context

allowSelfSignedHttps(True) # this line is needed if you use self-signed certificate in your scoring
service.

# Request data goes here
data = {
    "data": [
        {
            'date': "2000-01-01 00:00:00,000000",
            'season': "0",
            'yr': "0",
            'mnth': "0",
            'weekday': "0",
            'weathersit': "0",
            'temp': "0",
            'atemp': "0",
            'hum': "0",
            'windspeed': "0",
        },
    ],
}
body = str.encode(json.dumps(data))

url = 'http://ce339fd9-a27d-4090-b334-d8b098fad965.northeurope.azurecontainer.io/score'
api_key = "" # Replace this with the API key for the web service
headers = {'Content-Type':'application/json', 'Authorization':('Bearer ' + api_key)}

req = urllib.request.Request(url, body, headers)

try:
    response = urllib.request.urlopen(req)

    result = response.read()
    print(result)
except urllib.error.HTTPError as error:
    print("The request failed with status code: " + str(error.code))

    # Print the headers - they include the request ID and the timestamp, which are useful for
    # debugging the failure
    print(error.info())
        print(json.loads(error.read().decode("utf8", 'ignore')))
```

Regression (self-work)

Step 1.

The screenshot shows the 'Create a new Automated ML run' wizard in Microsoft Azure Machine Learning Studio. The sidebar on the left has 'Automated ML' selected. The main area is titled 'Select dataset' and shows a list of datasets. One dataset, 'bike-no', is highlighted with a green box. Below the list are buttons for 'Back', 'Next', and 'Cancel'.

Step 2

The screenshot shows the 'Configure run' step of the wizard. The sidebar still has 'Automated ML' selected. The main area is titled 'Configure run' and contains fields for 'Experiment name', 'Target column', and 'Select compute cluster'. The 'Experiment name' field has 'ML_bakeshare_regression_Akmal' and the 'Create new' radio button is selected. The 'Target column' field has 'cnt (Integer)'. The 'Select compute cluster' field has 'GeneralPurpose'. Below these fields are buttons for 'Back', 'Next', and 'Cancel'.

Step 3

The screenshot shows the 'Create a new Automated ML run' wizard in Microsoft Azure Machine Learning Studio. The current step is 'Select task type'. The 'Regression' option is selected and highlighted with a green box. On the right side, there is a large panel titled 'Additional configurations' containing various machine learning parameters:

- Primary metric:** Normalized root mean squared error
- Explain best model:** Checked
- Blocked algorithms:** A list of algorithms that Automated ML will not use during training.
- Exit criterion:** Training job time (hours) set to 6, Metric score threshold set to Auto.
- Validation:** Validation type set to Auto.
- Concurrency:** Max concurrent iterations set to 1.

At the bottom of the screen, there are 'Back', 'Finish', 'Save', and 'Cancel' buttons.

The screenshot shows the 'Create a new Automated ML run' wizard in Microsoft Azure Machine Learning Studio. The current step is 'Creating a new Automated ML run...'. A progress bar is visible, and a message at the bottom of the screen reads: "Starting a new run with run ID AutoML_741bdfe8-c4d9-45f8-ad25-1e7a32ba6b12...". The left sidebar shows the navigation menu for the studio.

Microsoft Azure Machine Learning Studio

Home > Automated ML > ML_bakeshare_regression_Akmal > tidy_ship_2fl72hyd (Run 1)

tidy_ship_2fl72hyd (Run 1)

Refresh Cancel Delete

Details Data guardrails Models Outputs + logs Child runs Snapshot

Properties

Status	Created by Ахмал Холиколов
Created Sep 27, 2021 10:08 AM	Input datasets Input name: training_data. Dataset: bike-no: Version 1
Started --	Output datasets None
Compute target GeneralPurpose	Arguments None
Run ID AutoML_741bc6e8-c4a9-45f8-ad25-1e7a32ba6b12	See all properties
Script name --	

Tags

No tags

Description

Click edit icon to add a description

Best model summary

No data

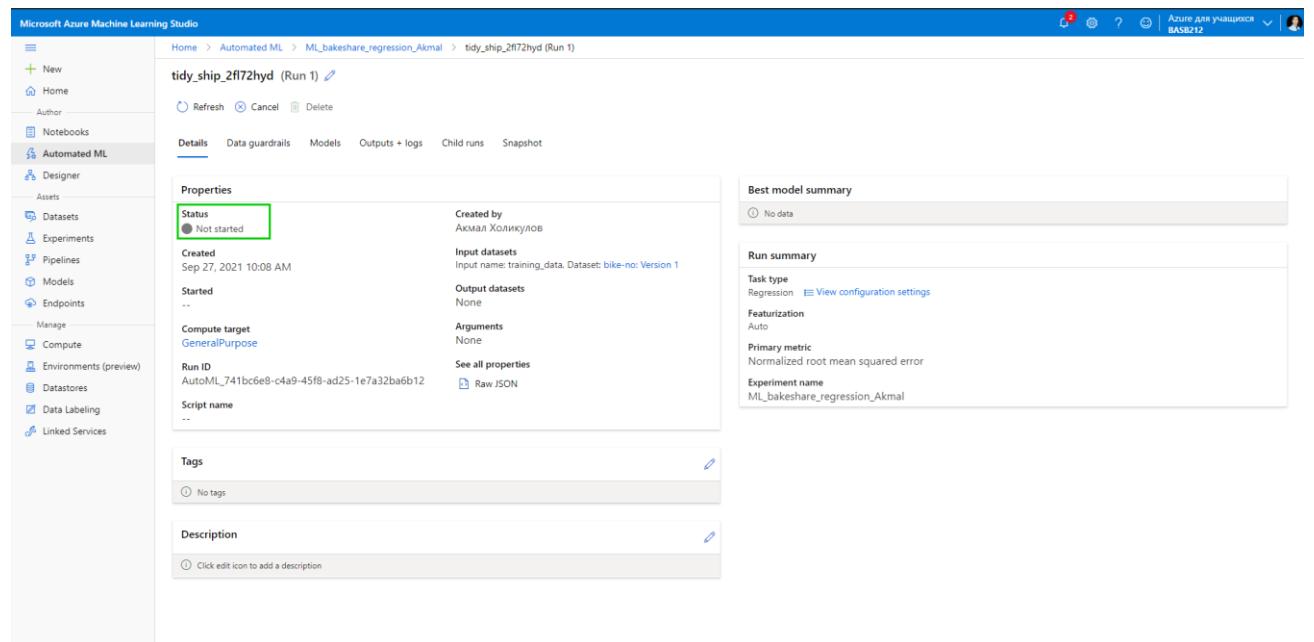
Run summary

Task type: Regression

Featurization: Auto

Primary metric: Normalized root mean squared error

Experiment name: ML_bakeshare_regression_Akmal



Microsoft Azure Machine Learning Studio

Home > Automated ML > ML_bakeshare_regression_Akmal > tidy_ship_2fl72hyd (Run 1)

tidy_ship_2fl72hyd (Run 1)

Refresh Cancel Delete

Details Data guardrails Models Outputs + logs Child runs Snapshot

Properties

Status	Script name --
Created Sep 27, 2021 10:08 AM	Created by Ахмал Холиколов
Started Sep 27, 2021 10:08 AM	Input datasets Input name: training_data. Dataset: bike-no: Version 1
Duration 49m 39.06s	Output datasets None
Compute duration 49m 39.06s	Arguments None
Compute target GeneralPurpose	See all properties
Run ID AutoML_741bc6e8-c4a9-45f8-ad25-1e7a32ba6b12	

Tags

No tags

Description

Click edit icon to add a description

Best model summary

Algorithm name: VotingEnsemble

Ensemble details

Normalized root mean squared error: 0.06714

Sampling: 100.00%

Registered models: No registration yet

Deploy status: No deployment yet

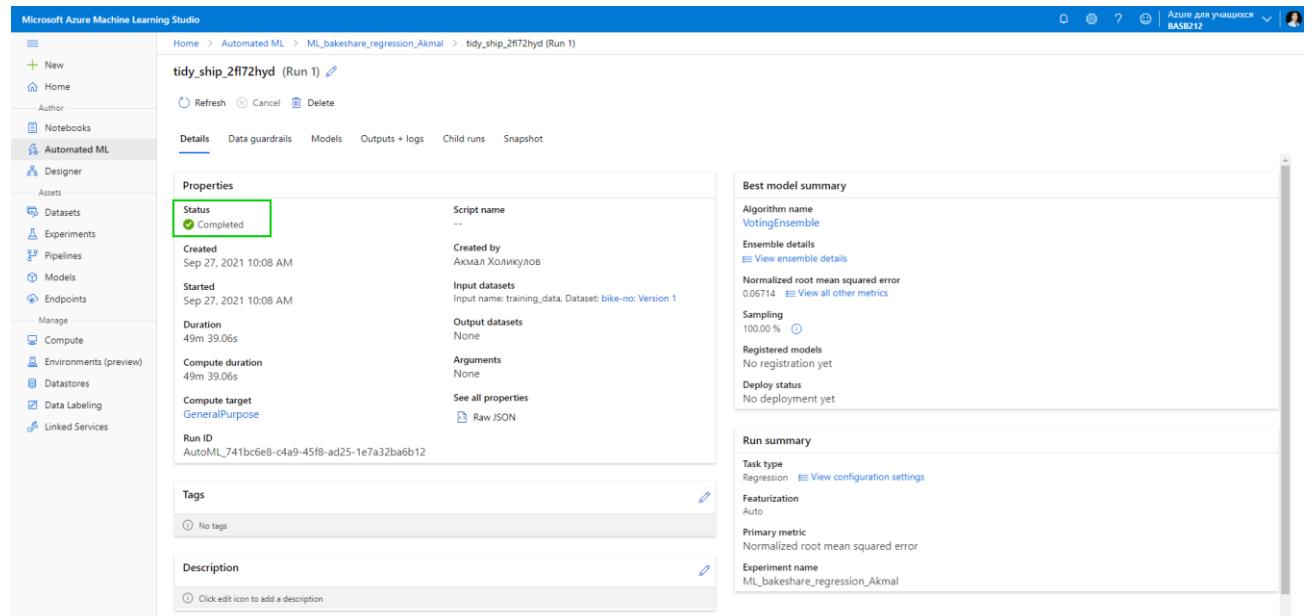
Run summary

Task type: Regression

Featurization: Auto

Primary metric: Normalized root mean squared error

Experiment name: ML_bakeshare_regression_Akmal



Now we need to deploy model. We have to repeat as before.

Microsoft Azure Machine Learning Studio

Home > Experiments > ML_bakeshare_regression_Akmal > tidy_ship_2fl7hyd (Run 1) > amusing_floor_st6p5j45 (Run 46)

amusing_floor_st6p5j45 (Run 46)

Metrics Data transformation (preview) Outputs + logs Images Child runs Snapshot Monitoring (preview)

Select a metric to see a visualization or table of the data.

View as: Chart Table

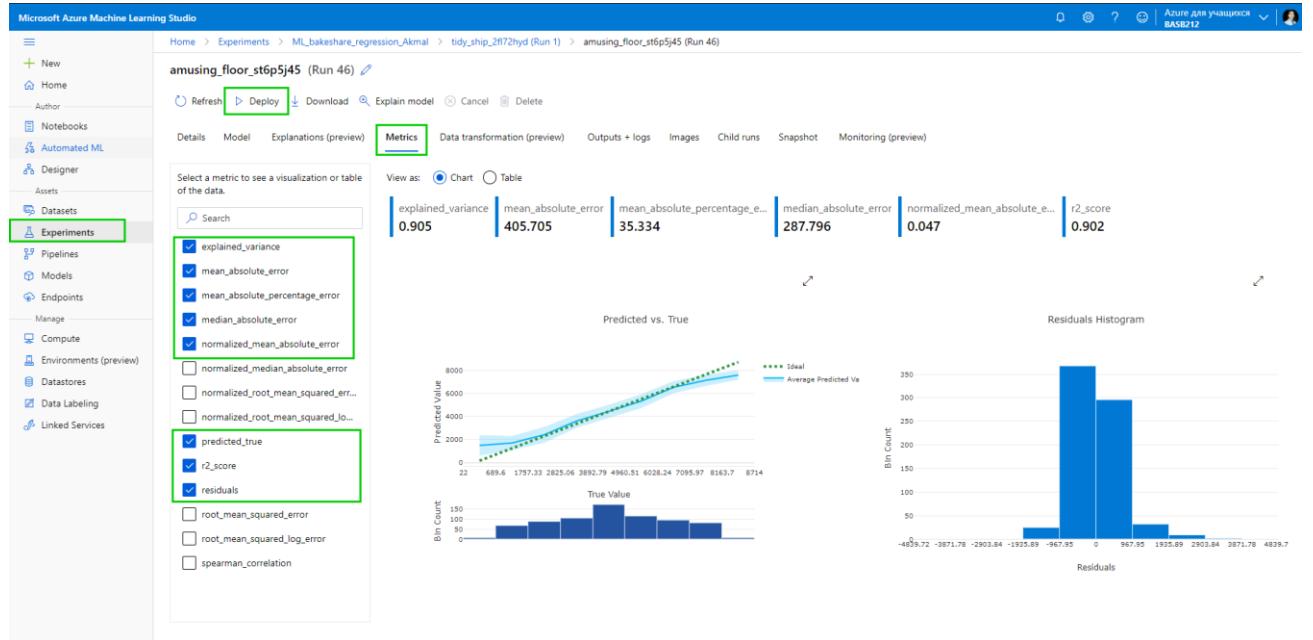
Search

explained_variance 0.905 mean_absolute_error 405.705 mean_absolute_percentage_error 35.334 median_absolute_error 287.796 normalized_mean_absolute_error 0.047 r2_score 0.902

explained_variance
 mean_absolute_error
 mean_absolute_percentage_error
 median_absolute_error
 normalized_mean_absolute_error
 normalized_median_absolute_error
 normalized_root_mean_squared_error
 normalized_root_mean_squared_log_error
 predicted_true
 r2_score
 residuals
 root_mean_squared_error
 root_mean_squared_log_error
 spearman_correlation

Predicted vs. True

Residuals Histogram



Microsoft Azure Machine Learning Studio

Home > Experiments > ML_bakeshare_regression_Akmal > tidy_ship_2fl7hyd (Run 1) > amusing_floor_st6p5j45 (Run 46)

amusing_floor_st6p5j45 (Run 46)

Metrics Data transformation (preview) Outputs + logs Images Child runs Snapshot Monitoring

Select a metric to see a visualization or table of the data.

View as: Chart Table

Search

explained_variance 0.905 mean_absolute_error 405.705 mean_absolute_percentage_error 35.334 median_absolute_error 287.796

explained_variance
 mean_absolute_error
 mean_absolute_percentage_error
 median_absolute_error
 normalized_mean_absolute_error
 normalized_median_absolute_error
 normalized_root_mean_squared_error
 normalized_root_mean_squared_log_error
 predicted_true
 r2_score
 residuals
 root_mean_squared_error
 root_mean_squared_log_error
 spearman_correlation

Predicted vs. True

Residuals Histogram

Deploy a model

Name *

Description

Compute type *

Model: AutoMLZ41bc6e8c40

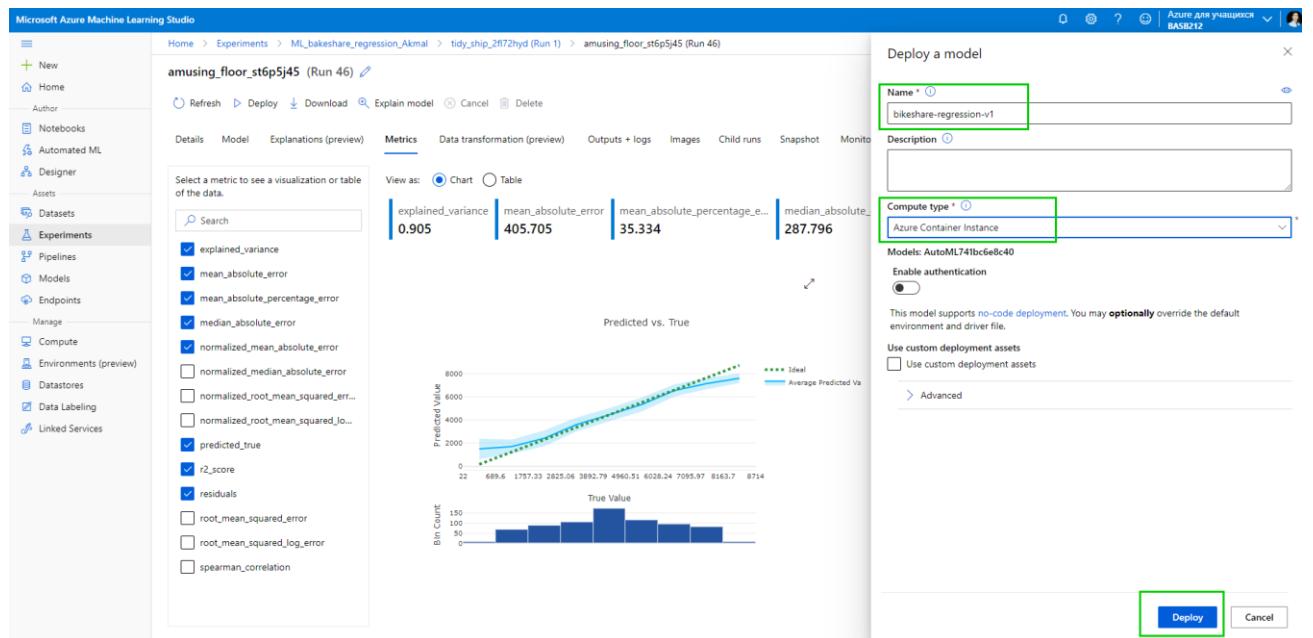
Enable authentication

This model supports no-code deployment. You may optionally override the default environment and driver file.

Use custom deployment assets

> Advanced

Deploy Cancel



Microsoft Azure Machine Learning Studio

Home > Endpoints > bikeshare-regression-v1

bikeshare-regression-v1

Details Test Consume Deployment logs

Attributes

Service ID: bikeshare-regression-v1
Description: --
Deployment state: Healthy (green box)
Compute type: Container instance
Created by: Акман Холиколов
Model ID: AutoML741bc6e8c40:1
Created on: 9/28/2021 11:59:05 вечера
Last updated on: 9/28/2021 11:59:05 вечера
Image ID: --
REST endpoint: http://dc12078a-7f0a-4b20-9dde-295e1f25ac0d.northeurope.azurecontainer.io/score (green box)
Key-based authentication enabled: false
Swagger URI: http://dc12078a-7f0a-4b20-9dde-295e1f25ac0d.northeurope.azurecontainer.io/swagger.json (green box)

Properties

runid: AutoML_741bc6e8-c4a9-45f8-ad25-1e7a32ba6b12_40
hasInferenceSchema: True
hasHttps: False

Microsoft Azure Machine Learning Studio

Home > Endpoints > bikeshare-regression-v1

bikeshare-regression-v1

Details **Test** Consume Deployment logs

Input data to test real-time endpoint

Select editor type: Form editor (radio button selected) JSON editor

Form editor fields:

- date: 2013-01-01 00:00:00.000000
- season: 1
- yr: 1
- mnth: 1
- weekday: 1
- weathersit: 1
- temp: 0.2
- atemp:

Test result

Test result output:

```
{ "result": [ 2594.950478941618 ] }
```

```
import urllib.request
import json
import os
import ssl

def allowSelfSignedHttps(allowed):
    # bypass the server certificate verification on client side
```

```

if allowed and not os.environ.get('PYTHONHTTPSVERIFY', '') and getattr(ssl, '_create_unverified_context', None):
    ssl._create_default_https_context = ssl._create_unverified_context

allowSelfSignedHttps(True) # this line is needed if you use self-signed certificate in your scoring service.

# Request data goes here
data = {
    "data": [
        {
            'date': "2000-01-01 00:00:00,000000",
            'season': "0",
            'yr': "0",
            'mnth': "0",
            'weekday': "0",
            'weathersit': "0",
            'temp': "0",
            'atemp': "0",
            'hum': "0",
            'windspeed': "0",
        },
    ],
}
body = str.encode(json.dumps(data))

url = 'http://dc12078a-7f0a-4b20-9dde-295e1f25ac0d.northeurope.azurecontainer.io/score'
api_key = '' # Replace this with the API key for the web service
headers = {'Content-Type':'application/json', 'Authorization':('Bearer ' + api_key)}

req = urllib.request.Request(url, body, headers)

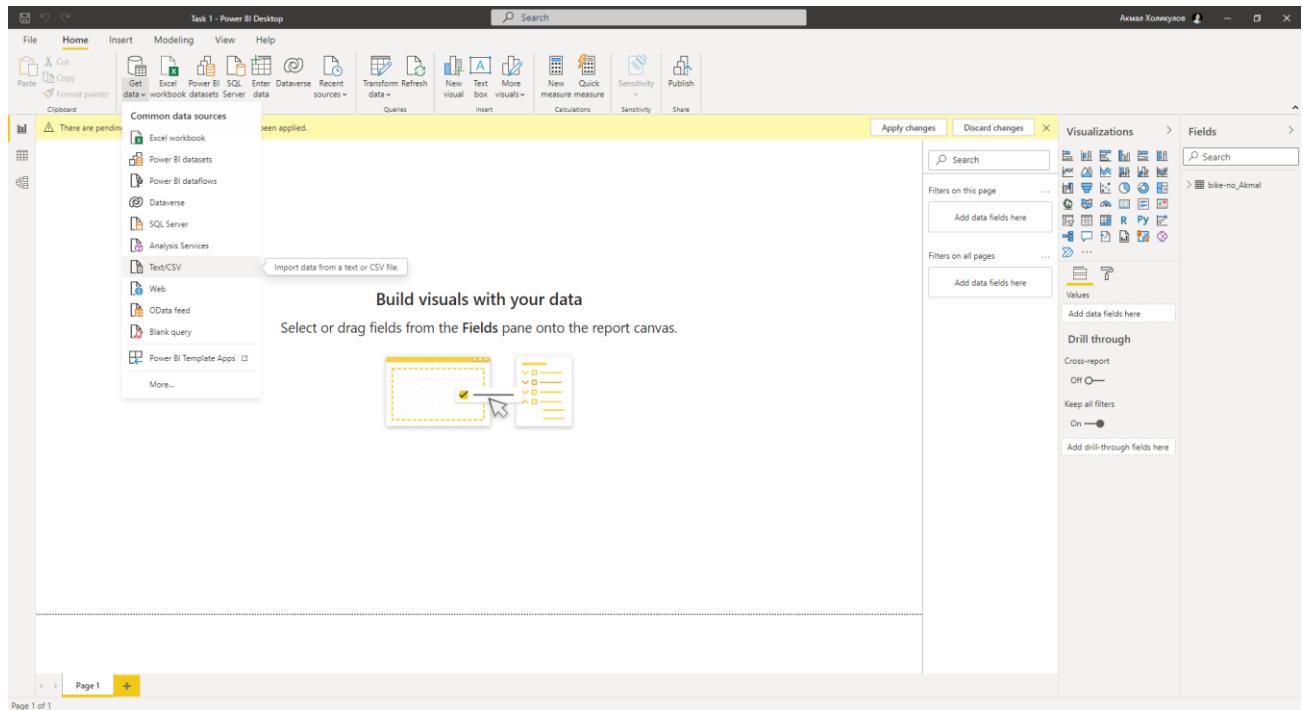
try:
    response = urllib.request.urlopen(req)

    result = response.read()
    print(result)
except urllib.error.HTTPError as error:
    print("The request failed with status code: " + str(error.code))

    # Print the headers - they include the request ID and the timestamp, which are useful for debugging the failure
    print(error.info())
    print(json.loads(error.read().decode("utf8", 'ignore')))
```

Consume Azure Machine Learning models in Power BI

First, we need to upload a data.



The screenshot shows the Power Query Editor within Power BI Desktop. It displays a table with 15 columns and 731 rows, with the first few rows of data visible. The columns include 'instant', 'date', 'season', 'yr', 'mnth', 'weekday', 'weathersit', and 'temp'. On the right side, there are 'Query Settings' and 'APPLIED STEPS' panes. The 'APPLIED STEPS' pane shows a step named 'Applied AzureML.bikeshare-regres...' which includes 'Source', 'Promoted Headers', and 'Changed Type'. The status bar at the bottom right indicates 'PREVIEW DOWNLOADED AT 11:12 AM'.

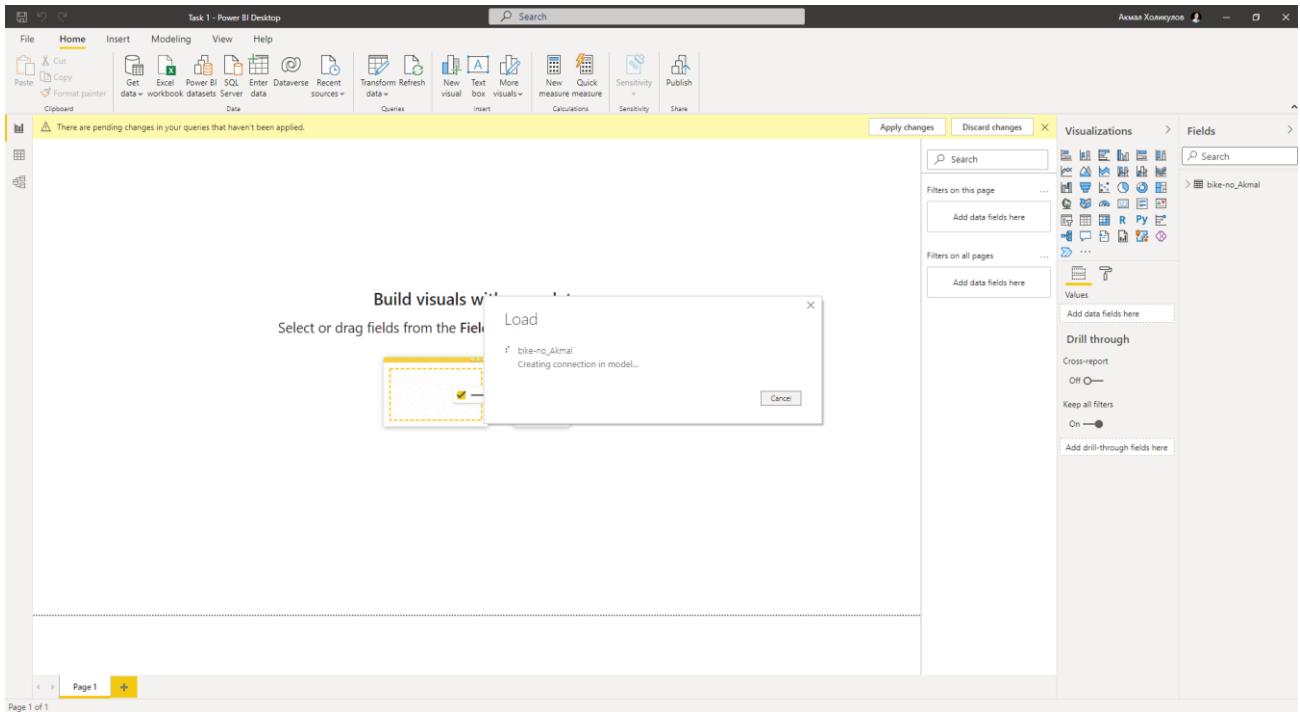
Then open that data by clicking “transform data”. In the window (Power Query Editor) we click on “Azure Machine Learning”. Choose deployed model from Azure ML (bikeshare-regression-v1), then put some data to columns. Click “OK”.

Here is the calculation made by model

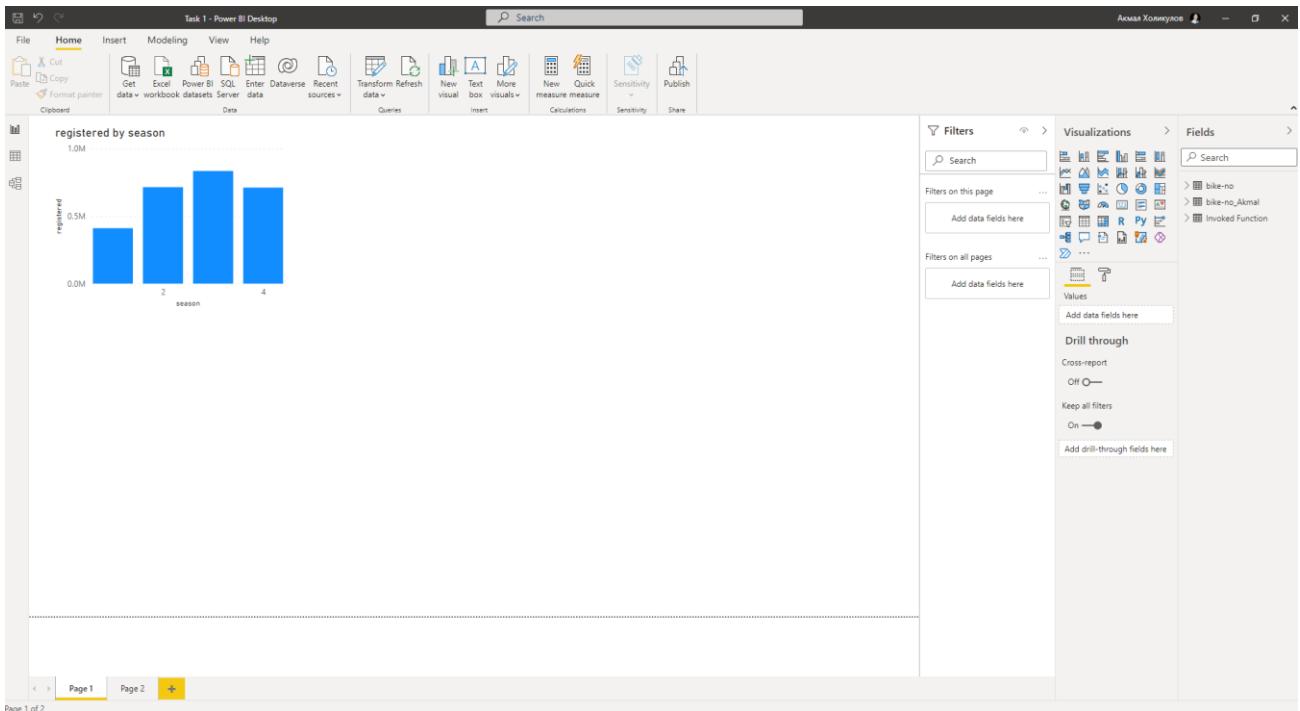
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29					
date	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011	1/2/2011						
temp	0.344287	0.369625	0.363979	0.186864	0.228997	0.204348	0.198522	0.165	0.188333	0.150833	0.165091	0.177277	0.165	0.16087	0.253333	0.231667	0.175883	0.234217	0.176771	0.216687	0.292174	0.261667	0.1775	0.0591304	0.0965217	0.0975913	0.223478	0.22775	0.195	0.205478				
atemp	0.369625	0.363979	0.35779	0.189405	0.22927	0.233029	0.208839	0.185	0.181675	0.150888	0.191464	0.160473	0.150883	0.168413	0.248112	0.234217	0.176771	0.234217	0.176771	0.232333	0.298422	0.25505	0.1775	0.0790696	0.0988391	0.0975913	0.234526	0.234526	0.2197	0.223317				
hum	0.805933	0.80946	0.248559	0.248309	0.180296	0.089582	0.167626	0.266804	0.341467	0.42917	0.122232	0.304627	0.309545	0.470417	0.126548	0.157983	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797	0.171797					
wind	0.1269	0.1229	0.131	0.120	0.108	0.088	0.1454	0.1518	0.1606	0.1562	0.1220	0.1137	0.1162	0.1406	0.1310	0.1206	0.1248	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204	0.1204			
casual	331	331	120	120	108	88	1454	1518	1606	1562	1209	1137	1162	1406	1310	1206	1248	959	959	959	959	959	959	959	959	959	959	959	959	959	959	959		
registered	654	654	670	670	670	670	801	801	801	801	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149	7149			
cnt	985	985	801	801	801	801	2463	2463	2463	2463	2109	2109	2109	2109	2109	2109	2109	2109	2109	2109	2227	2227	2227	2227	2227	2227	2227	2227	2227	2227	2227	2227	2227	2227

15 COLUMNS, 731 ROWS Column profiling based on top 1000 rows PREVIEW DOWNLOADED AT 1:12 AM

Click on “Close and apply”



After that we can visualize the data. For example, we take a “season” on X axis and “registered by season” on Y axis.



What other Data Science projects can be implemented for a bikeshare company? Share your ideas in the report

I think there are two data analysis algorithms that could be implemented for a bike rental company.

First, there is logistic regression. We have a lot of data on weather conditions over the past few years. By applying a logistic regression algorithm, we can use it to predict weather conditions, whether it is snow or rain (binary classification of data points). Bicycle rentals are directly dependent on weather conditions, because customers are not inclined to rent bicycles in bad weather conditions.

Second, linear regression. The number of people renting bicycles has increased since the start of the business. To measure the relationship between two continuous variables, we can apply a linear regression algorithm. For example, the number of rented bicycles on the Y-axis and the number of years on the X-axis. After applying linear regression, we can see the trend and based on that information we will know how many bikes we need to buy in advance for next years.