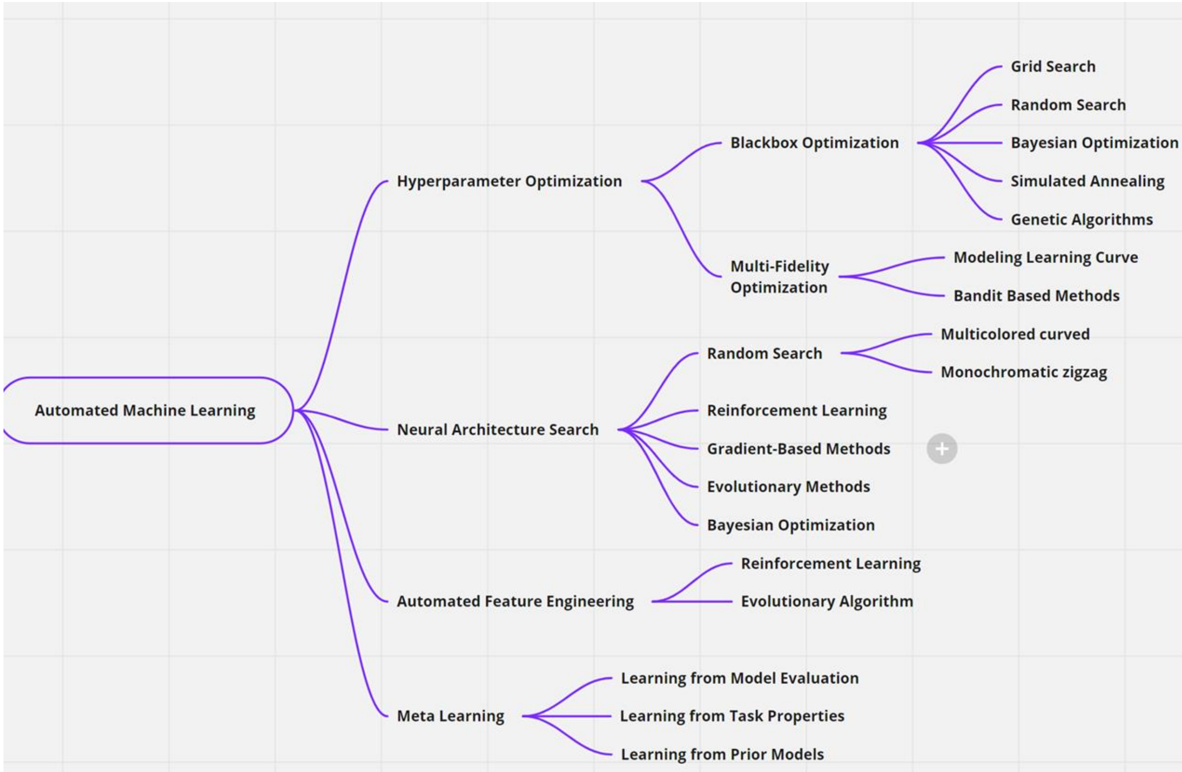
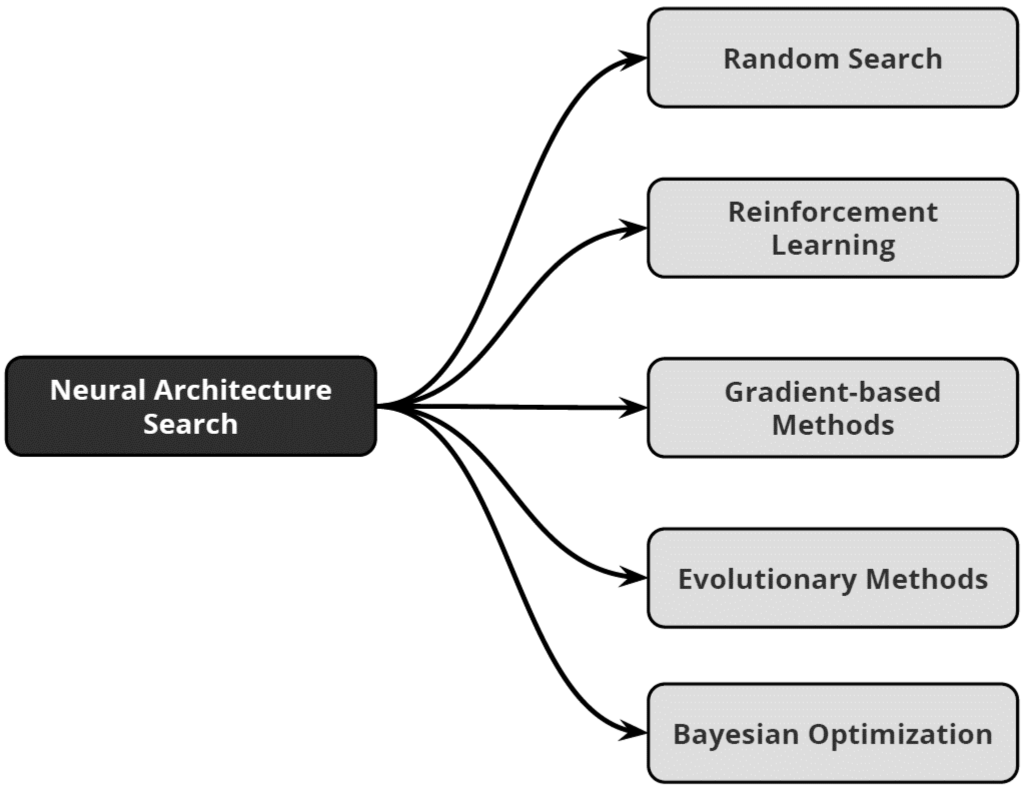
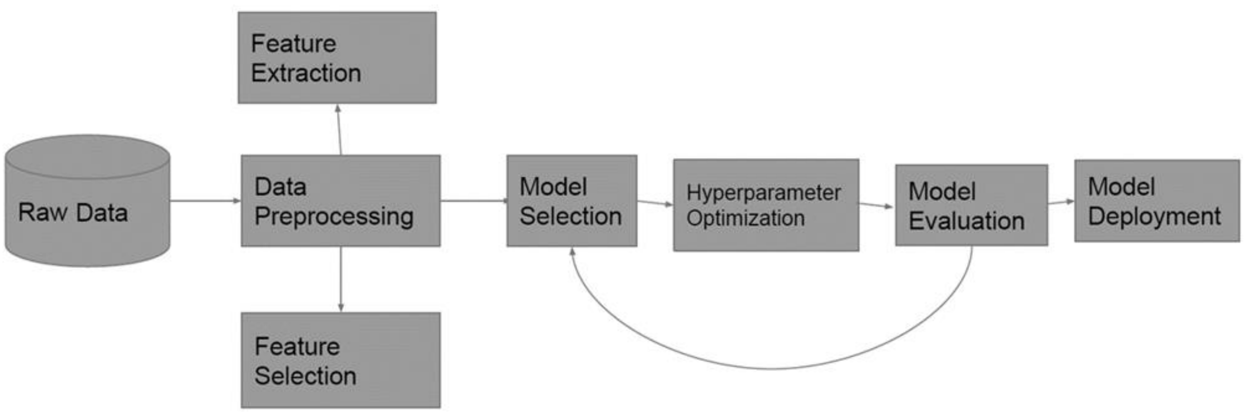
Chapter 1: A Lap around Automated Machine Learning

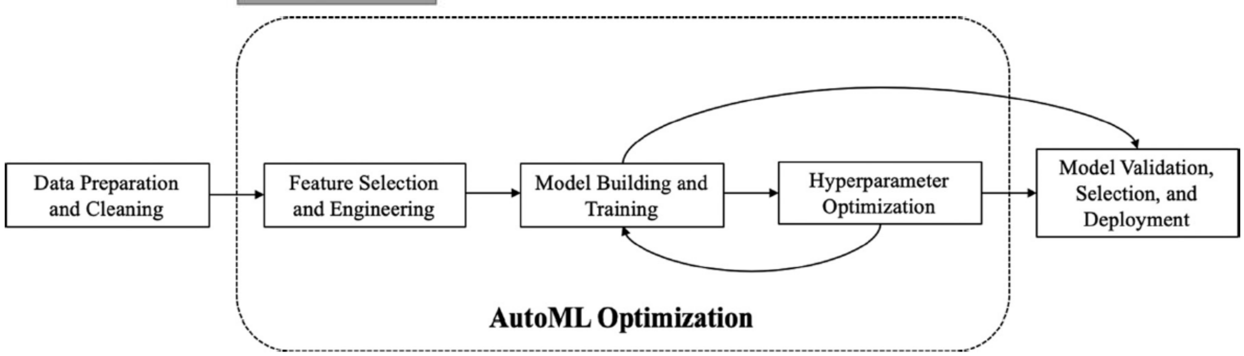


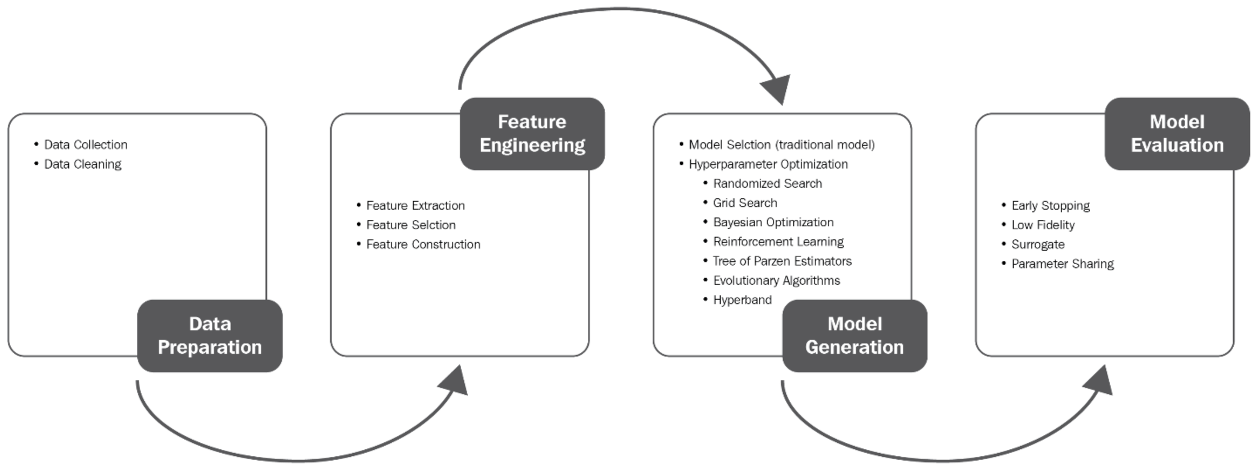
Project	Type	License
Auto-Keras	NAS	Custom
AutoML Vision	NAS	Commercial
AutoML Video Intelligence	NAS	Commercial
AutoML Natural Language	NAS	Commercial
AutoML Translation	NAS	Commercial
AutoML Tables	AutoFE, HPO	Commercial
auto-sklearn	HPO	Custom
auto_ml	HPO	MIT
BayesianOptimization	HPO	MIT
comet	HPO	Commercial
DataRobot	HPO	Commercial
Driverless AI	AutoFE	Commercial
H2O AutoML	HPO	Apache-2.0
Katib	HPO	Apache-2.0
MLJAR	HPO	Commercial
NNI	HPO, NAS	MIT
TPOT	AutoFE, HPO	LGPL-3.0
TransmogrifAI	HPO	BSD-3-Clause
MLBox	AutoFE, HPO	BSD-3 License
AutoAI Watson	AutoFE, HPO	Commercial

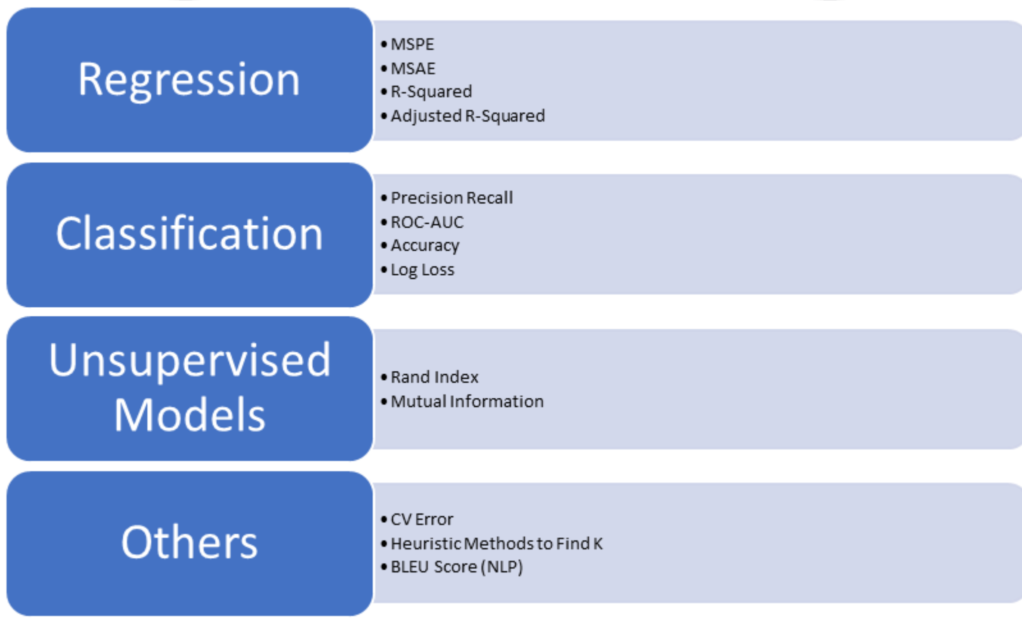
Chapter 2: Automated Machine Learning, Algorithms, and Techniques



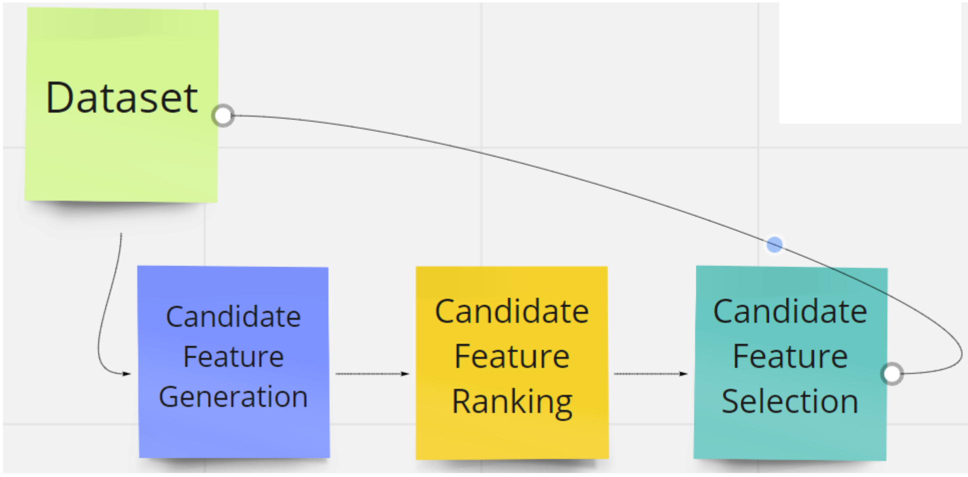





Data Preparation
Data Collection
Data Cleaning
Feature Engineering
Feature Extraction
Feature Selction
Feature Construction
Model Generation
Model Selction (traditional model)
Hyperparameter Optimization
Randomized Search
Grid Search
Bayesian Optimization
Reinforcement Learning
Tree of Parzen Estimators
Evolutionary Algorithms
Hyperband
Model Evaluation
Early Stopping
Low Fidelity
Surrogate
Parameter Sharing


Regression
MSPE
MSAE
R-Squared
Adjusted R-Squared
Classification
Precision Recall
ROC-AUC
Accuracy
Log Loss
Unsupervised Models
Rand Index
Mutual Information
Others
CV Error
Heuristic Methods to Find K
BLEU Score (NLP)

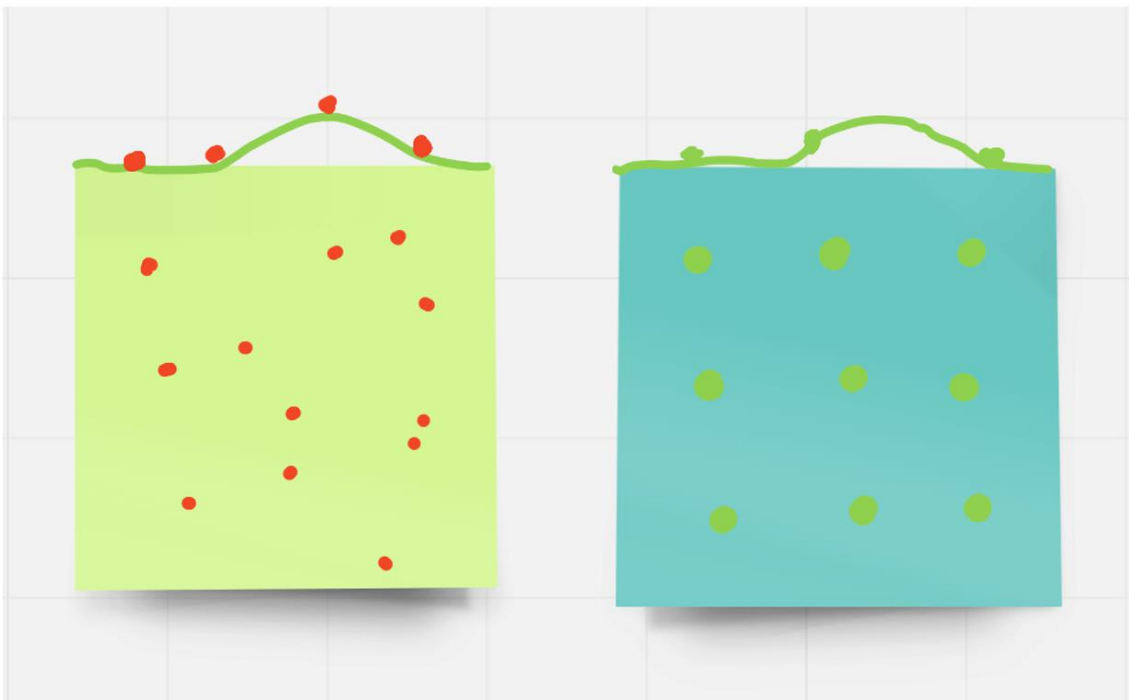
	Bayesian Optimization	Reinforcement Learning	Evolutionary Algorithms	Gradient - Based Approaches	Frameworks
Automated Feature Engineering		FeatureRL	GP (Genetic Programming) for Feature Engineering		FeatureTools
Automated Model and Hyper Parameter Search	TPE - Tree of Parzen Estimators SMAC (Sequential Model-Based Optimization for General Algorithm Configuration) Auto-SKLearn FABOLAS Fast Bayesian Optimization of Machine Learning Hyperparameters on Large Datasets BOHB: Robust and Efficient Hyperparameter Optimization at Scale	APRL (Autonomous Predictive Modeler via Reinforcement Learning) Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization	TPOT – Tree-based pipeline optimization. AutoStacker - Automatic Evolutionary Hierarchical Machine Learning System DarwinML - Graph-based Evolutionary Algorithm for Automated Machine Learning.		Hyperopt: Distributed Asynchronous Hyper-Parameter Optimization SMAC (Sequential Model-Based Optimization for General Algorithm Configuration) Auto-Sklearn TPOT – Tree based pipeline optimization.
Automated Deep Learning or Neural Architecture Search	AutoKeras NASBot	NAS – Neural Architecture Search NASNET (Neural Architecture Search Network) ENAS - Efficient Neural Architecture Search via Parameter Sharing		DARTS: Differentiable Architecture Search ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware NAONet (Neural Architecture Optimization NET)	AutoKeras AdaNet Neural Network Intelligence (NNI)



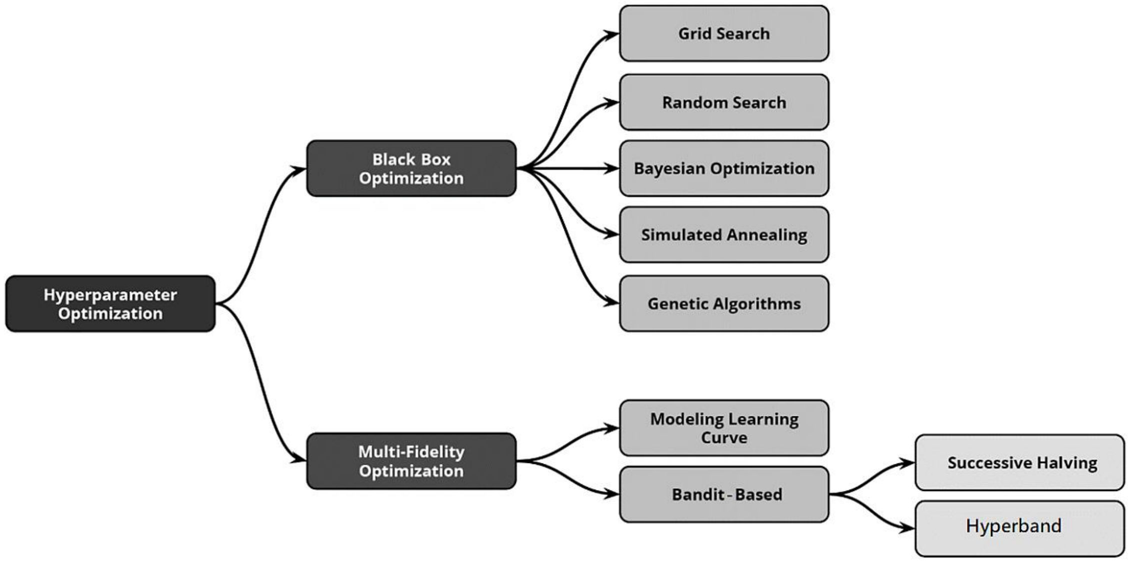
Random Layout

Grid Layout

Unimportant Parameters

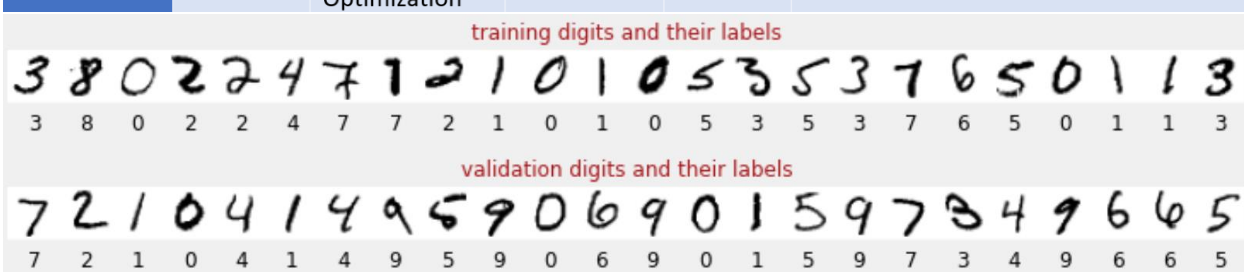


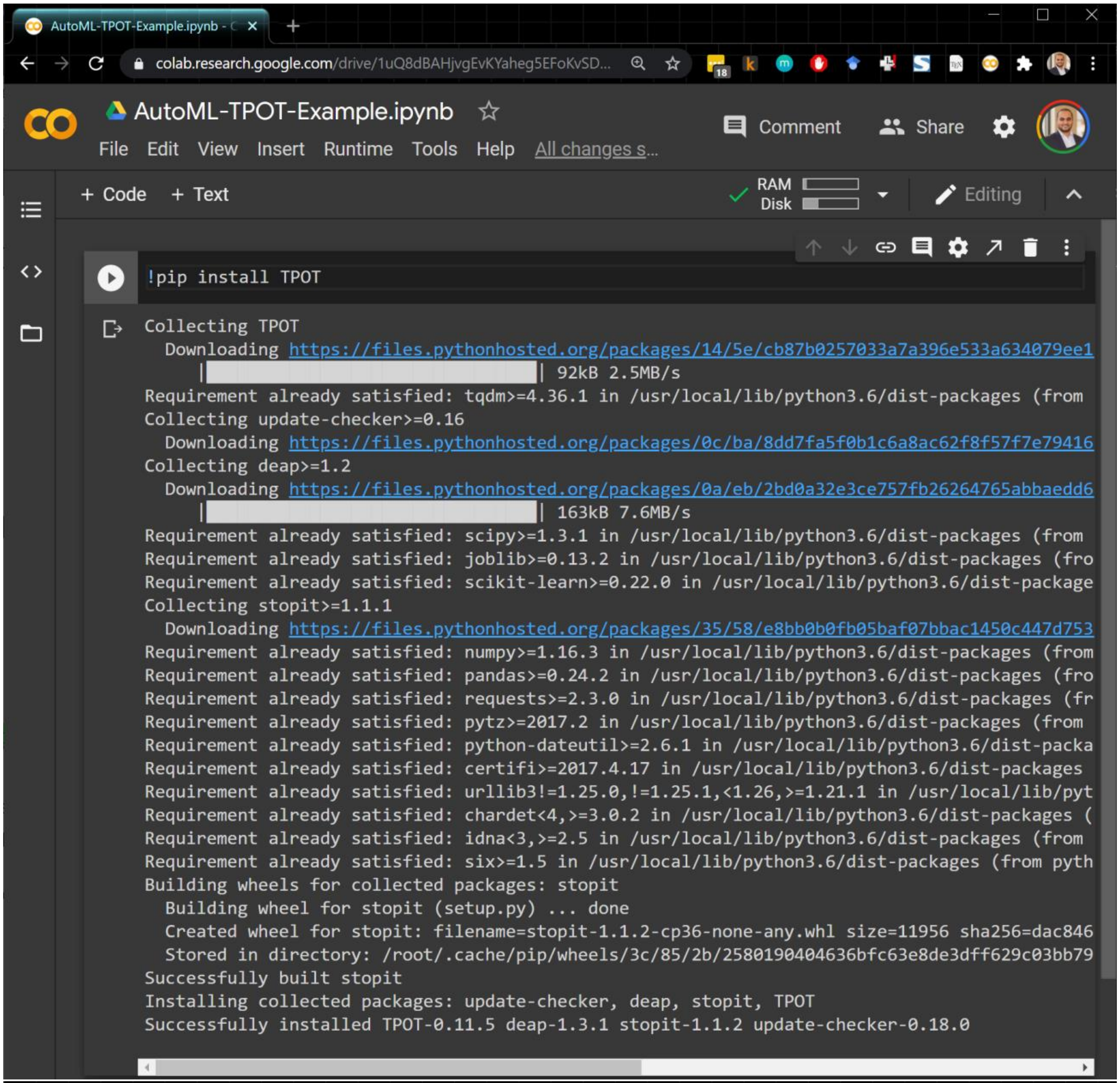
Important Parameters

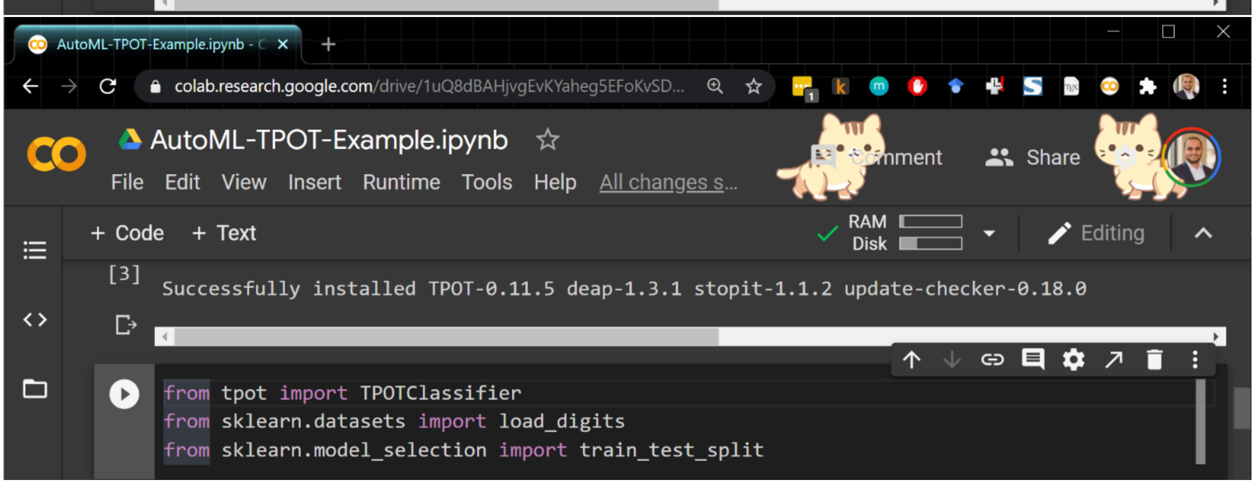

Hyperparameter Optimization
Black Box Optimization
Grid Search
Random Search
Bayesian Optimization
Simulated Annealing
Genetic Algorithms
Multi-Fidelity Optimization
Modeling Learning Curve
Bandit - Based
Successive Halving
Hyperband

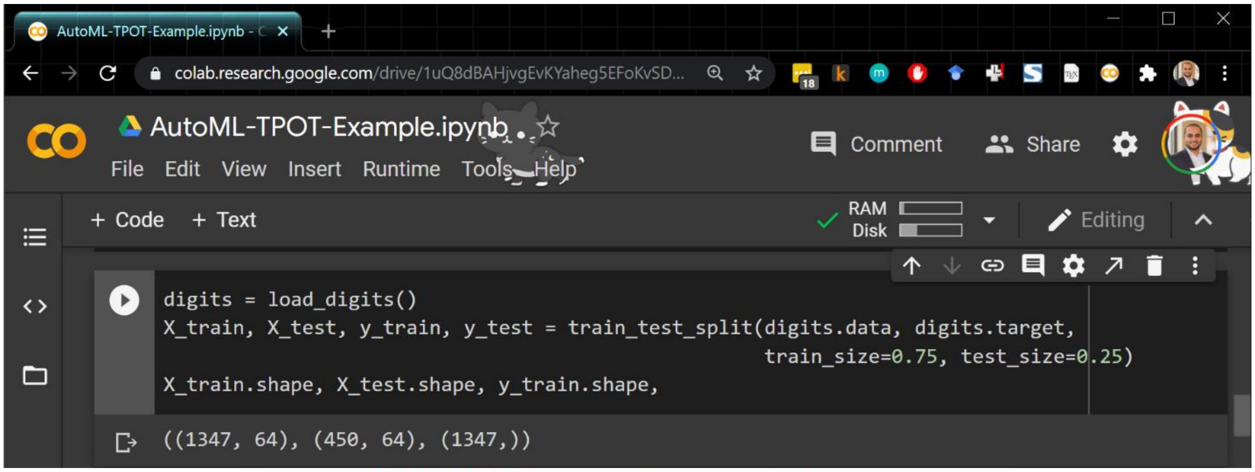
Chapter 3: Automated Machine Learning with Open Source Tools and Libraries

	Language	Automated Machine Learning Technique	Automated Feature Extraction	Meta Learning	Link
AutoWeka	Java	Bayesian Optimization	Yes	No	https://github.com/automl/autoweka
AutoSklearn	Python	Bayesian Optimization	Yes	Yes	https://automl.github.io/auto-sklearn/master/
TPOT	Python	Genetic Algorithm	Yes	No	http://epistasislab.github.io/tpot/
Hyperopt-Sklearn	Python	Bayesian Optimization & Random Search	Yes	No	https://github.com/hyperopt/hyperopt-sklearn
AutoStacker	Python	Genetic Algorithm	Yes	No	https://arxiv.org/abs/1803.00684
AlphaD3M	Python	Reinforcement Learning	Yes	Yes	https://www.cs.columbia.edu/~idrori/AlphaD3M.pdf
OBOE	Python	Collaborative Filtering	No	Yes	https://github.com/udellgroup/oboe
PMF	Python	Collaborative Filtering & Bayesian Optimization	Yes	Yes	https://github.com/rsheth80/pmf-automl

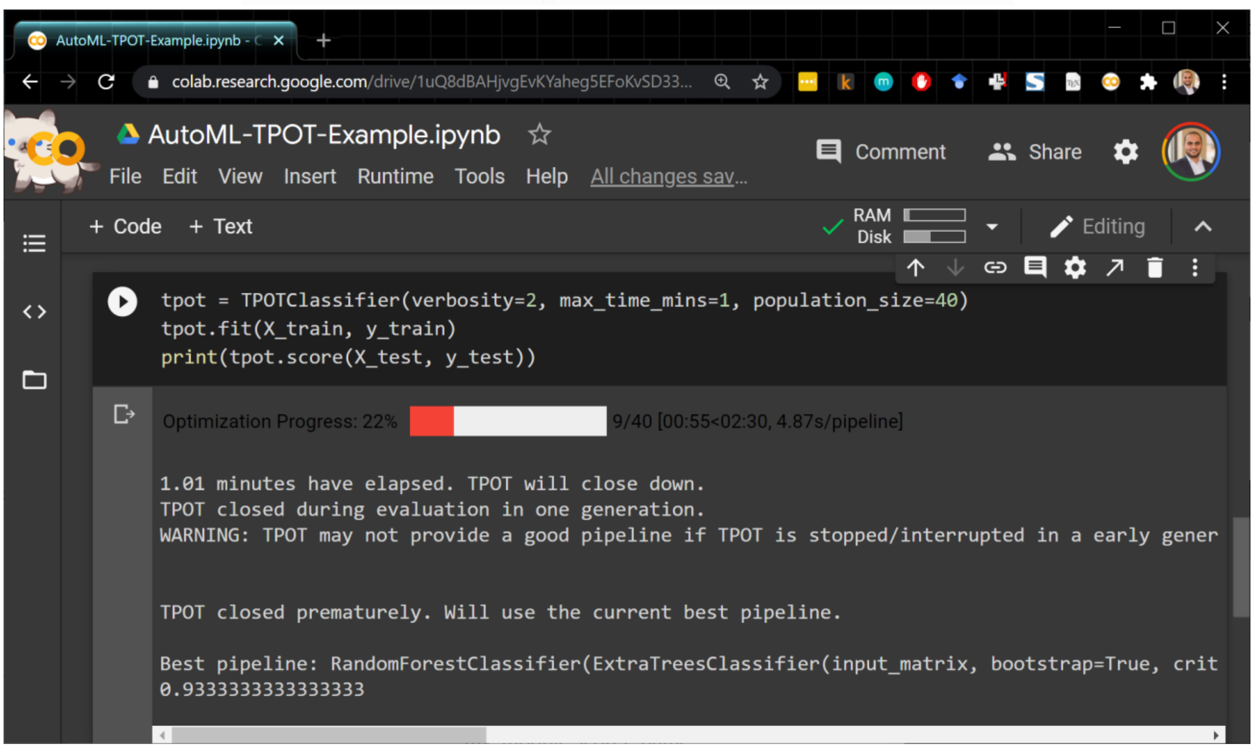


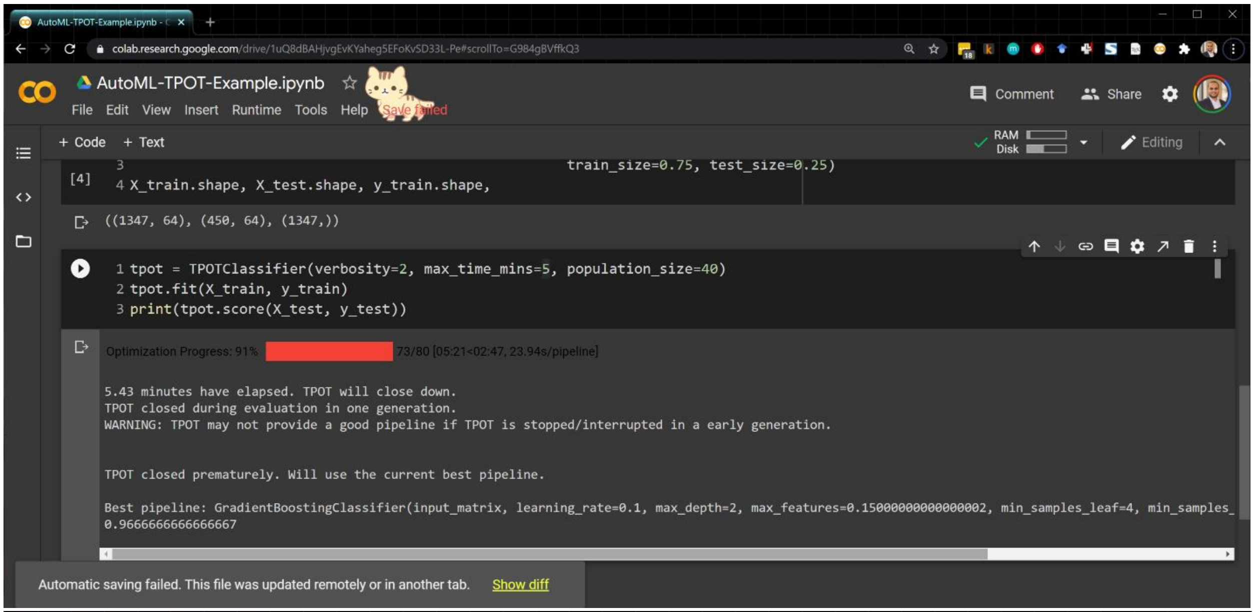

AutoML-TPOT-Example.ipynb - C
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD...
AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help All changes s...
Comment
Share
+ Code + Text
RAM
Disk
Editing
!pip install TPOT
Collecting TPOT
Downloading https://files.pythonhosted.org/packages/14/5e/cb87b0257033a7a396e533a634079ee1
| 92kB 2.5MB/s
Requirement already satisfied: tqdm>=4.36.1 in /usr/local/lib/python3.6/dist-packages (from
Collecting update-checker>=0.16
Downloading https://files.pythonhosted.org/packages/0c/ba/8dd7fa5f0b1c6a8ac62f8f57f7e79416
Collecting deap>=1.2
Downloading https://files.pythonhosted.org/packages/0a/eb/2bd0a32e3ce757fb26264765abbaedd6
| 163kB 7.6MB/s
Requirement already satisfied: scipy>=1.3.1 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: joblib>=0.13.2 in /usr/local/lib/python3.6/dist-packages (fro
Requirement already satisfied: scikit-learn>=0.22.0 in /usr/local/lib/python3.6/dist-package
Collecting stopit>=1.1.1
Downloading https://files.pythonhosted.org/packages/35/58/e8bb0b0fb05baf07bbac1450c447d753
Requirement already satisfied: numpy>=1.16.3 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: pandas>=0.24.2 in /usr/local/lib/python3.6/dist-packages (fro
Requirement already satisfied: requests>=2.3.0 in /usr/local/lib/python3.6/dist-packages (fr
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.6/dist-packa
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/pyt
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-packages (from pyth
Building wheels for collected packages: stopit
Building wheel for stopit (setup.py) ... done
Created wheel for stopit: filename=stopit-1.1.2-cp36-none-any.whl size=11956 sha256=dac846
Stored in directory: /root/.cache/pip/wheels/3c/85/2b/2580190404636bfc63e8de3dff629c03bb79
Successfully built stopit
Installing collected packages: update-checker, deap, stopit, TPOT
Successfully installed TPOT-0.11.5 deap-1.3.1 stopit-1.1.2 update-checker-0.18.0

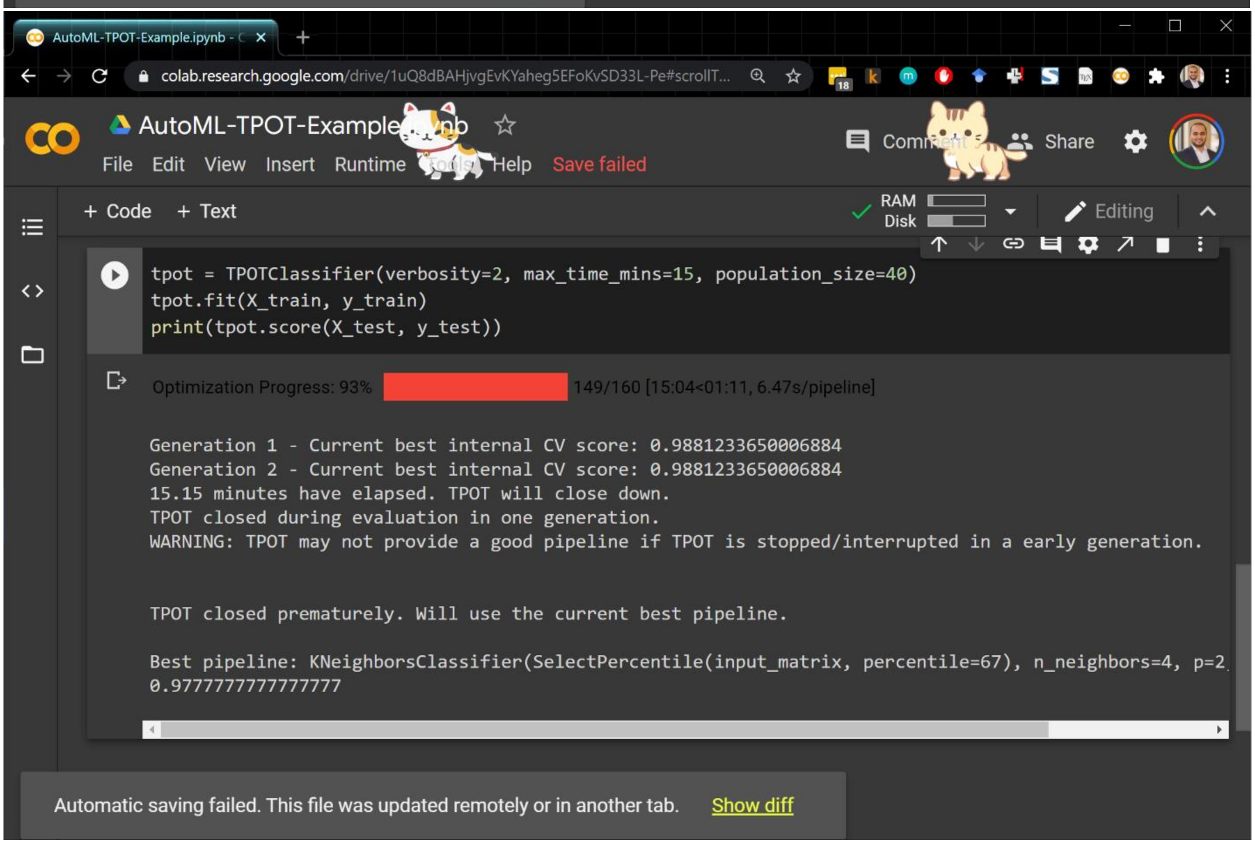

AutoML-TPOT-Example.ipynb - C
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD...
AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help All changes s...
Share
+ Code + Text
RAM
Disk
Editing
[3]
Successfully installed TPOT-0.11.5 deap-1.3.1 stopit-1.1.2 update-checker-0.18.0
from tpot import TPOTClassifier
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split

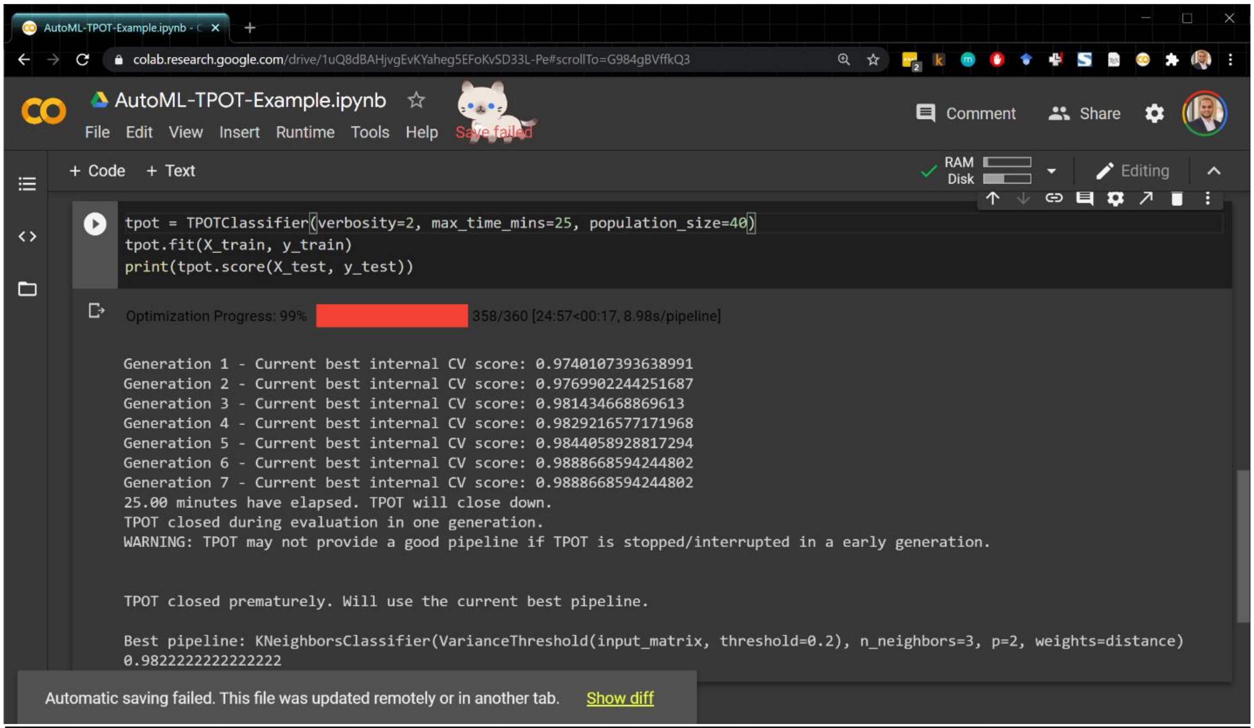


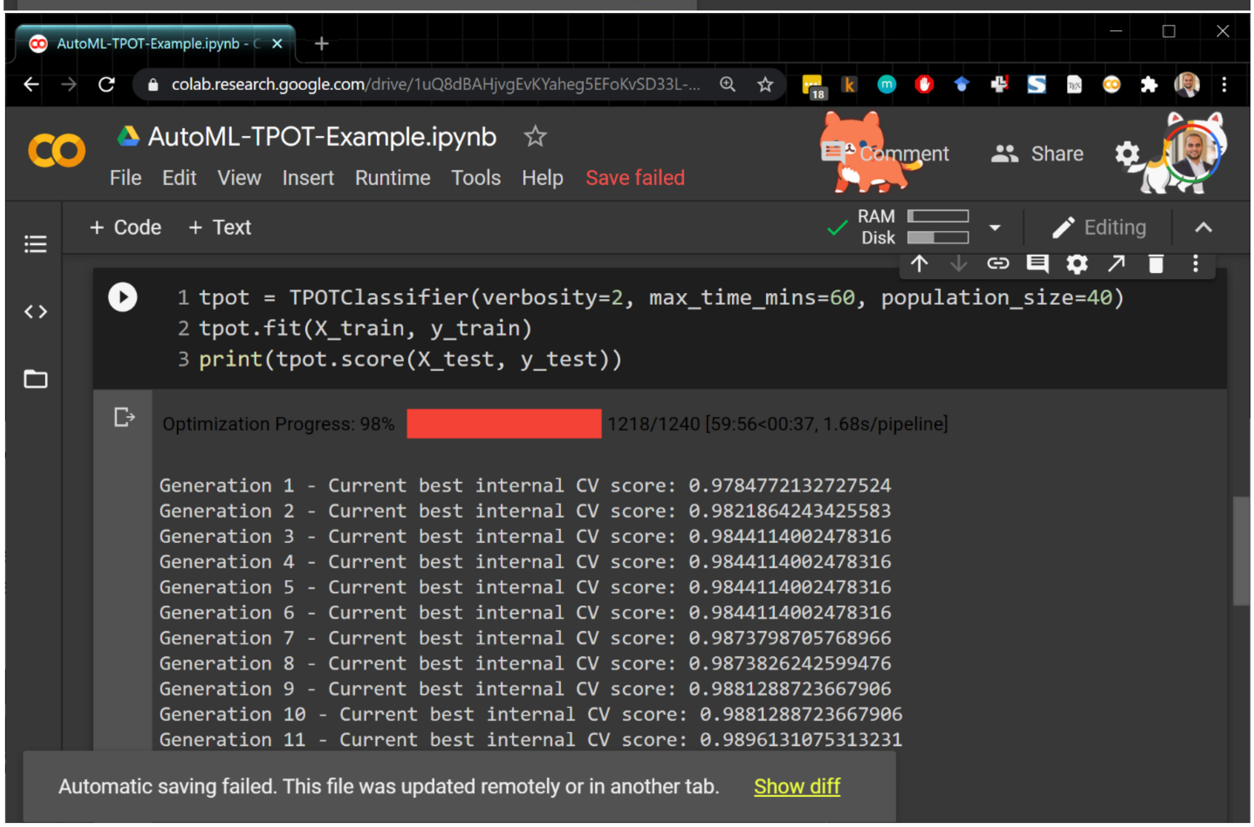

```
class tpot.TPOTClassifier(generations=100, population_size=100,
                          offspring_size=None, mutation_rate=0.9,
                          crossover_rate=0.1,
                          scoring='accuracy', cv=5,
                          subsample=1.0, n_jobs=1,
                          max_time_mins=None, max_eval_time_mins=5,
                          random_state=None, config_dict=None,
                          template=None,
                          warm_start=False,
                          memory=None,
                          use_dask=False,
                          periodic_checkpoint_folder=None,
                          early_stop=None,
                          verbosity=0,
                          disable_update_check=False,
                          log_file=None
                          )
```

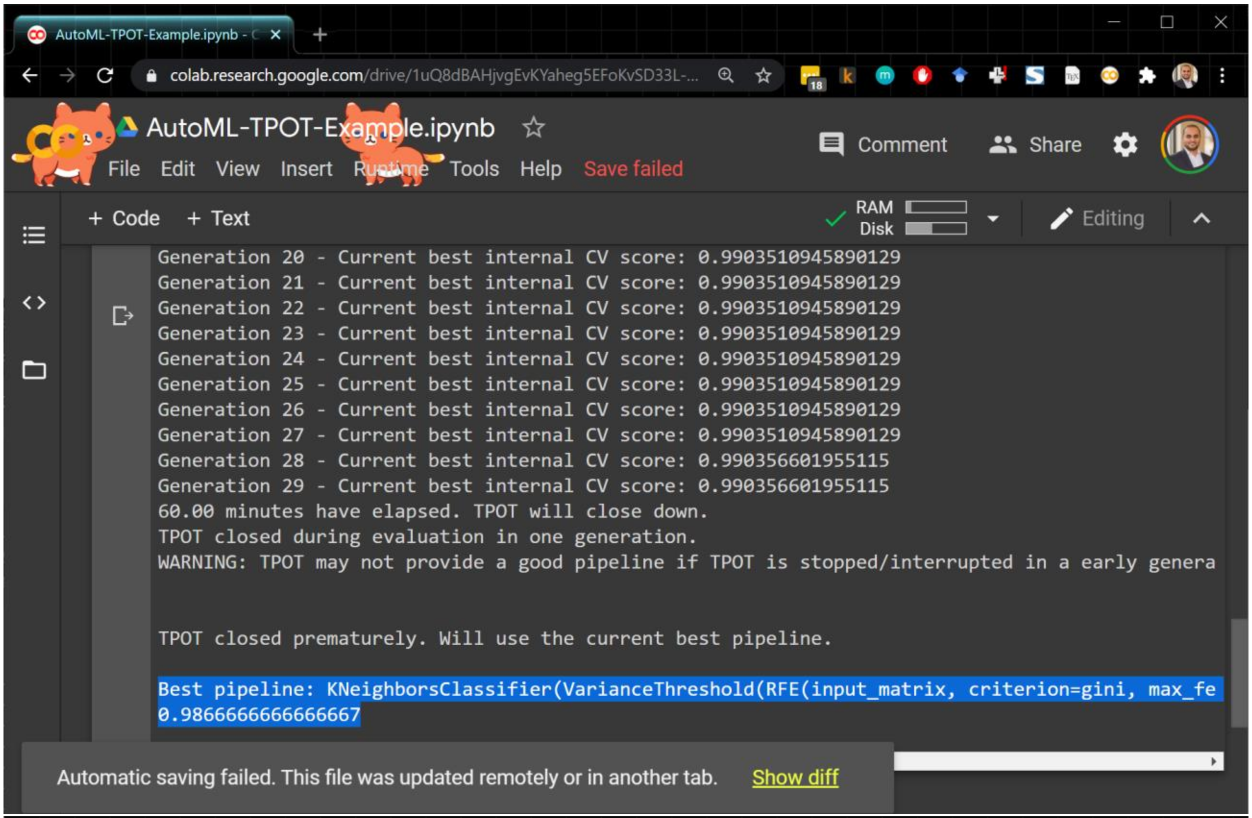
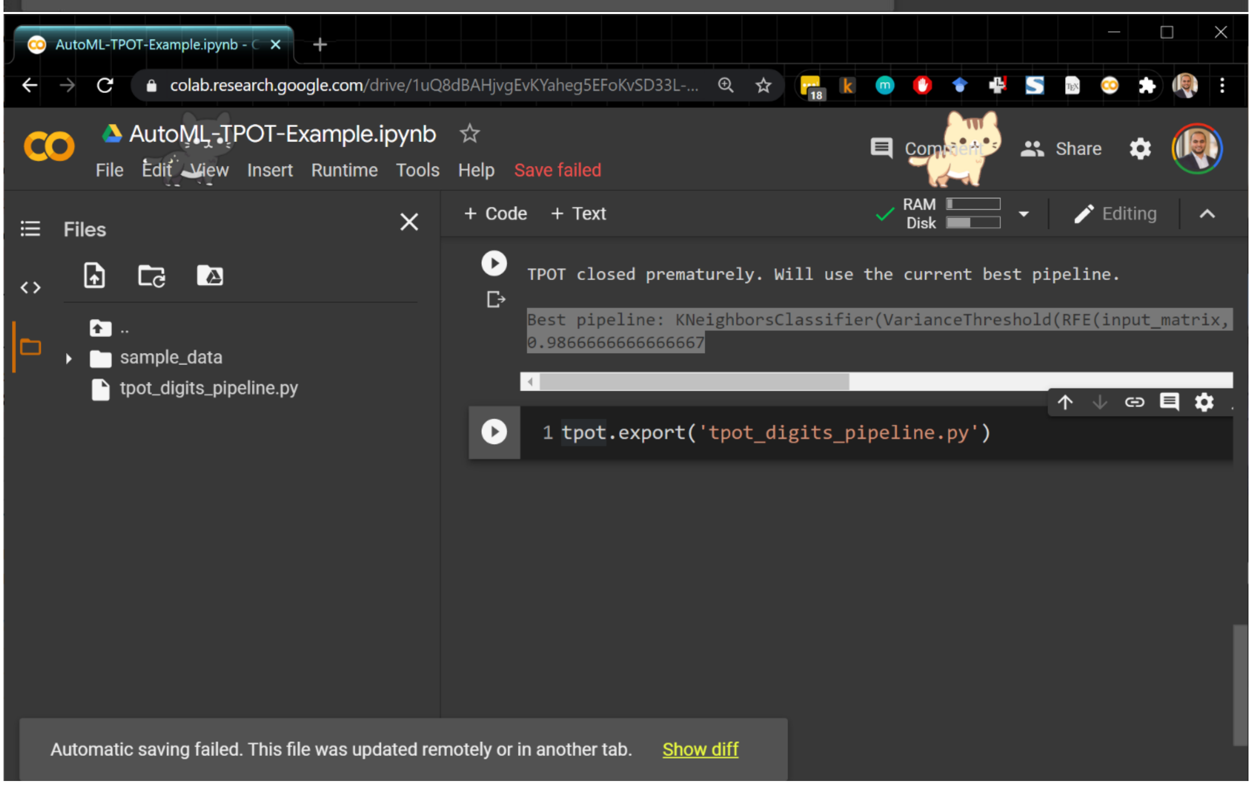


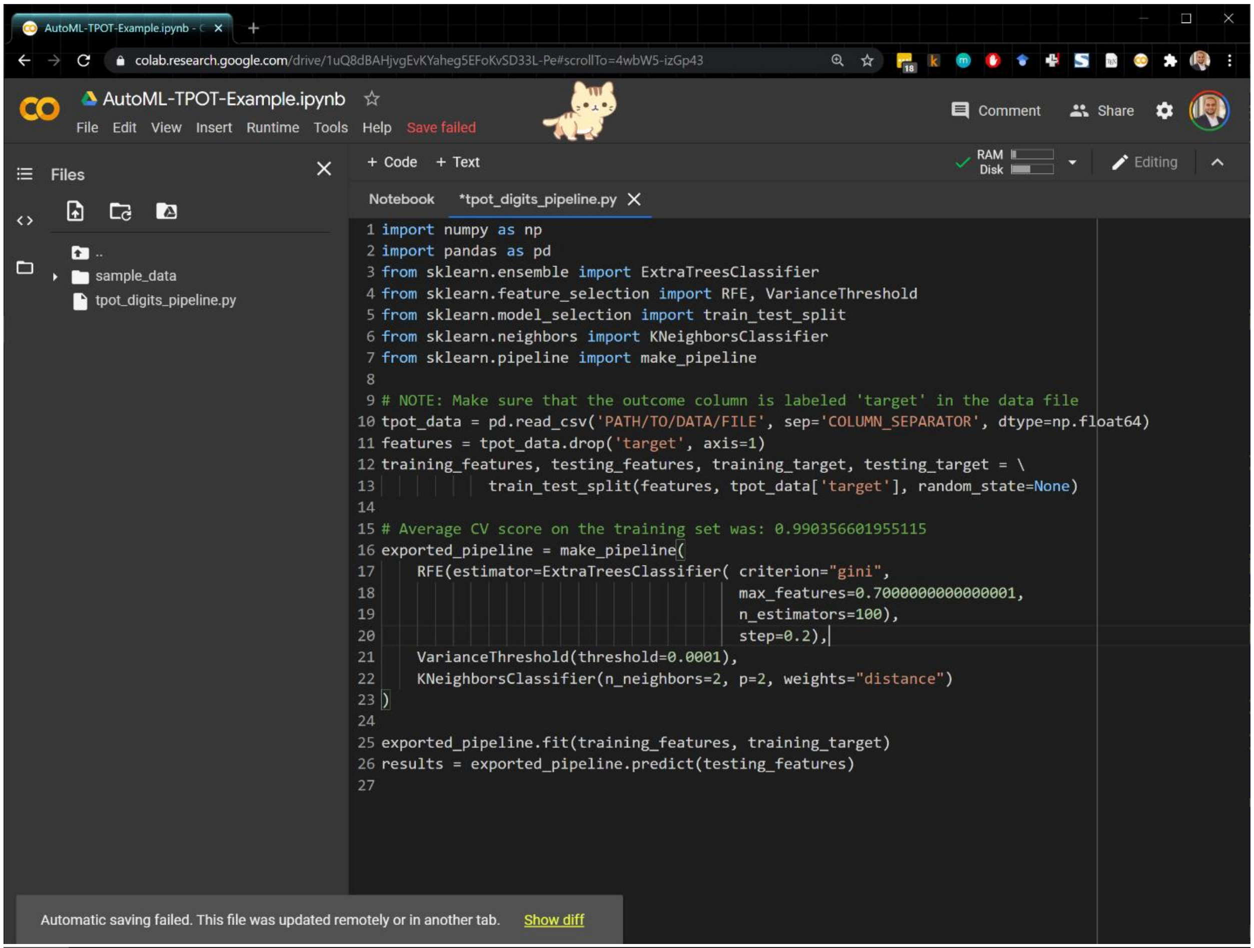

AutoML-TPOT-Example.ipynb
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD33L-Pe#scrollTo=G984gBVffkQ3
File Edit View Insert Runtime Tools Help Save failed
Comment Share
+ Code + Text
RAM Disk
Editing
train_size=0.75, test_size=0.25)
[4]
X_train.shape, X_test.shape, y_train.shape,
((1347, 64), (450, 64), (1347,))
tpot = TPOTClassifier(verbosity=2, max_time_mins=5, population_size=40)
tpot.fit(X_train, y_train)
print(tpot.score(X_test, y_test))
Optimization Progress: 91%
73/80 [05:21<02:47, 23.94s/pipeline]
5.43 minutes have elapsed. TPOT will close down.
TPOT closed during evaluation in one generation.
WARNING: TPOT may not provide a good pipeline if TPOT is stopped/interrupted in a early generation.
TPOT closed prematurely. Will use the current best pipeline.
Best pipeline: GradientBoostingClassifier(input_matrix, learning_rate=0.1, max_depth=2, max_features=0.15000000000000002, min_samples_leaf=4, min_samples_
0.9666666666666667
Automatic saving failed. This file was updated remotely or in another tab.
Show diff


AutoML-TPOT-Example.ipynb
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD33L-Pe#scrollT...
File Edit View Insert Runtime Help Save failed
Share
+ Code + Text
RAM Disk
Editing
tpot = TPOTClassifier(verbosity=2, max_time_mins=15, population_size=40)
tpot.fit(X_train, y_train)
print(tpot.score(X_test, y_test))
Optimization Progress: 93%
149/160 [15:04<01:11, 6.47s/pipeline]
Generation 1 - Current best internal CV score: 0.9881233650006884
Generation 2 - Current best internal CV score: 0.9881233650006884
15.15 minutes have elapsed. TPOT will close down.
TPOT closed during evaluation in one generation.
WARNING: TPOT may not provide a good pipeline if TPOT is stopped/interrupted in a early generation.
TPOT closed prematurely. Will use the current best pipeline.
Best pipeline: KNeighborsClassifier(SelectPercentile(input_matrix, percentile=67), n_neighbors=4, p=2
0.9777777777777777
Automatic saving failed. This file was updated remotely or in another tab.
Show diff

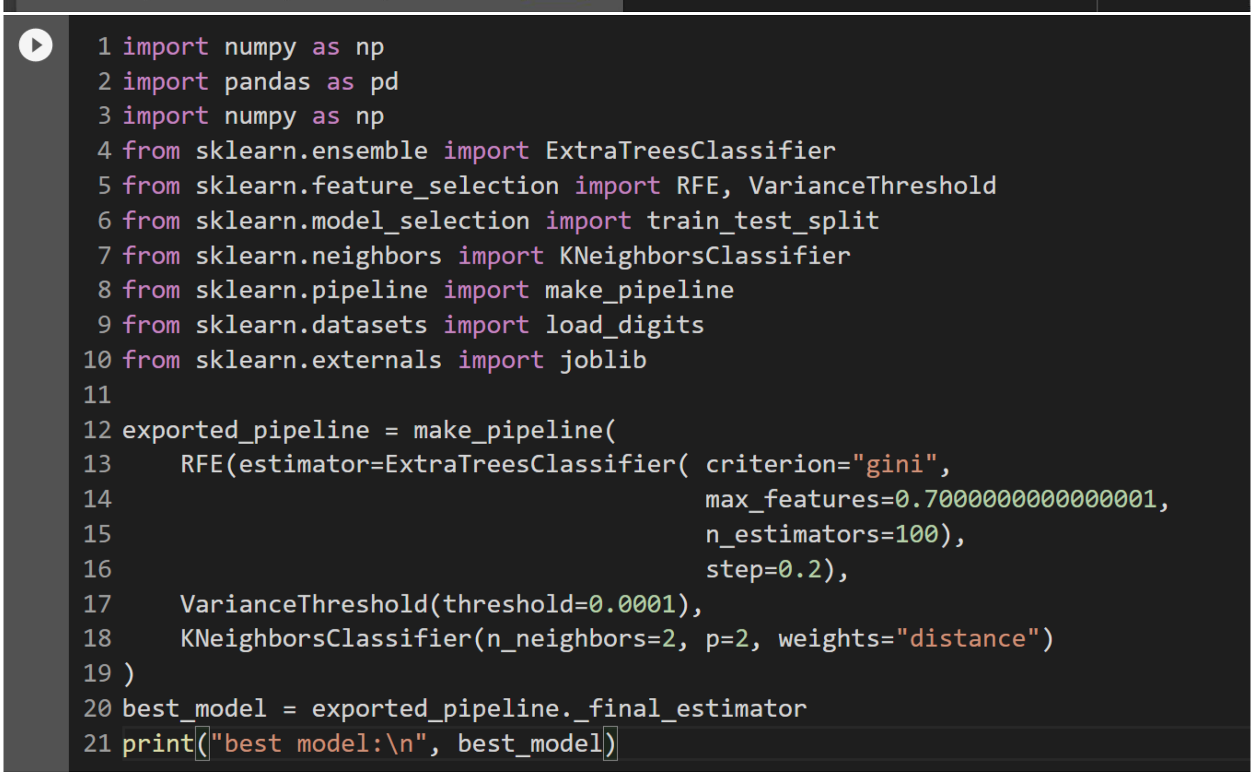

AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
+ Code + Text
RAM Disk
Editing
tpot = TPOTClassifier(verbosity=2, max_time_mins=25, population_size=40)
tpot.fit(X_train, y_train)
print(tpot.score(X_test, y_test))
Optimization Progress: 99% 358/360 [24:57<00:17, 8.98s/pipeline]
Generation 1 - Current best internal CV score: 0.9740107393638991
Generation 2 - Current best internal CV score: 0.9769902244251687
Generation 3 - Current best internal CV score: 0.981434668869613
Generation 4 - Current best internal CV score: 0.9829216577171968
Generation 5 - Current best internal CV score: 0.9844058928817294
Generation 6 - Current best internal CV score: 0.9888668594244802
Generation 7 - Current best internal CV score: 0.9888668594244802
25.00 minutes have elapsed. TPOT will close down.
TPOT closed during evaluation in one generation.
WARNING: TPOT may not provide a good pipeline if TPOT is stopped/interrupted in a early generation.
TPOT closed prematurely. Will use the current best pipeline.
Best pipeline: KNeighborsClassifier(VarianceThreshold(input_matrix, threshold=0.2), n_neighbors=3, p=2, weights=distance)
0.9822222222222222
Automatic saving failed. This file was updated remotely or in another tab. Show diff


AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
+ Code + Text
RAM Disk
Editing
1 tpot = TPOTClassifier(verbosity=2, max_time_mins=60, population_size=40)
2 tpot.fit(X_train, y_train)
3 print(tpot.score(X_test, y_test))
Optimization Progress: 98% 1218/1240 [59:56<00:37, 1.68s/pipeline]
Generation 1 - Current best internal CV score: 0.9784772132727524
Generation 2 - Current best internal CV score: 0.9821864243425583
Generation 3 - Current best internal CV score: 0.9844114002478316
Generation 4 - Current best internal CV score: 0.9844114002478316
Generation 5 - Current best internal CV score: 0.9844114002478316
Generation 6 - Current best internal CV score: 0.9844114002478316
Generation 7 - Current best internal CV score: 0.9873798705768966
Generation 8 - Current best internal CV score: 0.9873826242599476
Generation 9 - Current best internal CV score: 0.9881288723667906
Generation 10 - Current best internal CV score: 0.9881288723667906
Generation 11 - Current best internal CV score: 0.9896131075313231
Automatic saving failed. This file was updated remotely or in another tab. Show diff


AutoML-TPOT-Example.ipynb
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD33L-...
AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
+ Code + Text
RAM
Disk
Editing
Generation 20 - Current best internal CV score: 0.9903510945890129
Generation 21 - Current best internal CV score: 0.9903510945890129
Generation 22 - Current best internal CV score: 0.9903510945890129
Generation 23 - Current best internal CV score: 0.9903510945890129
Generation 24 - Current best internal CV score: 0.9903510945890129
Generation 25 - Current best internal CV score: 0.9903510945890129
Generation 26 - Current best internal CV score: 0.9903510945890129
Generation 27 - Current best internal CV score: 0.9903510945890129
Generation 28 - Current best internal CV score: 0.990356601955115
Generation 29 - Current best internal CV score: 0.990356601955115
60.00 minutes have elapsed. TPOT will close down.
TPOT closed during evaluation in one generation.
WARNING: TPOT may not provide a good pipeline if TPOT is stopped/interrupted in a early genera
TPOT closed prematurely. Will use the current best pipeline.
Best pipeline: KNeighborsClassifier(VarianceThreshold(RFE(input_matrix, criterion=gini, max_fe
0.9866666666666667
Automatic saving failed. This file was updated remotely or in another tab. Show diff

AutoML-TPOT-Example.ipynb
colab.research.google.com/drive/1uQ8dBAHjvgEvKYaheg5EFoKvSD33L-...
AutoML-TPOT-Example.ipynb
File Edit View Insert Runtime Tools Help Save failed
Share
Files
+ Code + Text
RAM
Disk
Editing
..
sample_data
tpot_digits_pipeline.py
TPOT closed prematurely. Will use the current best pipeline.
Best pipeline: KNeighborsClassifier(VarianceThreshold(RFE(input_matrix,
0.9866666666666667
1 tpot.export('tpot_digits_pipeline.py')
Automatic saving failed. This file was updated remotely or in another tab. Show diff



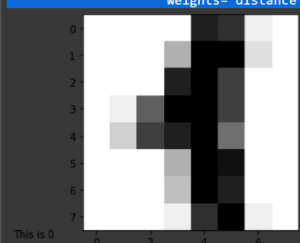

```python
import numpy as np
import pandas as pd
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.feature_selection import RFE, VarianceThreshold
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline

# NOTE: Make sure that the outcome column is labeled 'target' in the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE', sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target, testing_target = \
            train_test_split(features, tpot_data['target'], random_state=None)

# Average CV score on the training set was: 0.990356601955115
exported_pipeline = make_pipeline(
    RFE(estimator=ExtraTreesClassifier( criterion="gini",
                                        max_features=0.7000000000000001,
                                        n_estimators=100),
                                        step=0.2),
    VarianceThreshold(threshold=0.0001),
    KNeighborsClassifier(n_neighbors=2, p=2, weights="distance")
)

exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```



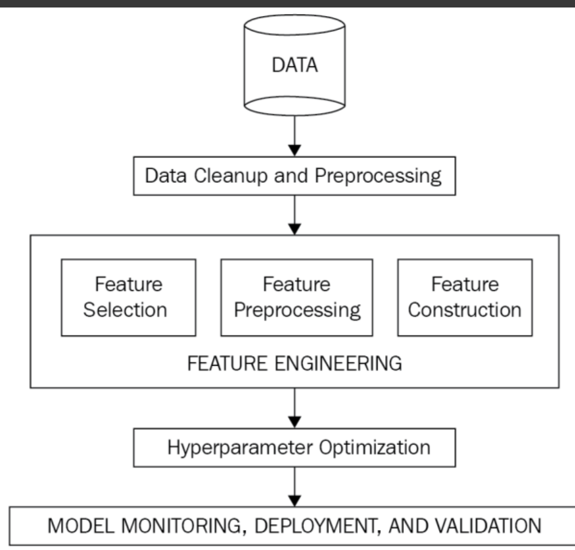
```python
import numpy as np
import pandas as pd
import numpy as np
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.feature_selection import RFE, VarianceThreshold
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline
from sklearn.datasets import load_digits
from sklearn.externals import joblib

exported_pipeline = make_pipeline(
    RFE(estimator=ExtraTreesClassifier( criterion="gini",
                                        max_features=0.7000000000000001,
                                        n_estimators=100),
                                        step=0.2),
    VarianceThreshold(threshold=0.0001),
    KNeighborsClassifier(n_neighbors=2, p=2, weights="distance")
)
best_model = exported_pipeline._final_estimator
print("best model:\n", best_model)
```

```python
arr = np.zeros(64).reshape(1,64)
arr[0] = digits.images[11].reshape(1, 64)
fig = plt.figure()
plt.imshow(digits.images[11],cmap = plt.cm.gray_r)
txt = "This is %d"%digits.target[10]
fig.text(0.1,0.1,txt)
plt.show()

exported_pipeline.fit(training_features, training_target)
digits = load_digits()
training_features, testing_features, training_target, testing_target = train_test_split(digit
                                                    train_size=0.8, test_size=0.2)

results = exported_pipeline.predict(arr)
print ("The number is predicted to be " + str(results))

joblib.dump(exported_pipeline, 'digits_model.pkl')
```

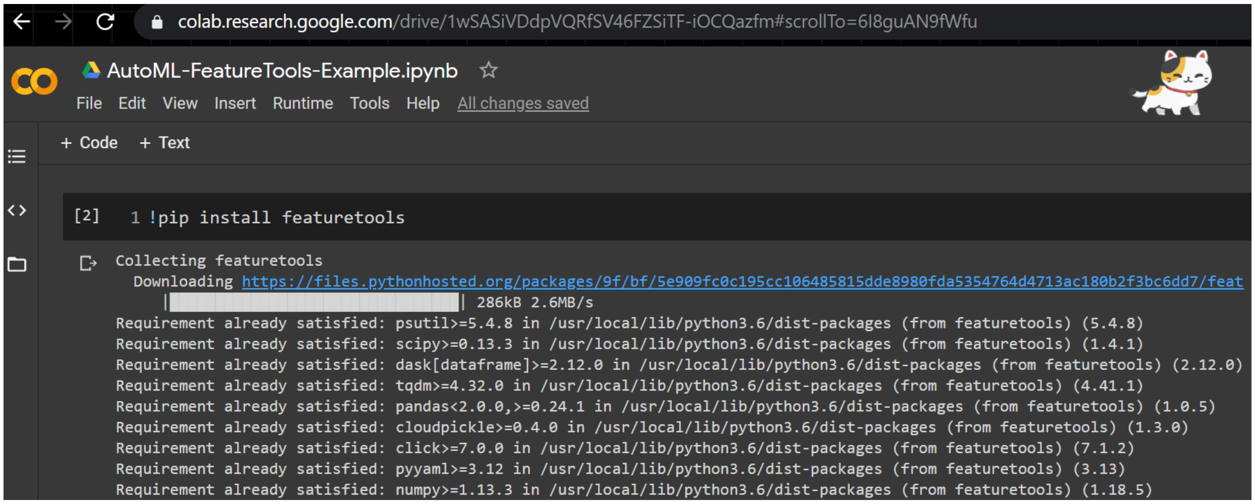
```
best model:
 KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                      metric_params=None, n_jobs=None, n_neighbors=2, p=2,
                      weights='distance')
```




```
The number is predicted to be [1]
['digits_model.pkl']
```



Supervised Classification Operators	Feature Preprocessing Operators	Feature Selection Operators
Decision Tree, RandomForest, eXtreme Gradient Boosting Classifier, LogisticRegression, and KNearestNeighborClassifier.	StandardScaler, RobustScaler, MinMaxScaler, MaxAbsScaler, RandomizedPCA, Binarizer, and PolynomialFeatures.	VarianceThreshold, SelectKBest, SelectPercentile, SelectFwe, and Recursive Feature Elimination (RFE).
Classification operators store the classifier's predictions as a new feature as well as the classification for the pipeline.	Preprocessing operators modify the dataset in some way and return the modified dataset.	Feature selection operators reduce the number of features in the data set using some criteria and return the modified dataset.



7.2.1. Boston house prices dataset

Data Set Characteristics:

Number of Instances:	506
Number of Attributes:	13 numeric/categorical predictive. Median Value (attribute 14) is usually the target.
Attribute Information (in order):	• CRIM per capita crime rate by town • ZN proportion of residential land zoned for lots over 25,000 sq.ft. • INDUS proportion of non-retail business acres per town • CHAS Charles River dummy variable (= 1 if tract bounds river; 0 otherwise) • NOX nitric oxides concentration (parts per 10 million) • RM average number of rooms per dwelling • AGE proportion of owner-occupied units built prior to 1940 • DIS weighted distances to five Boston employment centres • RAD index of accessibility to radial highways • TAX full-value property-tax rate per $10,000 • PTRATIO pupil-teacher ratio by town • B 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town • LSTAT % lower status of the population • MEDV Median value of owner-occupied homes in $1000's
Missing Attribute Values:	None
Creator:	Harrison, D. and Rubinfeld, D.L.

This is a copy of UCI ML housing dataset. https://archive.ics.uci.edu/ml/machine-learning-databases/housing/

```python
from sklearn.datasets import load_boston
import pandas as pd
import featuretools as ft
```

```python
# Load data and put into dataframe
boston = load_boston()
df = pd.DataFrame(boston.data, columns = boston.feature_names)
df['MEDV'] = boston.target
print (df.head(5))
```

```
      CRIM    ZN  INDUS  CHAS    NOX  ...    TAX  PTRATIO       B  LSTAT  MEDV
0  0.00632  18.0   2.31   0.0  0.538  ...  296.0     15.3  396.90   4.98  24.0
1  0.02731   0.0   7.07   0.0  0.469  ...  242.0     17.8  396.90   9.14  21.6
2  0.02729   0.0   7.07   0.0  0.469  ...  242.0     17.8  392.83   4.03  34.7
3  0.03237   0.0   2.18   0.0  0.458  ...  222.0     18.7  394.63   2.94  33.4
4  0.06905   0.0   2.18   0.0  0.458  ...  222.0     18.7  396.90   5.33  36.2

[5 rows x 14 columns]
```

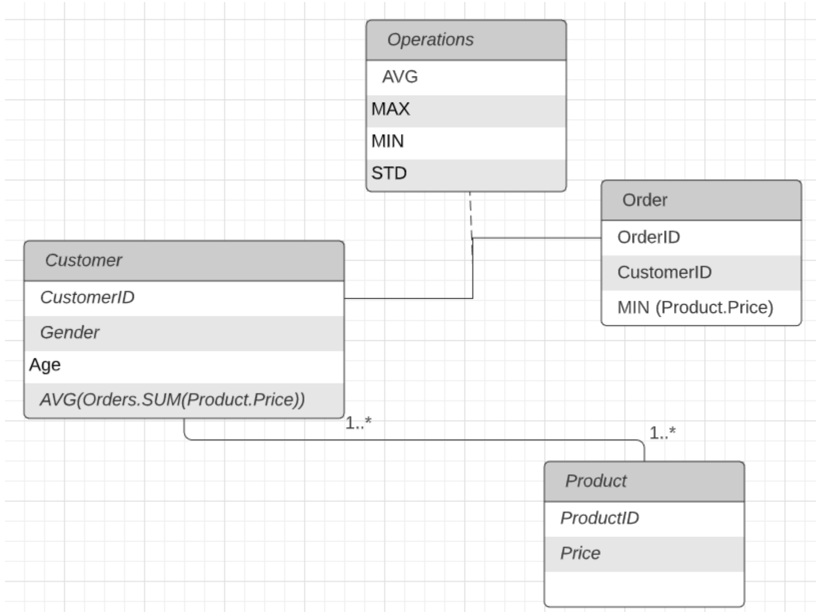
```python
# Make an entityset and add the entity
es = ft.EntitySet(id = 'boston')
es.entity_from_dataframe(entity_id = 'data', dataframe = df,
                         make_index = True, index = 'index')

# Run deep feature synthesis with transformation primitives
feature_matrix, feature_defs = ft.dfs(entityset = es, target_entity = 'data',
                                      trans_primitives = ['add_numeric', 'multiply_numeric'])
```

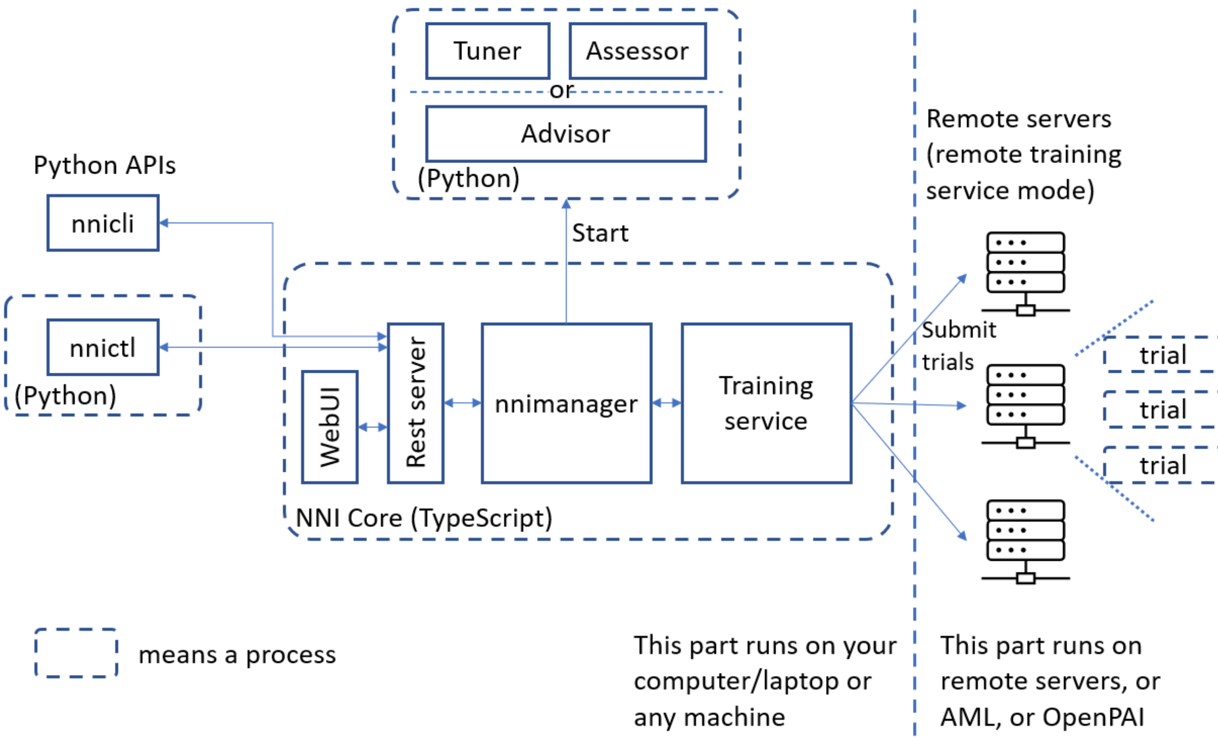
```python
                                      trans_primitives = ['add_numeric', 'multiply_numeric'])

feature_matrix.head()
```

index	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	AGE + B	AGE + CHAS	AGE + CRIM	AGE + DIS	AGE + INDUS	AGE + LSTAT	AGE + MEDV	AGE + NOX	AGE + PTRATIO	AGE + RAD	AGE + RM	AGE + TAX	AGE + ZN	B + CHAS	B + CRIM	B + DIS	B + INDUS
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0	462.10	65.2	65.20632	69.2900	67.51	70.18	89.2	65.738	80.5	66.2	71.775	361.2	83.2	396.90	396.90632	400.9900	399.21
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6	475.80	78.9	78.92731	83.8671	85.97	88.04	100.5	79.369	96.7	80.9	85.321	320.9	78.9	396.90	396.92731	401.8671	403.97
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7	453.93	61.1	61.12729	66.0671	68.17	65.13	95.8	61.569	78.9	63.1	68.285	303.1	61.1	392.83	392.85729	397.7971	399.90
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4	440.43	45.8	45.83237	51.8622	47.98	48.74	79.2	46.258	64.5	48.8	52.798	267.8	45.8	394.63	394.66237	400.6922	396.81
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2	451.10	54.2	54.26905	60.2622	56.38	59.53	90.4	54.658	72.9	57.2	61.347	276.2	54.2	396.90	396.96905	402.9622	399.08

5 rows × 196 columns

B + TAX	B + ZN	CHAS + CRIM	...	DIS * RAD	DIS * RM	DIS * TAX	DIS * ZN	INDUS * LSTAT	INDUS * MEDV	INDUS * NOX	INDUS * PTRATIO	INDUS * RAD	INDUS * RM	INDUS * TAX	INDUS * ZN	LSTAT * MEDV	LSTAT * NOX	LSTAT * PTRATIO	LSTAT * RAD	LSTAT * RM	LSTAT * TAX	LSTAT * ZN	MEDV * NOX	MEDV * PTRATIO	MEDV * RAD	MEDV * RM	MEDV * TAX
592.90	414.90	0.00632	...	4.0900	26.891750	1210.6400	73.62	11.5038	55.440	1.24278	35.343	2.31	15.18825	683.76	41.58	119.520	2.67924	76.194	4.98	32.74350	1474.08	89.64	12.9120	367.20	24.0	157.8000	7104.0
538.90	396.90	0.02731	...	9.9342	31.893749	1202.0382	0.00	64.6198	152.712	3.31583	125.846	14.14	45.39647	1710.94	0.00	197.424	4.28666	162.692	18.28	58.68794	2211.88	0.00	10.1304	384.48	43.2	138.6936	5227.2
534.83	392.83	0.02729	...	9.9342	35.688614	1202.0382	0.00	28.4921	245.329	3.31583	125.846	14.14	50.79795	1710.94	0.00	139.841	1.89007	71.734	8.06	28.95555	975.26	0.00	16.2743	617.66	69.4	249.3195	8397.4
516.63	394.63	0.03237	...	18.1866	42.423276	1345.8084	0.00	6.4092	72.812	0.99844	40.766	6.54	15.25564	483.96	0.00	98.196	1.34652	54.978	8.82	20.57412	652.68	0.00	15.2972	624.58	100.2	233.7332	7414.8
518.90	396.90	0.06905	...	18.1866	43.326543	1345.8084	0.00	11.6194	78.916	0.99844	40.766	6.54	15.58046	483.96	0.00	192.946	2.44114	99.671	15.99	38.09351	1183.26	0.00	16.5796	676.94	108.6	258.7214	8036.4


Operations
AVG
MAX
MIN
STD
Order
OrderID
CustomerID
MIN (Product.Price)
Customer
CustomerID
Gender
Age
AVG(Orders.SUM(Product.Price))
1..*
1..*
Product
ProductID
Price

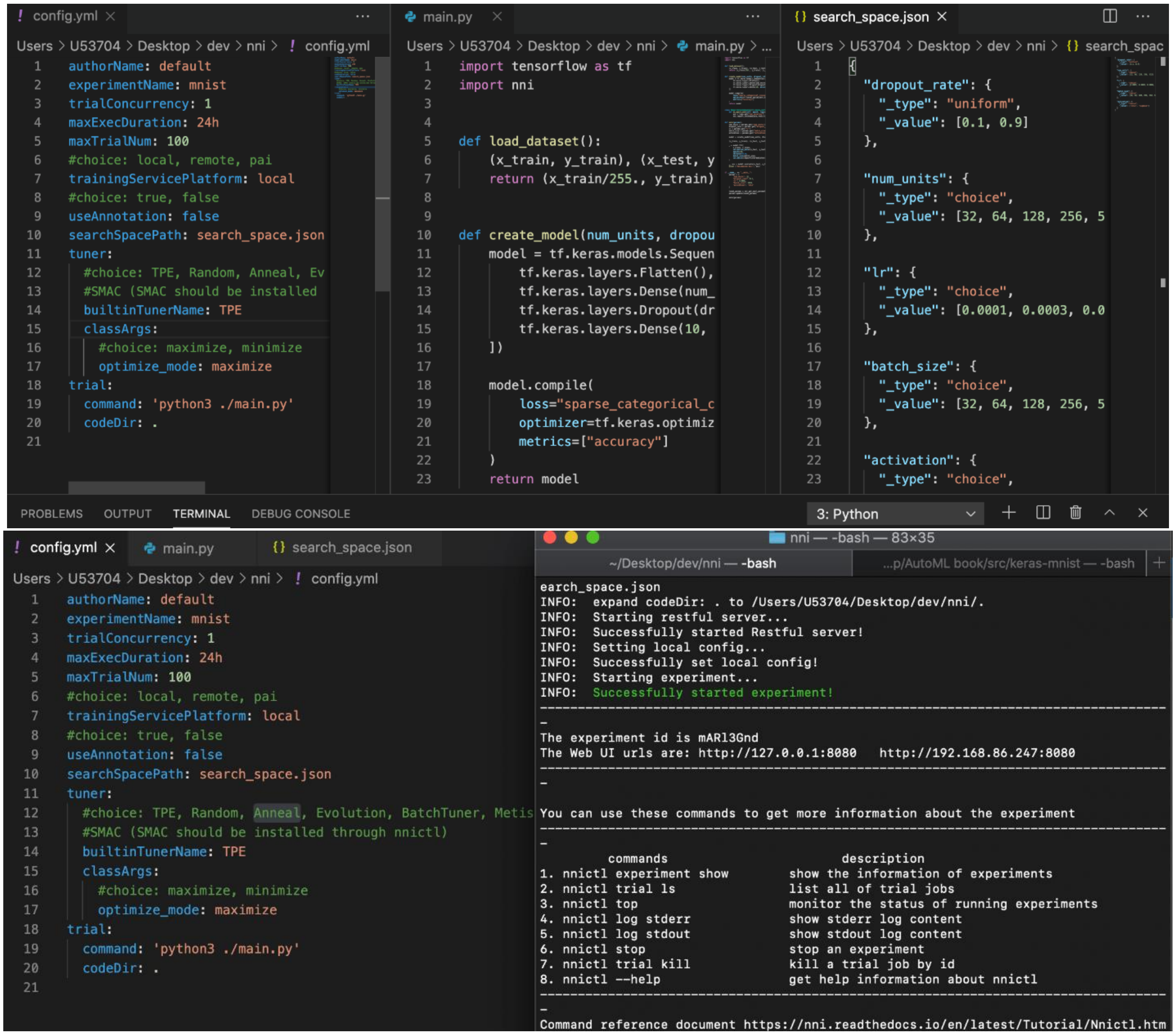

Tuner
Assessor
or
Advisor
(Python)
Python APIs
nnicli
nnictl
(Python)
Start
WebUI
Rest server
nnimanager
Training service
NNI Core (TypeScript)
Remote servers (remote training service mode)
Submit trials
trial
trial
trial
means a process
This part runs on your computer/laptop or any machine
This part runs on remote servers, or AML, or OpenPAI

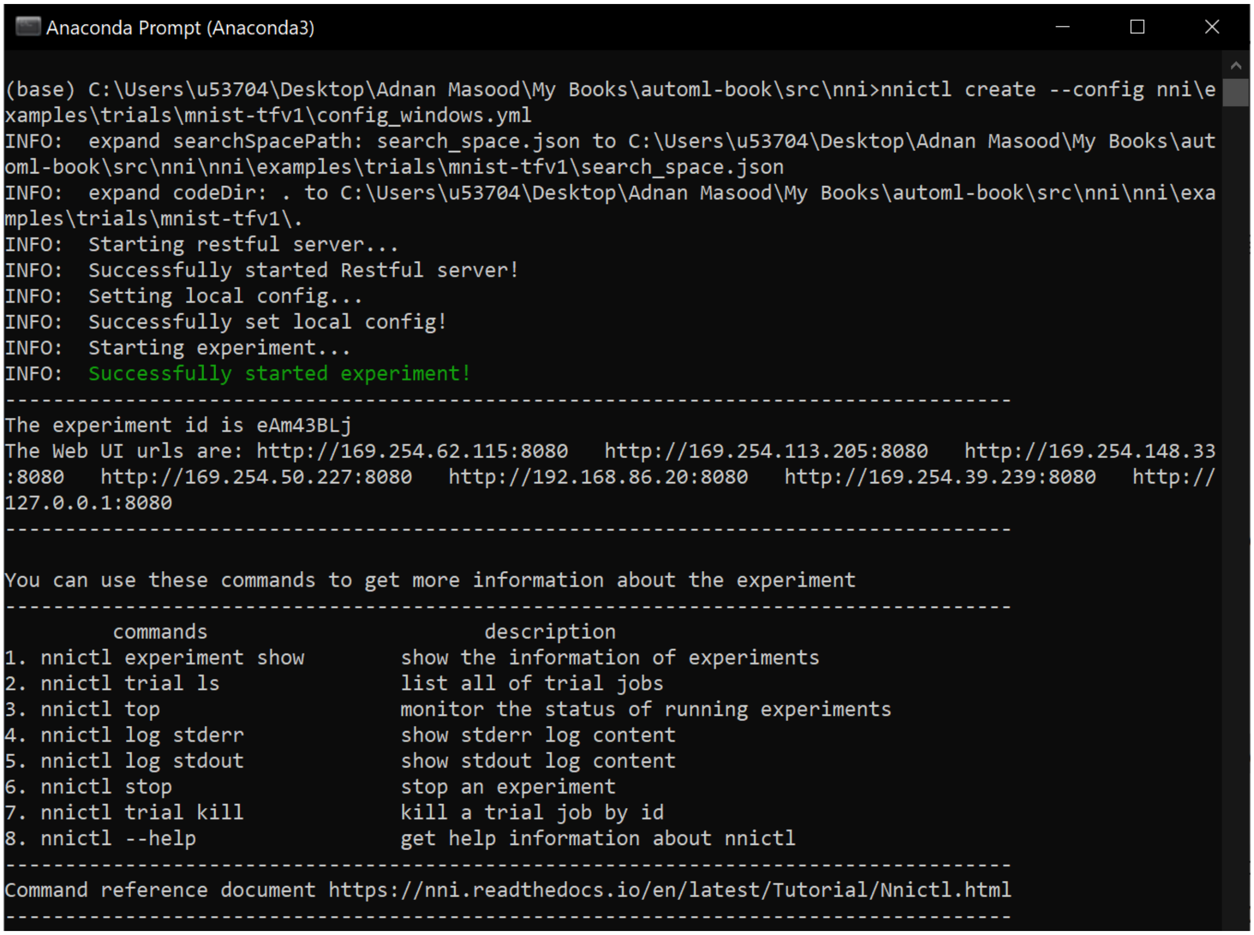
Anaconda Prompt (Anaconda3)

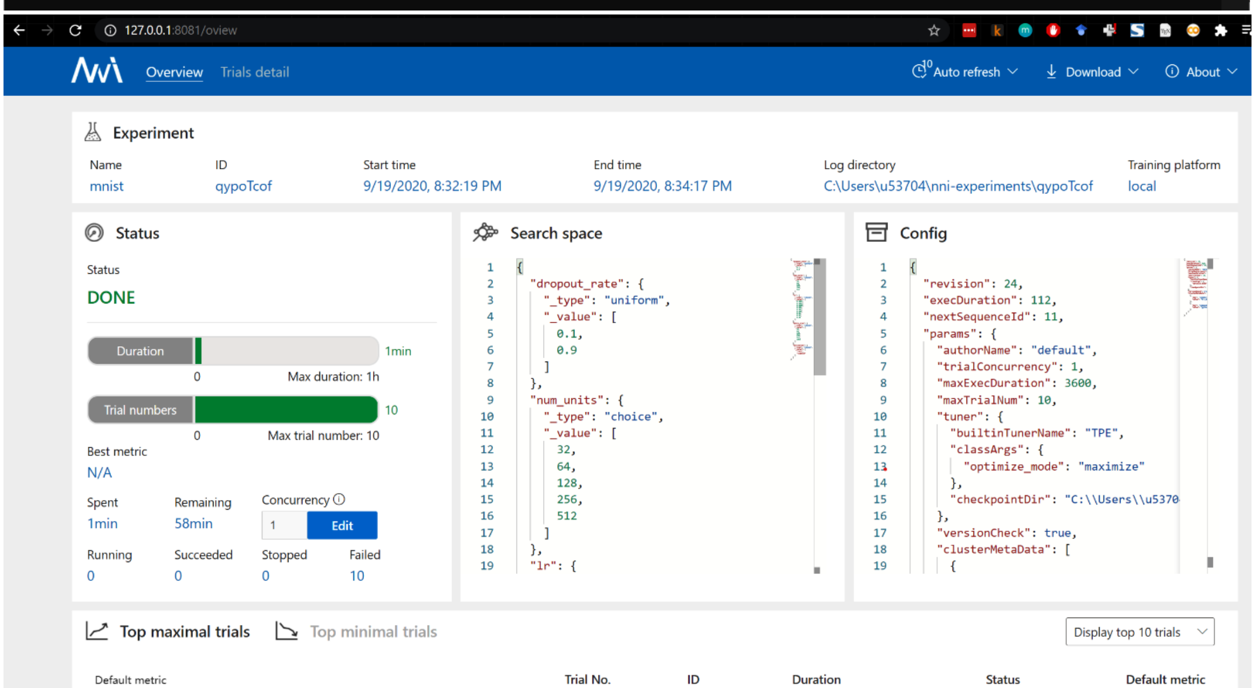
```
(base) C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni>python -m pip install --upgrade nni
Collecting nni
  Downloading nni-1.8-py3-none-win_amd64.whl (32.9 MB)
     |████████████████████████████████| 32.9 MB 3.2 MB/s
Requirement already satisfied, skipping upgrade: scipy in d:\anaconda3\lib\site-packages (from nni) (1.4.1)
Requirement already satisfied, skipping upgrade: numpy in d:\anaconda3\lib\site-packages (from nni) (1.18.1)
Collecting astor
  Downloading astor-0.8.1-py2.py3-none-any.whl (27 kB)
Collecting hyperopt==0.1.2
  Downloading hyperopt-0.1.2-py3-none-any.whl (115 kB)
     |████████████████████████████████| 115 kB 3.3 MB/s
Requirement already satisfied, skipping upgrade: requests in d:\anaconda3\lib\site-packages (from nni) (2.22.0)
Requirement already satisfied, skipping upgrade: psutil in d:\anaconda3\lib\site-packages (from nni) (5.6.7)
Collecting coverage
  Downloading coverage-5.3-cp37-cp37m-win_amd64.whl (208 kB)
     |████████████████████████████████| 208 kB 6.4 MB/s
Requirement already satisfied, skipping upgrade: pkginfo in d:\anaconda3\lib\site-packages (from nni) (1.5.0.1)
Collecting websockets
  Downloading websockets-8.1-cp37-cp37m-win_amd64.whl (66 kB)
     |████████████████████████████████| 66 kB 4.5 MB/s
Requirement already satisfied, skipping upgrade: colorama in d:\anaconda3\lib\site-packages (from nni) (0.4.3)
Collecting netifaces
  Downloading netifaces-0.10.9-cp37-cp37m-win_amd64.whl (16 kB)
Collecting schema
  Downloading schema-0.7.2-py2.py3-none-any.whl (16 kB)
Collecting PythonWebHDFS
  Downloading PythonWebHDFS-0.2.3-py3-none-any.whl (10 kB)
Collecting scikit-learn>=0.23.2
  Downloading scikit_learn-0.23.2-cp37-cp37m-win_amd64.whl (6.8 MB)
```

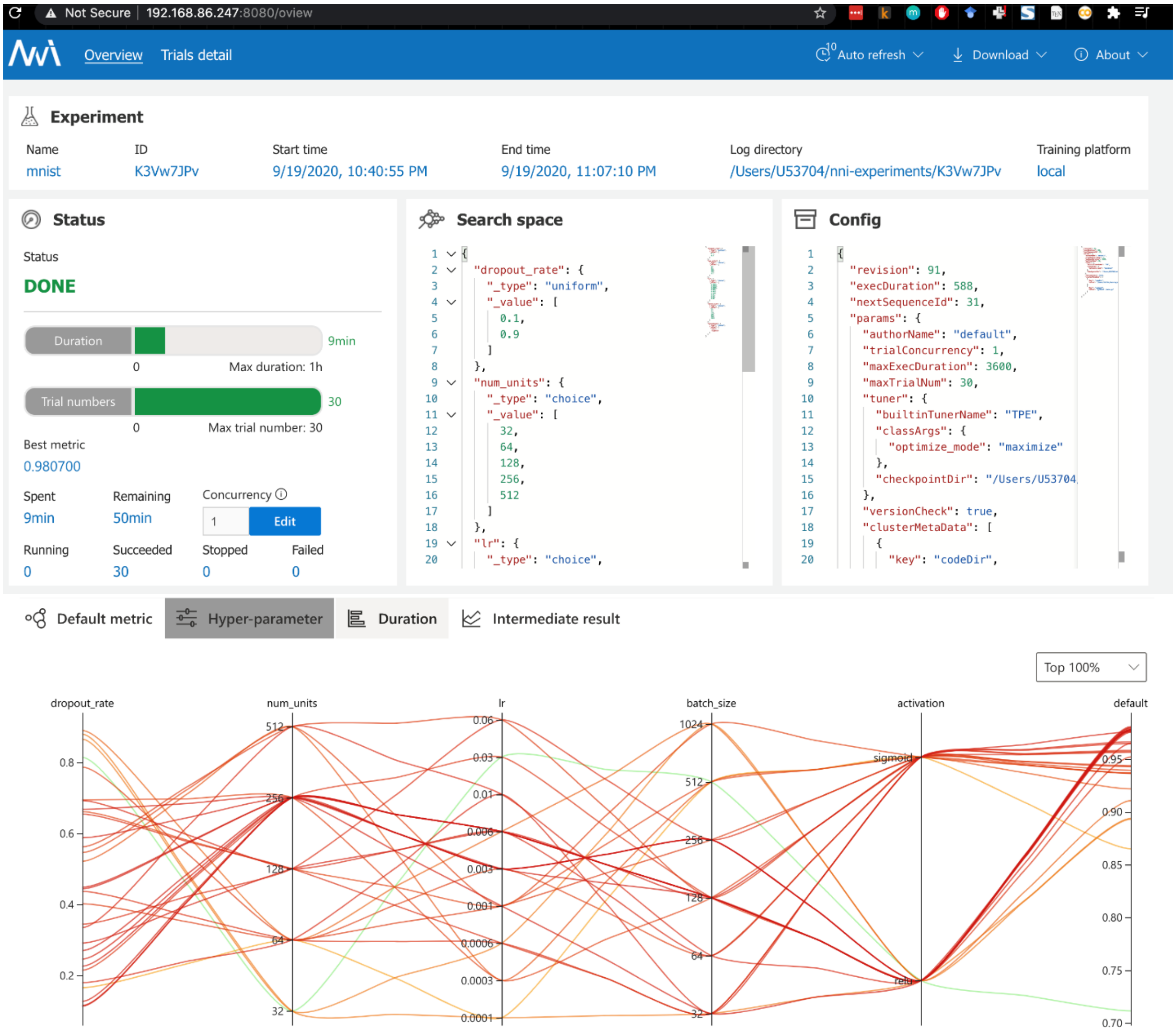
Anaconda Prompt (Anaconda3)

```
(base) C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni\keras-mnist>nnictl --help
usage: nnictl [-h] [--version]
              {ss_gen,create,resume,view,update,stop,trial,experiment,platform,webui,config,log,package,tensorboard
,top}
              ...

use nnictl command to control nni experiments

positional arguments:
  {ss_gen,create,resume,view,update,stop,trial,experiment,platform,webui,config,log,package,tensorboard,top}
    ss_gen              automatically generate search space file from trial
                        code
    create              create a new experiment
    resume              resume a new experiment
    view                view a stopped experiment
    update              update the experiment
    stop                stop the experiment
    trial               get trial information
    experiment          get experiment information
    platform            get platform information
    webui               get web ui information
    config              get config information
    log                 get log information
    package             control nni tuner and assessor packages
    tensorboard         manage tensorboard
    top                 monitor the experiment

optional arguments:
  -h, --help            show this help message and exit
  --version, -v

(base) C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni\keras-mnist>
```

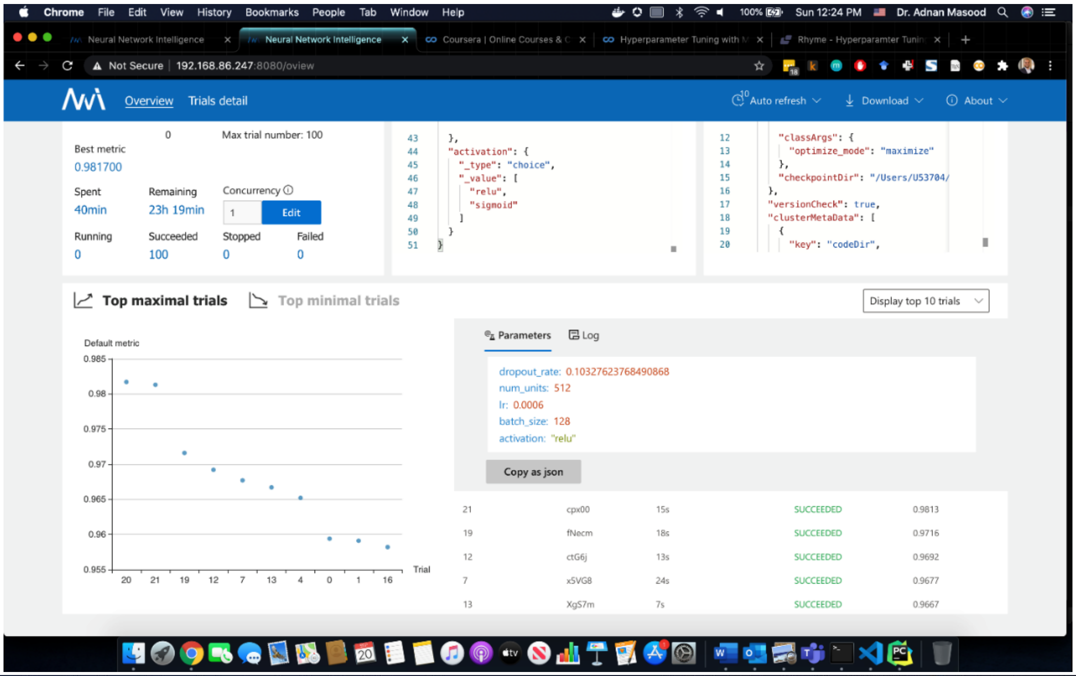

config.yml
main.py
search_space.json
Users > U53704 > Desktop > dev > nni > config.yml
authorName: default
experimentName: mnist
trialConcurrency: 1
maxExecDuration: 24h
maxTrialNum: 100
#choice: local, remote, pai
trainingServicePlatform: local
#choice: true, false
useAnnotation: false
searchSpacePath: search_space.json
tuner:
#choice: TPE, Random, Anneal, Evolution, BatchTuner, Metis
#SMAC (SMAC should be installed through nnictl)
builtinTunerName: TPE
classArgs:
#choice: maximize, minimize
optimize_mode: maximize
trial:
command: 'python3 ./main.py'
codeDir: .
Users > U53704 > Desktop > dev > nni > main.py > ...
import tensorflow as tf
import nni
def load_dataset():
(x_train, y_train), (x_test, y
return (x_train/255., y_train)
def create_model(num_units, dropou
model = tf.keras.models.Sequen
tf.keras.layers.Flatten(),
tf.keras.layers.Dense(num_
tf.keras.layers.Dropout(dr
tf.keras.layers.Dense(10,
])
model.compile(
loss="sparse_categorical_c
optimizer=tf.keras.optimiz
metrics=["accuracy"]
)
return model
Users > U53704 > Desktop > dev > nni > search_spac
{
"dropout_rate": {
"_type": "uniform",
"_value": [0.1, 0.9]
},
"num_units": {
"_type": "choice",
"_value": [32, 64, 128, 256, 5
},
"lr": {
"_type": "choice",
"_value": [0.0001, 0.0003, 0.0
},
"batch_size": {
"_type": "choice",
"_value": [32, 64, 128, 256, 5
},
"activation": {
"_type": "choice",
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
3: Python
nni — -bash — 83×35
~/Desktop/dev/nni — -bash
...p/AutoML book/src/keras-mnist — -bash
earch_space.json
INFO: expand codeDir: . to /Users/U53704/Desktop/dev/nni/.
INFO: Starting restful server...
INFO: Successfully started Restful server!
INFO: Setting local config...
INFO: Successfully set local config!
INFO: Starting experiment...
INFO: Successfully started experiment!
The experiment id is mAR13Gnd
The Web UI urls are: http://127.0.0.1:8080 http://192.168.86.247:8080
You can use these commands to get more information about the experiment
commands description
1. nnictl experiment show show the information of experiments
2. nnictl trial ls list all of trial jobs
3. nnictl top monitor the status of running experiments
4. nnictl log stderr show stderr log content
5. nnictl log stdout show stdout log content
6. nnictl stop stop an experiment
7. nnictl trial kill kill a trial job by id
8. nnictl --help get help information about nnictl
Command reference document https://nni.readthedocs.io/en/latest/Tutorial/Nnictl.htm

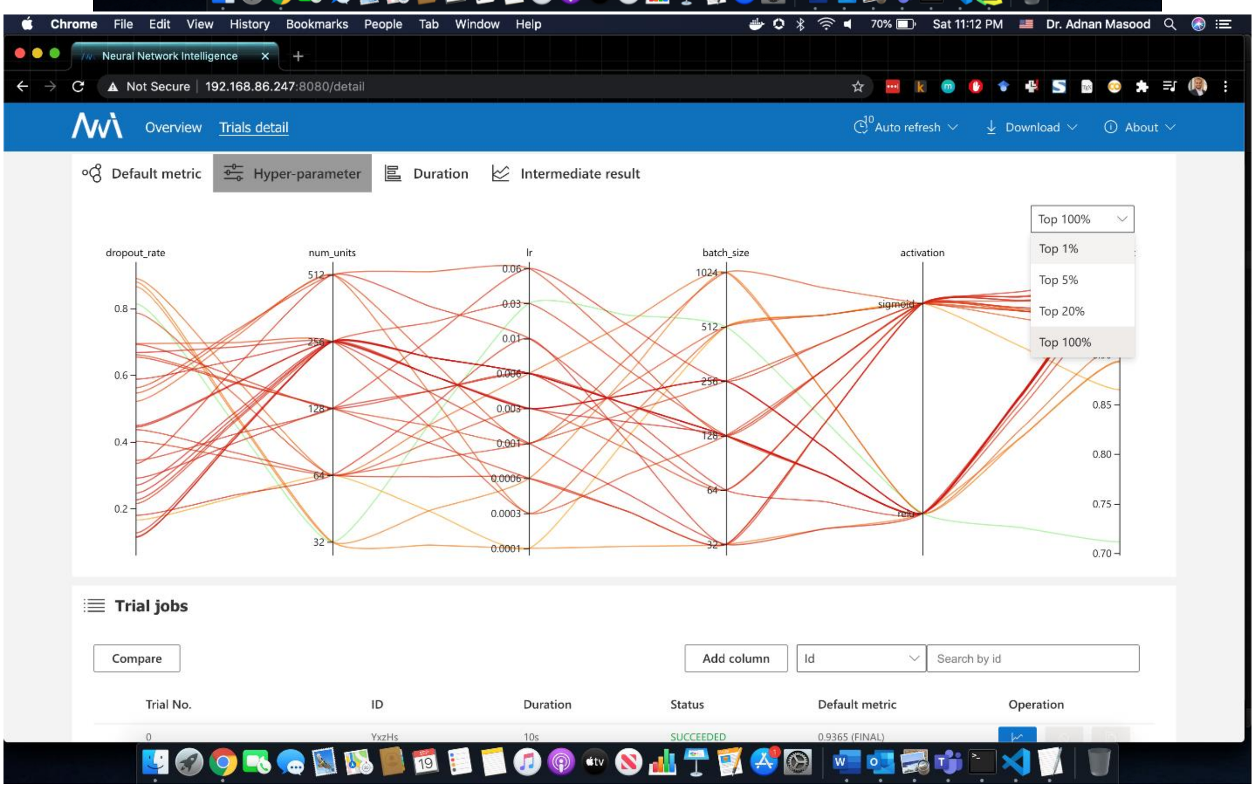

Anaconda Prompt (Anaconda3)
(base) C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni>nnictl create --config nni\examples\trials\mnist-tfv1\config_windows.yml
INFO: expand searchSpacePath: search_space.json to C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni\nni\examples\trials\mnist-tfv1\search_space.json
INFO: expand codeDir: . to C:\Users\u53704\Desktop\Adnan Masood\My Books\automl-book\src\nni\nni\examples\trials\mnist-tfv1\.
INFO: Starting restful server...
INFO: Successfully started Restful server!
INFO: Setting local config...
INFO: Successfully set local config!
INFO: Starting experiment...
INFO: Successfully started experiment!
The experiment id is eAm43BLj
The Web UI urls are: http://169.254.62.115:8080 http://169.254.113.205:8080 http://169.254.148.33:8080 http://169.254.50.227:8080 http://192.168.86.20:8080 http://169.254.39.239:8080 http://127.0.0.1:8080
You can use these commands to get more information about the experiment
commands description
1. nnictl experiment show show the information of experiments
2. nnictl trial ls list all of trial jobs
3. nnictl top monitor the status of running experiments
4. nnictl log stderr show stderr log content
5. nnictl log stdout show stdout log content
6. nnictl stop stop an experiment
7. nnictl trial kill kill a trial job by id
8. nnictl --help get help information about nnictl
Command reference document https://nni.readthedocs.io/en/latest/Tutorial/Nnictl.html


127.0.0.1:8081/oview
Overview
Trials detail
Auto refresh
Download
About
Experiment
Name mnist
ID qypoTcof
Start time 9/19/2020, 8:32:19 PM
End time 9/19/2020, 8:34:17 PM
Log directory C:\Users\u53704\nni-experiments\qypoTcof
Training platform local
Status
Status DONE
Duration 1min
0 Max duration: 1h
Trial numbers 10
0 Max trial number: 10
Best metric N/A
Spent 1min
Remaining 58min
Concurrency 1 Edit
Running 0
Succeeded 0
Stopped 0
Failed 10
Search space
{
"dropout_rate": {
"_type": "uniform",
"_value": [
0.1,
0.9
]
},
"num_units": {
"_type": "choice",
"_value": [
32,
64,
128,
256,
512
]
},
"lr": {
Config
{
"revision": 24,
"execDuration": 112,
"nextSequenceId": 11,
"params": {
"authorName": "default",
"trialConcurrency": 1,
"maxExecDuration": 3600,
"maxTrialNum": 10,
"tuner": {
"builtinTunerName": "TPE",
"classArgs": {
"optimize_mode": "maximize"
},
"checkpointDir": "C:\\Users\\u5370
},
"versionCheck": true,
"clusterMetaData": [
{
Top maximal trials
Top minimal trials
Display top 10 trials
Default metric
Trial No.
ID
Duration
Status
Default metric

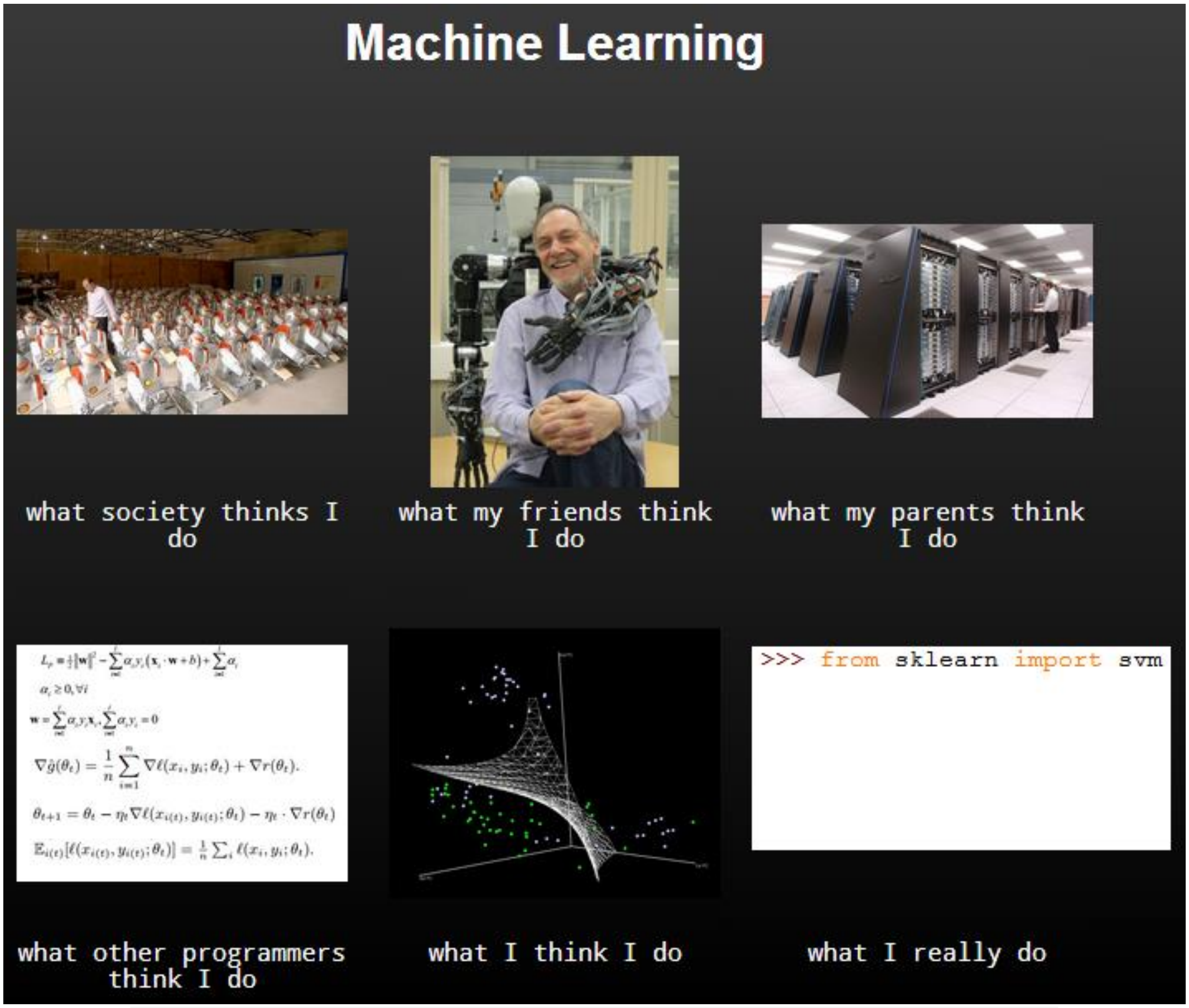


Not Secure | 192.168.86.247:8080/oview
Overview
Trials detail
Auto refresh
Download
About
Experiment
Name
mnist
ID
K3Vw7JPv
Start time
9/19/2020, 10:40:55 PM
End time
9/19/2020, 11:07:10 PM
Log directory
/Users/U53704/nni-experiments/K3Vw7JPv
Training platform
local
Status
Status
DONE
Duration
9min
Max duration: 1h
Trial numbers
30
Max trial number: 30
Best metric
0.980700
Spent
9min
Remaining
50min
Concurrency
Edit
Running
0
Succeeded
30
Stopped
0
Failed
0
Search space
{
"dropout_rate": {
"_type": "uniform",
"_value": [
0.1,
0.9
]
},
"num_units": {
"_type": "choice",
"_value": [
32,
64,
128,
256,
512
]
},
"lr": {
"_type": "choice",
Config
{
"revision": 91,
"execDuration": 588,
"nextSequenceId": 31,
"params": {
"authorName": "default",
"trialConcurrency": 1,
"maxExecDuration": 3600,
"maxTrialNum": 30,
"tuner": {
"builtinTunerName": "TPE",
"classArgs": {
"optimize_mode": "maximize"
},
"checkpointDir": "/Users/U53704
},
"versionCheck": true,
"clusterMetaData": [
{
"key": "codeDir",
Default metric
Hyper-parameter
Duration
Intermediate result
Top 100%
dropout_rate
num_units
lr
batch_size
activation
default

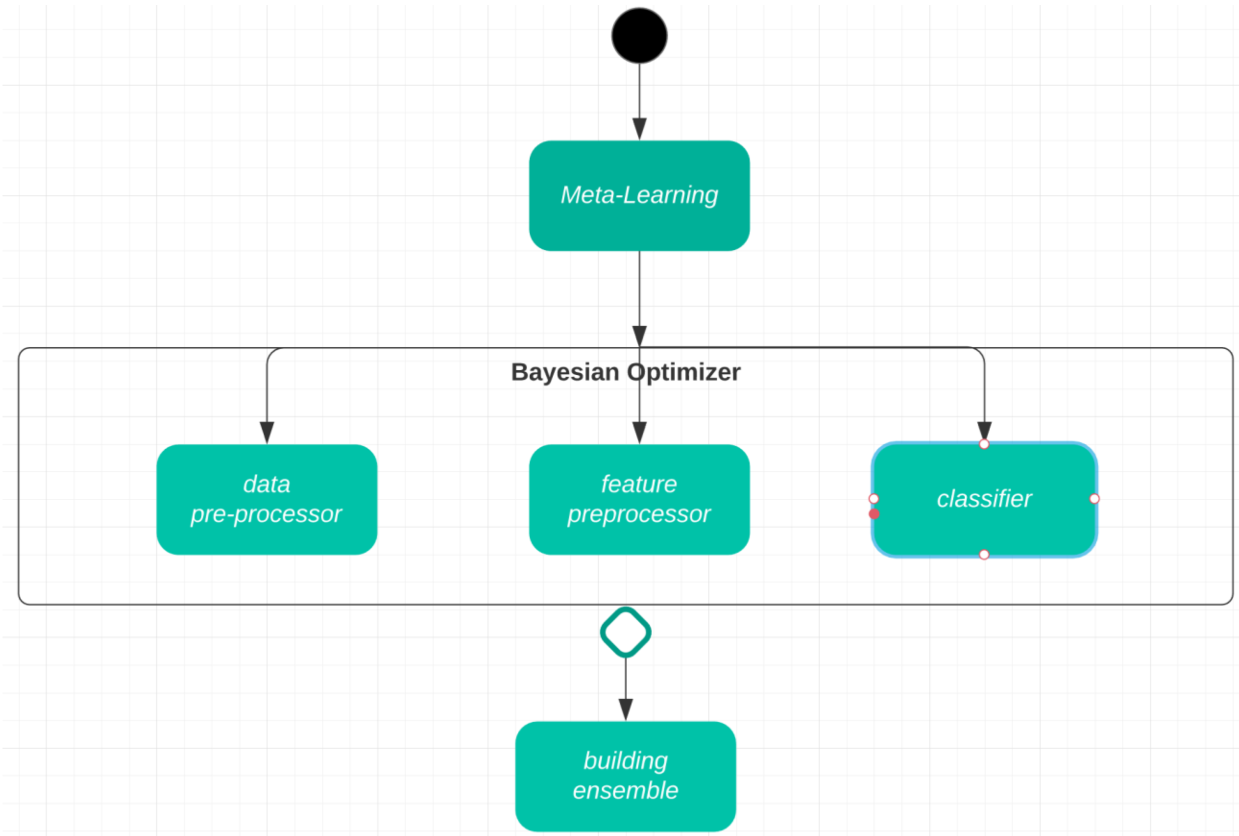

Chrome File Edit View History Bookmarks People Tab Window Help
Sun 12:24 PM
Dr. Adnan Masood
Neural Network Intelligence
Not Secure 192.168.86.247:8080/oview
Overview
Trials detail
Auto refresh
Download
About
Max trial number: 100
Best metric
0.981700
Spent
40min
Remaining
23h 19min
Concurrency
Edit
Running
0
Succeeded
100
Stopped
0
Failed
0
Top maximal trials
Top minimal trials
Display top 10 trials
Default metric
Trial
Parameters
Log
dropout_rate: 0.10327623768490868
num_units: 512
lr: 0.0006
batch_size: 128
activation: "relu"
Copy as json
21 cpx00 15s SUCCEEDED 0.9813
19 fNecm 18s SUCCEEDED 0.9716
12 ctG6j 13s SUCCEEDED 0.9692
7 x5VG8 24s SUCCEEDED 0.9677
13 XgS7m 7s SUCCEEDED 0.9667

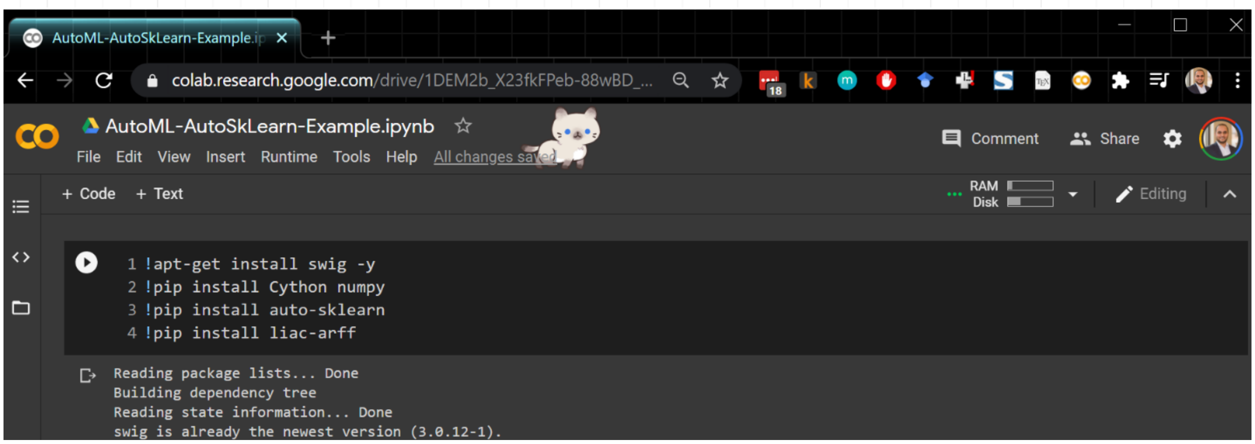

Chrome File Edit View History Bookmarks People Tab Window Help
Sat 11:12 PM
Dr. Adnan Masood
Neural Network Intelligence
Not Secure 192.168.86.247:8080/detail
Overview
Trials detail
Auto refresh
Download
About
Default metric
Hyper-parameter
Duration
Intermediate result
Top 100%
Top 1%
Top 5%
Top 20%
Top 100%
dropout_rate
num_units
lr
batch_size
activation
sigmoid
relu
Trial jobs
Compare
Add column
Id
Search by id
Trial No.
ID
Duration
Status
Default metric
Operation
0 YxzHs 10s SUCCEEDED 0.9365 (FINAL)

Trial No.	ID	Duration	Status	Default metric ↓	Operation
28	IS2fj	11s	SUCCEEDED	0.9807 (FINAL)	
27	mxn9S	8s	SUCCEEDED	0.9797 (FINAL)	
26	tMNvv	9s	SUCCEEDED	0.9794 (FINAL)	
21	IJJC1	8s	SUCCEEDED	0.9781 (FINAL)	
20	Rdm9Z	9s	SUCCEEDED	0.978 (FINAL)	
29	Ztsfn	9s	SUCCEEDED	0.9778 (FINAL)	
24	GVCUo	9s	SUCCEEDED	0.9777 (FINAL)	
14	z5bYY	16s	SUCCEEDED	0.9769 (FINAL)	
22	BHaar	8s	SUCCEEDED	0.9766 (FINAL)	
17	hHsqt	8s	SUCCEEDED	0.9757 (FINAL)	
23	pRRcL	8s	SUCCEEDED	0.9754 (FINAL)	
3	bbNIr	11s	SUCCEEDED	0.9663 (FINAL)	
18	sNSbm	14s	SUCCEEDED	0.9648 (FINAL)	
8	pof1Y	16s	SUCCEEDED	0.9648 (FINAL)	




```
>>> import autosklearn.classification
>>> cls = autosklearn.classification.AutoSklearnClassifier()
>>> cls.fit(X_train, y_train)
>>> predictions = cls.predict(X_test)
```


Meta-Learning
Bayesian Optimizer
data
pre-processor
feature
preprocessor
classifier
building
ensemble


AutoML-AutoSkLearn-Example.ipynb
colab.research.google.com/drive/1DEM2b_X23fkFPeb-88wBD_...
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
Editing
!apt-get install swig -y
!pip install Cython numpy
!pip install auto-sklearn
!pip install liac-arff
Reading package lists... Done
Building dependency tree
Reading state information... Done
swig is already the newest version (3.0.12-1).

```
import autosklearn.classification
import sklearn.model_selection as cv
import sklearn.datasets
import sklearn.metrics
#from autosklearn.experimental.askl2 import AutoSklearn2Classifier


X, y = sklearn.datasets.load_digits(return_X_y=True)
X_train, X_test, y_train, y_test = \
        sklearn.model_selection.train_test_split(X, y, random_state=1)
automl = autosklearn.classification.AutoSklearnClassifier()
automl.fit(X_train, y_train)
y_hat = automl.predict(X_test)
print("Accuracy score", sklearn.metrics.accuracy_score(y_test, y_hat))
```

```
Accuracy score 0.9888888888888889
```

```
!pip install autokeras
!pip install git+https://github.com/keras-team/keras-tuner.git@1.0.2rc1
!pip install tensorflow
```

```
import tensorflow as tf
from tensorflow.keras.datasets import mnist

(x_train, y_train), (x_test, y_test) = mnist.load_data()
print(x_train.shape)
print(y_train.shape)
print(y_train[:3])
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11493376/11490434 [==============================] - 0s 0us/step
(60000, 28, 28)
(60000,)
[5 0 4]
```

```python
import autokeras as ak

# Initialize the image classifier.
clf = ak.ImageClassifier(
    overwrite=True,
    max_trials=1)
# Feed the image classifier with training data.
clf.fit(x_train, y_train, epochs=10)
```

```
...
Search: Running Trial #1

Hyperparameter     |Value     |Best Value So Far
image_block_1/block_type|vanilla   |?
image_block_1/normalize|True      |?
image_block_1/augment|False     |?
image_block_1/conv_block_1/kernel_size|3         |?
image_block_1/conv_block_1/num_blocks|1         |?
image_block_1/conv_block_1/num_layers|2         |?
image_block_1/conv_block_1/max_pooling|True      |?
image_block_1/conv_block_1/separable|False     |?
image_block_1/conv_block_1/dropout|0.25      |?
image_block_1/conv_block_1/filters_0_0|32        |?
image_block_1/conv_block_1/filters_0_1|64        |?
classification_head_1/spatial_reduction_1/reduction_type|flatten   |?
classification_head_1/dropout|0.5       |?
optimizer          |adam      |?
learning_rate      |0.001     |?

Epoch 1/10
  90/1500 [>.............................] - ETA: 1:42 - loss: 0.7316 - accuracy: 0.7750
```

```python
# Predict with the best model.
print (x_test)
predicted_y = clf.predict(x_test)
print(predicted_y)
```

```
[[[0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]
  ...
  [0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]]

 [[0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]
  [0 0 0 ... 0 0 0]
```

```
[[7]
 [2]
 [1]
 ...
 [4]
 [5]
 [6]]
```

```python
# Evaluate the best model with testing data.
print(clf.evaluate(x_test, y_test))
```

```
WARNING:tensorflow:Unresolved object in checkpoint: (root).optimizer.iter
WARNING:tensorflow:Unresolved object in checkpoint: (root).optimizer.beta_1
WARNING:tensorflow:Unresolved object in checkpoint: (root).optimizer.beta_2
WARNING:tensorflow:Unresolved object in checkpoint: (root).optimizer.decay
WARNING:tensorflow:Unresolved object in checkpoint: (root).optimizer.learning_rate
WARNING:tensorflow:A checkpoint was restored (e.g. tf.train.Checkpoint.restore or tf.keras.Mo
313/313 [==============================] - 5s 16ms/step - loss: 0.0332 - accuracy: 0.9893
[0.03324095159769058, 0.989300012588501]
```

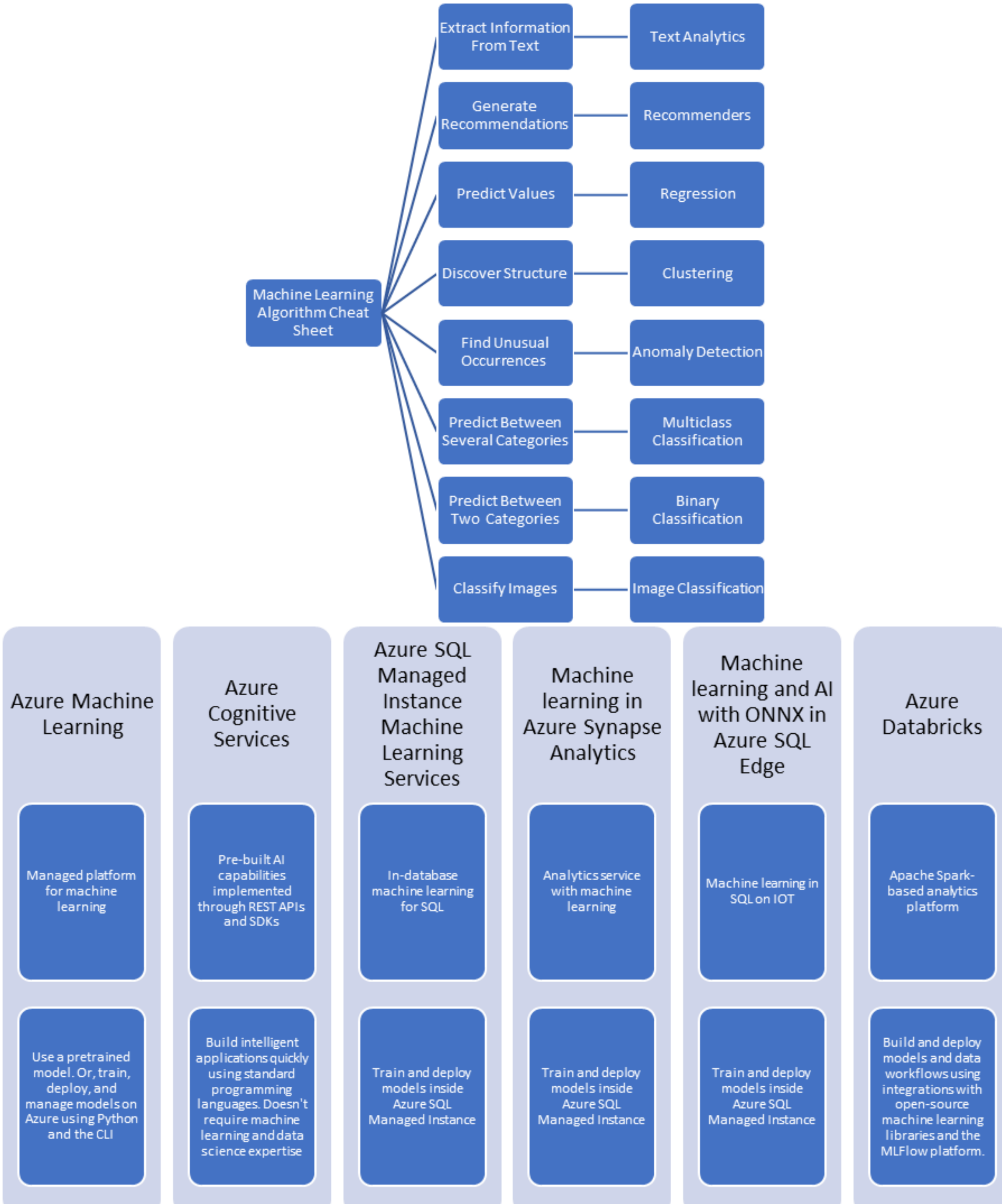
- image_classifier
- model_autokeras
  - assets
  - variables
  - saved_model.pb
- sample_data

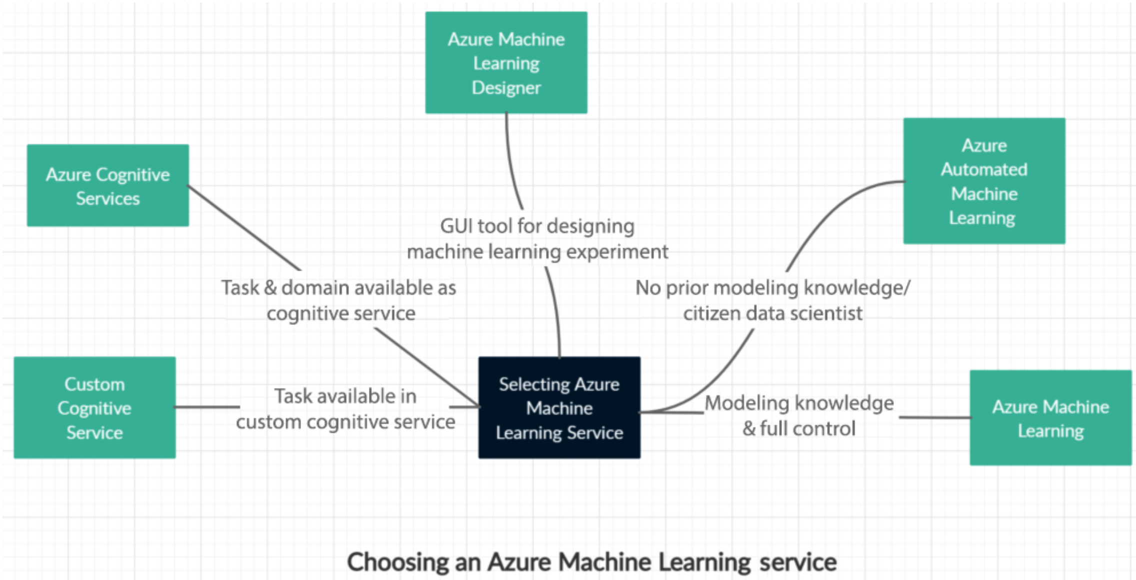
```python
# Export as a Keras Model.
model = clf.export_model()
print(type(model))  # <class 'tensorflow.python.keras.engine.training.Model'>

try:
    model.save("model_autokeras", save_format="tf")
except:
    model.save("model_autokeras.h5")
```

```python
from tensorflow.keras.models import load_model

loaded_model = load_model("model_autokeras", custom_objects=ak.CUSTOM_OBJECTS)

predicted_y = loaded_model.predict(x_test)
print(predicted_y)
```

```
[[1.04031775e-11 9.33228213e-13 3.23597726e-09 ... 1.00000000e+00
  1.17060192e-12 1.85679241e-08]
 [4.87752505e-10 1.90604410e-07 9.99998689e-01 ... 7.58147953e-14
  1.23664350e-08 2.22702485e-13]
 [6.41350306e-10 9.99989390e-01 3.39003947e-07 ... 2.34114125e-07
  9.20917387e-08 2.21865425e-11]
 ...
 [4.93693844e-13 1.76908531e-12 3.52099534e-14 ... 6.91528346e-09
  1.48345379e-07 4.70826649e-08]
 [1.18143204e-10 1.15603967e-15 7.66210359e-12 ... 1.39525295e-12
  4.68984922e-07 8.58040028e-11]
 [1.64026692e-09 3.16855014e-16 2.26211161e-09 ... 1.00495569e-16
  8.35135960e-09 3.97002526e-12]]
```

Chapter 4: Getting Started with Azure Machine Learning




Azure Machine Learning Designer
Azure Cognitive Services
Azure Automated Machine Learning
GUI tool for designing machine learning experiment
Task & domain available as cognitive service
No prior modeling knowledge/ citizen data scientist
Custom Cognitive Service
Task available in custom cognitive service
Selecting Azure Machine Learning Service
Modeling knowledge & full control
Azure Machine Learning
Choosing an Azure Machine Learning service

Collaborative Notebooks

- Maximize productivity with intellisense, easy compute and kernel switching and offline notebook editing.

Drag and Drop ML

- Use designer with modules for data transformation, model training and evaluation, or to create and publish ML pipelines with a few clicks.

MLOPS

- Use the central registry to store and track data. models, and metadata.
- Automatically capture lineage and governance data. Use Git to track work and GitHub Actions to implement workflows. Manage and monitor runs or compare multiple runs for training and experimentation.

RStudio Integration

- Built in R support and RStudio Server (Open Source edition) integration to build and deploy models and monitor runs.

Reinforcement learning

- Scale reinforcement learning to powerful compute clusters. support multi-agent scenarios, access open source RL algorithms, frameworks and environments.

Enterprise Grade Security

- Build and deploy models securely with capabilities like network isolation and Private Link. role-based access control for resources and actions, custom roles, and managed identity for compute resources.

Automated ML

- Rapidly create accurate models for classification, regression and time series forecasting, use model interpretability to understand how the model was built.

Data Labeling

- Prepare data quickly, manage and monitor labeling projects and automate iterative tasks with machine learning assisted labeling.

Autoscaling Compute

- Use managed compute to distribute training and rapidly test, validate and deploy models. CPU and GPU clusters can be shared across a workspace and automatically scale to meet your ML needs.

integration with other Azure services

- Accelerate productivity with built-in integration with Azure services such as Azure Synapse Analytics, Cognitive Search, Power BI, Azure Data Factory, Azure Data Lake, and Azure Databricks.

Responsible ML

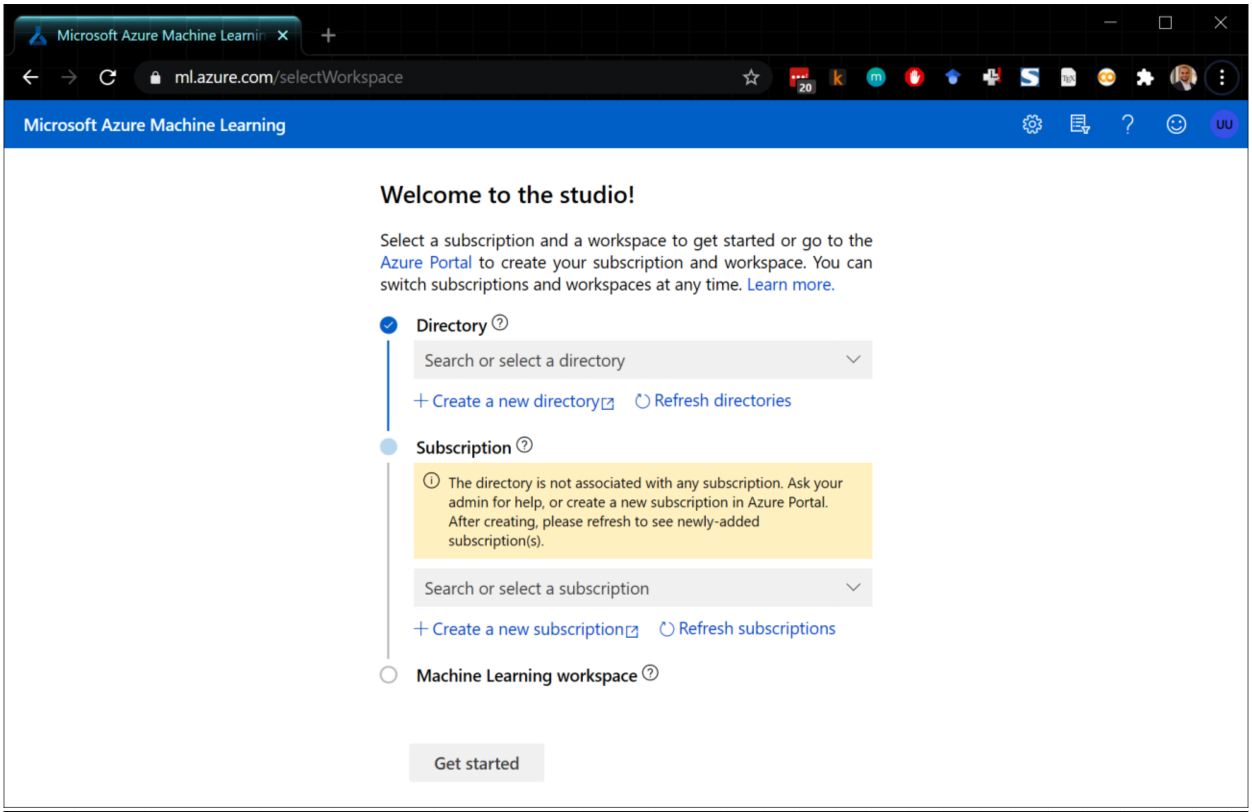
- Get model transparency at training and inferencjng with interpretability capabilities. Assess model fairness through disparity metrics and mitigate unfairness. Protect data with differential privacy.

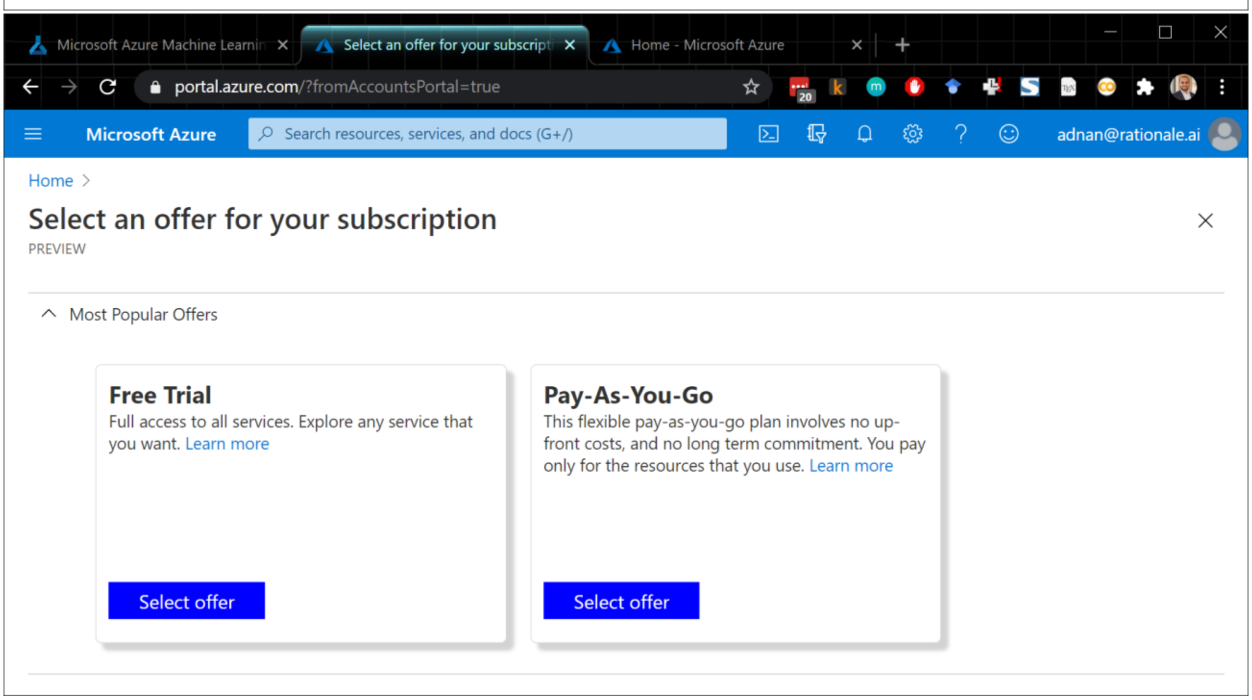
Cost management

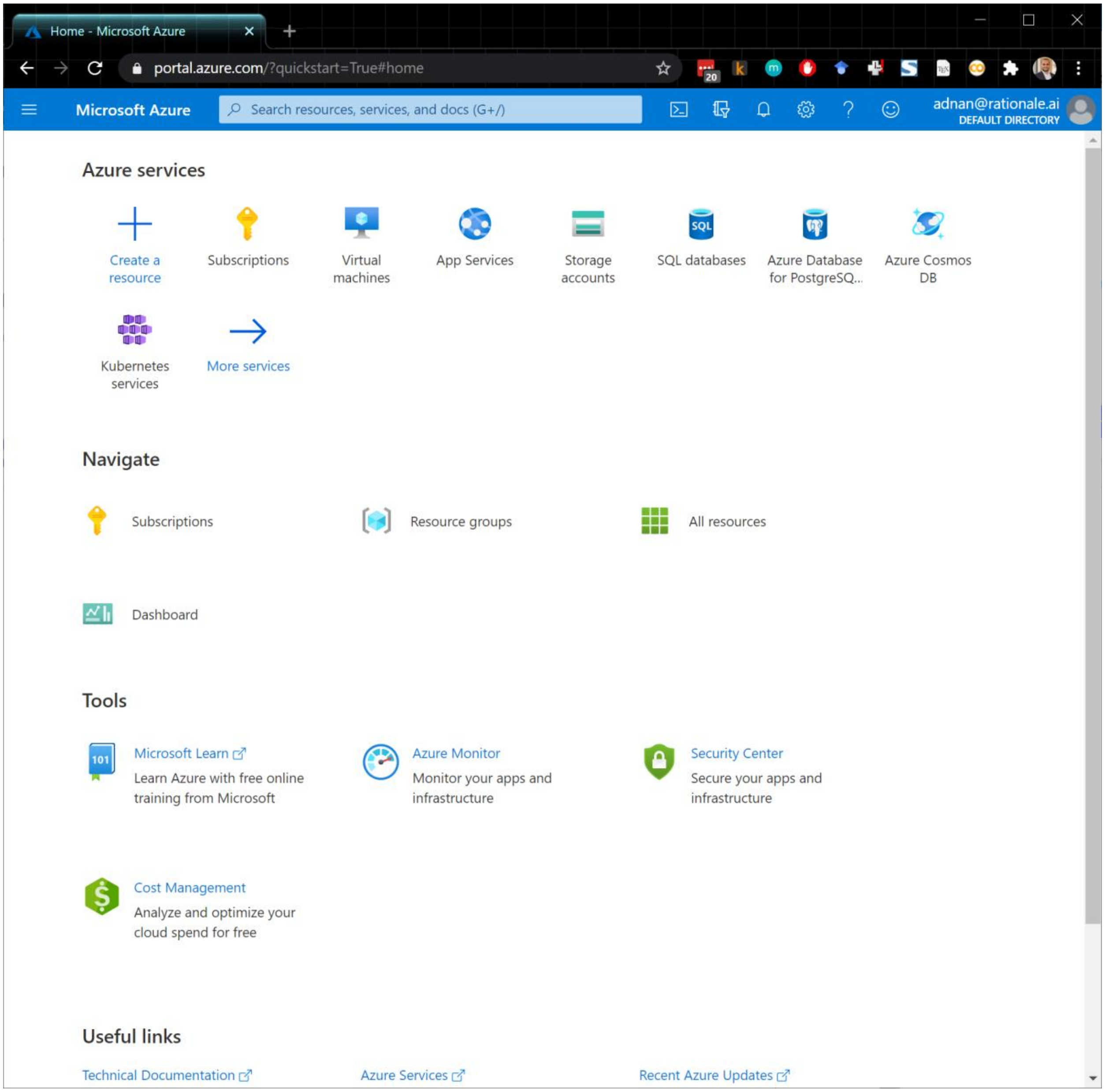
- Setter manage resource allocations for Azure Machine Learning
- Compute with workspace and resource level quota limits.

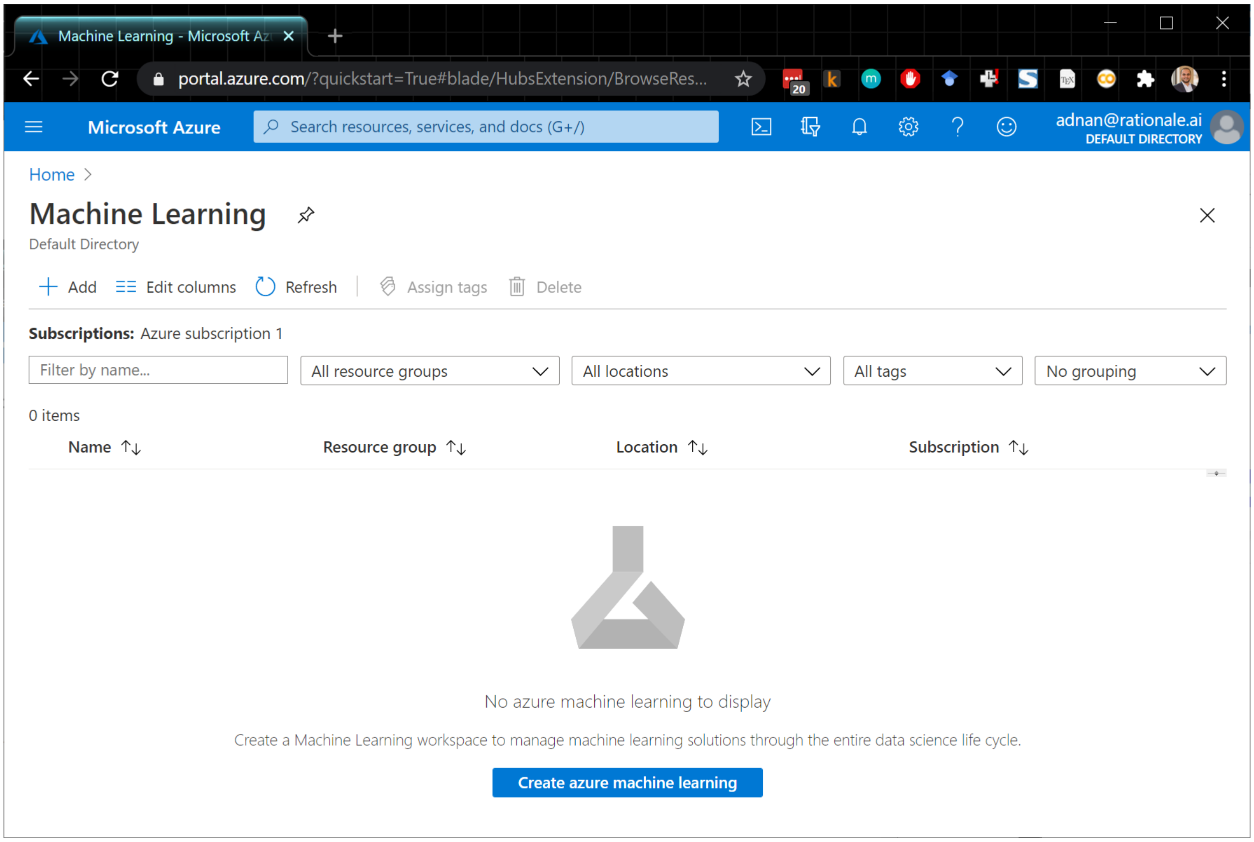
Training Targets	Automated Machine Learning	Machine Learning Pipelines
Local Computer	Supported	
Azure Machine Learning Compute Cluster	Supported with Hyperparameter Tuning	Supported
Azure Machine Learning Compute Instance	Supported with Hyperparameter Tuning	Supported
Remote VM	Supported with Hyperparameter Tuning	Supported
Azure Databricks	Supported (SDK Local Mode Only)	Supported
Azure Data Lake Analytics		Supported
Azure HDInsight		Supported
Azure Batch		Supported

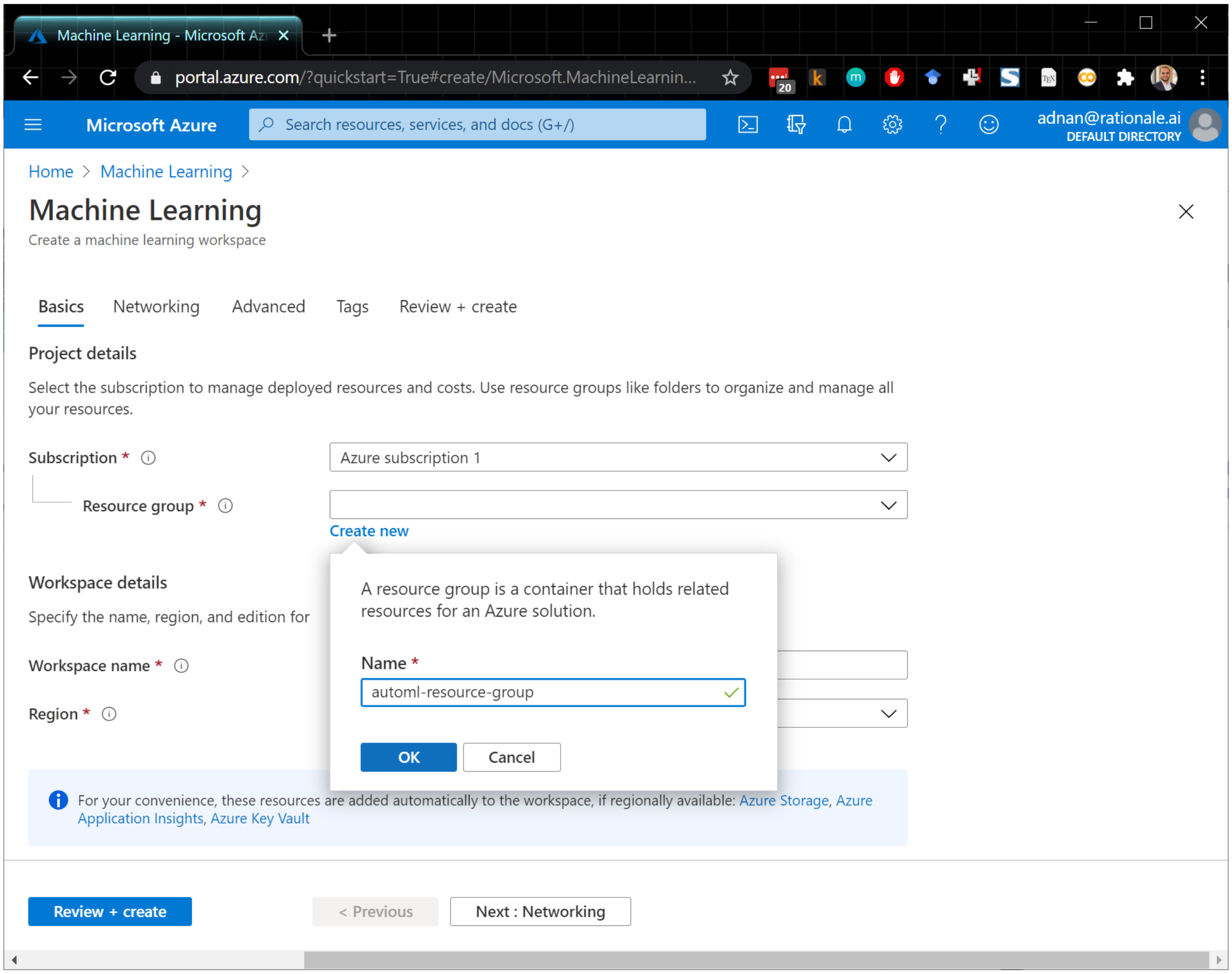
Compute Target	Usage	GPU / FPGA Support	Description
Local web service	Testing/debugging		Use for limited testing and troubleshooting. Hardware acceleration depends on use of libraries in the local system.
Azure Machine Learning compute instance web service	Testing/debugging		Use for limited testing and troubleshooting.
Azure Kubernetes Service (AKS)	Real-time inference	GPU supported with web service deployment. FPGA supported.	Use for high-scale production deployments. Provides fast response time and autoscaling of the deployed service. Cluster autoscaling isn't supported through the Azure Machine Learning SDK. To change the nodes in the AKS cluster, use the Ul for your AKS cluster in the Azure portal. AKS is the only option available for the designer.
Azure Container Instances	Testing or development		Use for low-scale CPU-based workloads that require less than 48 GB of RAM.
Azure Machine Learning compute clusters	Batch inference	GPU supported via machine learning pipeline.	Run batch scoring on serverless compute. Supports normal and low-priority VMs.
Azure Functions	(Preview) Real-time inference		
Azure IOT Edge	(Preview) IOT module		Deploy and serve ML models on IOT devices.
Azure Data Box Edge	Via IOT Edge	FPGA support	Deploy and serve ML models on IOT devices.

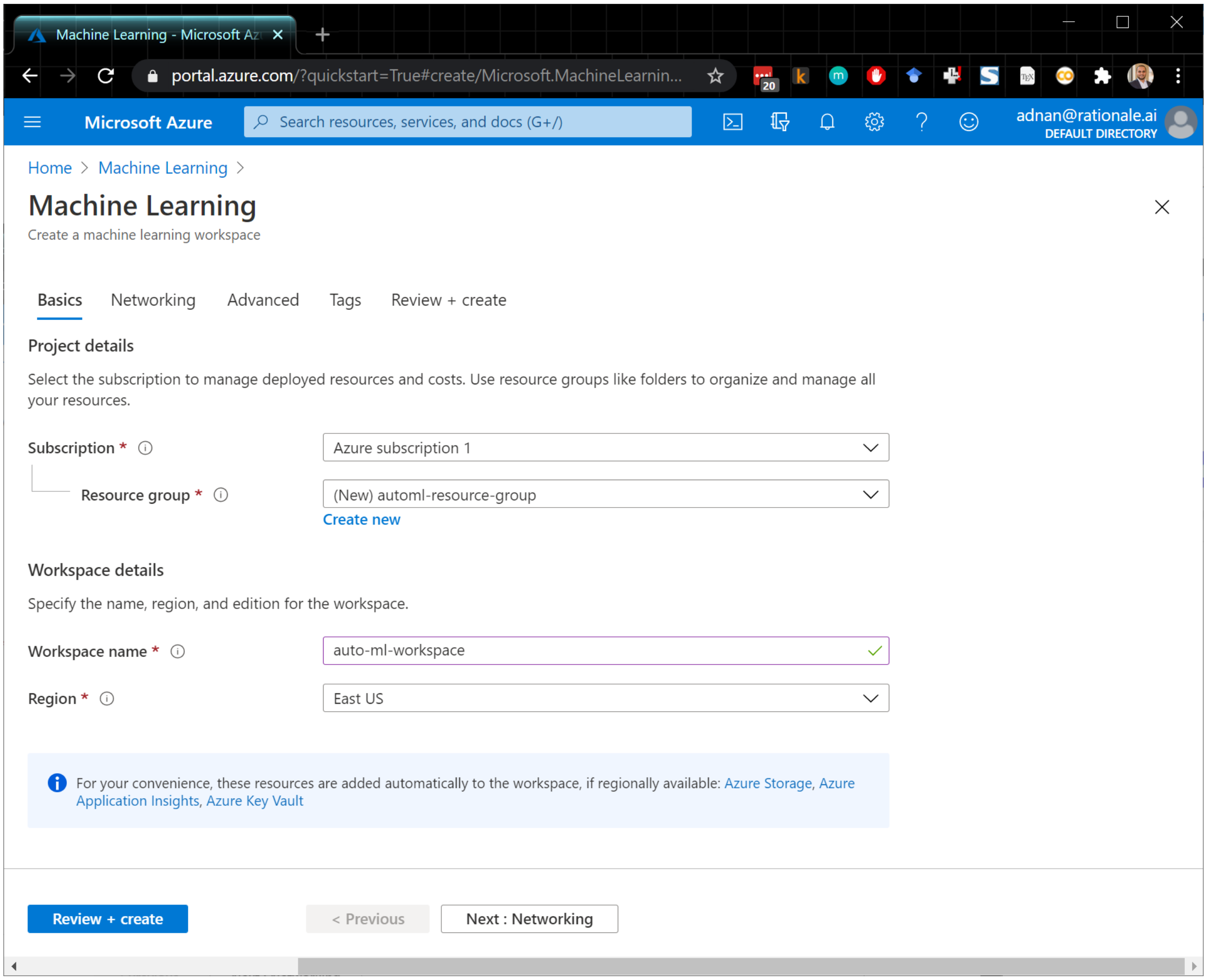

Microsoft Azure Machine Learnin
ml.azure.com/selectWorkspace
Microsoft Azure Machine Learning
UU
Welcome to the studio!
Select a subscription and a workspace to get started or go to the Azure Portal to create your subscription and workspace. You can switch subscriptions and workspaces at any time. Learn more.
Directory
Search or select a directory
+ Create a new directory
Refresh directories
Subscription
The directory is not associated with any subscription. Ask your admin for help, or create a new subscription in Azure Portal. After creating, please refresh to see newly-added subscription(s).
Search or select a subscription
+ Create a new subscription
Refresh subscriptions
Machine Learning workspace
Get started

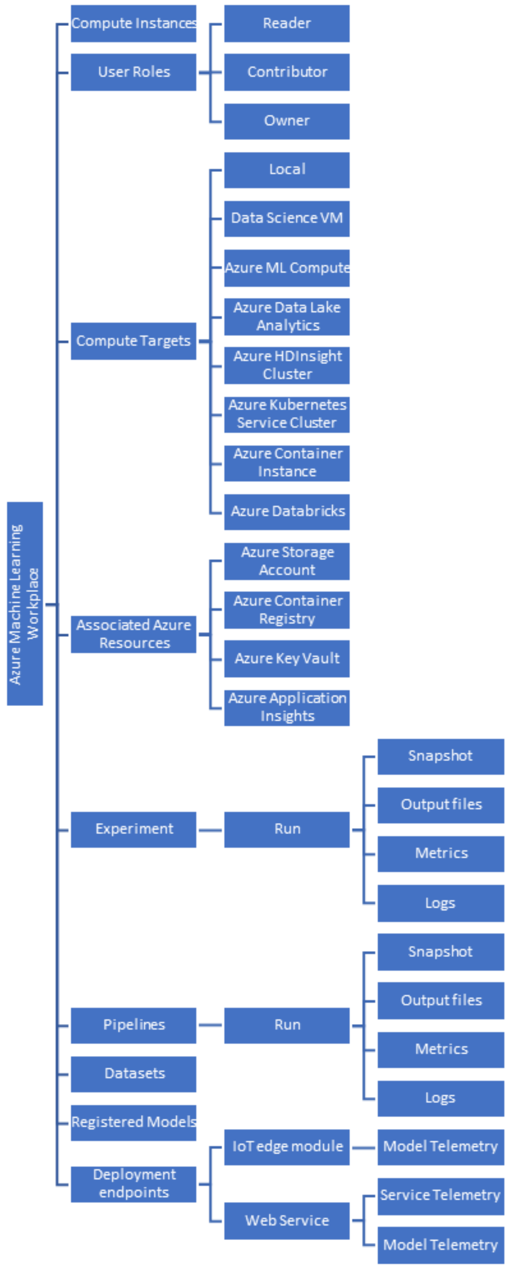

Microsoft Azure Machine Learnin
Select an offer for your subscript
Home - Microsoft Azure
portal.azure.com/?fromAccountsPortal=true
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
Home
Select an offer for your subscription
PREVIEW
Most Popular Offers
Free Trial
Full access to all services. Explore any service that you want. Learn more
Select offer
Pay-As-You-Go
This flexible pay-as-you-go plan involves no up-front costs, and no long term commitment. You pay only for the resources that you use. Learn more
Select offer

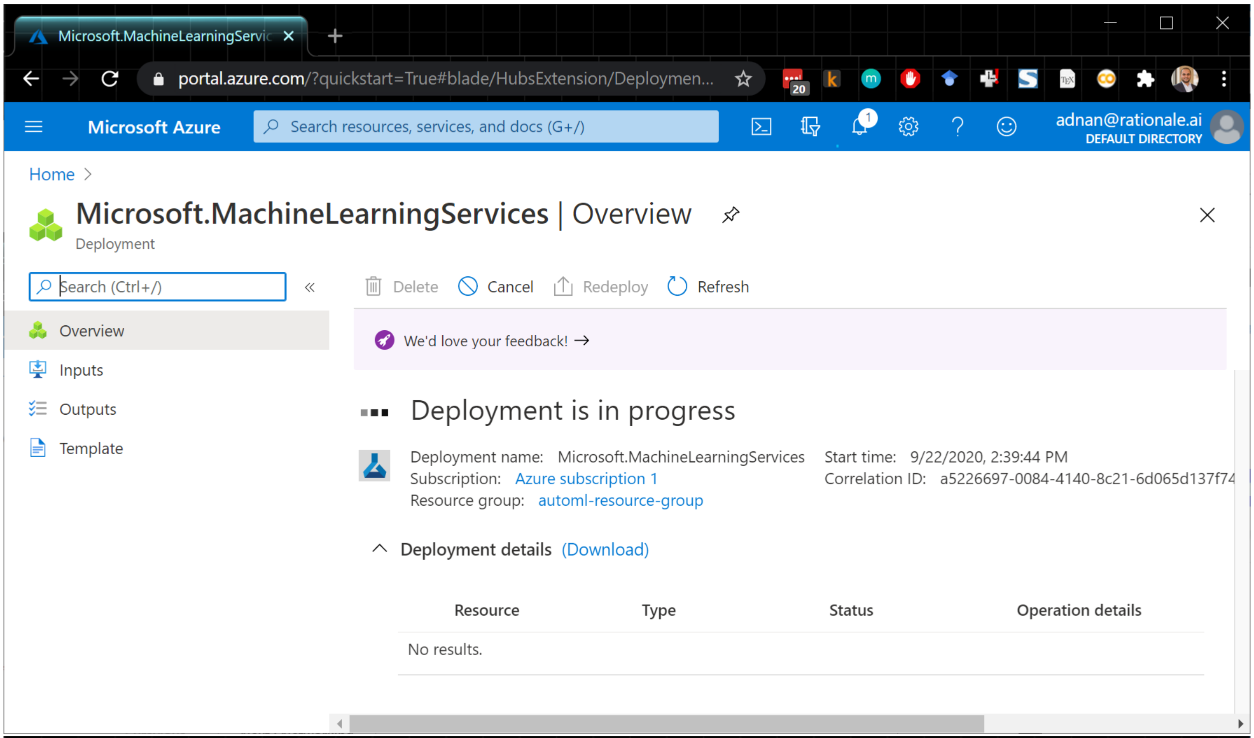
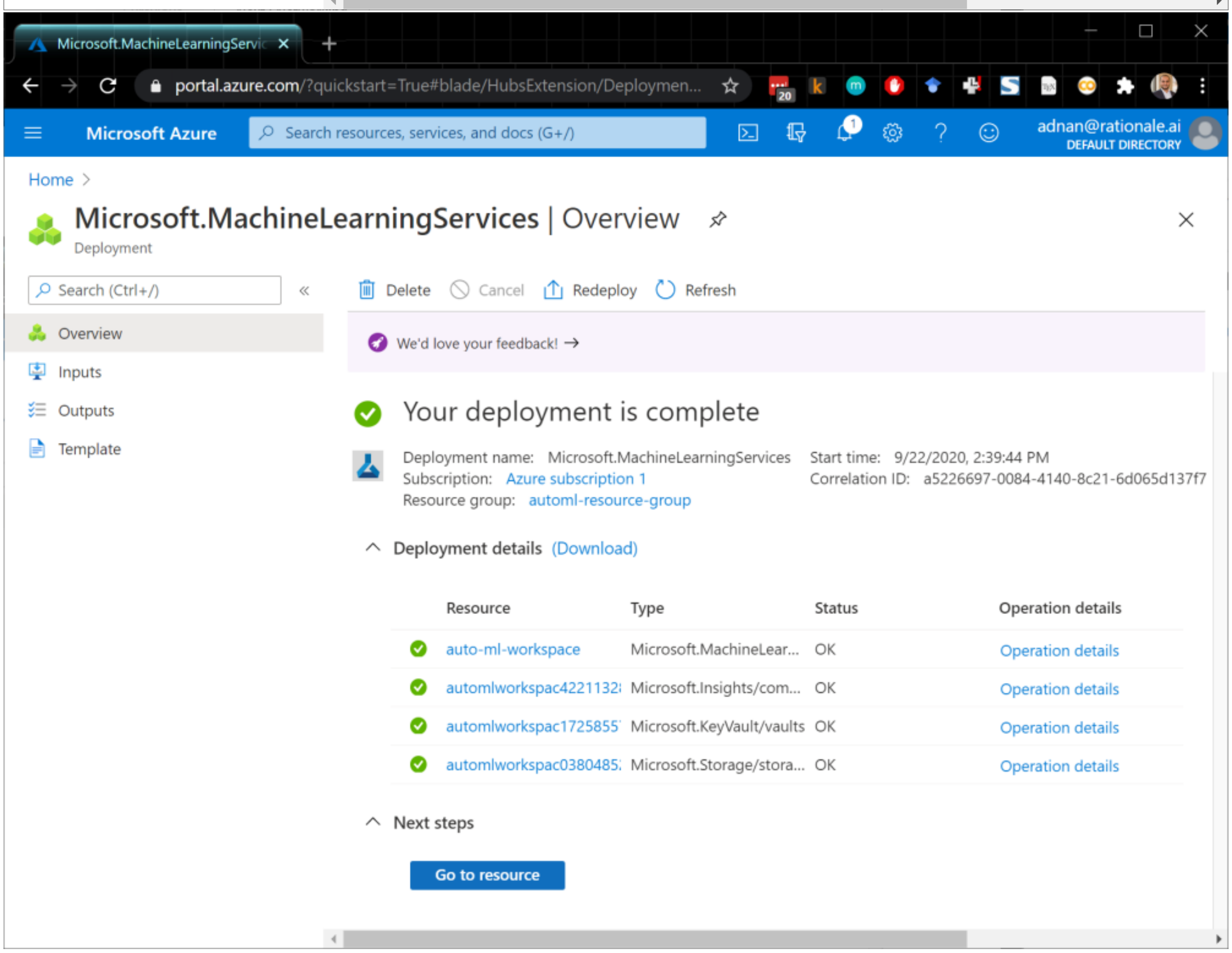

Home - Microsoft Azure
portal.azure.com/?quickstart=True#home
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Azure services
Create a resource
Subscriptions
Virtual machines
App Services
Storage accounts
SQL databases
Azure Database for PostgreSQ...
Azure Cosmos DB
Kubernetes services
More services
Navigate
Subscriptions
Resource groups
All resources
Dashboard
Tools
Microsoft Learn
Learn Azure with free online training from Microsoft
Azure Monitor
Monitor your apps and infrastructure
Security Center
Secure your apps and infrastructure
Cost Management
Analyze and optimize your cloud spend for free
Useful links
Technical Documentation
Azure Services
Recent Azure Updates

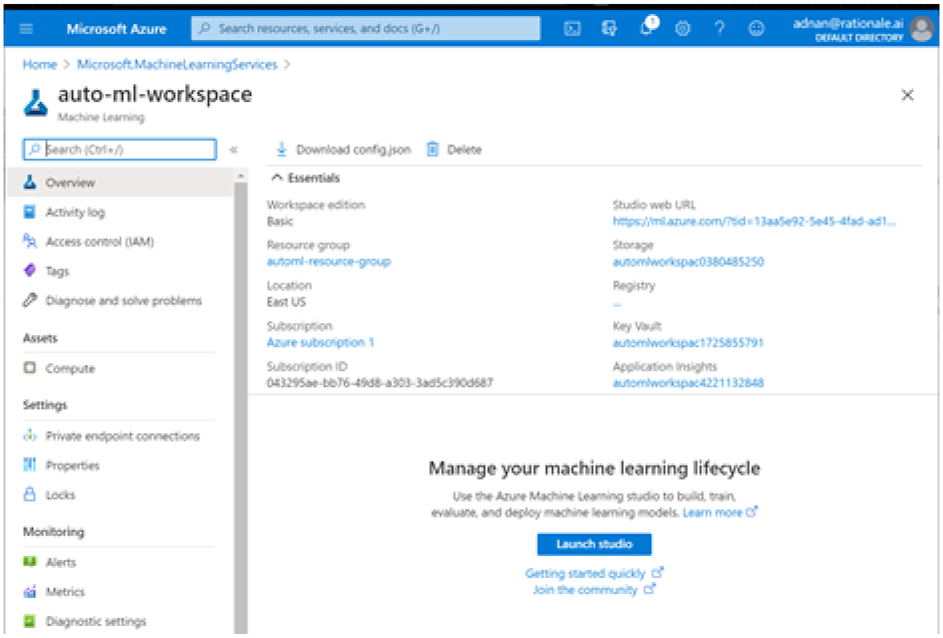

Machine Learning - Microsoft Az
portal.azure.com/?quickstart=True#blade/HubsExtension/BrowseRes...
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home
Machine Learning
Default Directory
Add
Edit columns
Refresh
Assign tags
Delete
Subscriptions: Azure subscription 1
Filter by name...
All resource groups
All locations
All tags
No grouping
0 items
Name
Resource group
Location
Subscription
No azure machine learning to display
Create a Machine Learning workspace to manage machine learning solutions through the entire data science life cycle.
Create azure machine learning


Machine Learning - Microsoft Az
portal.azure.com/?quickstart=True#create/Microsoft.MachineLearnin...
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home > Machine Learning >
Machine Learning
Create a machine learning workspace
Basics
Networking
Advanced
Tags
Review + create
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Azure subscription 1
Resource group *
Create new
Workspace details
Specify the name, region, and edition for
Workspace name *
Region *
A resource group is a container that holds related resources for an Azure solution.
Name *
automl-resource-group
OK
Cancel
For your convenience, these resources are added automatically to the workspace, if regionally available: Azure Storage, Azure Application Insights, Azure Key Vault
Review + create
< Previous
Next : Networking

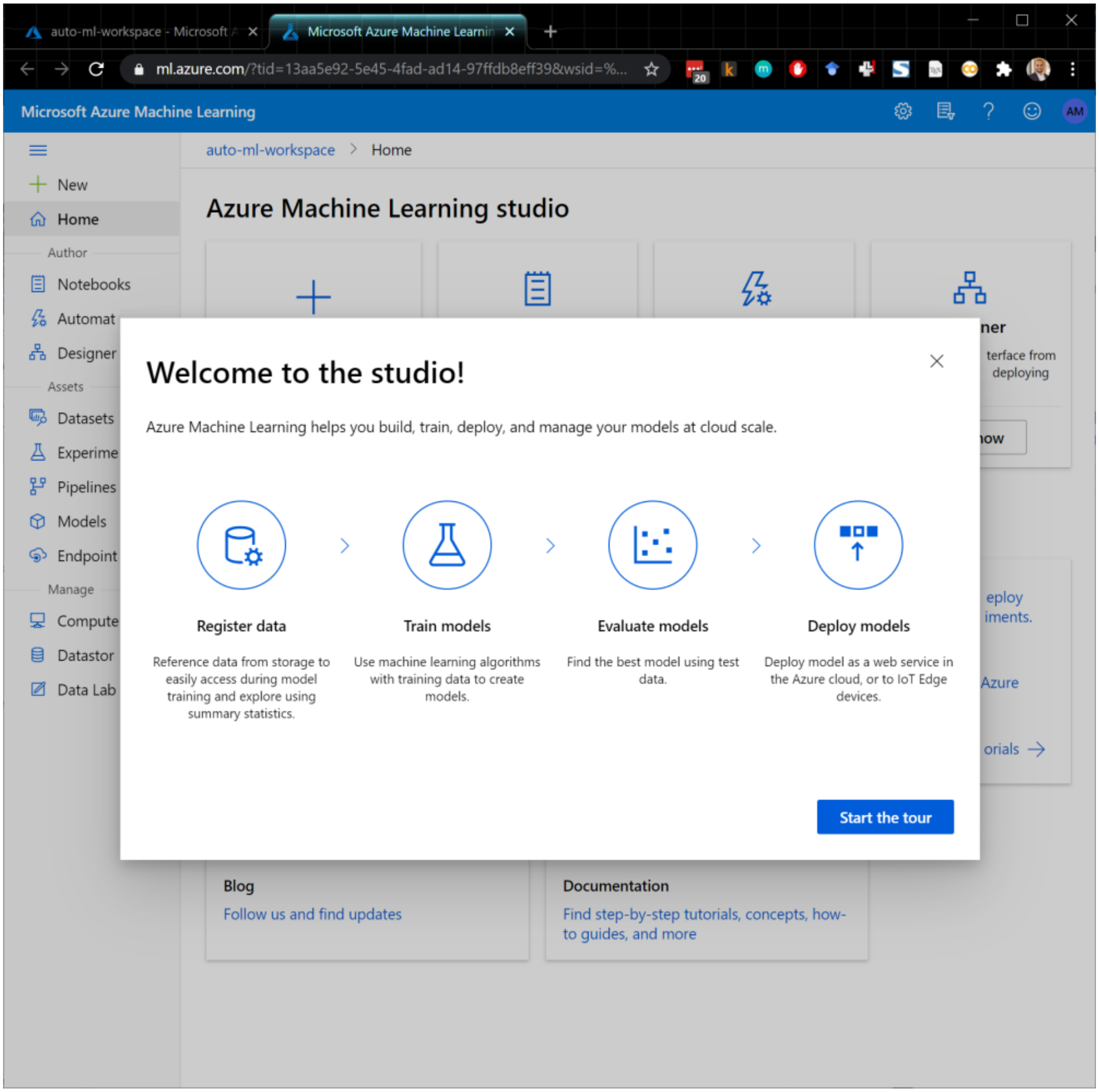

Machine Learning - Microsoft Az
portal.azure.com/?quickstart=True#create/Microsoft.MachineLearnin...
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home > Machine Learning >
Machine Learning
Create a machine learning workspace
Basics
Networking
Advanced
Tags
Review + create
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Azure subscription 1
Resource group *
(New) automl-resource-group
Create new
Workspace details
Specify the name, region, and edition for the workspace.
Workspace name *
auto-ml-workspace
Region *
East US
For your convenience, these resources are added automatically to the workspace, if regionally available: Azure Storage, Azure Application Insights, Azure Key Vault
Review + create
< Previous
Next : Networking

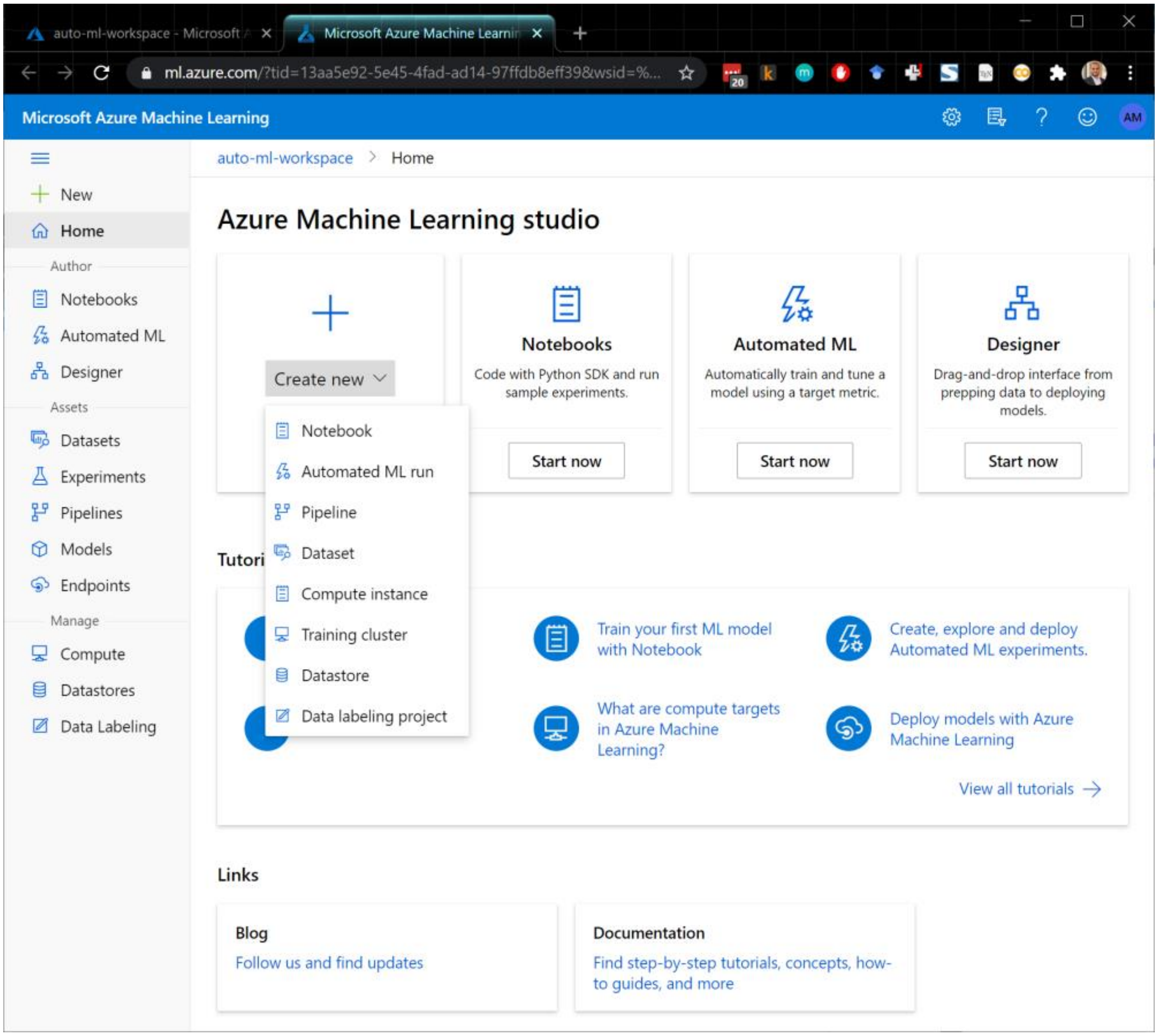

Azure Machine Learning Workplace
Compute Instances
User Roles
Reader
Contributor
Owner
Compute Targets
Local
Data Science VM
Azure ML Compute
Azure Data Lake Analytics
Azure HDInsight Cluster
Azure Kubernetes Service Cluster
Azure Container Instance
Azure Databricks
Associated Azure Resources
Azure Storage Account
Azure Container Registry
Azure Key Vault
Azure Application Insights
Experiment
Run
Snapshot
Output files
Metrics
Logs
Pipelines
Run
Snapshot
Output files
Metrics
Logs
Datasets
Registered Models
Deployment endpoints
IoT edge module
Model Telemetry
Web Service
Service Telemetry
Model Telemetry

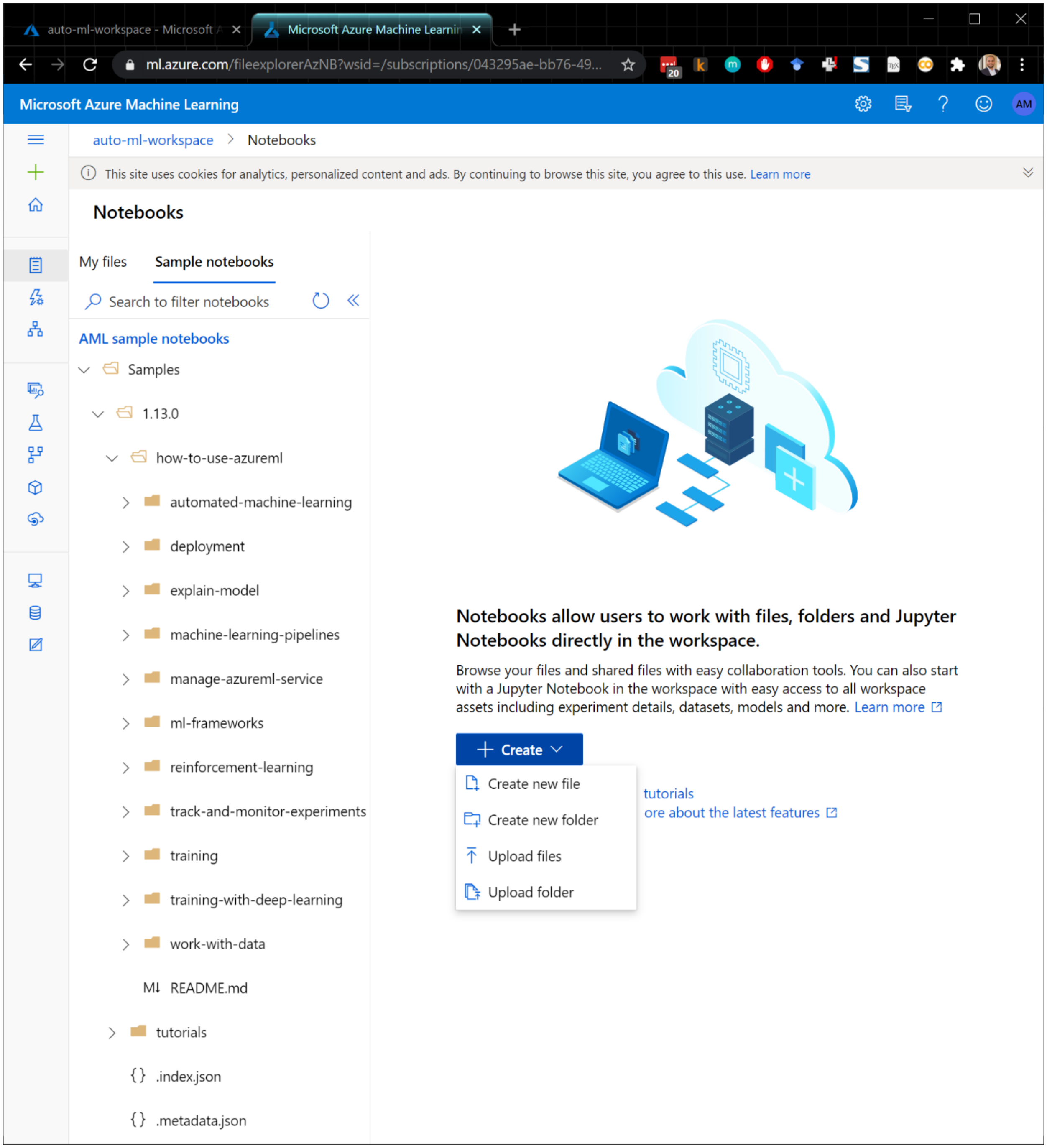

Microsoft.MachineLearningServic
portal.azure.com/?quickstart=True#blade/HubsExtension/Deploymen...
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home
Microsoft.MachineLearningServices | Overview
Deployment
Search (Ctrl+/)
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Refresh
We'd love your feedback!
Deployment is in progress
Deployment name: Microsoft.MachineLearningServices
Subscription: Azure subscription 1
Resource group: automl-resource-group
Start time: 9/22/2020, 2:39:44 PM
Correlation ID: a5226697-0084-4140-8c21-6d065d137f74
Deployment details (Download)
Resource
Type
Status
Operation details
No results.

Microsoft.MachineLearningServic
portal.azure.com/?quickstart=True#blade/HubsExtension/Deploymen...
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home
Microsoft.MachineLearningServices | Overview
Deployment
Search (Ctrl+/)
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Refresh
We'd love your feedback!
Your deployment is complete
Deployment name: Microsoft.MachineLearningServices
Subscription: Azure subscription 1
Resource group: automl-resource-group
Start time: 9/22/2020, 2:39:44 PM
Correlation ID: a5226697-0084-4140-8c21-6d065d137f7
Deployment details (Download)
Resource
Type
Status
Operation details
auto-ml-workspace
Microsoft.MachineLear...
OK
Operation details
automlworkspac4221132
Microsoft.Insights/com...
OK
Operation details
automlworkspac1725855
Microsoft.KeyVault/vaults
OK
Operation details
automlworkspac0380485
Microsoft.Storage/stora...
OK
Operation details
Next steps
Go to resource

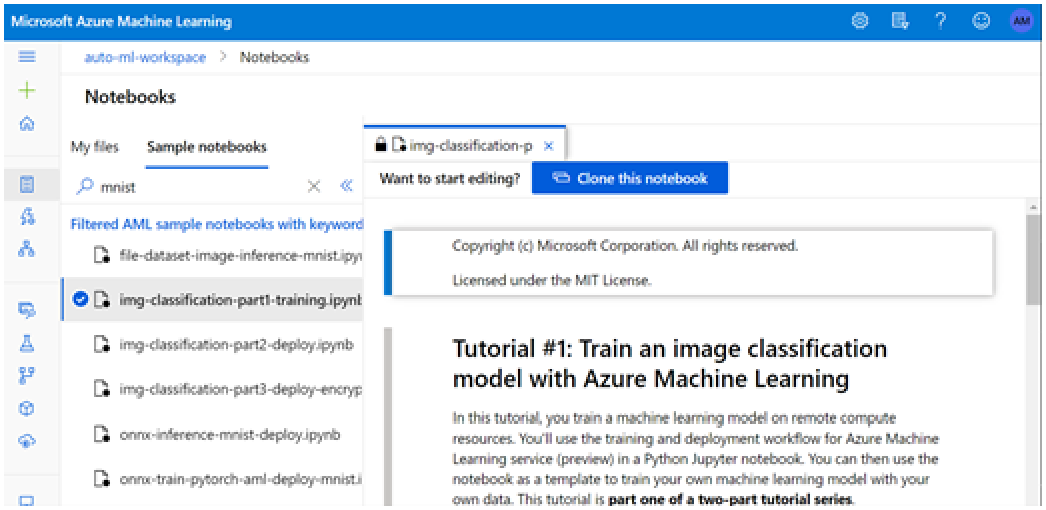


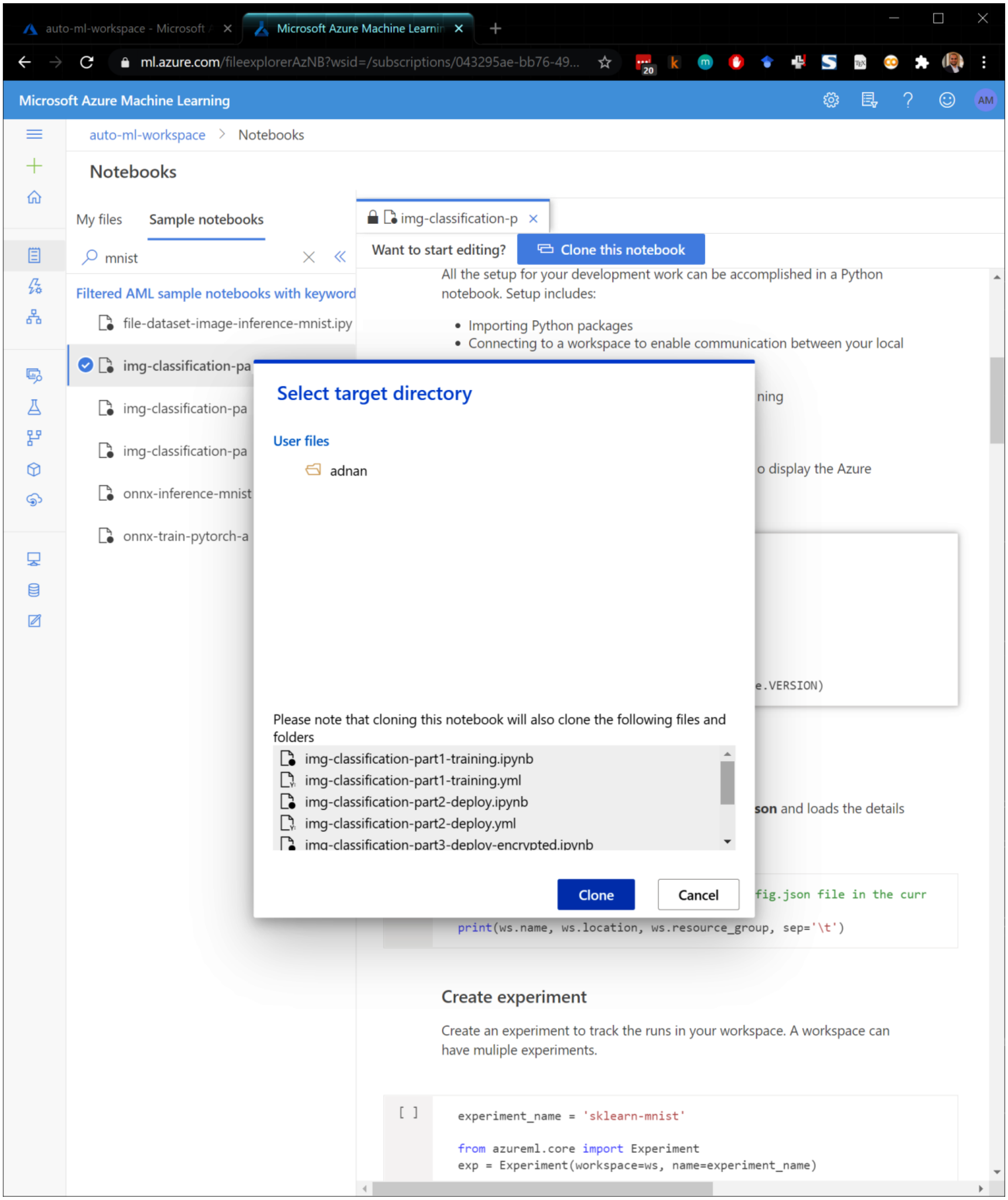
Microsoft Azure
Search resources, services, and docs (G+/)
adnan@rationale.ai
DEFAULT DIRECTORY
Home > Microsoft.MachineLearningServices >
auto-ml-workspace
Machine Learning
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Assets
Compute
Settings
Private endpoint connections
Properties
Locks
Monitoring
Alerts
Metrics
Diagnostic settings
Download config.json
Delete
Essentials
Workspace edition
Basic
Resource group
automl-resource-group
Location
East US
Subscription
Azure subscription 1
Subscription ID
043295ae-bb76-49d8-a303-3ad5c390d687
Studio web URL
https://ml.azure.com/?tid=13aa5e92-5e45-4fad-ad1...
Storage
automlworkspac0380485250
Registry
Key Vault
automlworkspac1725855791
Application Insights
automlworkspac4221132848
Manage your machine learning lifecycle
Use the Azure Machine Learning studio to build, train, evaluate, and deploy machine learning models. Learn more
Launch studio
Getting started quickly
Join the community


auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/?tid=13aa5e92-5e45-4fad-ad14-97ffdb8eff39&wsid=%...
Microsoft Azure Machine Learning
auto-ml-workspace
Home
New
Home
Author
Notebooks
Automat
Designer
Assets
Datasets
Experime
Pipelines
Models
Endpoint
Manage
Compute
Datastor
Data Lab
Azure Machine Learning studio
Welcome to the studio!
Azure Machine Learning helps you build, train, deploy, and manage your models at cloud scale.
Register data
Reference data from storage to easily access during model training and explore using summary statistics.
Train models
Use machine learning algorithms with training data to create models.
Evaluate models
Find the best model using test data.
Deploy models
Deploy model as a web service in the Azure cloud, or to IoT Edge devices.
Start the tour
Blog
Follow us and find updates
Documentation
Find step-by-step tutorials, concepts, how-to guides, and more

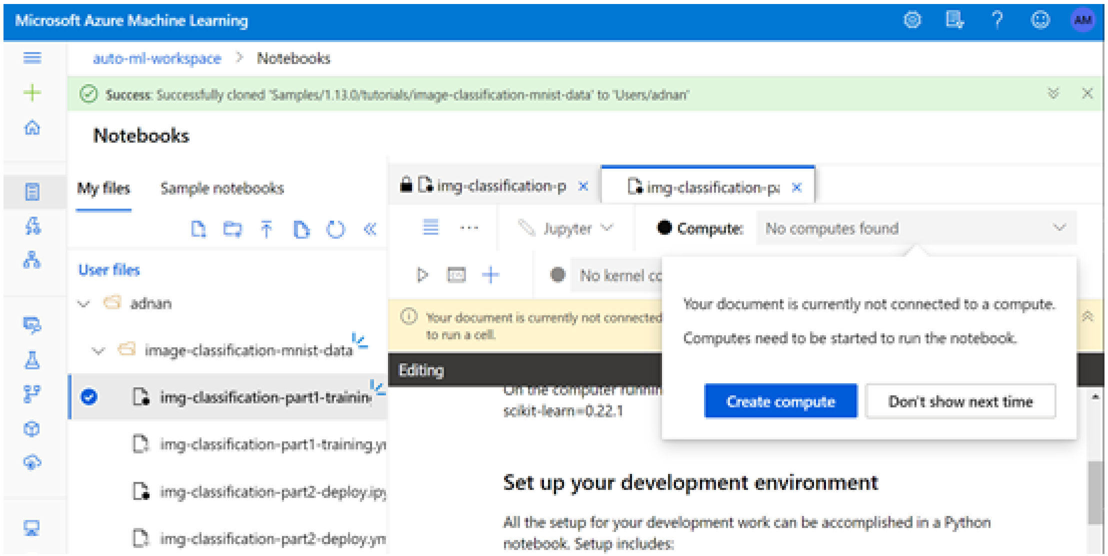

auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/?tid=13aa5e92-5e45-4fad-ad14-97ffdb8eff39&wsid=%...
Microsoft Azure Machine Learning
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
auto-ml-workspace
Home
Azure Machine Learning studio
Create new
Notebook
Automated ML run
Pipeline
Dataset
Compute instance
Training cluster
Datastore
Data labeling project
Notebooks
Code with Python SDK and run sample experiments.
Start now
Automated ML
Automatically train and tune a model using a target metric.
Start now
Designer
Drag-and-drop interface from prepping data to deploying models.
Start now
Tutori
Train your first ML model with Notebook
Create, explore and deploy Automated ML experiments.
What are compute targets in Azure Machine Learning?
Deploy models with Azure Machine Learning
View all tutorials
Links
Blog
Follow us and find updates
Documentation
Find step-by-step tutorials, concepts, how-to guides, and more


auto-ml-workspace - Microsoft A
Microsoft Azure Machine Learnin
ml.azure.com/fileexplorerAzNB?wsid=/subscriptions/043295ae-bb76-49...
Microsoft Azure Machine Learning
AM
auto-ml-workspace
Notebooks
This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use. Learn more
Notebooks
My files
Sample notebooks
Search to filter notebooks
AML sample notebooks
Samples
1.13.0
how-to-use-azureml
automated-machine-learning
deployment
explain-model
machine-learning-pipelines
manage-azureml-service
ml-frameworks
reinforcement-learning
track-and-monitor-experiments
training
training-with-deep-learning
work-with-data
README.md
tutorials
.index.json
.metadata.json
Notebooks allow users to work with files, folders and Jupyter Notebooks directly in the workspace.
Browse your files and shared files with easy collaboration tools. You can also start with a Jupyter Notebook in the workspace with easy access to all workspace assets including experiment details, datasets, models and more. Learn more
Create
Create new file
Create new folder
Upload files
Upload folder
tutorials
ore about the latest features

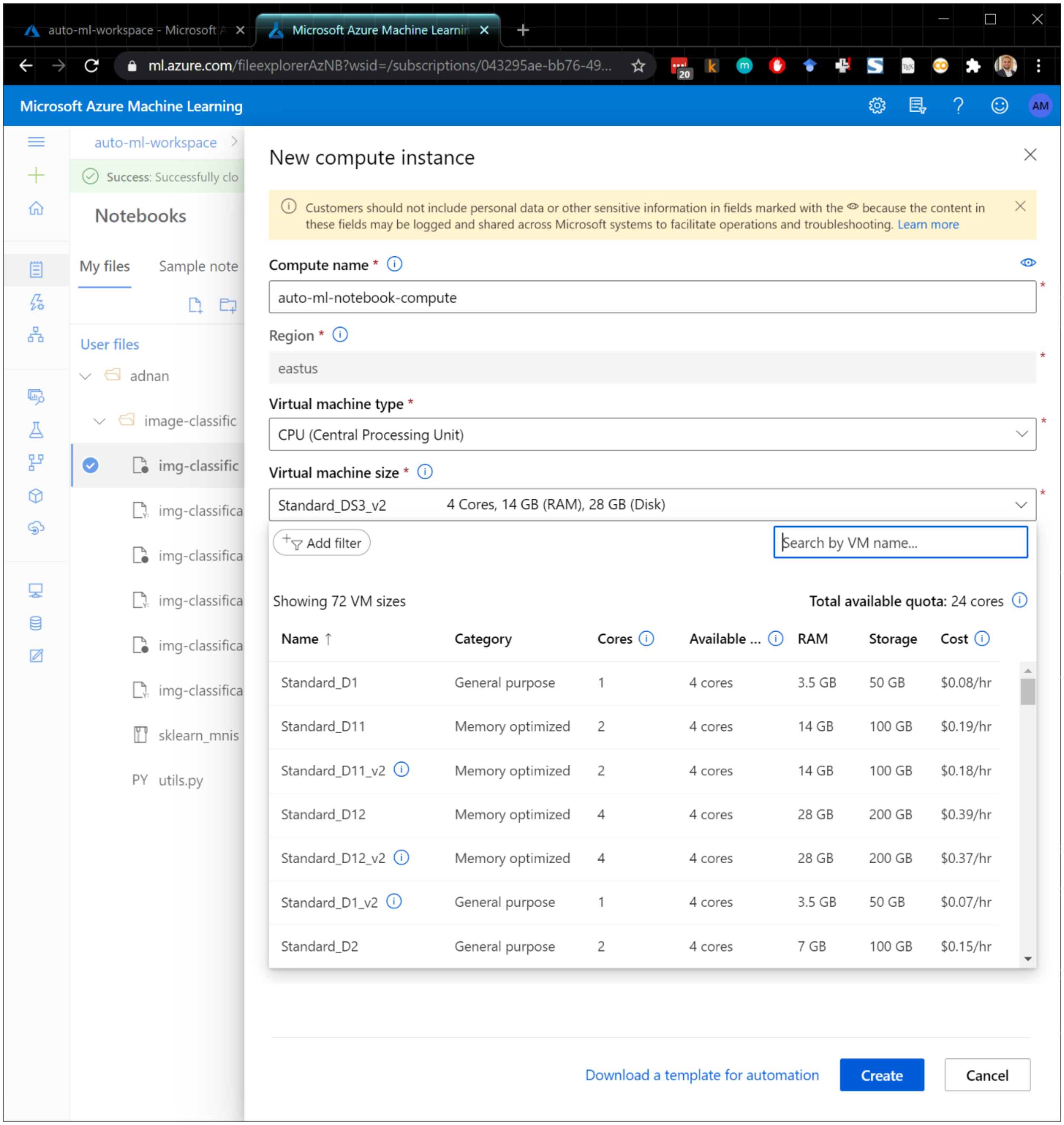

Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
Notebooks
My files
Sample notebooks
mnist
Filtered AML sample notebooks with keyword
file-dataset-image-inference-mnist.ipy
img-classification-part1-training.ipynb
img-classification-part2-deploy.ipynb
img-classification-part3-deploy-encryp
onnx-inference-mnist-deploy.ipynb
onnx-train-pytorch-aml-deploy-mnist.i
img-classification-p
Want to start editing?
Clone this notebook
Copyright (c) Microsoft Corporation. All rights reserved.
Licensed under the MIT License.
Tutorial #1: Train an image classification model with Azure Machine Learning
In this tutorial, you train a machine learning model on remote compute resources. You'll use the training and deployment workflow for Azure Machine Learning service (preview) in a Python Jupyter notebook. You can then use the notebook as a template to train your own machine learning model with your own data. This tutorial is part one of a two-part tutorial series.

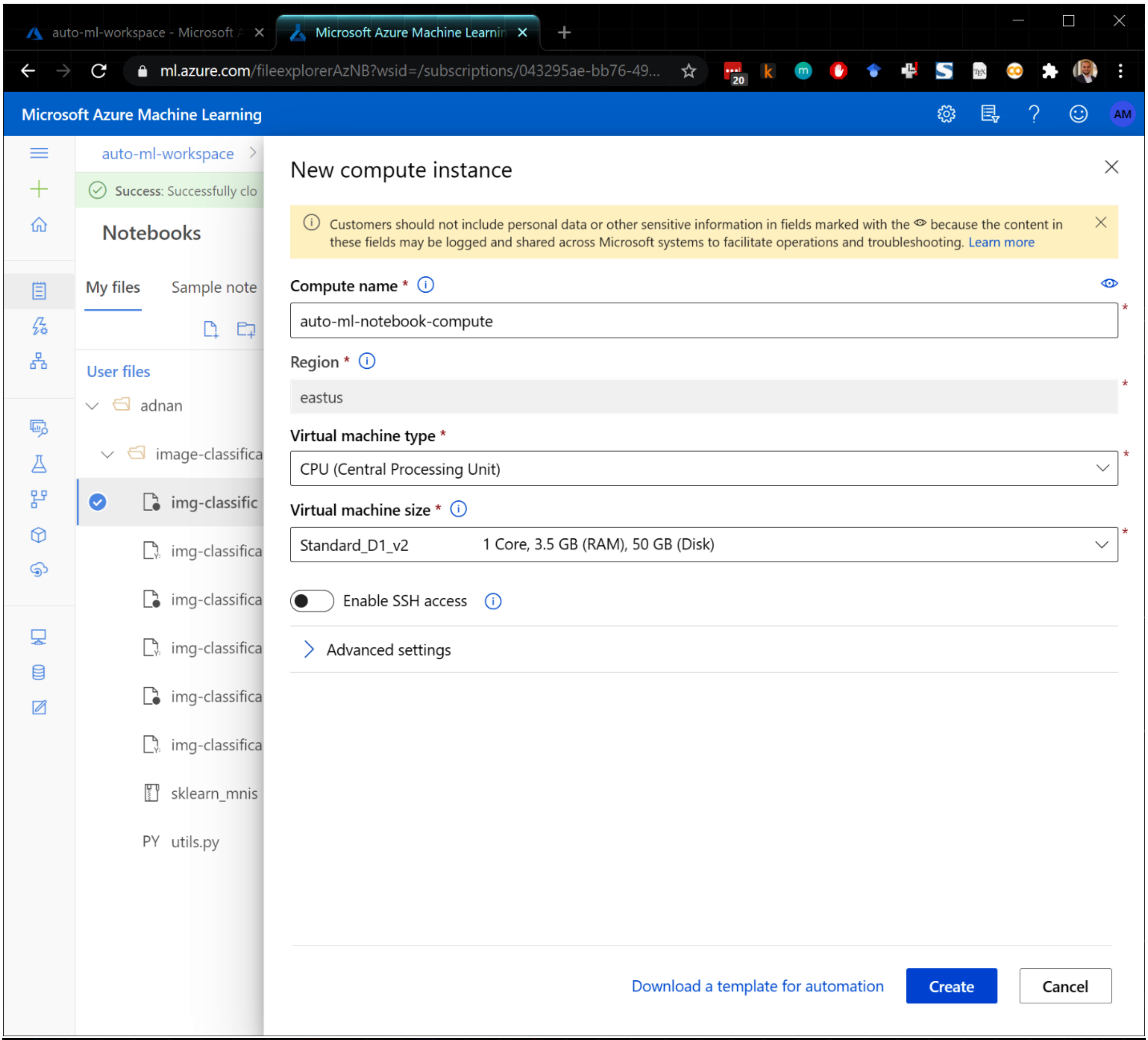
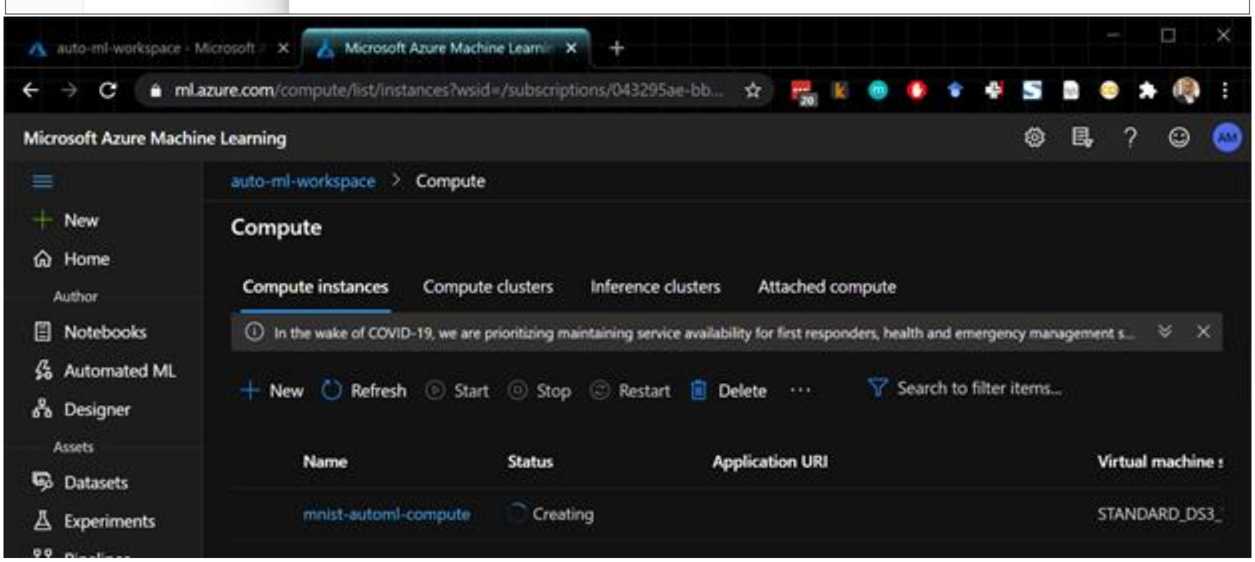


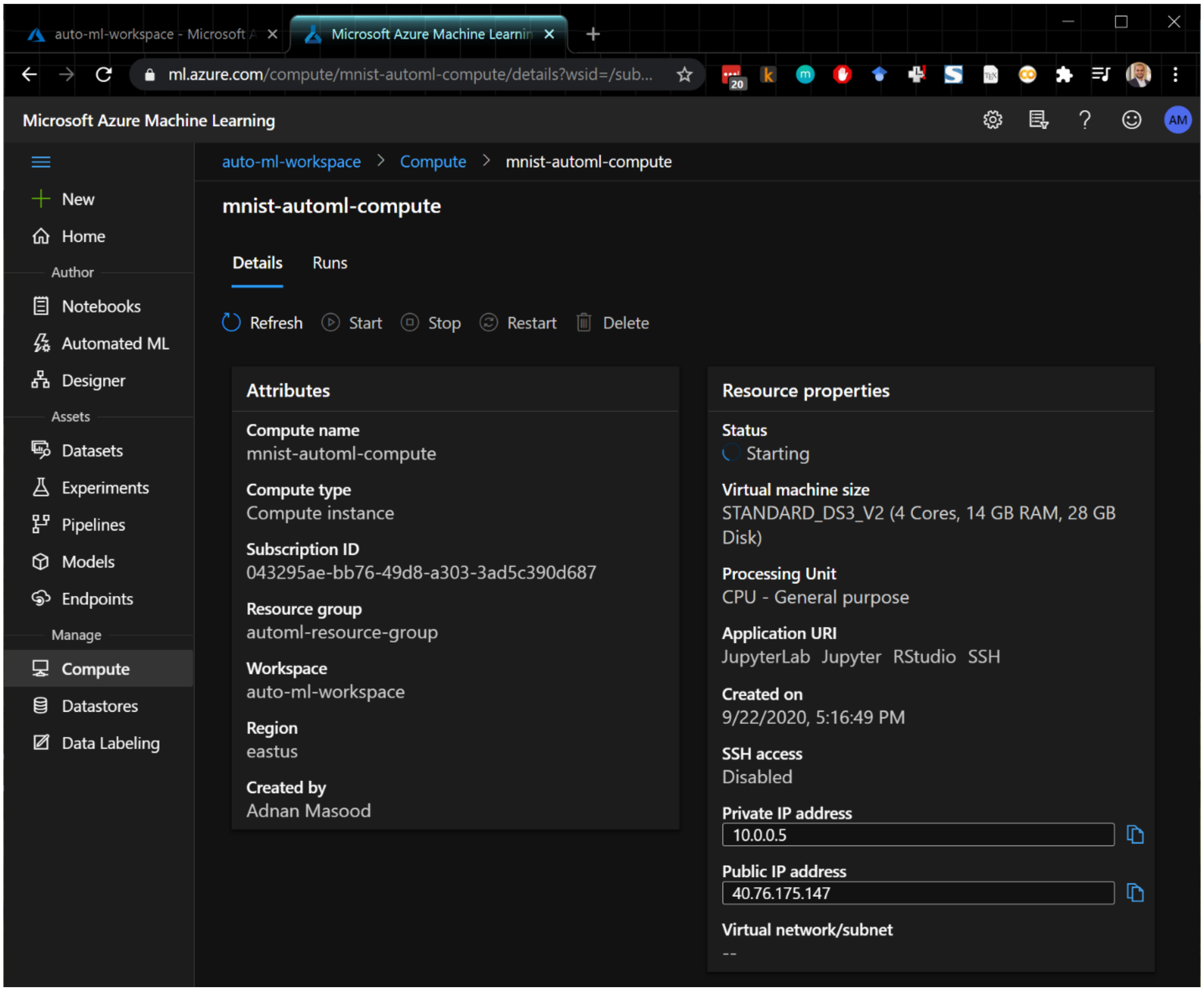
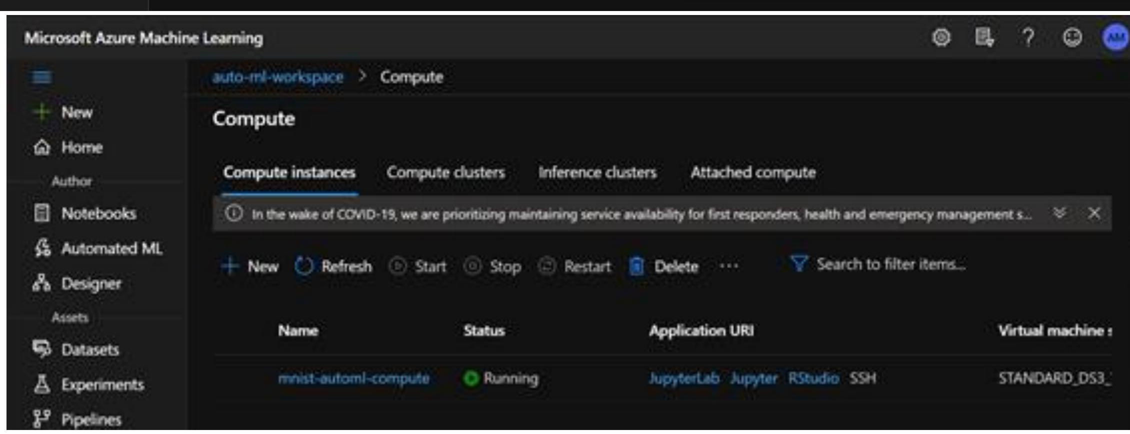
auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/fileexplorerAzNB?wsid=/subscriptions/043295ae-bb76-49...
Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
Notebooks
My files
Sample notebooks
mnist
Filtered AML sample notebooks with keyword
file-dataset-image-inference-mnist.ipy
img-classification-pa
img-classification-pa
img-classification-pa
onnx-inference-mnist
onnx-train-pytorch-a
img-classification-p
Want to start editing?
Clone this notebook
All the setup for your development work can be accomplished in a Python notebook. Setup includes:
Importing Python packages
Connecting to a workspace to enable communication between your local
Select target directory
User files
adnan
Please note that cloning this notebook will also clone the following files and folders
img-classification-part1-training.ipynb
img-classification-part1-training.yml
img-classification-part2-deploy.ipynb
img-classification-part2-deploy.yml
img-classification-part3-deploy-encrypted.ipynb
Clone
Cancel
print(ws.name, ws.location, ws.resource_group, sep='\t')
Create experiment
Create an experiment to track the runs in your workspace. A workspace can have muliple experiments.
experiment_name = 'sklearn-mnist'
from azureml.core import Experiment
exp = Experiment(workspace=ws, name=experiment_name)

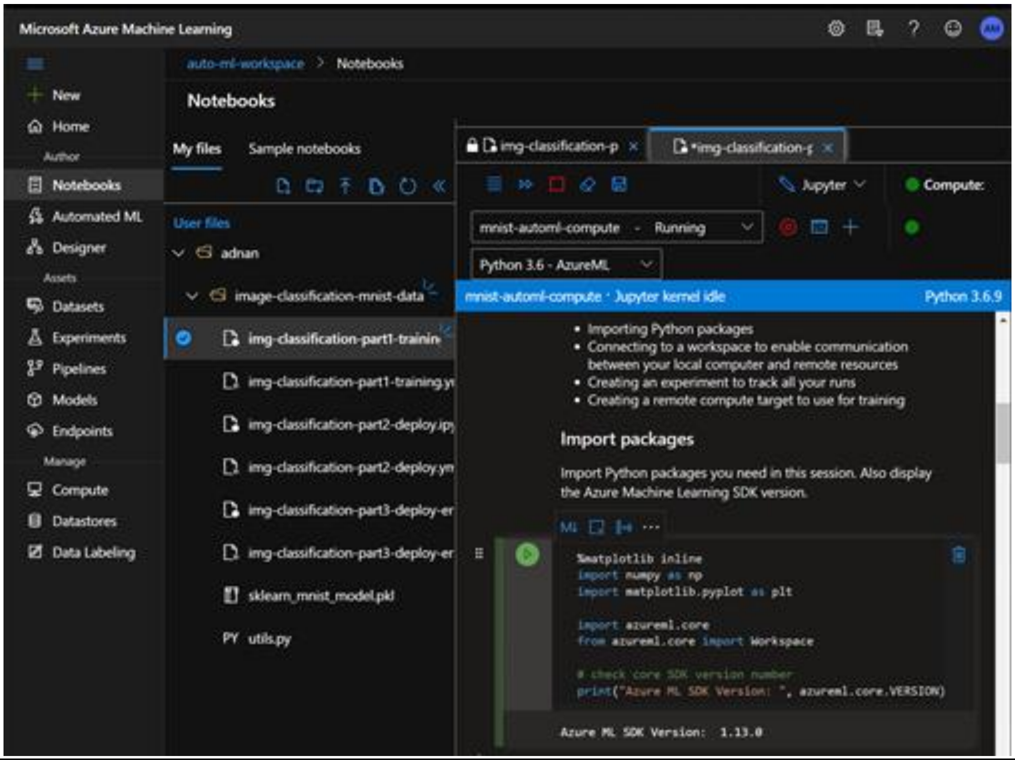


Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
Success: Successfully cloned 'Samples/1.13.0/tutorials/image-classification-mnist-data' to 'Users/adnan'
Notebooks
My files
Sample notebooks
User files
adnan
image-classification-mnist-data
img-classification-part1-trainin
img-classification-part1-training.y
img-classification-part2-deploy.ip
img-classification-part2-deploy.ym
img-classification-p
img-classification-p
Jupyter
Compute:
No computes found
No kernel c
Your document is currently not connected
to run a cell.
Editing
scikit-learn=0.22.1
Your document is currently not connected to a compute.
Computes need to be started to run the notebook.
Create compute
Don't show next time
Set up your development environment
All the setup for your development work can be accomplished in a Python notebook. Setup includes:

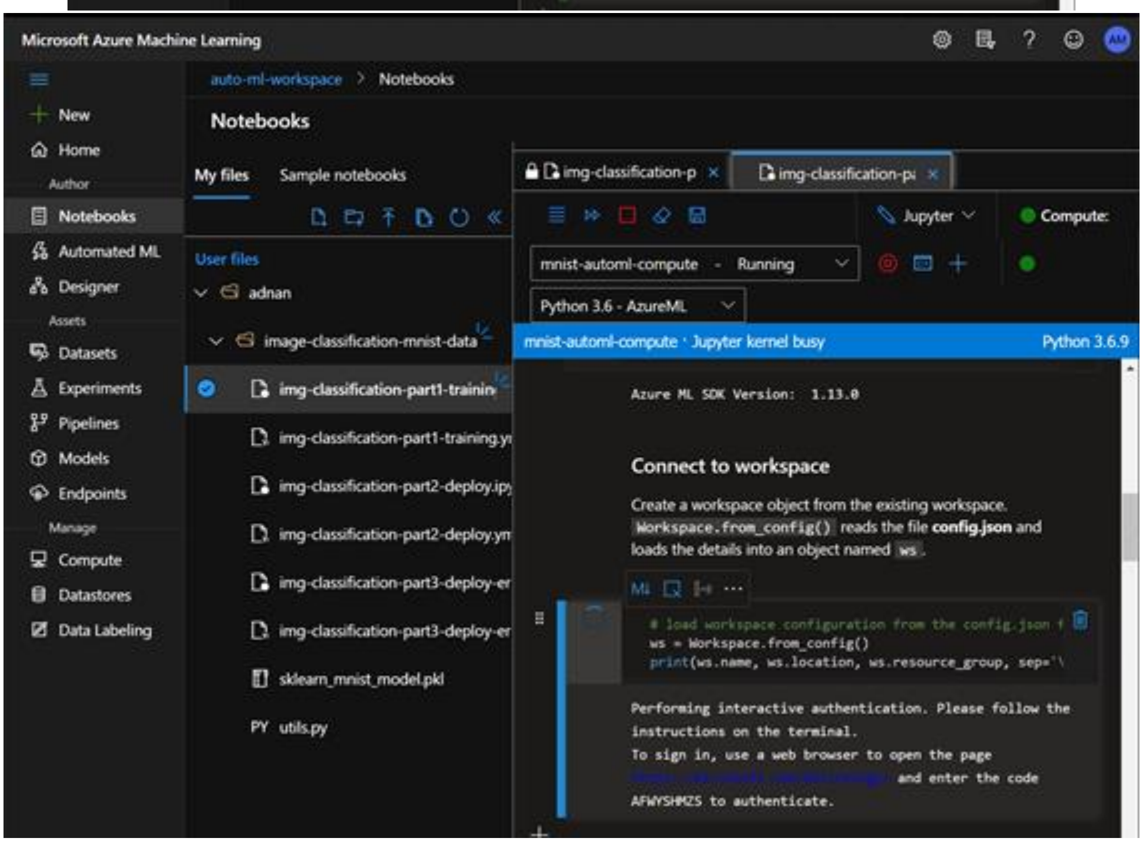

auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/fileexplorerAzNB?wsid=/subscriptions/043295ae-bb76-49...
Microsoft Azure Machine Learning
auto-ml-workspace
Success: Successfully clo
Notebooks
My files
Sample note
User files
adnan
image-classific
img-classific
img-classifica
img-classifica
img-classifica
img-classifica
img-classifica
sklearn_mnis
utils.py
New compute instance
Customers should not include personal data or other sensitive information in fields marked with the because the content in these fields may be logged and shared across Microsoft systems to facilitate operations and troubleshooting. Learn more
Compute name *
auto-ml-notebook-compute
Region *
eastus
Virtual machine type *
CPU (Central Processing Unit)
Virtual machine size *
Standard_DS3_v2 4 Cores, 14 GB (RAM), 28 GB (Disk)
Add filter
Search by VM name...
Showing 72 VM sizes
Total available quota: 24 cores
Name Category Cores Available ... RAM Storage Cost
Standard_D1 General purpose 1 4 cores 3.5 GB 50 GB $0.08/hr
Standard_D11 Memory optimized 2 4 cores 14 GB 100 GB $0.19/hr
Standard_D11_v2 Memory optimized 2 4 cores 14 GB 100 GB $0.18/hr
Standard_D12 Memory optimized 4 4 cores 28 GB 200 GB $0.39/hr
Standard_D12_v2 Memory optimized 4 4 cores 28 GB 200 GB $0.37/hr
Standard_D1_v2 General purpose 1 4 cores 3.5 GB 50 GB $0.07/hr
Standard_D2 General purpose 2 4 cores 7 GB 100 GB $0.15/hr
Download a template for automation
Create
Cancel


auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/fileexplorerAzNB?wsid=/subscriptions/043295ae-bb76-49...
Microsoft Azure Machine Learning
auto-ml-workspace
Success: Successfully clo
Notebooks
My files
Sample note
User files
adnan
image-classifica
img-classific
img-classifica
img-classifica
img-classifica
img-classifica
img-classifica
sklearn_mnis
PY utils.py
New compute instance
Customers should not include personal data or other sensitive information in fields marked with the because the content in these fields may be logged and shared across Microsoft systems to facilitate operations and troubleshooting. Learn more
Compute name *
auto-ml-notebook-compute
Region *
eastus
Virtual machine type *
CPU (Central Processing Unit)
Virtual machine size *
Standard_D1_v2
1 Core, 3.5 GB (RAM), 50 GB (Disk)
Enable SSH access
Advanced settings
Download a template for automation
Create
Cancel

auto-ml-workspace - Microsoft
Microsoft Azure Machine Learn
ml.azure.com/compute/list/instances?wsid=/subscriptions/043295ae-bb...
Microsoft Azure Machine Learning
auto-ml-workspace
Compute
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Compute
Compute instances
Compute clusters
Inference clusters
Attached compute
In the wake of COVID-19, we are prioritizing maintaining service availability for first responders, health and emergency management s...
New
Refresh
Start
Stop
Restart
Delete
Search to filter items...
Name
Status
Application URI
Virtual machine s
mnist-automl-compute
Creating
STANDARD_DS3_

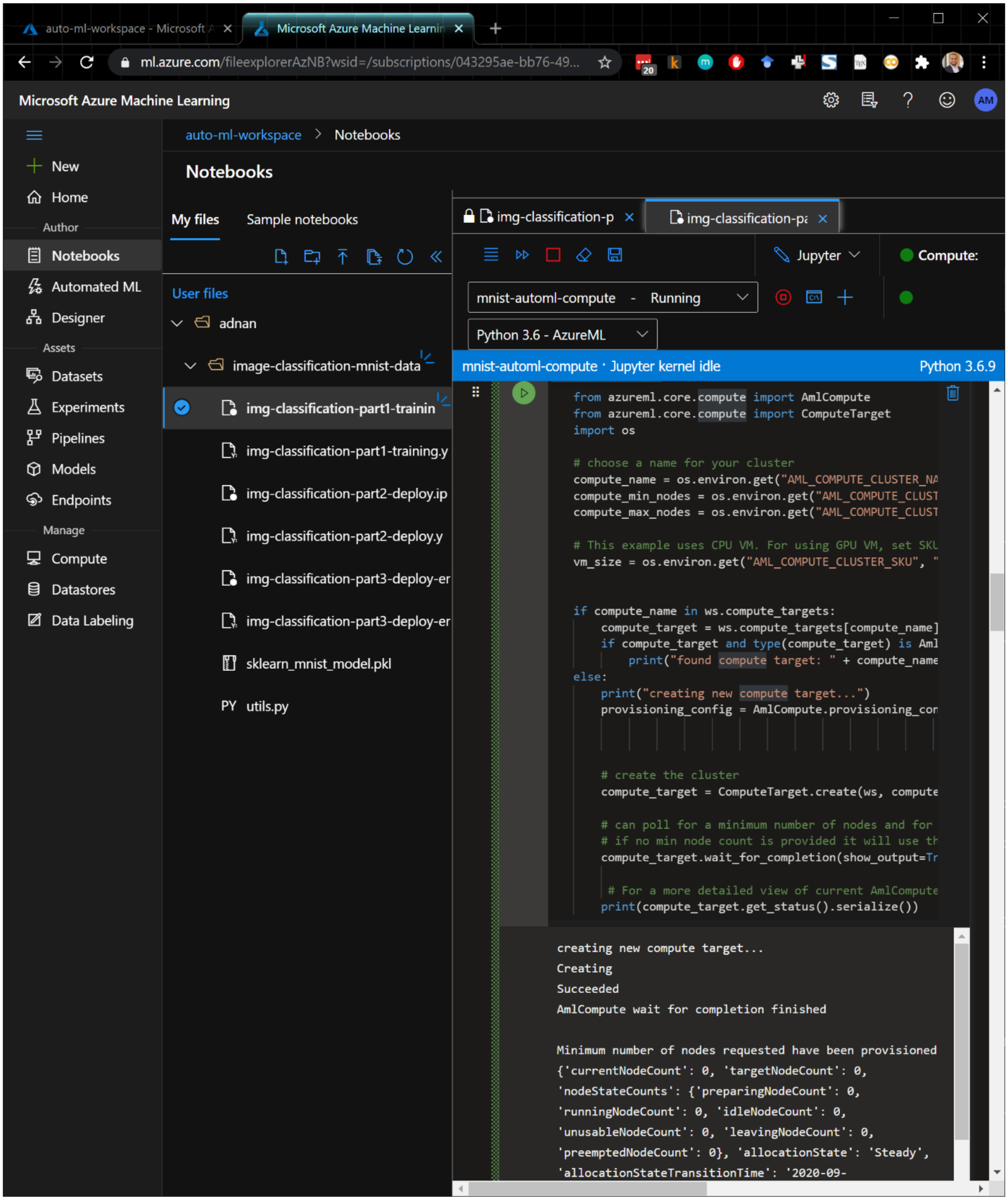

auto-ml-workspace - Microsoft A
Microsoft Azure Machine Learnin
ml.azure.com/compute/mnist-automl-compute/details?wsid=/sub...
Microsoft Azure Machine Learning
auto-ml-workspace > Compute > mnist-automl-compute
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
mnist-automl-compute
Details
Runs
Refresh
Start
Stop
Restart
Delete
Attributes
Compute name
mnist-automl-compute
Compute type
Compute instance
Subscription ID
043295ae-bb76-49d8-a303-3ad5c390d687
Resource group
automl-resource-group
Workspace
auto-ml-workspace
Region
eastus
Created by
Adnan Masood
Resource properties
Status
Starting
Virtual machine size
STANDARD_DS3_V2 (4 Cores, 14 GB RAM, 28 GB Disk)
Processing Unit
CPU - General purpose
Application URI
JupyterLab Jupyter RStudio SSH
Created on
9/22/2020, 5:16:49 PM
SSH access
Disabled
Private IP address
10.0.0.5
Public IP address
40.76.175.147
Virtual network/subnet
--

Microsoft Azure Machine Learning
auto-ml-workspace > Compute
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Compute
Compute instances
Compute clusters
Inference clusters
Attached compute
In the wake of COVID-19, we are prioritizing maintaining service availability for first responders, health and emergency management s...
New
Refresh
Start
Stop
Restart
Delete
Search to filter items...
Name
Status
Application URI
Virtual machine s
mnist-automl-compute
Running
JupyterLab Jupyter RStudio SSH
STANDARD_DS3_

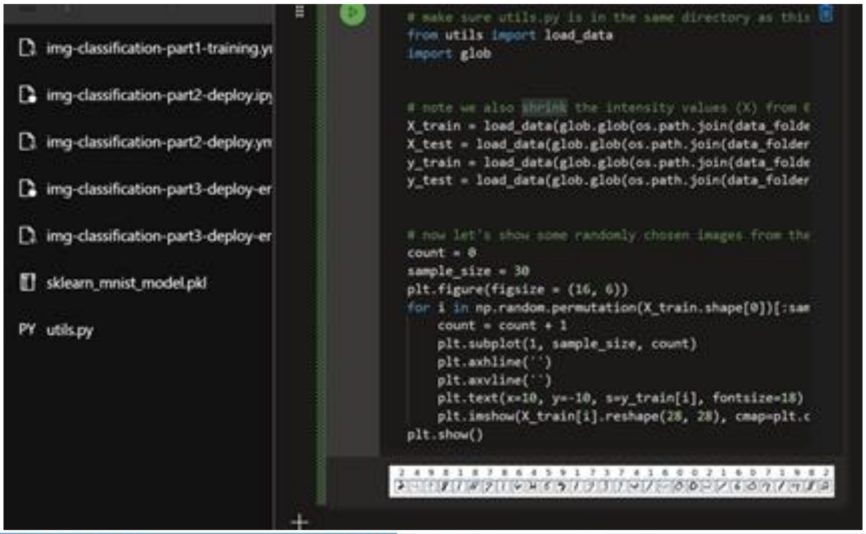


Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
Notebooks
My files
Sample notebooks
User files
adnan
image-classification-mnist-data
img-classification-part1-trainin
img-classification-part1-training.y
img-classification-part2-deploy.ip
img-classification-part2-deploy.ym
img-classification-part3-deploy-er
img-classification-part3-deploy-er
sklearn_mnist_model.pkl
utils.py
img-classification-p
*img-classification-p
Jupyter
Compute:
mnist-automl-compute - Running
Python 3.6 - AzureML
mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
• Importing Python packages
• Connecting to a workspace to enable communication between your local computer and remote resources
• Creating an experiment to track all your runs
• Creating a remote compute target to use for training
Import packages
Import Python packages you need in this session. Also display the Azure Machine Learning SDK version.
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
import azureml.core
from azureml.core import Workspace
# check core SDK version number
print("Azure ML SDK Version: ", azureml.core.VERSION)
Azure ML SDK Version: 1.13.0

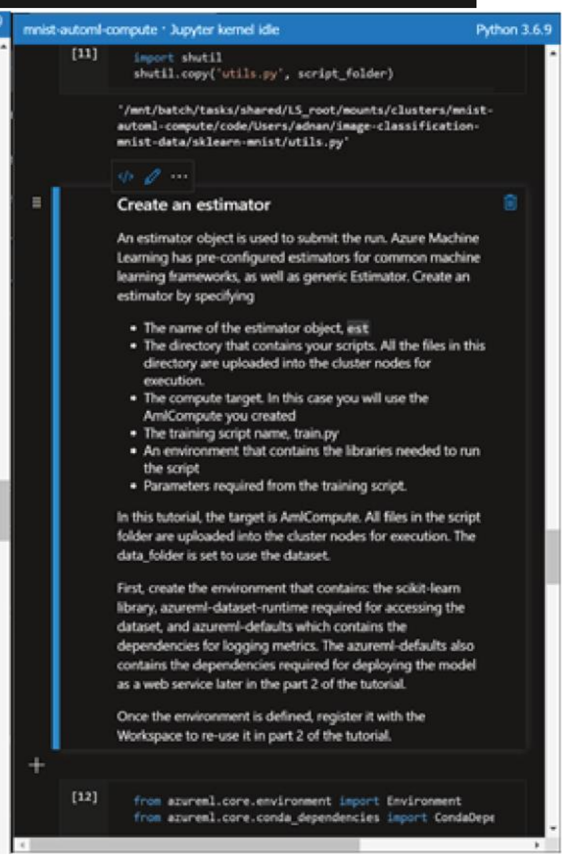


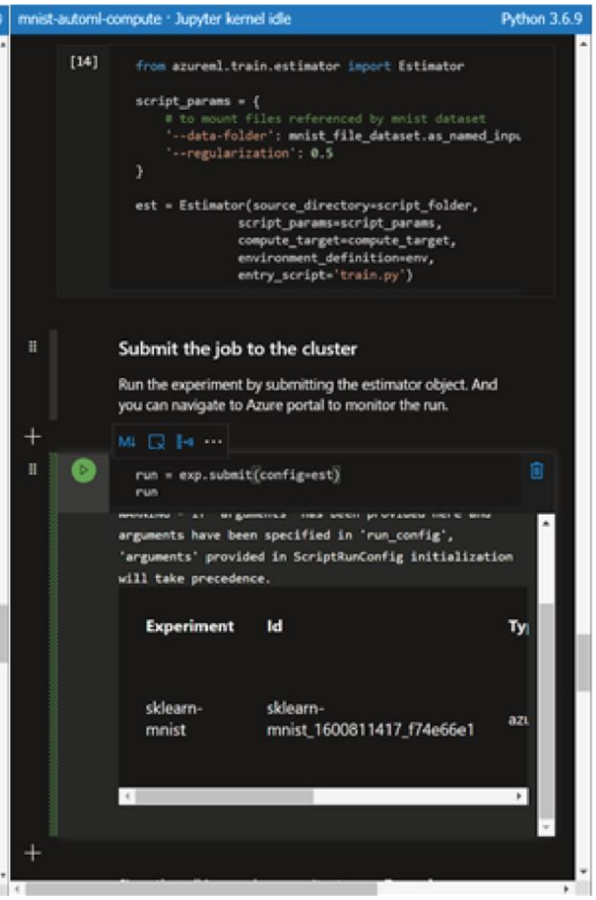
Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
Notebooks
My files
Sample notebooks
User files
adnan
image-classification-mnist-data
img-classification-part1-trainin
img-classification-part1-training.y
img-classification-part2-deploy.ip
img-classification-part2-deploy.ym
img-classification-part3-deploy-er
img-classification-part3-deploy-er
sklearn_mnist_model.pkl
utils.py
img-classification-p
img-classification-p
Jupyter
Compute:
mnist-automl-compute - Running
Python 3.6 - AzureML
mnist-automl-compute · Jupyter kernel busy
Python 3.6.9
Azure ML SDK Version: 1.13.0
Connect to workspace
Create a workspace object from the existing workspace. Workspace.from_config() reads the file config.json and loads the details into an object named ws.
# load workspace configuration from the config.json f
ws = Workspace.from_config()
print(ws.name, ws.location, ws.resource_group, sep='\
Performing interactive authentication. Please follow the instructions on the terminal.
To sign in, use a web browser to open the page
and enter the code AFWYSHMZS to authenticate.

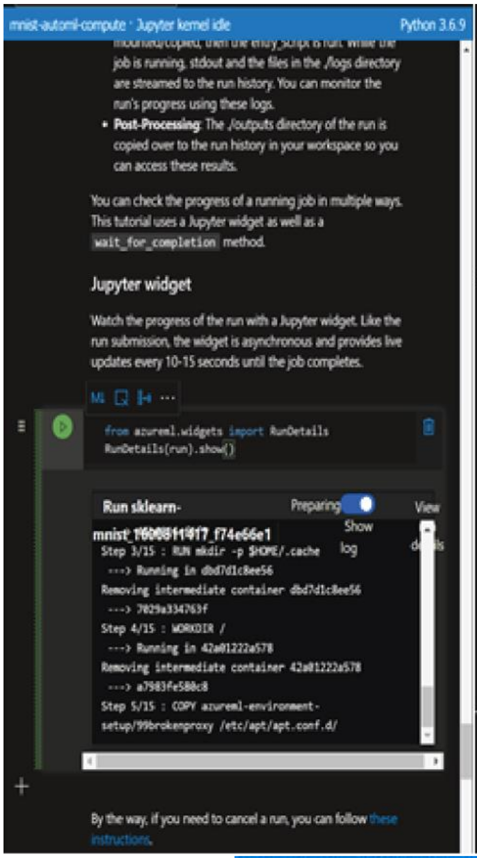

auto-ml-workspace - Microsoft
Microsoft Azure Machine Learnin
ml.azure.com/fileexplorerAzNB?wsid=/subscriptions/043295ae-bb76-49...
Microsoft Azure Machine Learning
auto-ml-workspace > Notebooks
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
Notebooks
My files
Sample notebooks
User files
adnan
image-classification-mnist-data
img-classification-part1-trainin
img-classification-part1-training.y
img-classification-part2-deploy.ip
img-classification-part2-deploy.y
img-classification-part3-deploy-er
img-classification-part3-deploy-er
sklearn_mnist_model.pkl
PY utils.py
img-classification-p
img-classification-pa
Jupyter
Compute:
mnist-automl-compute - Running
Python 3.6 - AzureML
mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
from azureml.core.compute import AmlCompute
from azureml.core.compute import ComputeTarget
import os
# choose a name for your cluster
compute_name = os.environ.get("AML_COMPUTE_CLUSTER_NA
compute_min_nodes = os.environ.get("AML_COMPUTE_CLUST
compute_max_nodes = os.environ.get("AML_COMPUTE_CLUST
# This example uses CPU VM. For using GPU VM, set SKU
vm_size = os.environ.get("AML_COMPUTE_CLUSTER_SKU", "
if compute_name in ws.compute_targets:
compute_target = ws.compute_targets[compute_name]
if compute_target and type(compute_target) is Aml
print("found compute target: " + compute_name
else:
print("creating new compute target...")
provisioning_config = AmlCompute.provisioning_cor
# create the cluster
compute_target = ComputeTarget.create(ws, compute
# can poll for a minimum number of nodes and for
# if no min node count is provided it will use th
compute_target.wait_for_completion(show_output=Tr
# For a more detailed view of current AmlCompute
print(compute_target.get_status().serialize())
creating new compute target...
Creating
Succeeded
AmlCompute wait for completion finished
Minimum number of nodes requested have been provisioned
{'currentNodeCount': 0, 'targetNodeCount': 0,
'nodeStateCounts': {'preparingNodeCount': 0,
'runningNodeCount': 0, 'idleNodeCount': 0,
'unusableNodeCount': 0, 'leavingNodeCount': 0,
'preemptedNodeCount': 0}, 'allocationState': 'Steady',
'allocationStateTransitionTime': '2020-09-

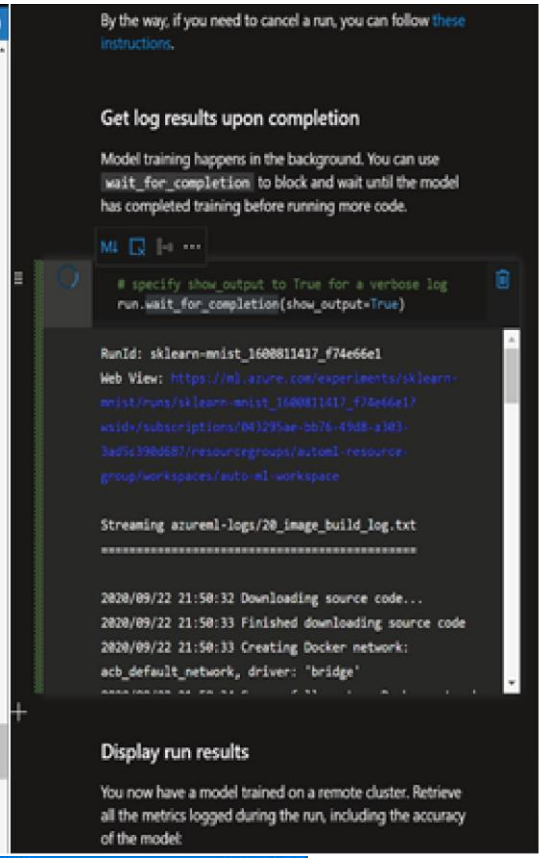

img-classification-part1-training.y
img-classification-part2-deploy.ip
img-classification-part2-deploy.ym
img-classification-part3-deploy-er
img-classification-part3-deploy-er
sklearn_mnist_model.pkl
PY utils.py
# make sure utils.py is in the same directory as this
from utils import load_data
import glob
# note we also shrink the intensity values (X) from 0
X_train = load_data(glob.glob(os.path.join(data_folde
X_test = load_data(glob.glob(os.path.join(data_folder
y_train = load_data(glob.glob(os.path.join(data_folde
y_test = load_data(glob.glob(os.path.join(data_folder
# now let's show some randomly chosen images from the
count = 0
sample_size = 30
plt.figure(figsize = (16, 6))
for i in np.random.permutation(X_train.shape[0])[:sam
count = count + 1
plt.subplot(1, sample_size, count)
plt.axhline('')
plt.axvline('')
plt.text(x=10, y=-10, s=y_train[i], fontsize=18)
plt.imshow(X_train[i].reshape(28, 28), cmap=plt.c
plt.show()


nist-automl-compute · Jupyter kernel idle
Python 3.6.9
# let user feed in 2 parameters, the dataset to mount
parser = argparse.ArgumentParser()
parser.add_argument('--data-folder', type=str, dest='
parser.add_argument('--regularization', type=float, d
args = parser.parse_args()
data_folder = args.data_folder
print('Data folder:', data_folder)
# load train and test set into numpy arrays
# note we scale the pixel intensity values to 0-1 (by
X_train = load_data(glob.glob(os.path.join(data_folde
X_test = load_data(glob.glob(os.path.join(data_folder
y_train = load_data(glob.glob(os.path.join(data_folde
y_test = load_data(glob.glob(os.path.join(data_folder
print(X_train.shape, y_train.shape, X_test.shape, y_t
# get hold of the current run
run = Run.get_context()
print('Train a logistic regression model with regular
clf = LogisticRegression(C=1.0/args.reg, solver="libl
clf.fit(X_train, y_train)
print('Predict the test set')
y_hat = clf.predict(X_test)
# calculate accuracy on the prediction
acc = np.average(y_hat == y_test)
print('Accuracy is', acc)
run.log('regularization rate', np.float(args.reg))
run.log('accuracy', np.float(acc))
os.makedirs('outputs', exist_ok=True)
# note file saved in the outputs folder is automatica
joblib.dump(value=clf, filename='outputs/sklearn_mnis
Writing
/mnt/batch/tasks/shared/LS_root/mounts/clusters/mnist-automl-compute/code/Users/adnan/image-classification-mnist-data/sklearn-mnist/train.py


mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
[11]
import shutil
shutil.copy('utils.py', script_folder)
'/mnt/batch/tasks/shared/LS_root/mounts/clusters/mnist-automl-compute/code/Users/adnan/image-classification-mnist-data/sklearn-mnist/utils.py'
Create an estimator
An estimator object is used to submit the run. Azure Machine Learning has pre-configured estimators for common machine learning frameworks, as well as generic Estimator. Create an estimator by specifying
• The name of the estimator object, est
• The directory that contains your scripts. All the files in this directory are uploaded into the cluster nodes for execution.
• The compute target. In this case you will use the AmlCompute you created
• The training script name, train.py
• An environment that contains the libraries needed to run the script
• Parameters required from the training script.
In this tutorial, the target is AmlCompute. All files in the script folder are uploaded into the cluster nodes for execution. The data_folder is set to use the dataset.
First, create the environment that contains: the scikit-learn library, azureml-dataset-runtime required for accessing the dataset, and azureml-defaults which contains the dependencies for logging metrics. The azureml-defaults also contains the dependencies required for deploying the model as a web service later in the part 2 of the tutorial.
Once the environment is defined, register it with the Workspace to re-use it in part 2 of the tutorial.
[12]
from azureml.core.environment import Environment
from azureml.core.conda_dependencies import CondaDepe


mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
Workspace to re-use it in part 2 of the tutorial.
from azureml.core.environment import Environment
from azureml.core.conda_dependencies import CondaDepe
# to install required packages
env = Environment('tutorial-env')
cd = CondaDependencies.create(pip_packages=['azureml-
env.python.conda_dependencies = cd
# Register environment to re-use later
env.register(workspace = ws)
"databricks": {
"eggLibraries": [],
"jarLibraries": [],
"mavenLibraries": [],
"pypiLibraries": [],
"rcranLibraries": []
"docker": {
"arguments": [],
"baseDockerfile": null,
"baseImage":
"mcr.microsoft.com/azureml/intelmpi2018.3-ubuntu16.04:20200821.v1",
"baseImageRegistry": {
"address": null,
"password": null,
"registryIdentity": null,
Then, create the estimator by specifying the training script, compute target and environment.
[13]
from azureml.train.estimator import Estimator
script_params = {
# to mount files referenced by mnist dataset
'--data-folder': mnist_file_dataset.as_named_inpu
'--regularization': 0.5

mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
[14]
from azureml.train.estimator import Estimator
script_params = {
# to mount files referenced by mnist dataset
'--data-folder': mnist_file_dataset.as_named_inpu
'--regularization': 0.5
}
est = Estimator(source_directory=script_folder,
script_params=script_params,
compute_target=compute_target,
environment_definition=env,
entry_script='train.py')
Submit the job to the cluster
Run the experiment by submitting the estimator object. And you can navigate to Azure portal to monitor the run.
run = exp.submit(config=est)
run
arguments have been specified in 'run_config',
'arguments' provided in ScriptRunConfig initialization
will take precedence.
Experiment
Id
sklearn-mnist
sklearn-mnist_1600811417_f74e66e1


mnist-automl-compute · Jupyter kernel idle
Python 3.6.9
job is running, stdout and the files in the ./logs directory are streamed to the run history. You can monitor the run's progress using these logs.
• Post-Processing: The ./outputs directory of the run is copied over to the run history in your workspace so you can access these results.
You can check the progress of a running job in multiple ways. This tutorial uses a Jupyter widget as well as a wait_for_completion method.
Jupyter widget
Watch the progress of the run with a Jupyter widget. Like the run submission, the widget is asynchronous and provides live updates every 10-15 seconds until the job completes.
from azureml.widgets import RunDetails
RunDetails(run).show()
Run sklearn-
Preparing
View
Show log
Step 3/15 : RUN mkdir -p $HOME/.cache
---> Running in dbd7d1c8ee56
Removing intermediate container dbd7d1c8ee56
---> 7029a334763f
Step 4/15 : WORKDIR /
---> Running in 42a01222a578
Removing intermediate container 42a01222a578
---> a7983fe580c8
Step 5/15 : COPY azureml-environment-setup/99brokenproxy /etc/apt/apt.conf.d/
By the way, if you need to cancel a run, you can follow these instructions.


By the way, if you need to cancel a run, you can follow these instructions.
Get log results upon completion
Model training happens in the background. You can use wait_for_completion to block and wait until the model has completed training before running more code.
# specify show_output to True for a verbose log
run.wait_for_completion(show_output=True)
RunId: sklearn-mnist_1600811417_f74e66e1
Web View: https://ml.azure.com/experiments/sklearn-mnist/runs/sklearn-mnist_1600811417_f74e66e1?wsid=/subscriptions/043295ae-bb76-49d8-a303-3ad5c390d687/resourcegroups/automl-resource-group/workspaces/auto-ml-workspace
Streaming azureml-logs/20_image_build_log.txt
2020/09/22 21:50:32 Downloading source code...
2020/09/22 21:50:33 Finished downloading source code
2020/09/22 21:50:33 Creating Docker network: acb_default_network, driver: 'bridge'
Display run results
You now have a model trained on a remote cluster. Retrieve all the metrics logged during the run, including the accuracy of the model:


mnist-automl-compute · Jupyter kernel busy
Python 3.6.9
has completed training before running more code.
# specify show_output to True for a verbose log
run.wait_for_completion(show_output=True)
mkl-2019.4 | 204.1 MB | ###3 | 33%
mkl-2019.4 | 204.1 MB | #### | 38%
mkl-2019.4 | 204.1 MB | ####3 | 43%
mkl-2019.4 | 204.1 MB | ####7 | 48%
mkl-2019.4 | 204.1 MB | #####2 | 53%
mkl-2019.4 | 204.1 MB | #####7 | 58%
mkl-2019.4 | 204.1 MB | ######2 | 63%
mkl-2019.4 | 204.1 MB | ######7 | 68%
mkl-2019.4 | 204.1 MB | #######2 | 73%
mkl-2019.4 | 204.1 MB | #######7 | 77%
mkl-2019.4 | 204.1 MB | ########2 | 82%
mkl-2019.4 | 204.1 MB | ########7 | 87%
mkl-2019.4 | 204.1 MB | #########1 | 92%
mkl-2019.4 | 204.1 MB | #########6 | 97%
Display run results
You now have a model trained on a remote cluster. Retrieve all the metrics logged during the run, including the accuracy of the model:
print(run.get_metrics())
In the next tutorial you will explore this model in more detail.
Register model
The last step in the training script wrote the file outputs/sklearn_mnist_model.pkl in a directory named

Get log results upon completion

Model training happens in the background. You can use `wait_for_completion` to block and wait until the model has completed training before running more code.

[14]

```
# specify show_output to True for a verbose log
run.wait_for_completion(show_output=True)
```

```
2020/09/23 02:00:41 The following dependencies were found:
2020/09/23 02:00:41
- image:
    registry: 934b9f82f47a4ac8bede2c0ab17f6a6a.azurecr.io
    repository: azureml/azureml_0a6838d76e7468052f3a857ca80cfaa3
    tag: latest
    digest: sha256:09cdccf921b046044cb49ecf0464d04435ff952cadc1c054a5ae28b913433103
  runtime-dependency:
    registry: mcr.microsoft.com
    repository: azureml/intelmpi2018.3-ubuntu16.04
    tag: 20200821.v1
    digest: sha256:8cee6f674276dddb23068d2710da7f7f95b119412cc482675ac79ba45a4acf99
  git: {}

Run ID: cal was successful after 5m2s
```

Submit the job to the cluster

Run the experiment by submitting the estimator object. And you can navigate to Azure portal to monitor the run.

```
run = exp.submit(config=est)
run
```

```
WARNING - If 'script' has been provided here and a script file name has been specified in 'run_config', 'script' provided in ScriptRunConfig initialization will take precedence.
WARNING - If 'arguments' has been provided here and arguments have been specified in 'run_config', 'arguments' provided in ScriptRunConfig initialization will take precedence.
```

Experiment	Id	Type	Status	Details Page	Docs Page
sklearn-mnist	sklearn-mnist_1600826131_a0367d7d	azureml.scriptrun	Starting	Link to Azure Machine Learning studio	Link to Documentation

mnist-automl-compute · Jupyter kernel busy Python 3.6.9

```
  runtime-dependency:
    registry: mcr.microsoft.com
    repository: azureml/intelmpi2018.3-ubuntu16.04
    tag: 20200821.v1
    digest:
sha256:8cee6f674276dddb23068d2710da7f7f95b119412cc482675
ac79ba45a4acf99
  git: {}


Run ID: ca1 was successful after 5m35s
```

Display run results

You now have a model trained on a remote cluster. Retrieve all the metrics logged during the run, including the accuracy of the model:

[-]
```python
print(run.get_metrics())
```

In the next tutorial you will explore this model in more detail.

Register model

The last step in the training script wrote the file `outputs/sklearn_mnist_model.pkl` in a directory named `outputs` in the VM of the cluster where the job is executed. `outputs` is a special directory in that all content in this directory is automatically uploaded to your workspace. This content appears in the run record in the experiment under your workspace. Hence, the model file is now also available in your workspace.

You can see files associated with that run.

[ ]
```python
print(run.get_file_names())
```

Display run results

You now have a model trained on a remote cluster. Retrieve all the metrics logged during the run, including the accuracy of the model:

[18]
```python
print(run.get_metrics())
```

```
{'regularization rate': 0.5, 'accuracy': 0.9193}
```

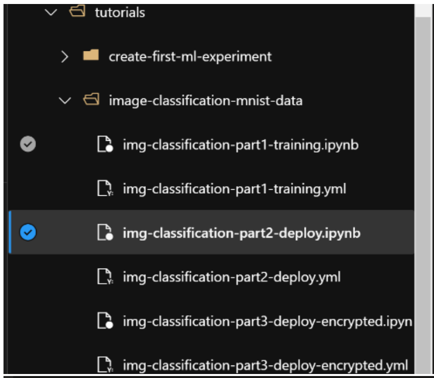
Register model

The last step in the training script wrote the file `outputs/sklearn_mnist_model.pkl` in a directory named `outputs` in the VM of the cluster where the job is executed. `outputs` is a special directory in that all content in this directory is automatically uploaded to your workspace. This content appears in the run record in the experiment under your workspace. Hence, the model file is now also available in your workspace.

You can see files associated with that run.

```python
print(run.get_file_names())
```

```
['azureml-logs/20_image_build_log.txt', 'azureml-logs/55_azureml-execution-
tvmps_4ac17b36679f8faa19f3b03f634710f765c4d13ba57a6c3e96b075965b4af794_d.txt', 'azureml-logs/65_job_prep-
tvmps_4ac17b36679f8faa19f3b03f634710f765c4d13ba57a6c3e96b075965b4af794_d.txt', 'azureml-
logs/70_driver_log.txt', 'azureml-logs/75_job_post-
tvmps_4ac17b36679f8faa19f3b03f634710f765c4d13ba57a6c3e96b075965b4af794_d.txt', 'azureml-
logs/process_info.json', 'azureml-logs/process_status.json', 'logs/azureml/109_azureml.log',
'logs/azureml/dataprep/backgroundProcess.log', 'logs/azureml/dataprep/backgroundProcess_Telemetry.log',
'logs/azureml/dataprep/engine_spans_l_8e589ded-fe2b-474a-b7b7-9d60265f5567.jsonl',
'logs/azureml/dataprep/engine_spans_l_964ab3d8-9263-416f-b59d-fc49bd448e5d.jsonl',
'logs/azureml/dataprep/python_span_l_8e589ded-fe2b-474a-b7b7-9d60265f5567.jsonl',
'logs/azureml/dataprep/python_span_l_964ab3d8-9263-416f-b59d-fc49bd448e5d.jsonl',
'logs/azureml/job_prep_azureml.log', 'logs/azureml/job_release_azureml.log',
'outputs/sklearn_mnist_model.pkl']
```




```
%%writefile score.py
import json
import numpy as np
import os
import pickle
import joblib

def init():
    global model
    # AZUREML_MODEL_DIR is an environment variable created during deployment.
    # It is the path to the model folder (./azureml-models/$MODEL_NAME/$VERSION)
    # For multiple models, it points to the folder containing all deployed models (./azureml-models)
    model_path = os.path.join(os.getenv('AZUREML_MODEL_DIR'), 'sklearn_mnist_model.pkl')
    model = joblib.load(model_path)

def run(raw_data):
    data = np.array(json.loads(raw_data)['data'])
    # make prediction
    y_hat = model.predict(data)
    # you can return any data type as long as it is JSON-serializable
    return y_hat.tolist()
```

```
ws = Workspace.from_config()
model = Model(ws, 'sklearn_mnist')


myenv = Environment.get(workspace=ws, name="tutorial-env", version="1")
inference_config = InferenceConfig(entry_script="score.py", environment=myenv)

service_name = 'sklearn-mnist-svc-' + str(uuid.uuid4())[:4]
service = Model.deploy(workspace=ws,
                       name=service_name,
                       models=[model],
                       inference_config=inference_config,
                       deployment_config=aciconfig)

service.wait_for_deployment(show_output=True)
```

```
Running...........................
Succeeded
ACI service creation operation finished, operation "Succeeded"
CPU times: user 279 ms, sys: 56.5 ms, total: 336 ms
Wall time: 2min 36s
```

```
print(service.scoring_uri)
```

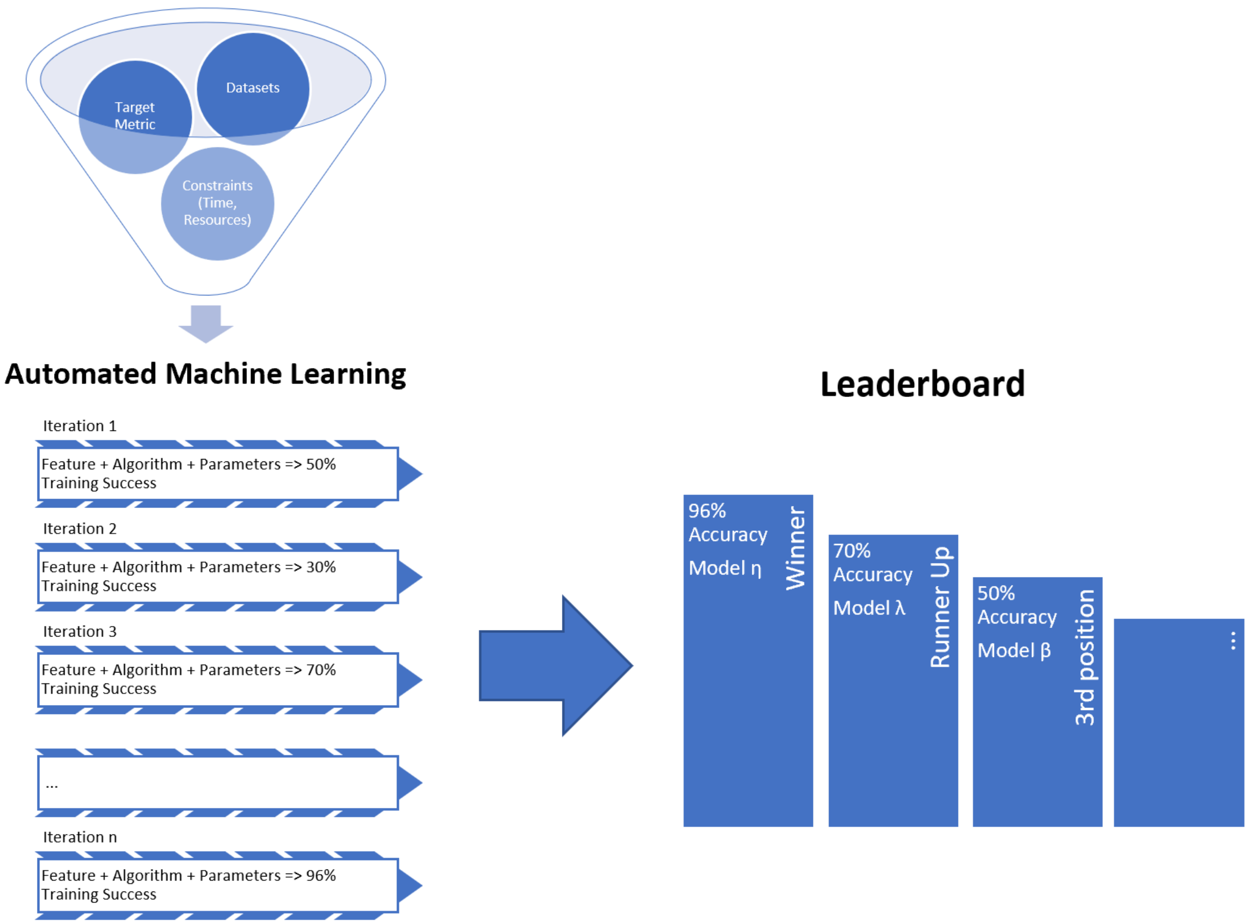
```
http://47f2f8ea-0a5b-4580-bdfe-6de578bcb5f3.eastus.azurecontainer.io/score
```

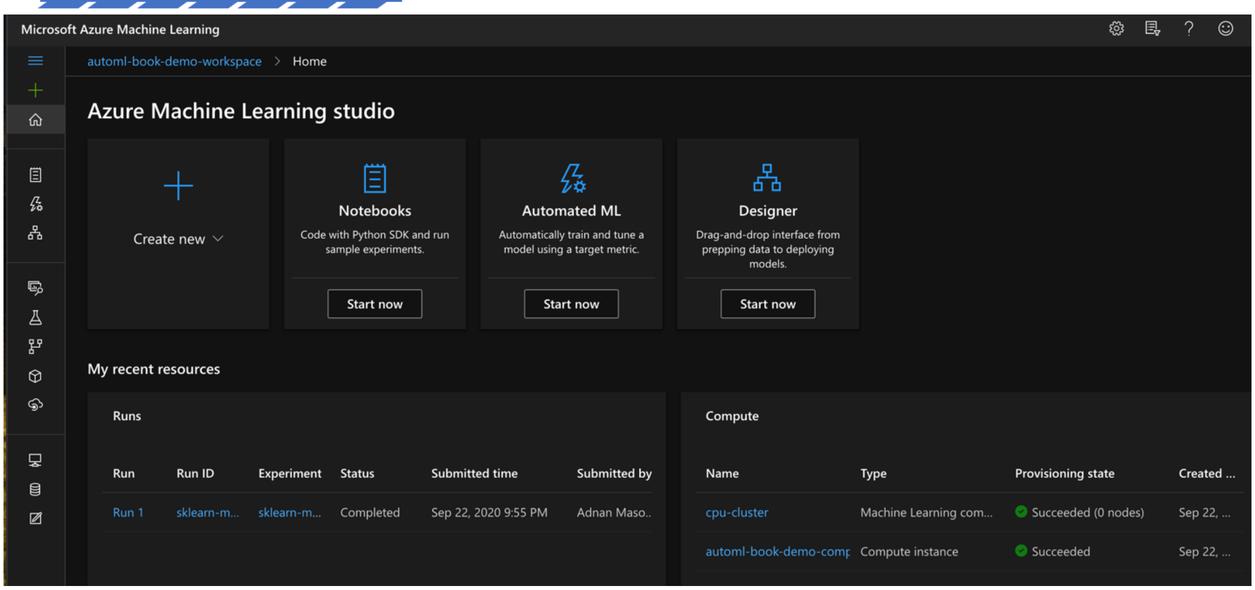
```
[ ]
import json
test = json.dumps({"data": X_test.tolist()})
test = bytes(test, encoding='utf8')
y_hat = service.run(input_data=test)
```

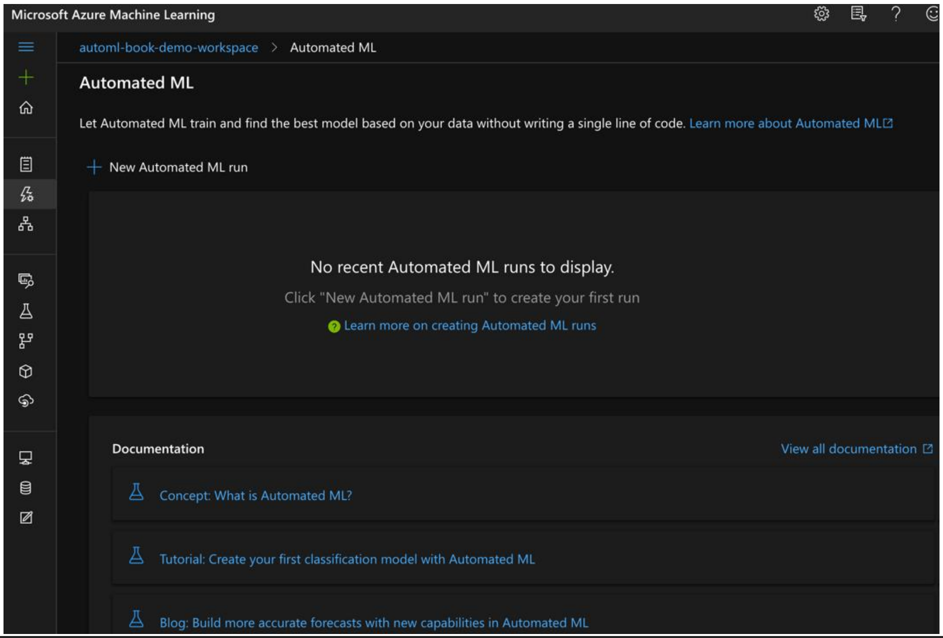
```
from sklearn.metrics import confusion_matrix

conf_mx = confusion_matrix(y_test, y_hat)
print(conf_mx)
print('Overall accuracy:', np.average(y_hat == y_test))
```

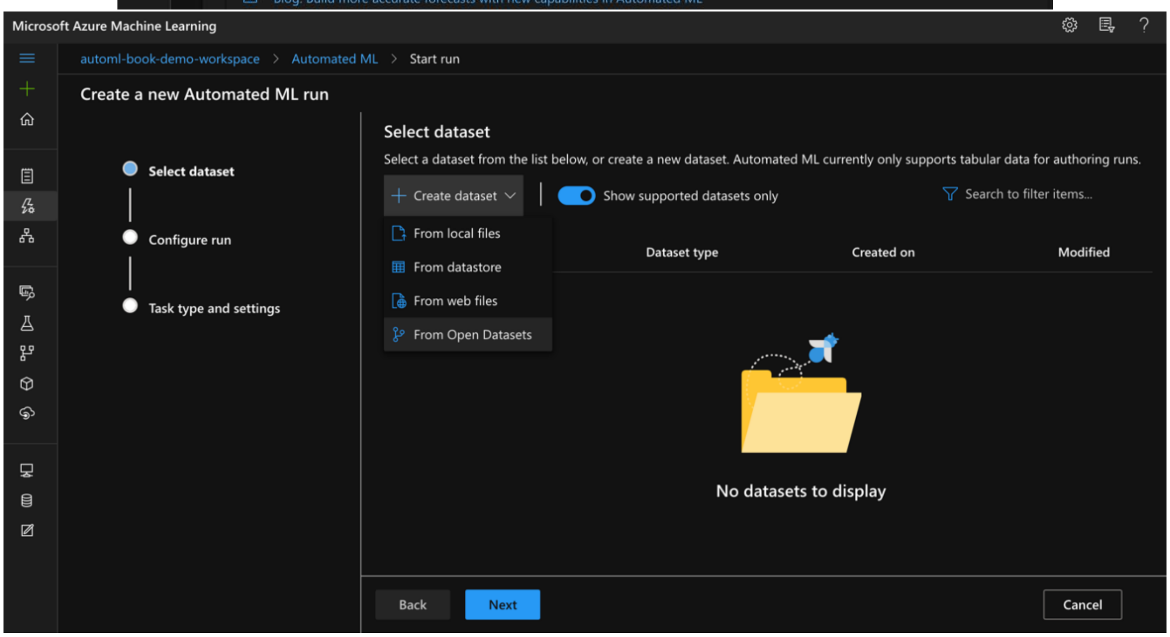
```
[[ 960    0    2    2    1    4    6    3    1    1]
 [   0 1113    3    1    0    1    5    1   11    0]
 [   9    8  919   20    9    5   10   12   37    3]
 [   4    0   17  918    2   24    4   11   21    9]
 [   1    4    4    3  913    0   10    3    5   39]
 [  10    2    0   42   11  768   17    7   28    7]
 [   9    3    7    2    6   20  907    1    3    0]
 [   2    9   22    5    8    1    1  948    5   27]
 [  10   15    5   21   15   26    7   11  852   12]
 [   7    8    2   14   32   13    0   26   12  895]]
Overall accuracy: 0.9193
```

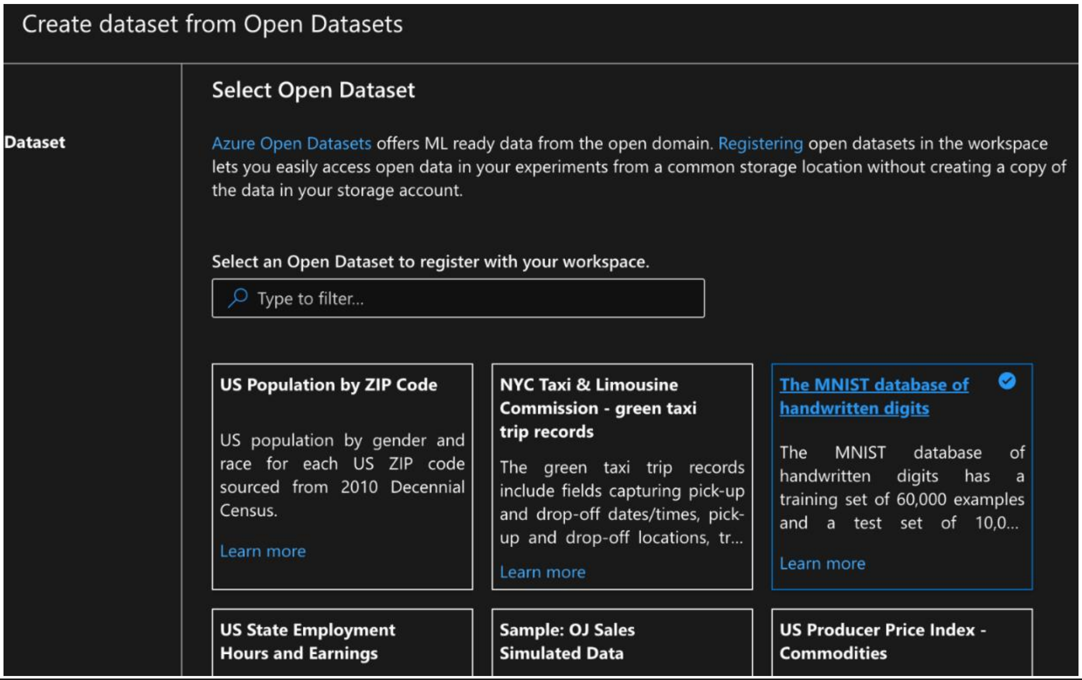
Chapter 5: Automated Machine Learning with Microsoft Azure

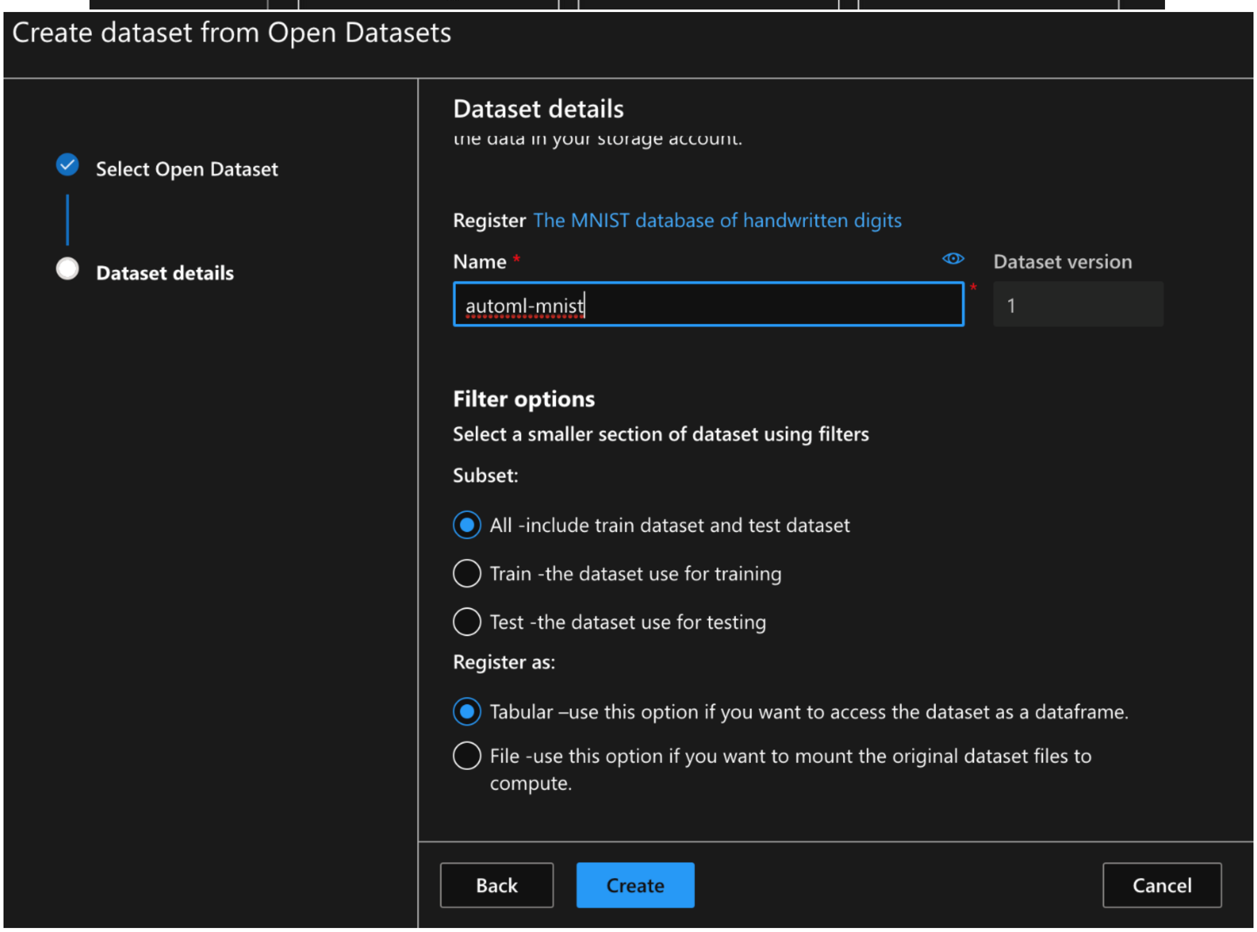


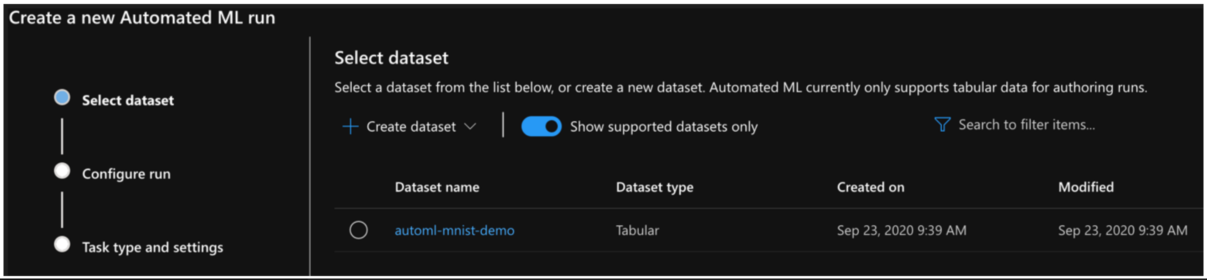


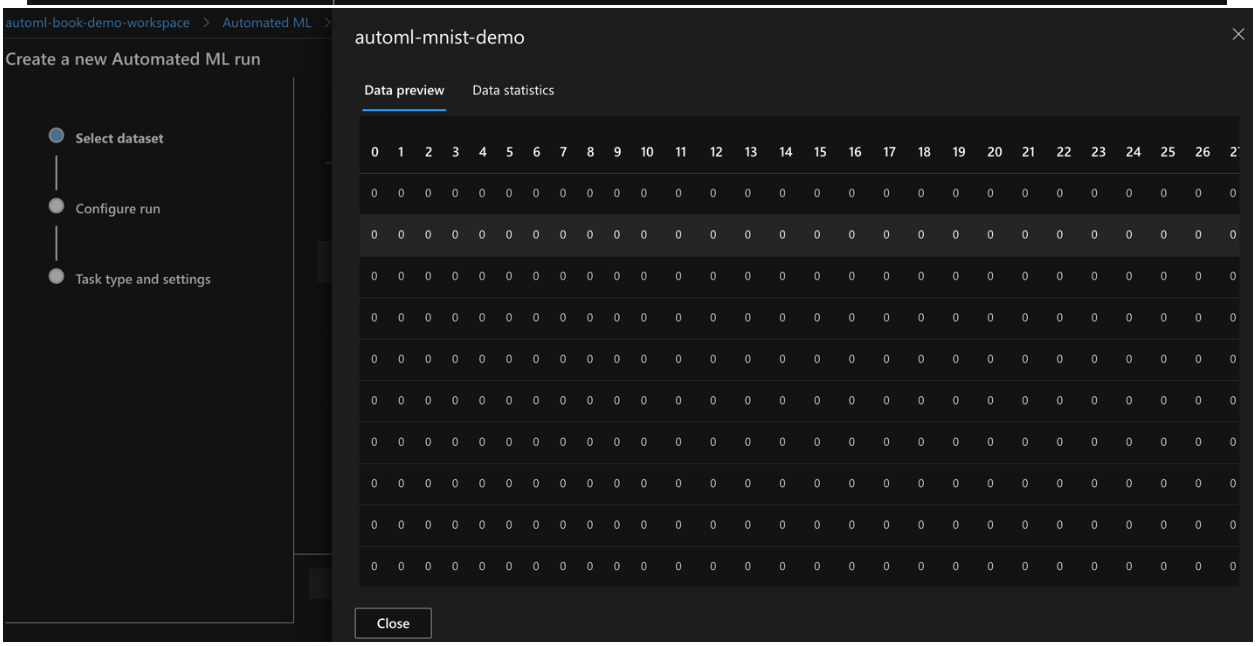

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML
Automated ML
Let Automated ML train and find the best model based on your data without writing a single line of code. Learn more about Automated ML
New Automated ML run
No recent Automated ML runs to display.
Click "New Automated 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

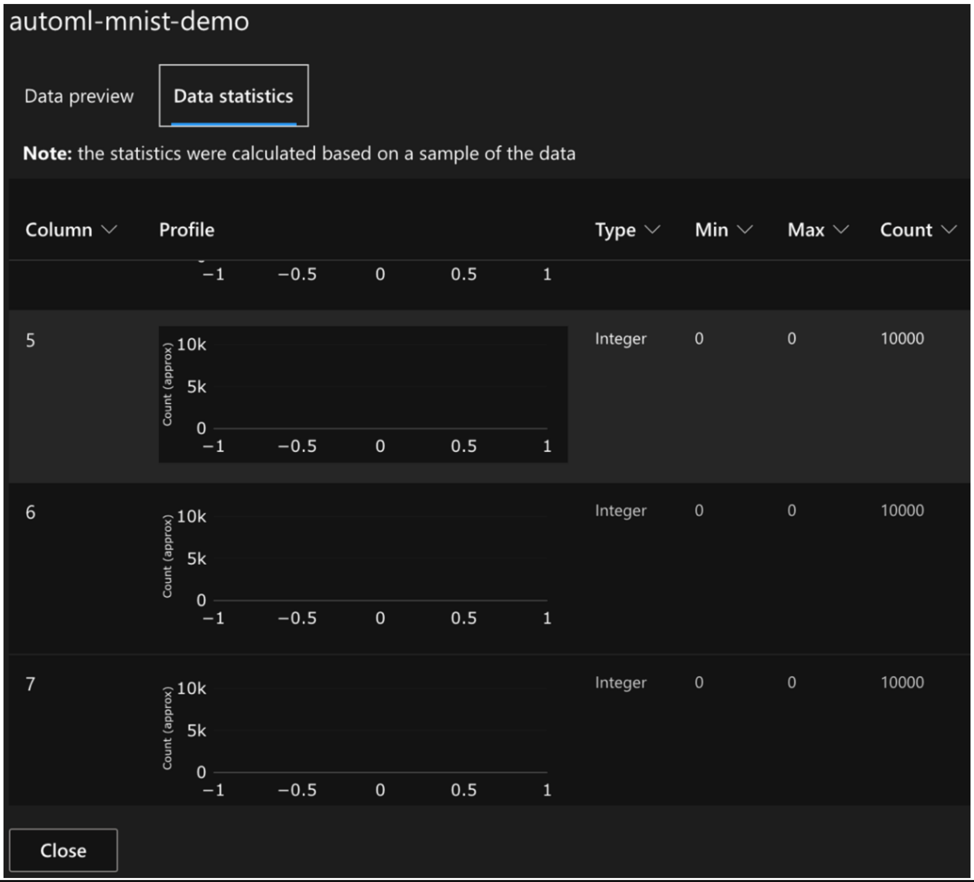

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > Start run
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Select dataset
Select a dataset from the list below, or create a new dataset. Automated ML currently only supports tabular data for authoring runs.
Create dataset
Show supported datasets only
Search to filter items...
From local files
From datastore
From web files
From Open Datasets
Dataset type
Created on
Modified
No datasets to display
Back
Next
Cancel

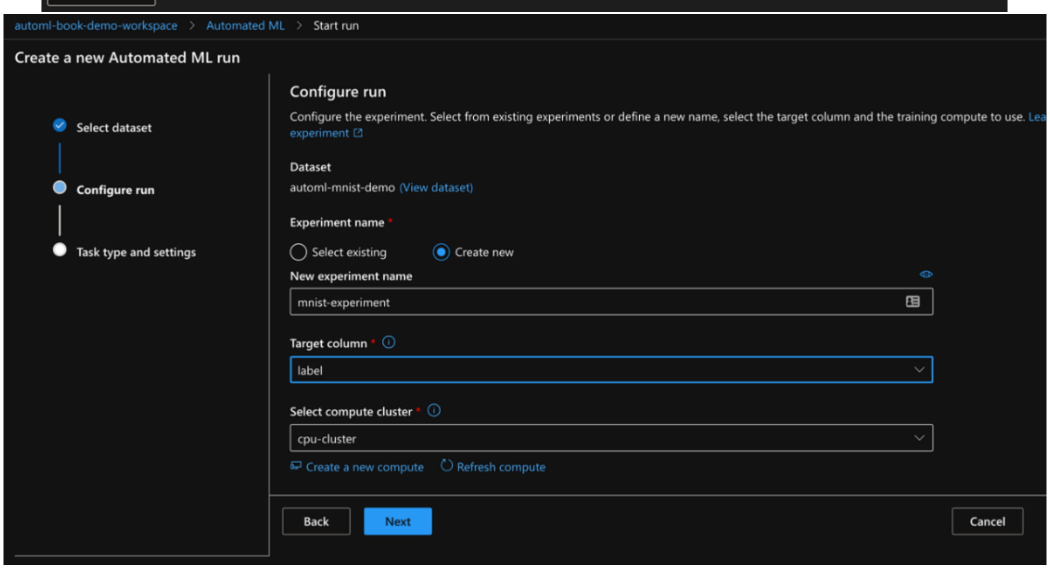

Create dataset from Open Datasets
Dataset
Select Open Dataset
Azure Open Datasets offers ML ready data from the open domain. Registering open datasets in the workspace lets you easily access open data in your experiments from a common storage location without creating a copy of the data in your storage account.
Select an Open Dataset to register with your workspace.
Type to filter...
US Population by ZIP Code
US population by gender and race for each US ZIP code sourced from 2010 Decennial Census.
Learn more
NYC Taxi & Limousine Commission - green taxi trip records
The green taxi trip records include fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, tr...
Learn more
The MNIST database of handwritten digits
The MNIST database of handwritten digits has a training set of 60,000 examples and a test set of 10,0...
Learn more
US State Employment Hours and Earnings
Sample: OJ Sales Simulated Data
US Producer Price Index - Commodities

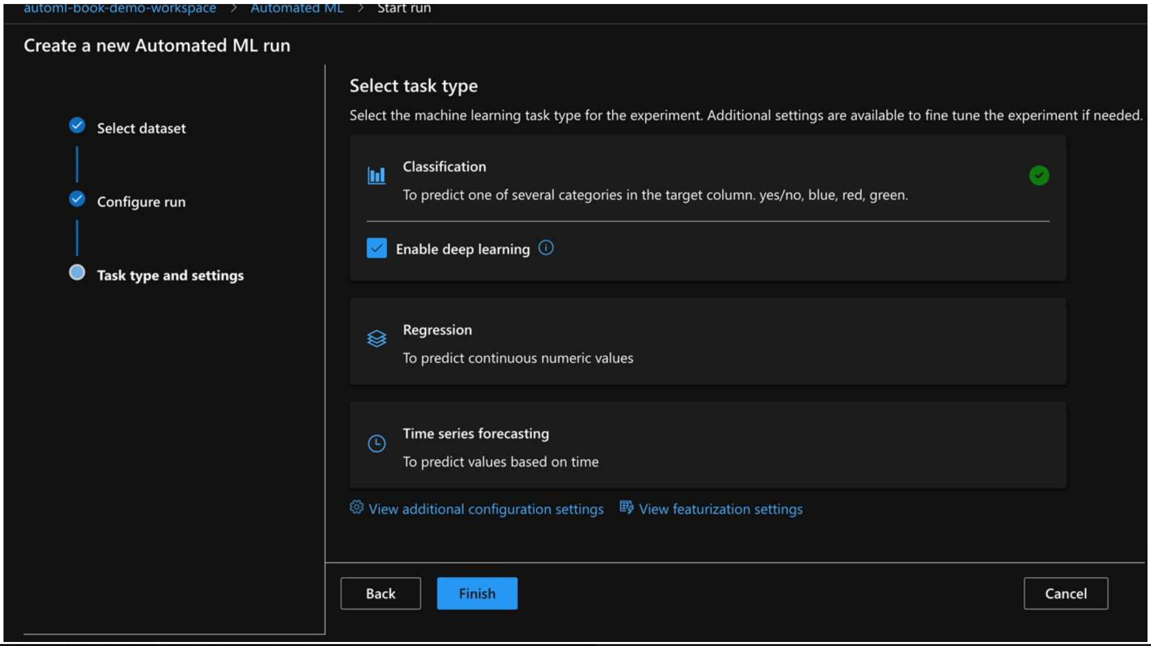

Create dataset from Open Datasets
Select Open Dataset
Dataset details
Dataset details
Register The MNIST database of handwritten digits
Name *
Dataset version
automl-mnist
1
Filter options
Select a smaller section of dataset using filters
Subset:
All -include train dataset and test dataset
Train -the dataset use for training
Test -the dataset use for testing
Register as:
Tabular –use this option if you want to access the dataset as a dataframe.
File -use this option if you want to mount the original dataset files to compute.
Back
Create
Cancel

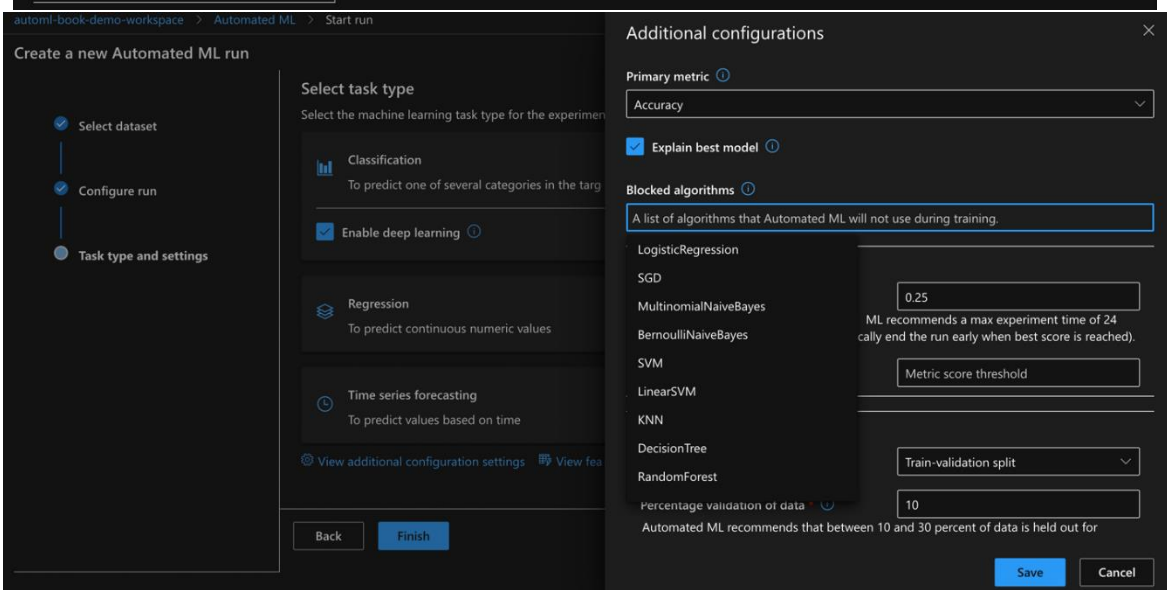

Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Select dataset
Select a dataset from the list below, or create a new dataset. Automated ML currently only supports tabular data for authoring runs.
Create dataset
Show supported datasets only
Search to filter items...
Dataset name
Dataset type
Created on
Modified
automl-mnist-demo
Tabular
Sep 23, 2020 9:39 AM
Sep 23, 2020 9:39 AM

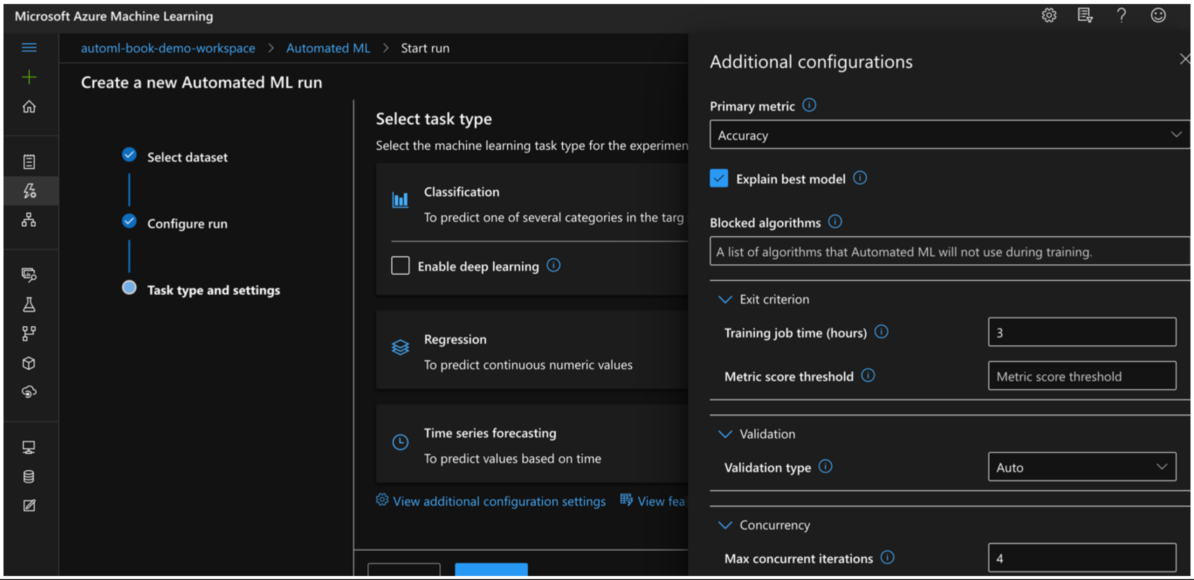

automl-book-demo-workspace
Automated ML
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
automl-mnist-demo
Data preview
Data statistics
Close

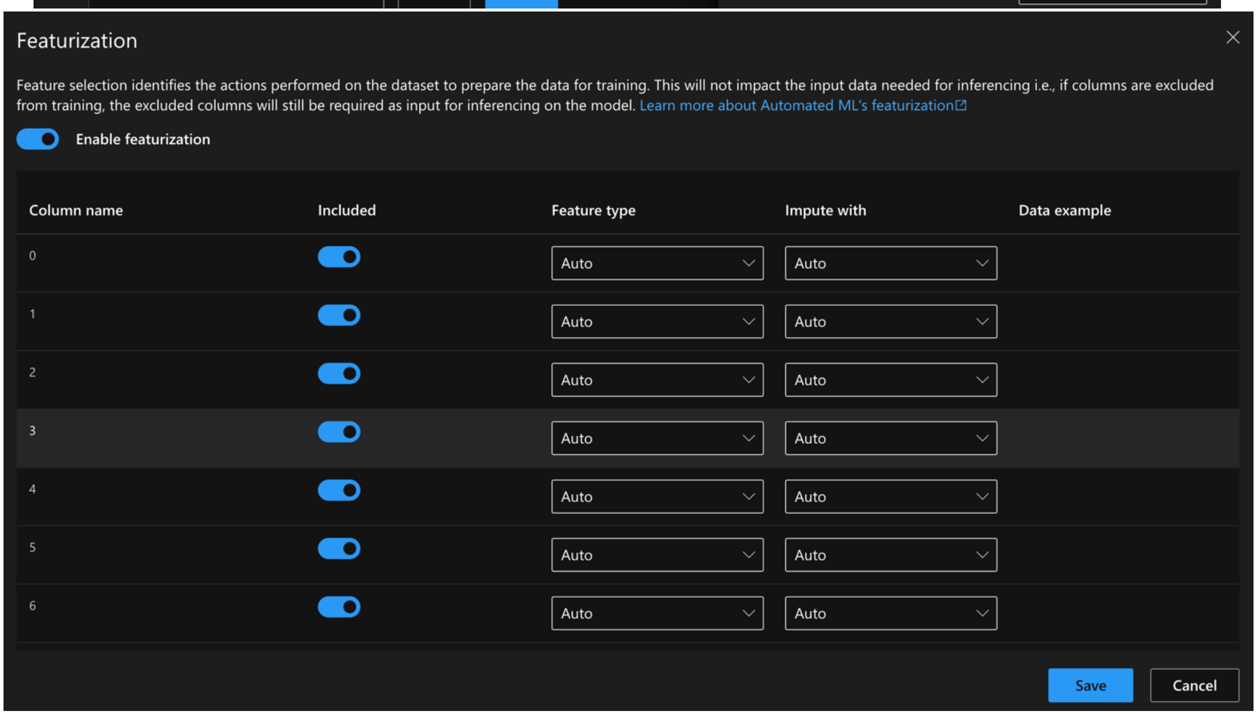

automl-mnist-demo
Data preview
Data statistics
Note: the statistics were calculated based on a sample of the data
Column
Profile
Type
Min
Max
Count
5
Integer
0
0
10000
6
Integer
0
0
10000
7
Integer
0
0
10000
Count (approx)
10k
5k
0
−1
−0.5
0
0.5
1
Close

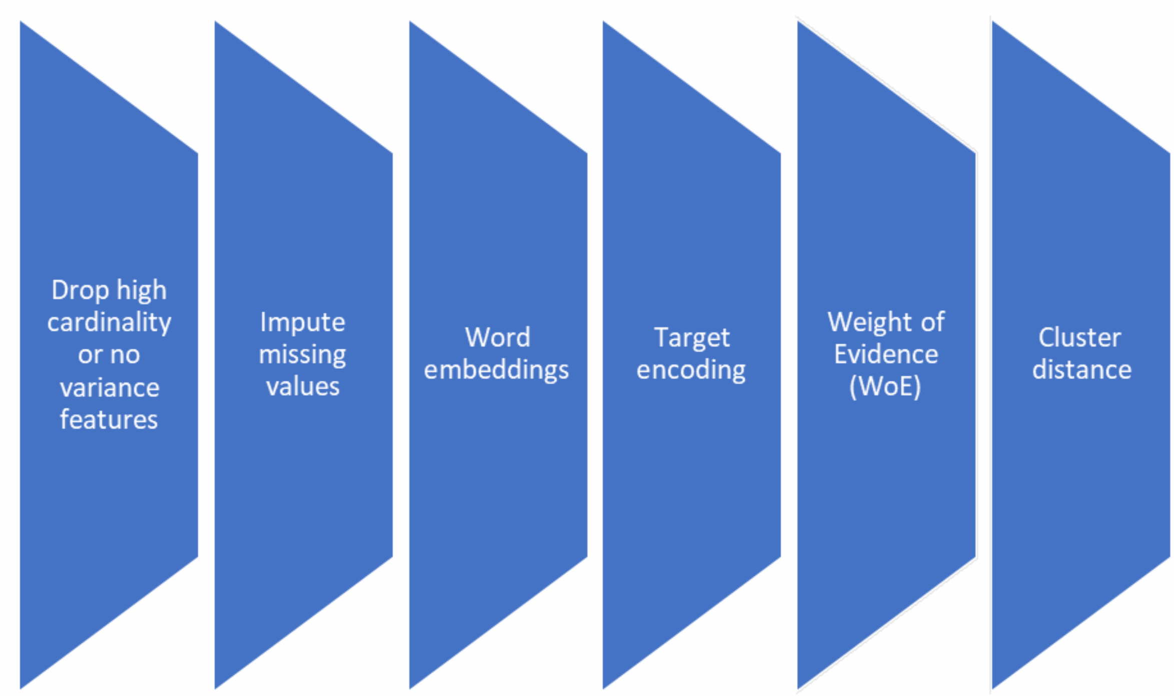

automl-book-demo-workspace > Automated ML > Start run
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Configure run
Configure the experiment. Select from existing experiments or define a new name, select the target column and the training compute to use.
Dataset
automl-mnist-demo (View dataset)
Experiment name
Select existing
Create new
New experiment name
mnist-experiment
Target column
label
Select compute cluster
cpu-cluster
Create a new compute
Refresh compute
Back
Next
Cancel

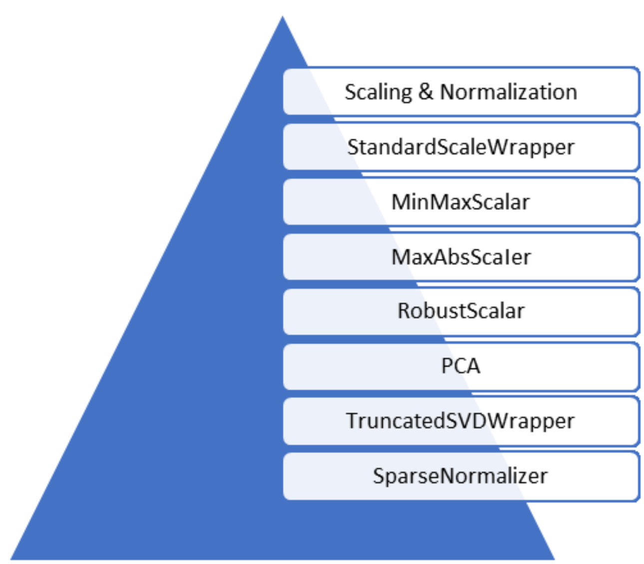

automl-book-demo-workspace > Automated ML > Start run
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Select task type
Select the machine learning task type for the experiment. Additional settings are available to fine tune the experiment if needed.
Classification
To predict one of several categories in the target column. yes/no, blue, red, green.
Enable deep learning
Regression
To predict continuous numeric values
Time series forecasting
To predict values based on time
View additional configuration settings
View featurization settings
Back
Finish
Cancel


automl-book-demo-workspace > Automated ML > Start run
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Select task type
Classification
Enable deep learning
Regression
To predict continuous numeric values
Time series forecasting
To predict values based on time
View additional configuration settings
Back
Finish
Additional configurations
Primary metric
Accuracy
Explain best model
Blocked algorithms
A list of algorithms that Automated ML will not use during training.
LogisticRegression
SGD
MultinomialNaiveBayes
BernoulliNaiveBayes
SVM
LinearSVM
KNN
DecisionTree
RandomForest
0.25
ML recommends a max experiment time of 24
cally end the run early when best score is reached).
Metric score threshold
Train-validation split
Percentage validation of data
10
Automated ML recommends that between 10 and 30 percent of data is held out for
Save
Cancel

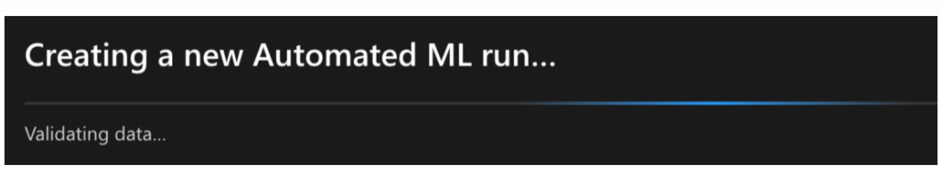

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > Start run
Create a new Automated ML run
Select dataset
Configure run
Task type and settings
Select task type
Select the machine learning task type for the experimen
Classification
To predict one of several categories in the targ
Enable deep learning
Regression
To predict continuous numeric values
Time series forecasting
To predict values based on time
View additional configuration settings
View fea
Additional configurations
Primary metric
Accuracy
Explain best model
Blocked algorithms
A list of algorithms that Automated ML will not use during training.
Exit criterion
Training job time (hours)
3
Metric score threshold
Metric score threshold
Validation
Validation type
Auto
Concurrency
Max concurrent iterations
4

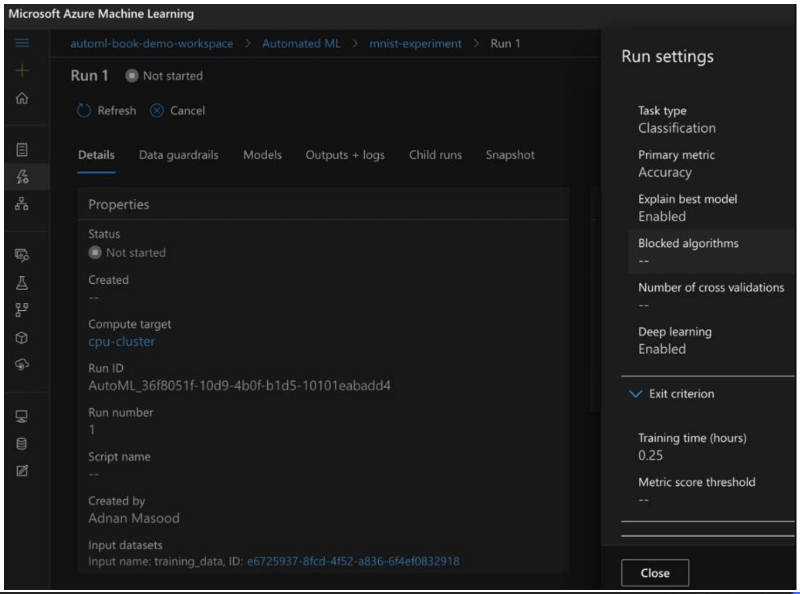
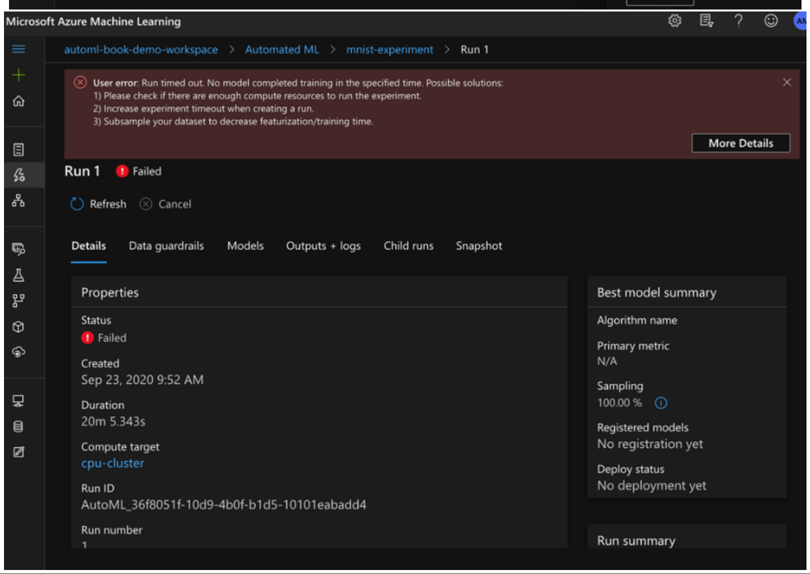
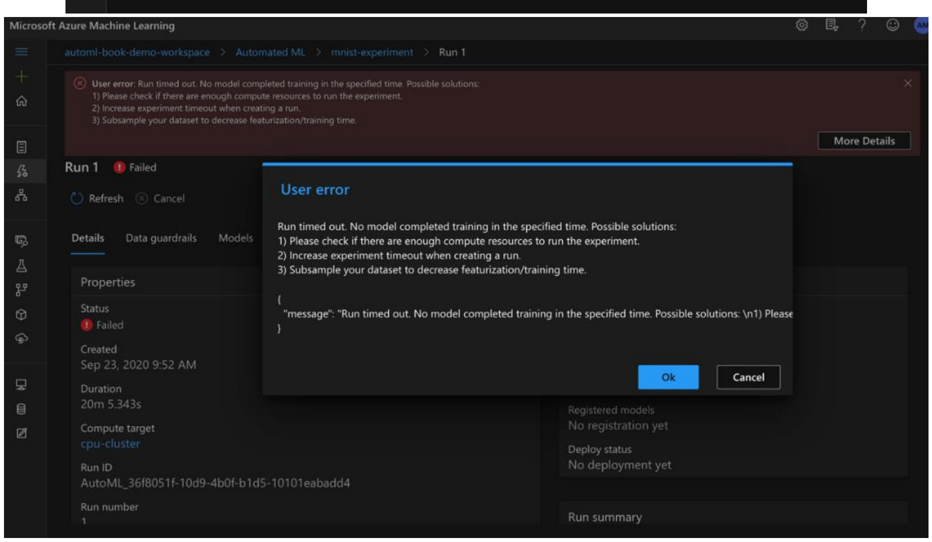

Featurization
Feature selection identifies the actions performed on the dataset to prepare the data for training. This will not impact the input data needed for inferencing i.e., if columns are excluded from training, the excluded columns will still be required as input for inferencing on the model. Learn more about Automated ML's featurization
Enable featurization
Column name
Included
Feature type
Impute with
Data example
0
1
2
3
4
5
6
Auto
Save
Cancel

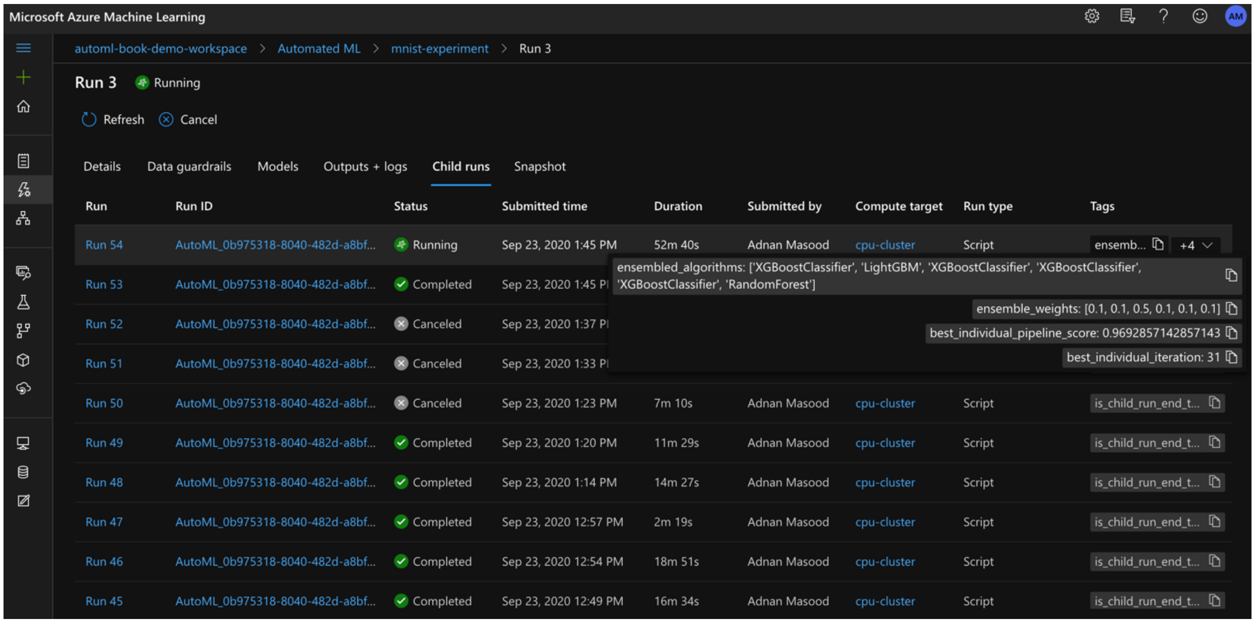

Drop high cardinality or no variance features
Impute missing values
Word embeddings
Target encoding
Weight of Evidence (WoE)
Cluster distance

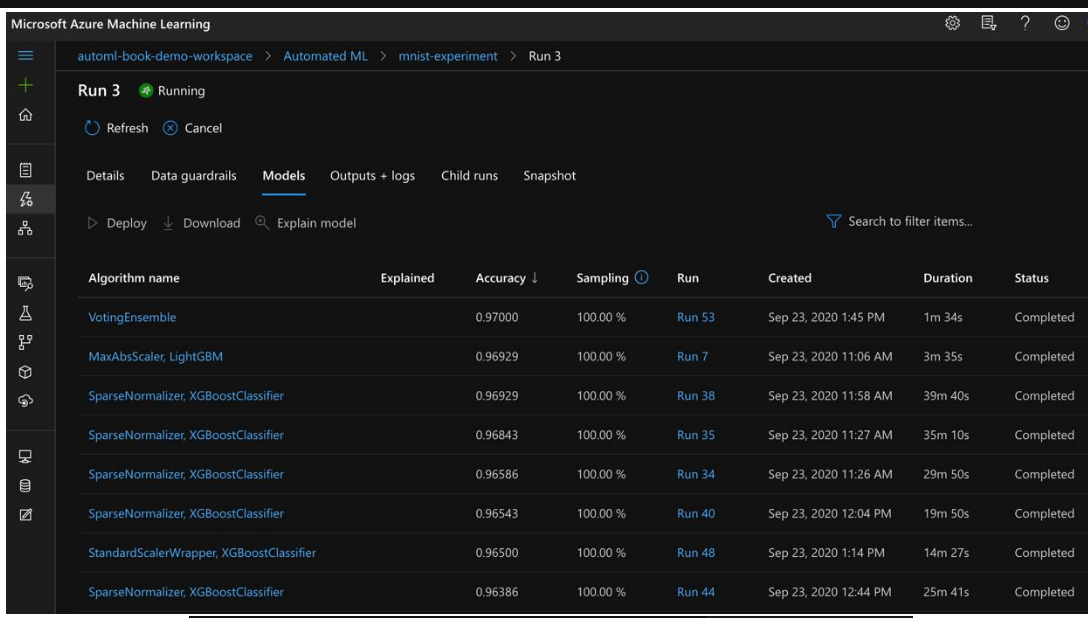

Scaling & Normalization
StandardScaleWrapper
MinMaxScalar
MaxAbsScaler
RobustScalar
PCA
TruncatedSVDWrapper
SparseNormalizer

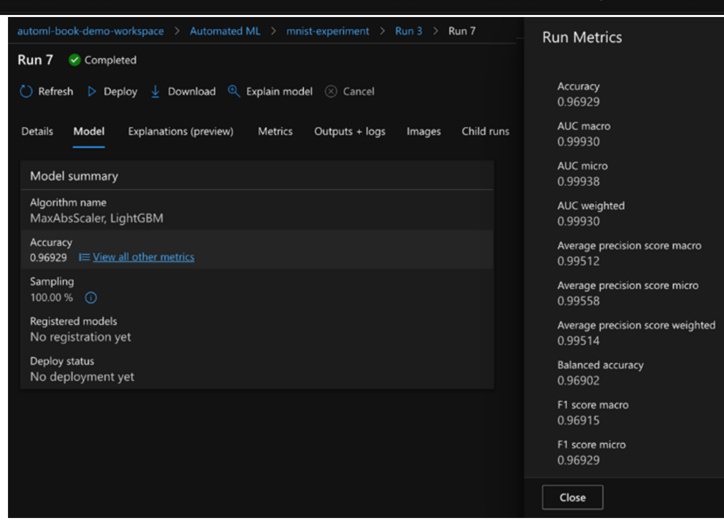

Missing feature values imputation
High cardinality feature handling
Validation split handling
Class balancing detection
Memory issues detection
Frequency detection

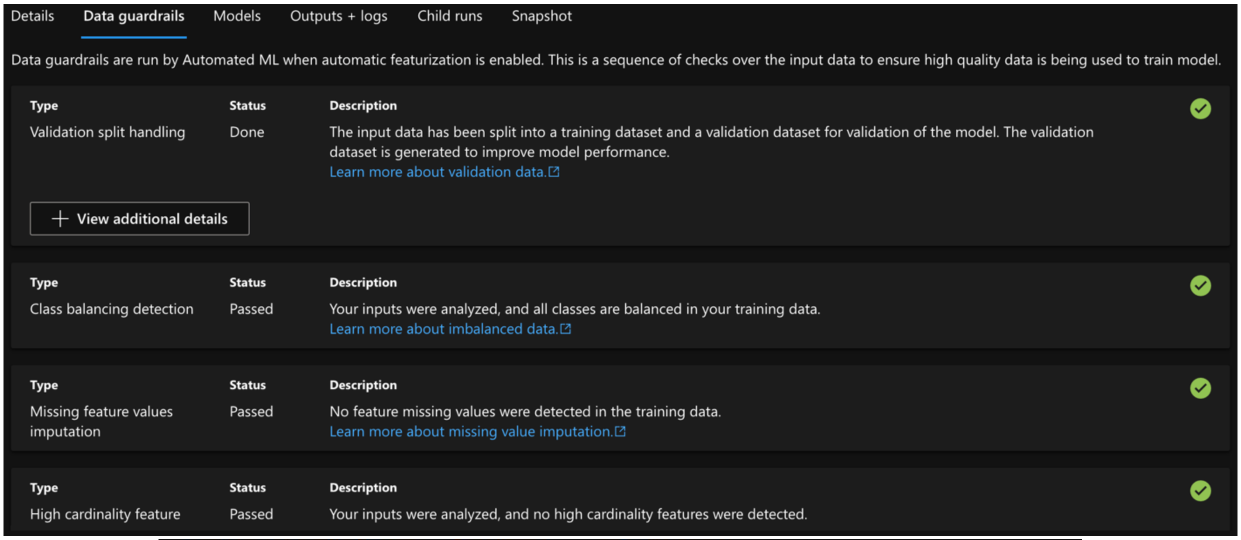

Creating a new Automated ML run...
Validating data...

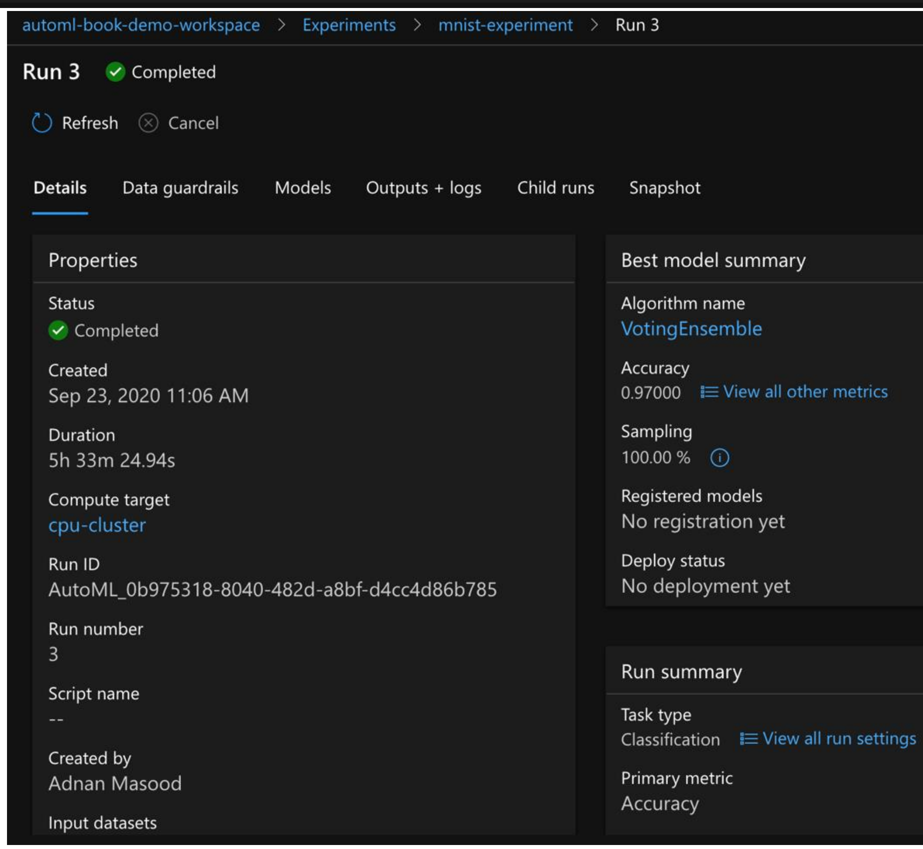

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > mnist-experiment > Run 1
Run 1 Not started
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Properties
Status
Not started
Created
--
Compute target
cpu-cluster
Run ID
AutoML_36f8051f-10d9-4b0f-b1d5-10101eabadd4
Run number
1
Script name
--
Created by
Adnan Masood
Input datasets
Input name: training_data, ID: e6725937-8fcd-4f52-a836-6f4ef0832918
Run settings
Task type
Classification
Primary metric
Accuracy
Explain best model
Enabled
Blocked algorithms
--
Number of cross validations
--
Deep learning
Enabled
Exit criterion
Training time (hours)
0.25
Metric score threshold
--
Close

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > mnist-experiment > Run 1
User error: Run timed out. No model completed training in the specified time. Possible solutions:
1) Please check if there are enough compute resources to run the experiment.
2) Increase experiment timeout when creating a run.
3) Subsample your dataset to decrease featurization/training time.
More Details
Run 1 Failed
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Properties
Status
Failed
Created
Sep 23, 2020 9:52 AM
Duration
20m 5.343s
Compute target
cpu-cluster
Run ID
AutoML_36f8051f-10d9-4b0f-b1d5-10101eabadd4
Run number
1
Best model summary
Algorithm name
Primary metric
N/A
Sampling
100.00 %
Registered models
No registration yet
Deploy status
No deployment yet
Run summary

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > mnist-experiment > Run 1
User error: Run timed out. No model completed training in the specified time. Possible solutions:
1) Please check if there are enough compute resources to run the experiment.
2) Increase experiment timeout when creating a run.
3) Subsample your dataset to decrease featurization/training time.
More Details
Run 1 Failed
Refresh Cancel
Details Data guardrails Models
Properties
Status
Failed
Created
Sep 23, 2020 9:52 AM
Duration
20m 5.343s
Compute target
cpu-cluster
Run ID
AutoML_36f8051f-10d9-4b0f-b1d5-10101eabadd4
Run number
1
User error
Run timed out. No model completed training in the specified time. Possible solutions:
1) Please check if there are enough compute resources to run the experiment.
2) Increase experiment timeout when creating a run.
3) Subsample your dataset to decrease featurization/training time.
{
"message": "Run timed out. No model completed training in the specified time. Possible solutions: \n1) Please
}
Ok
Cancel
Registered models
No registration yet
Deploy status
No deployment yet
Run summary

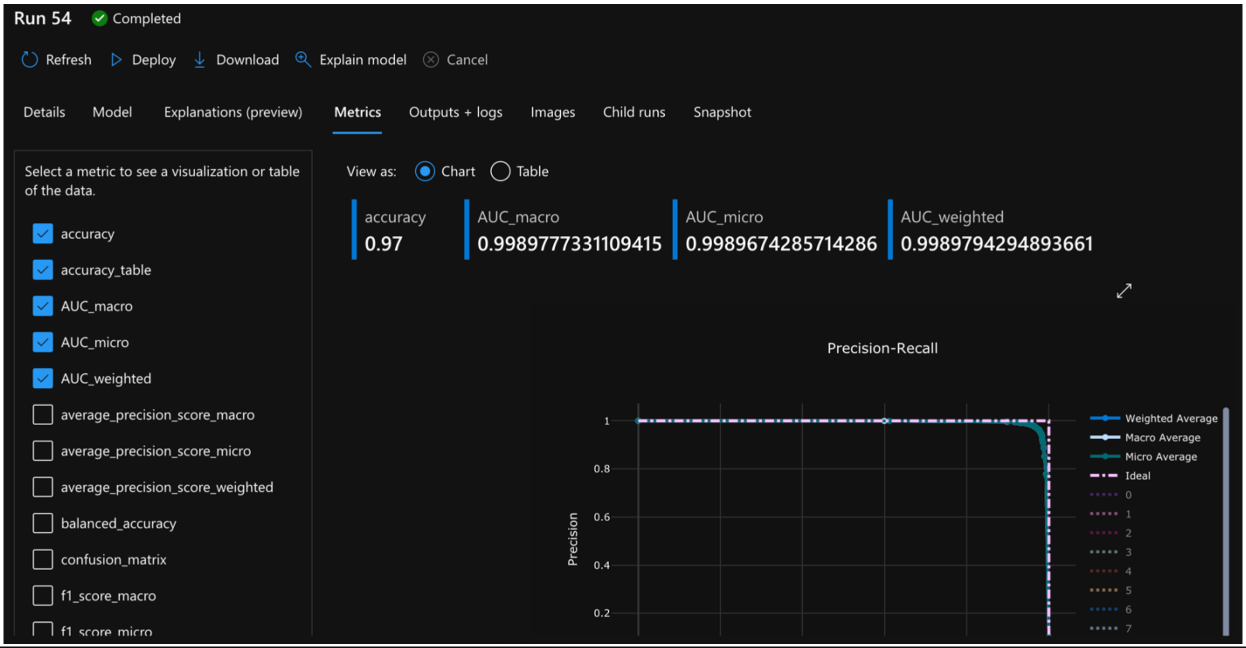

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > mnist-experiment > Run 3
Run 3 Running
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Run Run ID Status Submitted time Duration Submitted by Compute target Run type Tags
Run 54 AutoML_0b975318-8040-482d-a8bf... Running Sep 23, 2020 1:45 PM 52m 40s Adnan Masood cpu-cluster Script ensemb... +4
ensembled_algorithms: ['XGBoostClassifier', 'LightGBM', 'XGBoostClassifier', 'XGBoostClassifier', 'XGBoostClassifier', 'RandomForest']
ensemble_weights: [0.1, 0.1, 0.5, 0.1, 0.1, 0.1]
best_individual_pipeline_score: 0.9692857142857143
best_individual_iteration: 31
Run 53 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 1:45 PM
Run 52 AutoML_0b975318-8040-482d-a8bf... Canceled Sep 23, 2020 1:37 PM
Run 51 AutoML_0b975318-8040-482d-a8bf... Canceled Sep 23, 2020 1:33 PM
Run 50 AutoML_0b975318-8040-482d-a8bf... Canceled Sep 23, 2020 1:23 PM 7m 10s Adnan Masood cpu-cluster Script is_child_run_end_t...
Run 49 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 1:20 PM 11m 29s Adnan Masood cpu-cluster Script is_child_run_end_t...
Run 48 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 1:14 PM 14m 27s Adnan Masood cpu-cluster Script is_child_run_end_t...
Run 47 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 12:57 PM 2m 19s Adnan Masood cpu-cluster Script is_child_run_end_t...
Run 46 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 12:54 PM 18m 51s Adnan Masood cpu-cluster Script is_child_run_end_t...
Run 45 AutoML_0b975318-8040-482d-a8bf... Completed Sep 23, 2020 12:49 PM 16m 34s Adnan Masood cpu-cluster Script is_child_run_end_t...

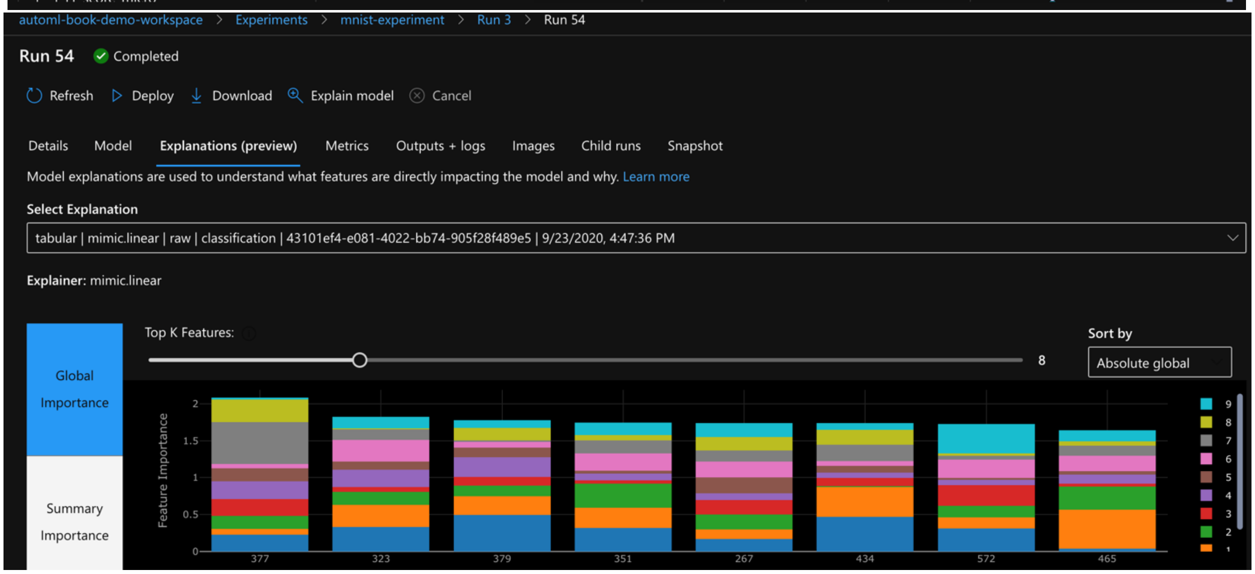

Microsoft Azure Machine Learning
automl-book-demo-workspace > Automated ML > mnist-experiment > Run 3
Run 3 Running
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Deploy Download Explain model
Search to filter items...
Algorithm name Explained Accuracy Sampling Run Created Duration Status
VotingEnsemble 0.97000 100.00 % Run 53 Sep 23, 2020 1:45 PM 1m 34s Completed
MaxAbsScaler, LightGBM 0.96929 100.00 % Run 7 Sep 23, 2020 11:06 AM 3m 35s Completed
SparseNormalizer, XGBoostClassifier 0.96929 100.00 % Run 38 Sep 23, 2020 11:58 AM 39m 40s Completed
SparseNormalizer, XGBoostClassifier 0.96843 100.00 % Run 35 Sep 23, 2020 11:27 AM 35m 10s Completed
SparseNormalizer, XGBoostClassifier 0.96586 100.00 % Run 34 Sep 23, 2020 11:26 AM 29m 50s Completed
SparseNormalizer, XGBoostClassifier 0.96543 100.00 % Run 40 Sep 23, 2020 12:04 PM 19m 50s Completed
StandardScalerWrapper, XGBoostClassifier 0.96500 100.00 % Run 48 Sep 23, 2020 1:14 PM 14m 27s Completed
SparseNormalizer, XGBoostClassifier 0.96386 100.00 % Run 44 Sep 23, 2020 12:44 PM 25m 41s Completed

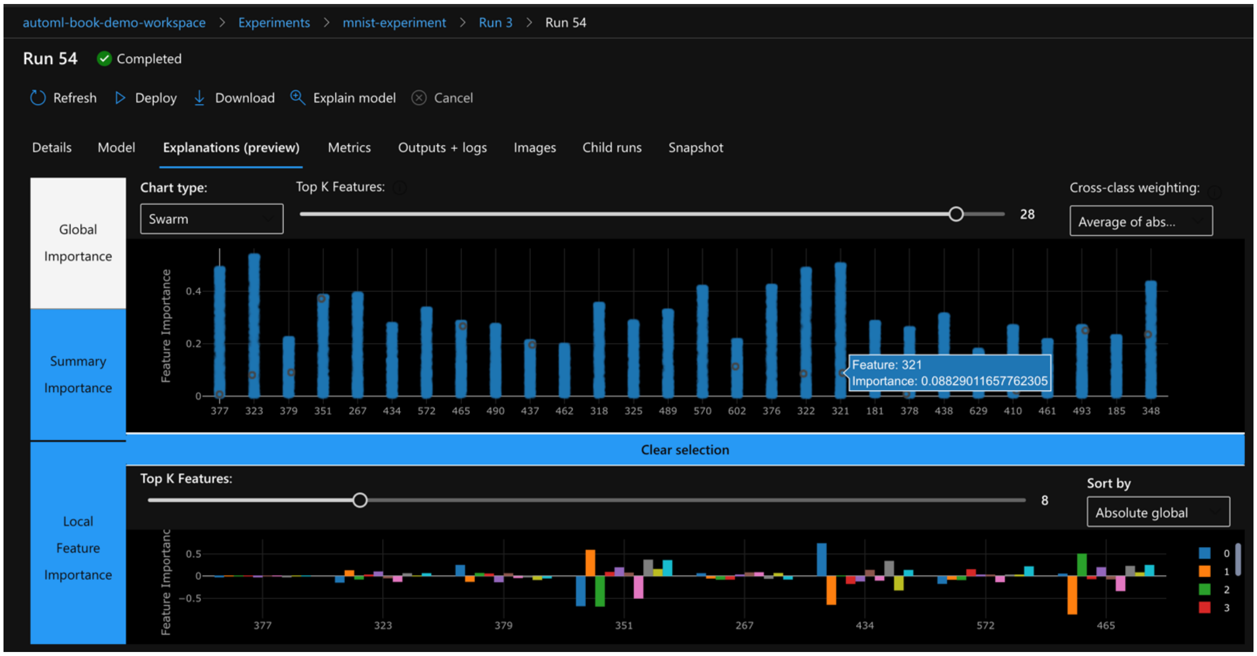

automl-book-demo-workspace > Automated ML > mnist-experiment > Run 3 > Run 7
Run 7 Completed
Refresh Deploy Download Explain model Cancel
Details Model Explanations (preview) Metrics Outputs + logs Images Child runs
Model summary
Algorithm name
MaxAbsScaler, LightGBM
Accuracy
0.96929 View all other metrics
Sampling
100.00 %
Registered models
No registration yet
Deploy status
No deployment yet
Run Metrics
Accuracy
0.96929
AUC macro
0.99930
AUC micro
0.99938
AUC weighted
0.99930
Average precision score macro
0.99512
Average precision score micro
0.99558
Average precision score weighted
0.99514
Balanced accuracy
0.96902
F1 score macro
0.96915
F1 score micro
0.96929
Close


Details
Data guardrails
Models
Outputs + logs
Child runs
Snapshot
Data guardrails are run by Automated ML when automatic featurization is enabled. This is a sequence of checks over the input data to ensure high quality data is being used to train model.
Type
Validation split handling
Status
Done
Description
The input data has been split into a training dataset and a validation dataset for validation of the model. The validation dataset is generated to improve model performance.
Learn more about validation data.
View additional details
Type
Class balancing detection
Status
Passed
Description
Your inputs were analyzed, and all classes are balanced in your training data.
Learn more about imbalanced data.
Type
Missing feature values imputation
Status
Passed
Description
No feature missing values were detected in the training data.
Learn more about missing value imputation.
Type
High cardinality feature
Status
Passed
Description
Your inputs were analyzed, and no high cardinality features were detected.

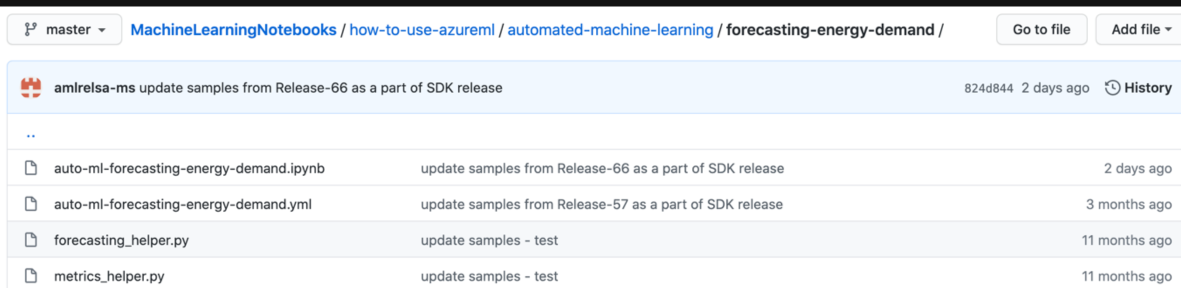

automl-book-demo-workspace > Experiments > mnist-experiment > Run 3
Run 3
Completed
Refresh
Cancel
Details
Data guardrails
Models
Outputs + logs
Child runs
Snapshot
Properties
Status
Completed
Created
Sep 23, 2020 11:06 AM
Duration
5h 33m 24.94s
Compute target
cpu-cluster
Run ID
AutoML_0b975318-8040-482d-a8bf-d4cc4d86b785
Run number
3
Script name
--
Created by
Adnan Masood
Input datasets
Best model summary
Algorithm name
VotingEnsemble
Accuracy
0.97000
View all other metrics
Sampling
100.00 %
Registered models
No registration yet
Deploy status
No deployment yet
Run summary
Task type
Classification
View all run settings
Primary metric
Accuracy


Run 54
Completed
Refresh
Deploy
Download
Explain model
Cancel
Details
Model
Explanations (preview)
Metrics
Outputs + logs
Images
Child runs
Snapshot
Select a metric to see a visualization or table of the data.
accuracy
accuracy_table
AUC_macro
AUC_micro
AUC_weighted
average_precision_score_macro
average_precision_score_micro
average_precision_score_weighted
balanced_accuracy
confusion_matrix
f1_score_macro
View as:
Chart
Table
accuracy
0.97
AUC_macro
0.9989777331109415
AUC_micro
0.9989674285714286
AUC_weighted
0.9989794294893661
Precision-Recall
Precision
Weighted Average
Macro Average
Micro Average
Ideal

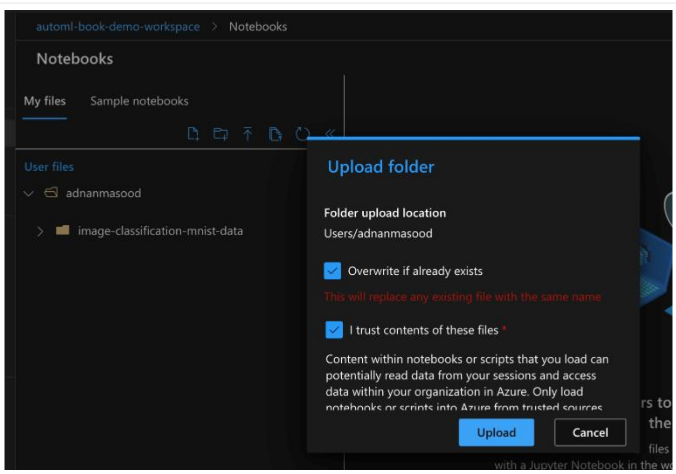

automl-book-demo-workspace > Experiments > mnist-experiment > Run 3 > Run 54
Run 54
Completed
Refresh
Deploy
Download
Explain model
Cancel
Details
Model
Explanations (preview)
Metrics
Outputs + logs
Images
Child runs
Snapshot
Model explanations are used to understand what features are directly impacting the model and why. Learn more
Select Explanation
tabular | mimic.linear | raw | classification | 43101ef4-e081-4022-bb74-905f28f489e5 | 9/23/2020, 4:47:36 PM
Explainer: mimic.linear
Global Importance
Summary Importance
Top K Features:
8
Sort by
Absolute global
Feature Importance
377
323
379
351
267
434
572
465



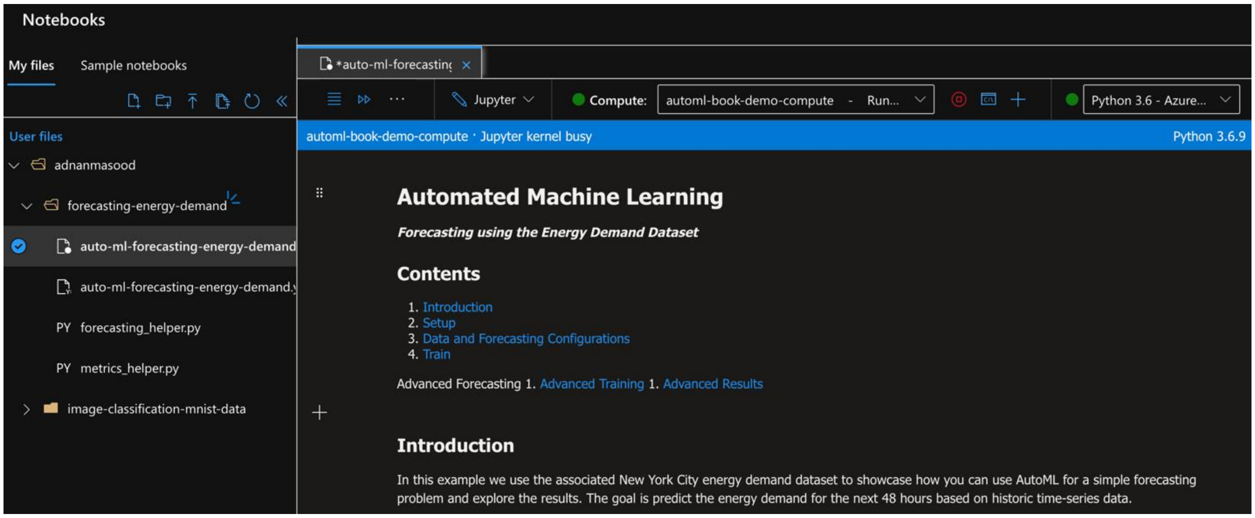
automl-book-demo-workspace > Experiments > mnist-experiment > Run 3 > Run 54
Run 54 Completed
Refresh Deploy Download Explain model Cancel
Details Model Explanations (preview) Metrics Outputs + logs Images Child runs Snapshot
Global Importance
Summary Importance
Local Feature Importance
Chart type:
Swarm
Top K Features:
28
Cross-class weighting:
Average of abs...
Feature Importance
Feature: 321
Importance: 0.08829011657762305
377 323 379 351 267 434 572 465 490 437 462 318 325 489 570 602 376 322 321 181 378 438 629 410 461 493 185 348
Clear selection
Top K Features:
8
Sort by
Absolute global
377 323 379 351 267 434 572 465
0 1 2 3

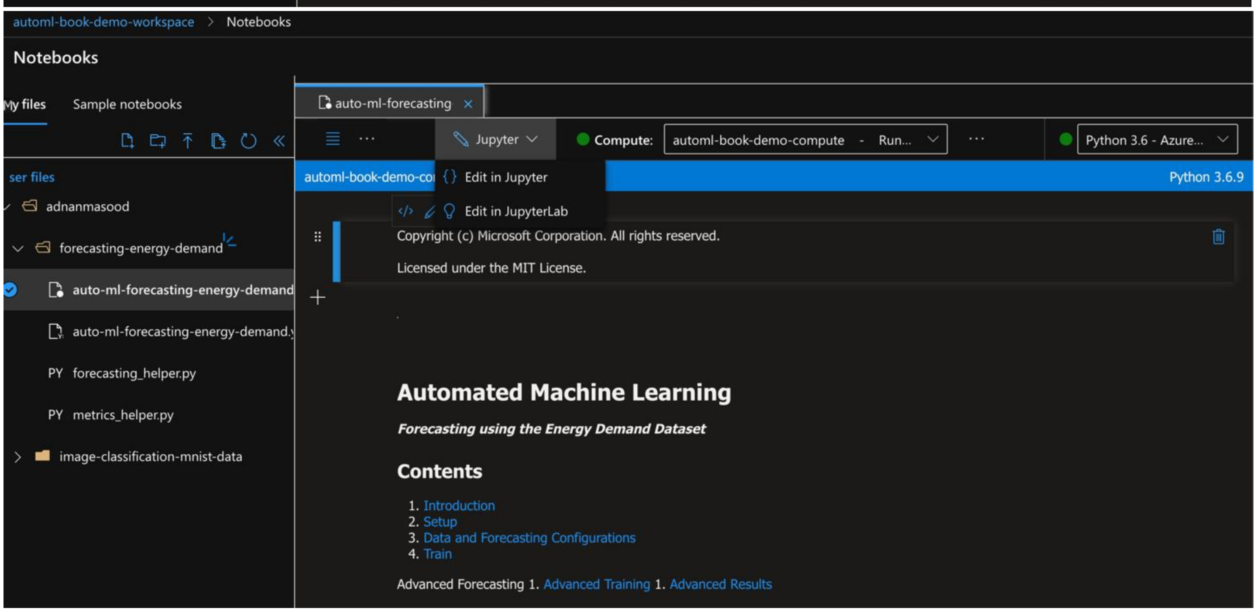


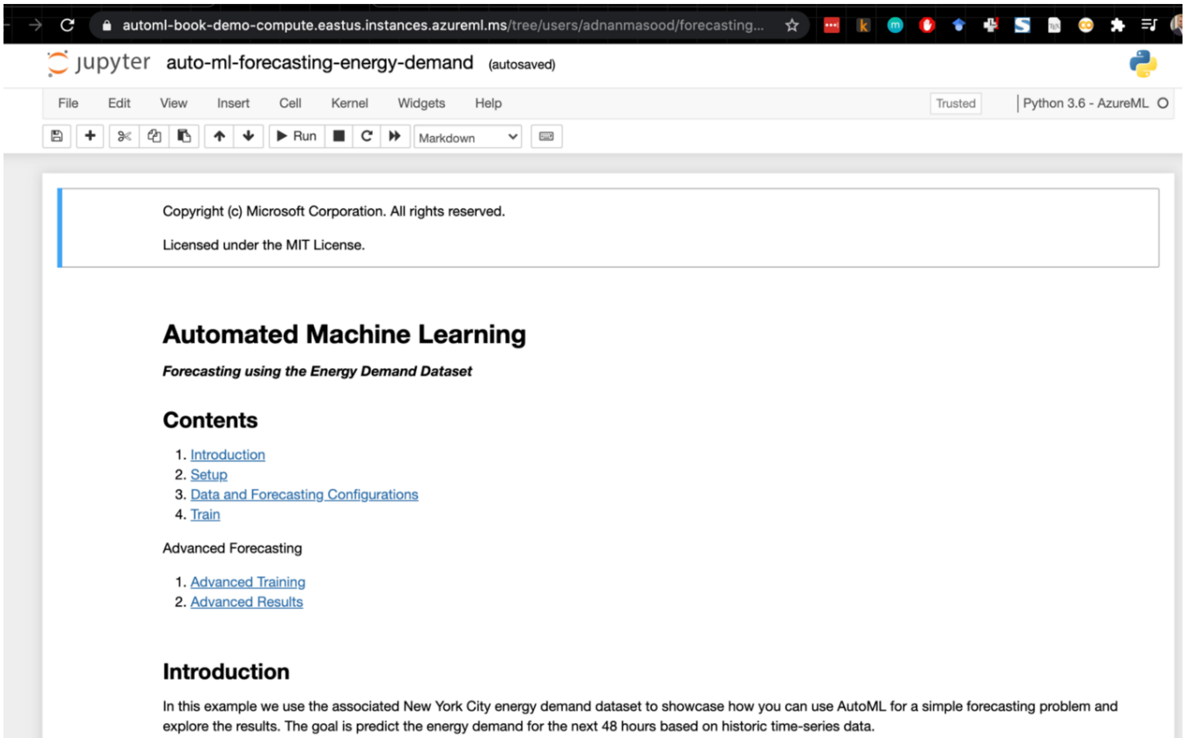
master
MachineLearningNotebooks / how-to-use-azureml / automated-machine-learning / forecasting-energy-demand /
Go to file
Add file
amlrelsa-ms update samples from Release-66 as a part of SDK release
824d844 2 days ago History
..
auto-ml-forecasting-energy-demand.ipynb
update samples from Release-66 as a part of SDK release
2 days ago
auto-ml-forecasting-energy-demand.yml
update samples from Release-57 as a part of SDK release
3 months ago
forecasting_helper.py
update samples - test
11 months ago
metrics_helper.py
update samples - test
11 months ago



automl-book-demo-workspace > Notebooks
Notebooks
My files Sample notebooks
User files
adnanmasood
image-classification-mnist-data
Upload folder
Folder upload location
Users/adnanmasood
Overwrite if already exists
This will replace any existing file with the same name
I trust contents of these files *
Content within notebooks or scripts that you load can potentially read data from your sessions and access data within your organization in Azure. Only load
Upload
Cancel


Notebooks
My files
Sample notebooks
*auto-ml-forecasting
Jupyter
Compute:
automl-book-demo-compute - Run...
Python 3.6 - Azure...
automl-book-demo-compute · Jupyter kernel busy
Python 3.6.9
User files
adnanmasood
forecasting-energy-demand
auto-ml-forecasting-energy-demand
auto-ml-forecasting-energy-demand.y
PY forecasting_helper.py
PY metrics_helper.py
image-classification-mnist-data
Automated Machine Learning
Forecasting using the Energy Demand Dataset
Contents
1. Introduction
2. Setup
3. Data and Forecasting Configurations
4. Train
Advanced Forecasting 1. Advanced Training 1. Advanced Results
Introduction
In this example we use the associated New York City energy demand dataset to showcase how you can use AutoML for a simple forecasting problem and explore the results. The goal is predict the energy demand for the next 48 hours based on historic time-series data.


automl-book-demo-workspace > Notebooks
Notebooks
My files
Sample notebooks
auto-ml-forecasting
Jupyter
Compute:
automl-book-demo-compute - Run...
Python 3.6 - Azure...
automl-book-demo-co
Edit in Jupyter
Edit in JupyterLab
Python 3.6.9
ser files
adnanmasood
forecasting-energy-demand
auto-ml-forecasting-energy-demand
auto-ml-forecasting-energy-demand.y
PY forecasting_helper.py
PY metrics_helper.py
image-classification-mnist-data
Copyright (c) Microsoft Corporation. All rights reserved.
Licensed under the MIT License.
Automated Machine Learning
Forecasting using the Energy Demand Dataset
Contents
1. Introduction
2. Setup
3. Data and Forecasting Configurations
4. Train
Advanced Forecasting 1. Advanced Training 1. Advanced Results



Automated ML provides users with both native time-series and deep learning models as part of the recommendation system.

Models	Description	Benefits
Prophet (Preview)	Prophet works best with time series that have strong seasonal effects and several seasons of historical data. To leverage this model, install it locally using `pip install fbprophet`.	Accurate & fast, robust to outliers, missing data, and dramatic changes in your time series.
Auto-ARIMA (Preview)	Auto-Regressive Integrated Moving Average (ARIMA) performs best, when the data is stationary. This means that its statistical properties like the mean and variance are constant over the entire set. For example, if you flip a coin, then the probability of you getting heads is 50%, regardless if you flip today, tomorrow or next year.	Great for univariate series, since the past values are used to predict the future values.
ForecastTCN (Preview)	ForecastTCN is a neural network model designed to tackle the most demanding forecasting tasks, capturing nonlinear local and global trends in your data as well as relationships between time series.	Capable of leveraging complex trends in your data and readily scales to the largest of datasets.

Classification	Regression	Time Series Forecasting
Logistic Regression*	Elastic Net*	Elastic Net
Light GBM*	Light GBM*	Light GBM
Gradient Boosting*	Gradient Boosting*	Gradient Boosting
Decision Tree*	Decision Tree*	Decision Tree
K Nearest Neighbors*	K Nearest Neighbors*	K Nearest Neighbors
Linear SVC*	LARS Lasso*	LARS Lasso
Support Vector Classification (SVC)*	Stochastic Gradient Descent (SGD)*	Stochastic Gradient Descent (SGD)
Random Forest*	Random Forest*	Random Forest
Extremely Randomized Trees*	Extremely Randomized Trees*	Extremely Randomized Trees
Xgboost*	Xgboost*	Xgboost
Averaged Perceptron Classifier	Online Gradient Descent Regressor	Auto-ARIMA
Naive Bayes*	Fast Linear Regressor	Prophet
Stochastic Gradient Descent (SGD)*		ForecastTCN
Linear SVM Classifier*		

Classification	Regression	Time Series Forecasting
accuracy	spearman_correlation	spearman_correlation
AUC_weighted	normalized_root_mean_squared_error	normalized_root_mean_squared_error
average_precision_score_weighted	r2_score	r2_score
norm_macro_recall	normalized_mean_absolute_error	normalized_mean_absolute_error
precision_score_weighted		

Target column is what we want to forecast.
Time column is the time axis along which to predict.

The other columns, "temp" and "precip", are implicitly designated as features.

In [ ]:
```
target_column_name = 'demand'
time_column_name = 'timeStamp'
```

In [ ]:
```
dataset = Dataset.Tabular.from_delimited_files(path = "https://automlsamplenotebookdata.blob.core.windows.net/automl-sample-notebook-dat
a/nyc_energy.csv").with_timestamp_columns(fine_grain_timestamp=time_column_name)
dataset.take(5).to_pandas_dataframe().reset_index(drop=True)
```

The NYC Energy dataset is missing energy demand values for all datetimes later than August 10th, 2017 5AM. Below, we trim the rows containing these missing values from the end of the dataset.

In [ ]:
```
# Cut off the end of the dataset due to large number of nan values
dataset = dataset.time_before(datetime(2017, 10, 10, 5))
```

Split the data into train and test sets

The first split we make is into train and test sets. Note that we are splitting on time. Data before and including August 8th, 2017 5AM will be used for training, and data after will be used for testing.

```
# split into train based on time
train = dataset.time_before(datetime(2017, 8, 8, 5), include_boundary=True)
train.to_pandas_dataframe().reset_index(drop=True).sort_values(time_column_nam
e).tail(5)
```

```
# split into test based on time
test = dataset.time_between(datetime(2017, 8, 8, 6), datetime(2017, 8, 10, 5))
test.to_pandas_dataframe().reset_index(drop=True).head(5)
```

```
forecast_horizon = 48
```

```
from azureml.automl.core.forecasting_parameters import ForecastingParameters
forecasting_parameters = ForecastingParameters(
    time_column_name=time_column_name, forecast_horizon=forecast_horizon
)

automl_config = AutoMLConfig(task='forecasting',
                             primary_metric='normalized_root_mean_squared_erro
r',
                             blocked_models = ['ExtremeRandomTrees', 'AutoArim
a', 'Prophet'],
                             experiment_timeout_hours=0.3,
                             training_data=train,
                             label_column_name=target_column_name,
                             compute_target=compute_target,
                             enable_early_stopping=True,
                             n_cross_validations=3,
                             verbosity=logging.INFO,
                             forecasting_parameters=forecasting_parameters)
```


```
forecast_horizon = 48
```

```
In [12]: remote_run = experiment.submit(automl_config, show_output=False)
```

```
Running on remote or ADB.
```

```
In [13]: remote_run
```

Out[13]:

Experiment	Id	Type	Status	Details Page	Docs Page
automl-forecasting-energydemand	AutoML_31651b62-6c60-4ccf-a145-be69dd4e95e3	automl	NotStarted	Link to Azure Machine Learning studio	Link to Documentation

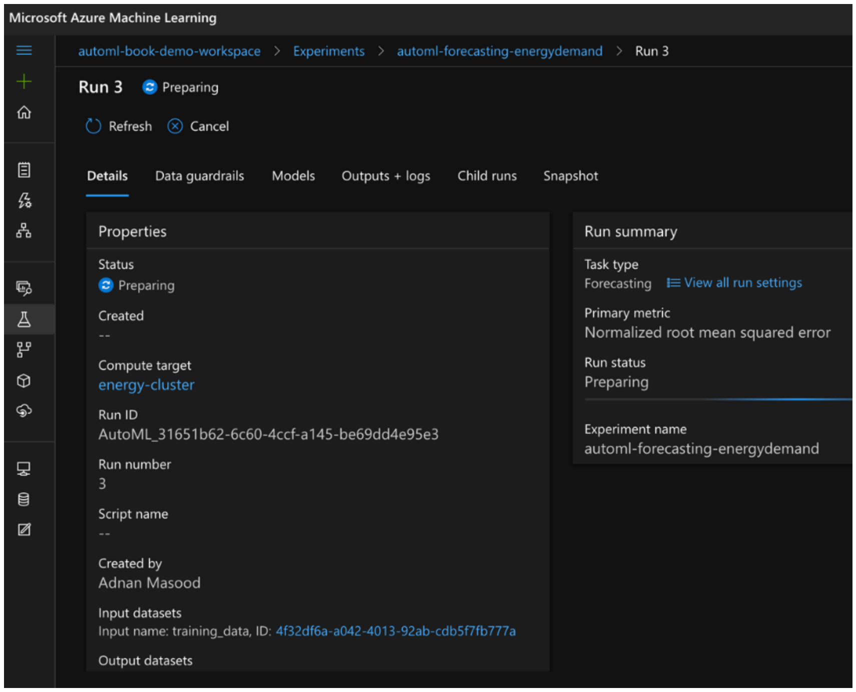
```
In [*]: remote_run.wait_for_completion()
```

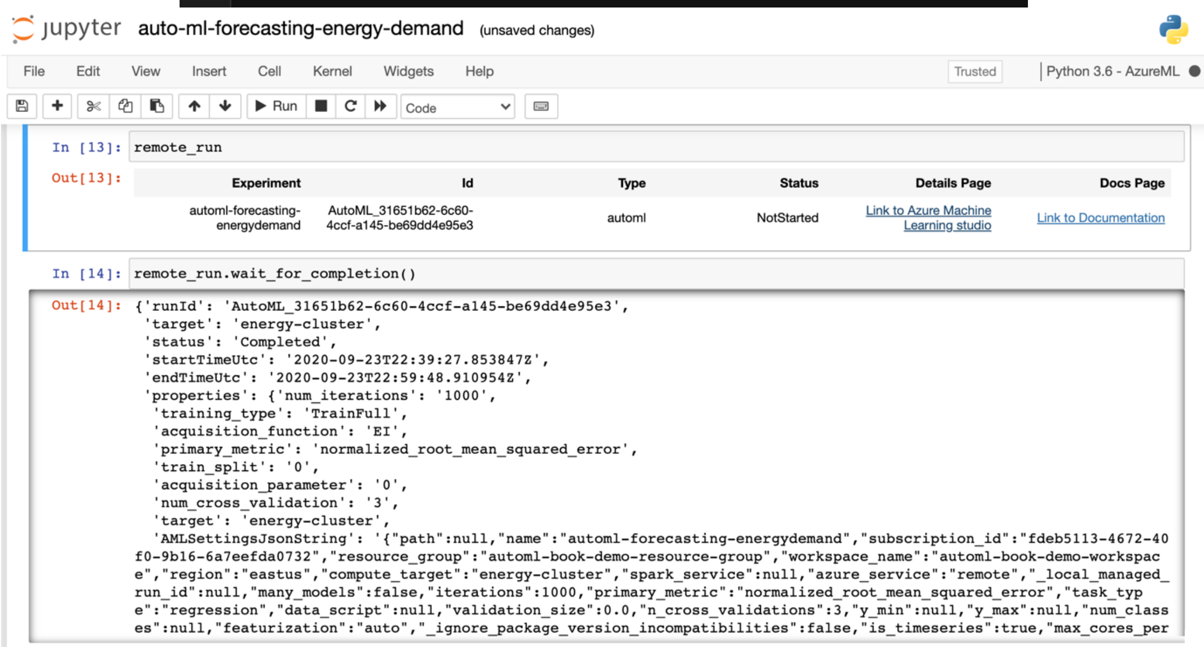
Retrieve the Best Model

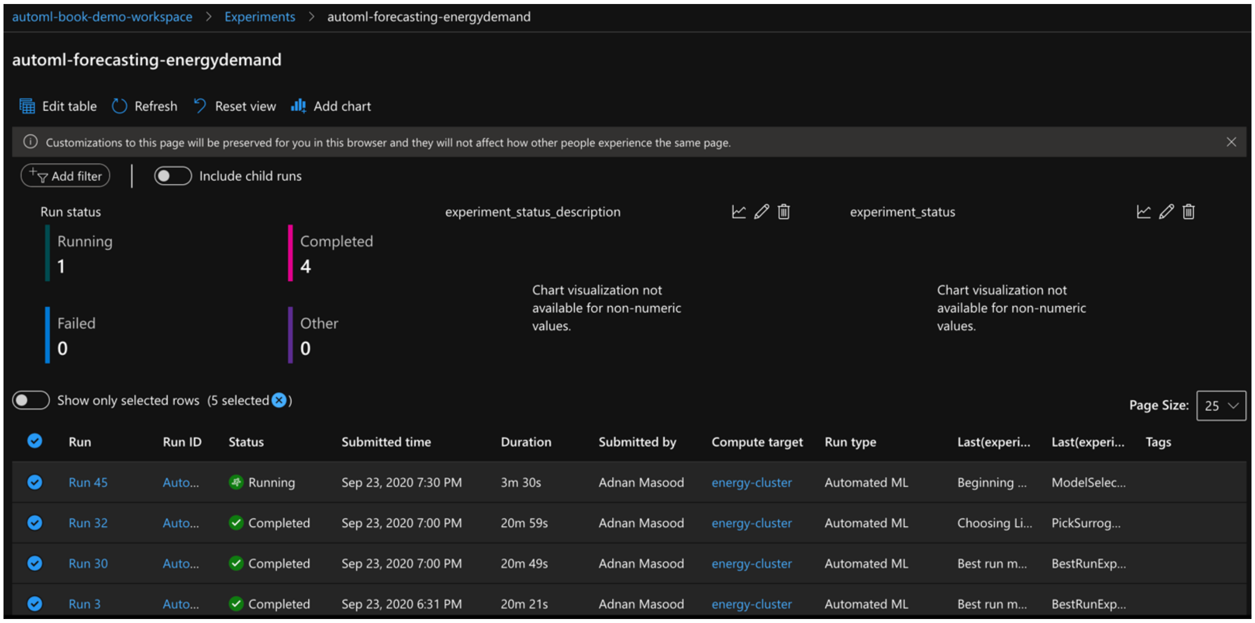
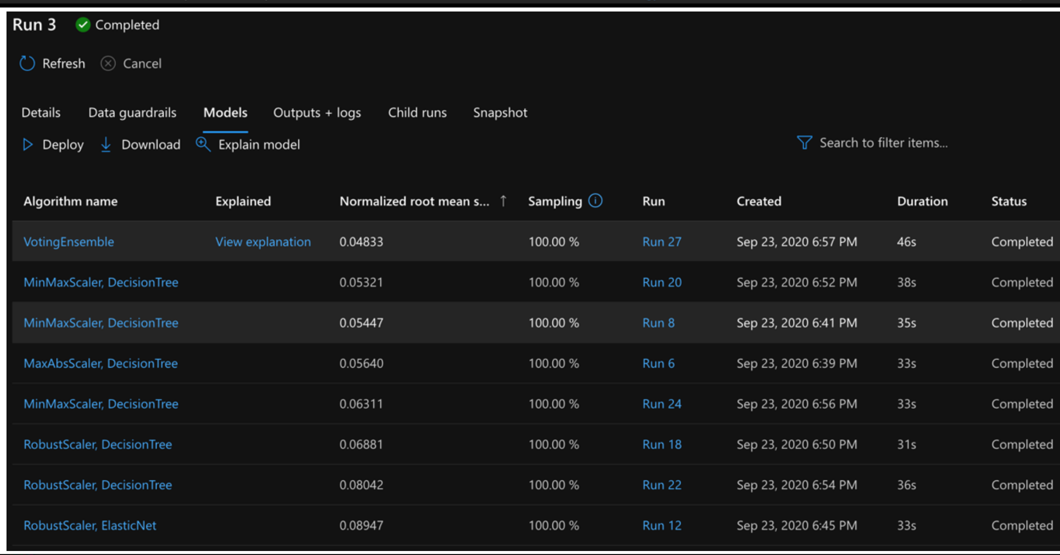
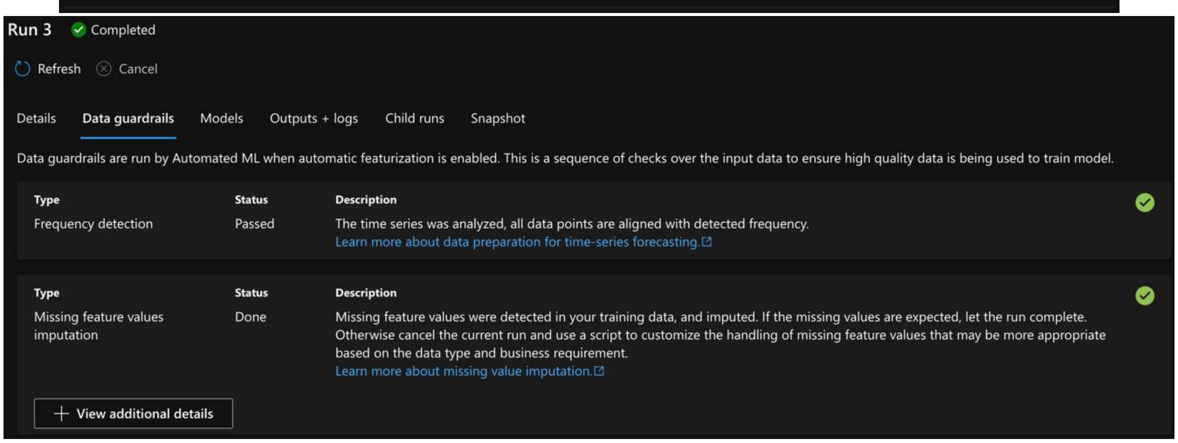
Below we select the best model from all the training iterations using get_output method.

```
In [*]: best_run, fitted_model = remote_run.get_output()
        fitted_model.steps
```

```
from azureml.automl.core.forecasting_parameters import ForecastingParameters
forecasting_parameters = ForecastingParameters(
    time_column_name=time_column_name, forecast_horizon=forecast_horizon
)

automl_config = AutoMLConfig(task='forecasting',
                             primary_metric='normalized_root_mean_squared_erro
r',
                             blocked_models = ['ExtremeRandomTrees', 'AutoArim
a', 'Prophet'],
                             experiment_timeout_hours=0.3,
                             training_data=train,
                             label_column_name=target_column_name,
                             compute_target=compute_target,
                             enable_early_stopping=True,
                             n_cross_validations=3,
                             verbosity=logging.INFO,
                             forecasting_parameters=forecasting_parameters)
```


Microsoft Azure Machine Learning
automl-book-demo-workspace > Experiments > automl-forecasting-energydemand > Run 3
Run 3 Preparing
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Properties
Status
Preparing
Created
--
Compute target
energy-cluster
Run ID
AutoML_31651b62-6c60-4ccf-a145-be69dd4e95e3
Run number
3
Script name
--
Created by
Adnan Masood
Input datasets
Input name: training_data, ID: 4f32df6a-a042-4013-92ab-cdb5f7fb777a
Output datasets
Run summary
Task type
Forecasting
View all run settings
Primary metric
Normalized root mean squared error
Run status
Preparing
Experiment name
automl-forecasting-energydemand


jupyter auto-ml-forecasting-energy-demand (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3.6 - AzureML
Run
Code
In [13]: remote_run
Out[13]:
Experiment Id Type Status Details Page Docs Page
automl-forecasting-energydemand AutoML_31651b62-6c60-4ccf-a145-be69dd4e95e3 automl NotStarted Link to Azure Machine Learning studio Link to Documentation
In [14]: remote_run.wait_for_completion()
Out[14]: {'runId': 'AutoML_31651b62-6c60-4ccf-a145-be69dd4e95e3',
'target': 'energy-cluster',
'status': 'Completed',
'startTimeUtc': '2020-09-23T22:39:27.853847Z',
'endTimeUtc': '2020-09-23T22:59:48.910954Z',
'properties': {'num_iterations': '1000',
'training_type': 'TrainFull',
'acquisition_function': 'EI',
'primary_metric': 'normalized_root_mean_squared_error',
'train_split': '0',
'acquisition_parameter': '0',
'num_cross_validation': '3',
'target': 'energy-cluster',
'AMLSettingsJsonString': '{"path":null,"name":"automl-forecasting-energydemand","subscription_id":"fdeb5113-4672-40
f0-9b16-6a7eefda0732","resource_group":"automl-book-demo-resource-group","workspace_name":"automl-book-demo-workspac
e","region":"eastus","compute_target":"energy-cluster","spark_service":null,"azure_service":"remote","_local_managed_
run_id":null,"many_models":false,"iterations":1000,"primary_metric":"normalized_root_mean_squared_error","task_typ
e":"regression","data_script":null,"validation_size":0.0,"n_cross_validations":3,"y_min":null,"y_max":null,"num_class
es":null,"featurization":"auto","_ignore_package_version_incompatibilities":false,"is_timeseries":true,"max_cores_per


automl-book-demo-workspace > Experiments > automl-forecasting-energydemand
automl-forecasting-energydemand
Edit table Refresh Reset view Add chart
Customizations to this page will be preserved for you in this browser and they will not affect how other people experience the same page.
Add filter
Include child runs
Run status
Running 1
Completed 4
Failed 0
Other 0
experiment_status_description
Chart visualization not available for non-numeric values.
experiment_status
Chart visualization not available for non-numeric values.
Show only selected rows (5 selected)
Page Size: 25
Run | Run ID | Status | Submitted time | Duration | Submitted by | Compute target | Run type | Last(experi... | Last(experi... | Tags
Run 45 | Auto... | Running | Sep 23, 2020 7:30 PM | 3m 30s | Adnan Masood | energy-cluster | Automated ML | Beginning ... | ModelSelec...
Run 32 | Auto... | Completed | Sep 23, 2020 7:00 PM | 20m 59s | Adnan Masood | energy-cluster | Automated ML | Choosing Li... | PickSurrog...
Run 30 | Auto... | Completed | Sep 23, 2020 7:00 PM | 20m 49s | Adnan Masood | energy-cluster | Automated ML | Best run m... | BestRunExp...
Run 3 | Auto... | Completed | Sep 23, 2020 6:31 PM | 20m 21s | Adnan Masood | energy-cluster | Automated ML | Best run m... | BestRunExp...

Run 3 Completed
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Deploy Download Explain model
Search to filter items...
Algorithm name | Explained | Normalized root mean s... | Sampling | Run | Created | Duration | Status
VotingEnsemble | View explanation | 0.04833 | 100.00 % | Run 27 | Sep 23, 2020 6:57 PM | 46s | Completed
MinMaxScaler, DecisionTree | 0.05321 | 100.00 % | Run 20 | Sep 23, 2020 6:52 PM | 38s | Completed
MinMaxScaler, DecisionTree | 0.05447 | 100.00 % | Run 8 | Sep 23, 2020 6:41 PM | 35s | Completed
MaxAbsScaler, DecisionTree | 0.05640 | 100.00 % | Run 6 | Sep 23, 2020 6:39 PM | 33s | Completed
MinMaxScaler, DecisionTree | 0.06311 | 100.00 % | Run 24 | Sep 23, 2020 6:56 PM | 33s | Completed
RobustScaler, DecisionTree | 0.06881 | 100.00 % | Run 18 | Sep 23, 2020 6:50 PM | 31s | Completed
RobustScaler, DecisionTree | 0.08042 | 100.00 % | Run 22 | Sep 23, 2020 6:54 PM | 36s | Completed
RobustScaler, ElasticNet | 0.08947 | 100.00 % | Run 12 | Sep 23, 2020 6:45 PM | 33s | Completed

Run 3 Completed
Refresh Cancel
Details Data guardrails Models Outputs + logs Child runs Snapshot
Data guardrails are run by Automated ML when automatic featurization is enabled. This is a sequence of checks over the input data to ensure high quality data is being used to train model.
Type Frequency detection
Status Passed
Description The time series was analyzed, all data points are aligned with detected frequency.
Learn more about data preparation for time-series forecasting.
Type Missing feature values imputation
Status Done
Description Missing feature values were detected in your training data, and imputed. If the missing values are expected, let the run complete. Otherwise cancel the current run and use a script to customize the handling of missing feature values that may be more appropriate based on the data type and business requirement.
Learn more about missing value imputation.
View additional details

Retrieve the Best Model

Below we select the best model from all the training iterations using get_output method.

In [15]:
```
best_run, fitted_model = remote_run.get_output()
fitted_model.steps
```

Out[15]:
```
[('timeseriestransformer',
  TimeSeriesTransformer(featurization_config=None,
                        pipeline_type=<TimeSeriesPipelineType.FULL: 1>)),
 ('prefittedsoftvotingregressor',
  PreFittedSoftVotingRegressor(estimators=[('7',
                                            Pipeline(memory=None,
                                                     steps=[('minmaxscaler',
                                                             MinMaxScaler(copy=True,
                                                                          feature_range=(0,
                                                                                         1))),
                                                            ('decisiontreeregressor',
                                                             DecisionTreeRegressor(ccp_alpha=0.0,
                                                                                   criterion='mse',
                                                                                   max_depth=None,
                                                                                   max_features=0.7,
                                                                                   max_leaf_nodes=None,
                                                                                   min_impurity_decrease=0.0,
                                                                                   min_impurity_split=None,
                                                                                   min_samples_leaf=0.001953125,
                                                                                   min_sam...
                                                                                   max_depth=None,
                                                                                   max_features=0.8,
                                                                                   max_leaf_nodes=None,
                                                                                   min_impurity_decrease=0.0,
                                                                                   min_impurity_split=None,
                                                                                   min_samples_leaf=0.018779547644135
22,
                                                                                   min_samples_split=0.00182615846827
02607,
                                                                                   min_weight_fraction_leaf=0.0,
                                                                                   presort='deprecated',
                                                                                   random_state=None,
                                                                                   splitter='best'))],
                                                     verbose=False))],
                               weights=[0.45454545454545453, 0.2727272727272727,
                                        0.2727272727272727]))]
```

Featurization

You can access the engineered feature names generated in time-series featurization.

In [16]:
```
fitted_model.named_steps['timeseriestransformer'].get_engineered_feature_names()
```

Out[16]:
```
['precip',
 'temp',
 'precip_WASNULL',
 'temp_WASNULL',
 'year',
 'half',
 'quarter',
 'month',
 'day',
 'hour',
 'am_pm',
 'hour12',
 'wday',
 'qday',
 'week']
```

View featurization summary

You can also see what featurization steps were performed on different raw features in the user data. For each raw feature in the user data, the following information is displayed:

- Raw feature name
- Number of engineered features formed out of this raw feature
- Type detected
- If feature was dropped
- List of feature transformations for the raw feature

In [17]:
```
# Get the featurization summary as a list of JSON
featurization_summary = fitted_model.named_steps['timeseriestransformer'].get_featurization_summary()
# View the featurization summary as a pandas dataframe
pd.DataFrame.from_records(featurization_summary)
```

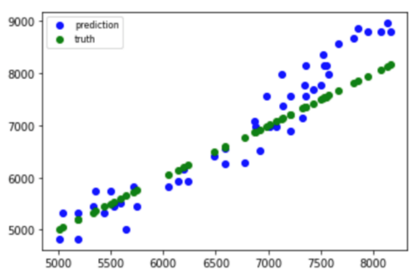
Out[17]:

	RawFeatureName	TypeDetected	Dropped	EngineeredFeatureCount	Transformations
0	precip	Numeric	No	2	[MedianImputer, ImputationMarker]
1	temp	Numeric	No	2	[MedianImputer, ImputationMarker]
2	timeStamp	DateTime	No	11	[DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer, DateTimeTransformer]

In [21]:
```python
from azureml.automl.core.shared import constants
from azureml.automl.runtime.shared.score import scoring
from matplotlib import pyplot as plt

# use automl metrics module
scores = scoring.score_regression(
    y_test=df_all[target_column_name],
    y_pred=df_all['predicted'],
    metrics=list(constants.Metric.SCALAR_REGRESSION_SET))

print("[Test data scores]\n")
for key, value in scores.items():
    print('{}:   {:.3f}'.format(key, value))

# Plot outputs
%matplotlib inline
test_pred = plt.scatter(df_all[target_column_name], df_all['predicted'], color='b')
test_test = plt.scatter(df_all[target_column_name], df_all[target_column_name], color='g')
plt.legend((test_pred, test_test), ('prediction', 'truth'), loc='upper left', fontsize=8)
plt.show()
```

```
[Test data scores]

normalized_root_mean_squared_error:   0.150
mean_absolute_percentage_error:   5.491
normalized_mean_absolute_error:   0.122
r2_score:   0.743
normalized_median_absolute_error:   0.097
root_mean_squared_log_error:   0.064
normalized_root_mean_squared_log_error:   0.130
explained_variance:   0.787
mean_absolute_error:   383.207
root_mean_squared_error:   473.089
spearman_correlation:   0.972
median_absolute_error:   305.623
```

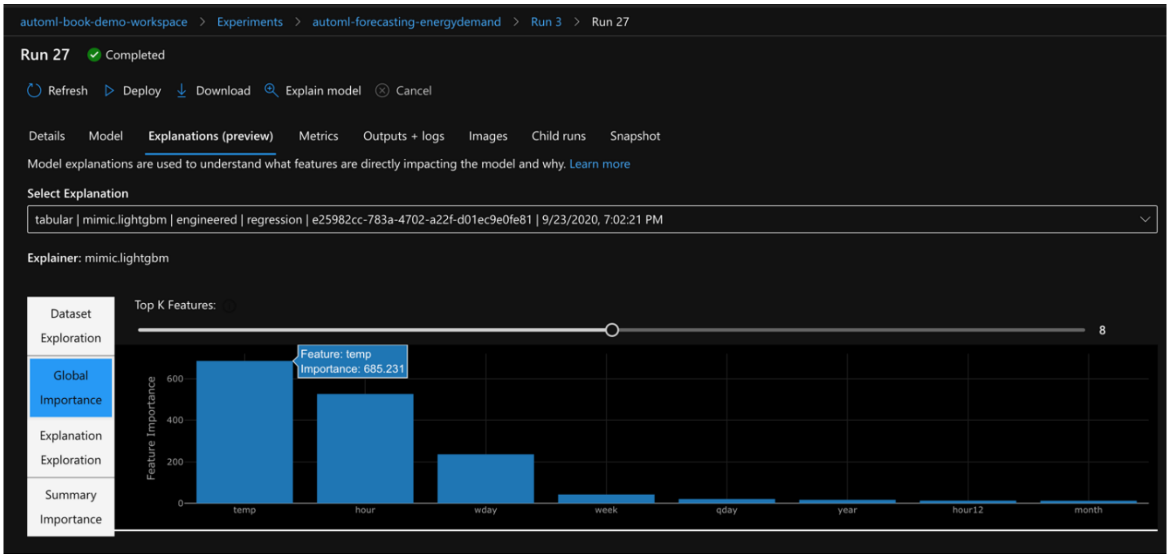
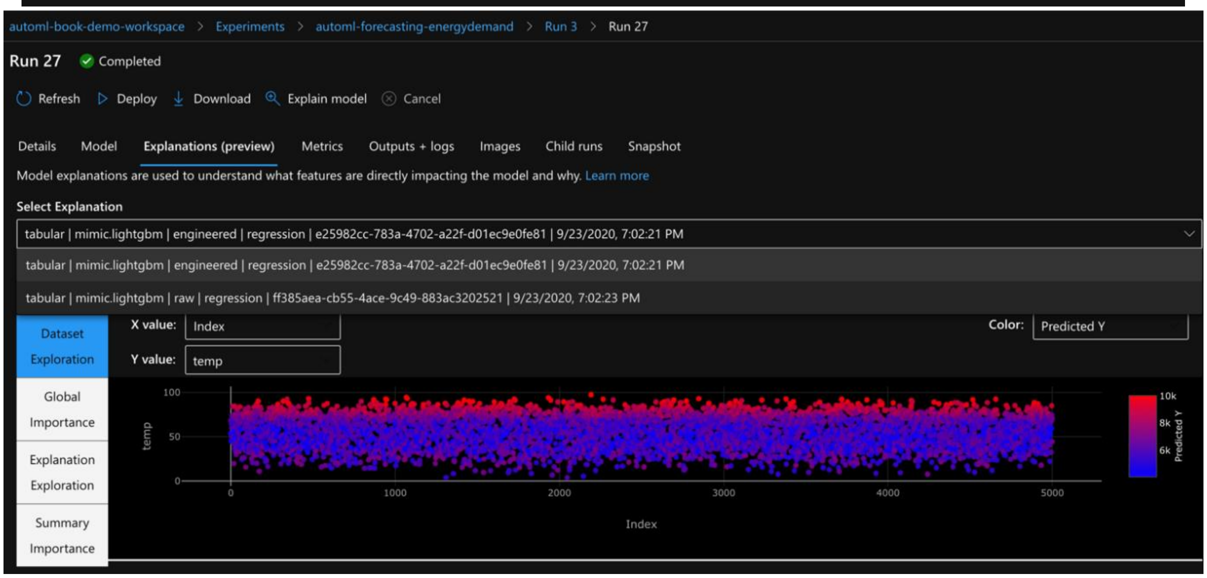

```
[Test data scores]

normalized_root_mean_squared_error:   0.150
mean_absolute_percentage_error:   5.491
normalized_mean_absolute_error:   0.122
r2_score:   0.743
normalized_median_absolute_error:   0.097
root_mean_squared_log_error:   0.064
normalized_root_mean_squared_log_error:   0.130
explained_variance:   0.787
mean_absolute_error:   383.207
root_mean_squared_error:   473.089
spearman_correlation:   0.972
median_absolute_error:   305.623
```

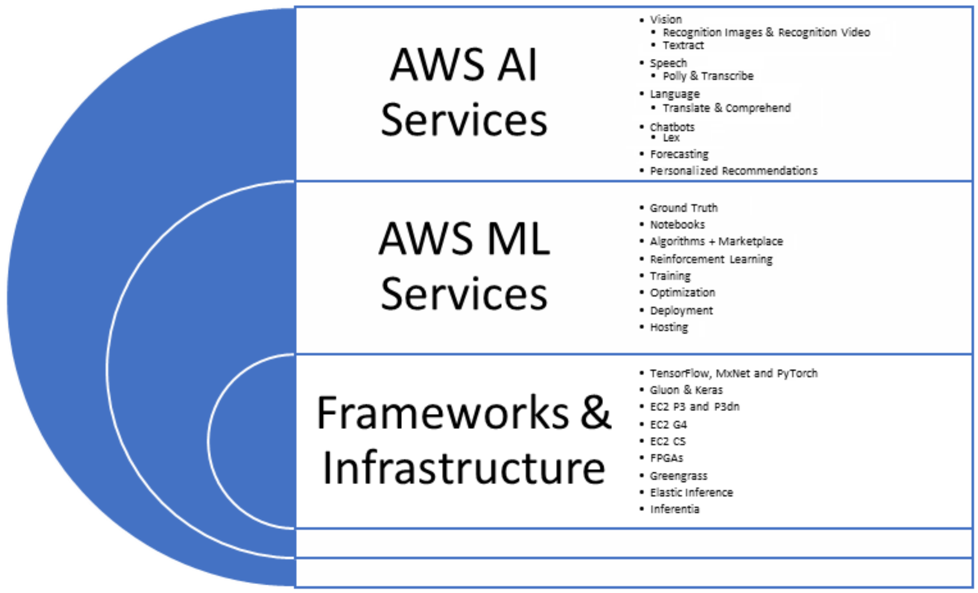
Looking at `X_trans` is also useful to see what featurization happened to the data.

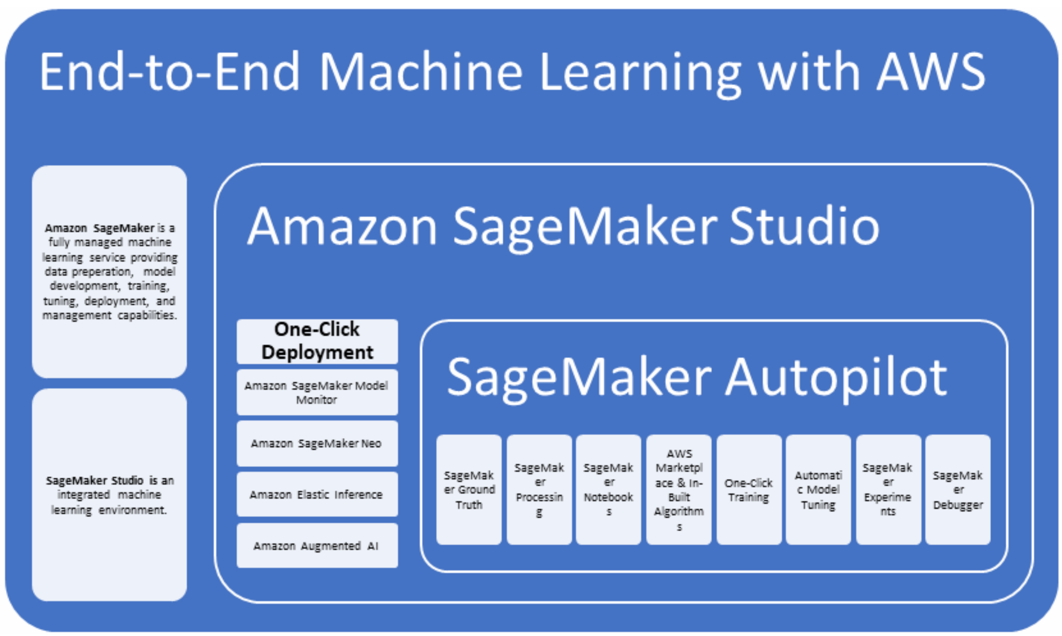
In [22]:
```python
X_trans
```

timeStamp	_automl_dummy_grain_col	precip	temp	precip_WASNULL	temp_WASNULL	year	half	quarter	month	day	hour	am_pm	hour12	wday	qday	week	_automl_ta
2017-08-08 06:00:00	_automl_dummy_grain_col	0.00	66.17	0	0	2017	2	3	8	8	6	0	6	1	39	32	
2017-08-08 07:00:00	_automl_dummy_grain_col	0.00	66.29	0	0	2017	2	3	8	8	7	0	7	1	39	32	
2017-08-08 08:00:00	_automl_dummy_grain_col	0.00	66.72	0	0	2017	2	3	8	8	8	0	8	1	39	32	
2017-08-08 09:00:00	_automl_dummy_grain_col	0.00	67.37	0	0	2017	2	3	8	8	9	0	9	1	39	32	
2017-08-08 10:00:00	_automl_dummy_grain_col	0.00	68.30	0	0	2017	2	3	8	8	10	0	10	1	39	32	

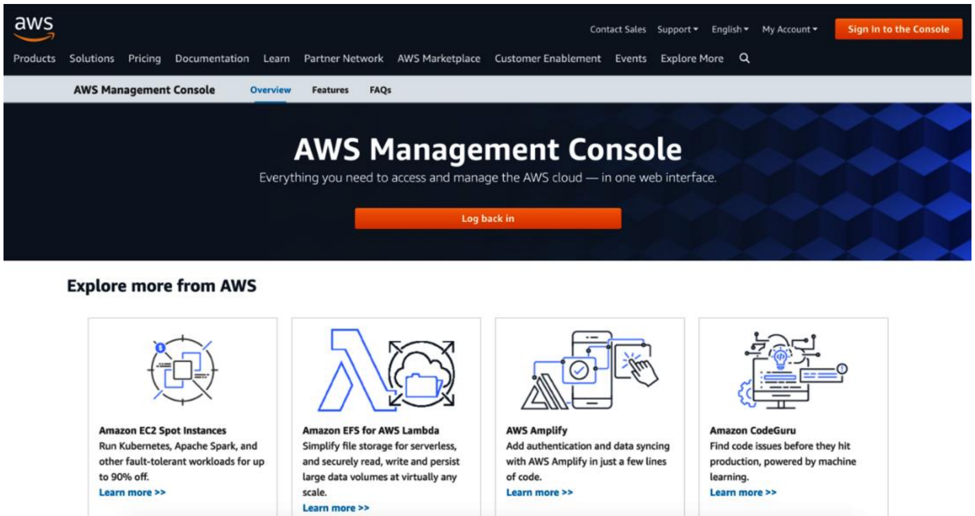

automl-book-demo-workspace > Experiments > automl-forecasting-energydemand > Run 3 > Run 27
Run 27
Completed
Refresh
Deploy
Download
Explain model
Cancel
Details
Model
Explanations (preview)
Metrics
Outputs + logs
Images
Child runs
Snapshot
Model explanations are used to understand what features are directly impacting the model and why. Learn more
Select Explanation
tabular | mimic.lightgbm | engineered | regression | e25982cc-783a-4702-a22f-d01ec9e0fe81 | 9/23/2020, 7:02:21 PM
Explainer: mimic.lightgbm
Dataset Exploration
Global Importance
Explanation Exploration
Summary Importance
Top K Features:
8
Feature: temp
Importance: 685.231
Feature Importance
600
400
200
0
temp
hour
wday
week
qday
year
hour12
month

automl-book-demo-workspace > Experiments > automl-forecasting-energydemand > Run 3 > Run 27
Run 27
Completed
Refresh
Deploy
Download
Explain model
Cancel
Details
Model
Explanations (preview)
Metrics
Outputs + logs
Images
Child runs
Snapshot
Model explanations are used to understand what features are directly impacting the model and why. Learn more
Select Explanation
tabular | mimic.lightgbm | engineered | regression | e25982cc-783a-4702-a22f-d01ec9e0fe81 | 9/23/2020, 7:02:21 PM
tabular | mimic.lightgbm | engineered | regression | e25982cc-783a-4702-a22f-d01ec9e0fe81 | 9/23/2020, 7:02:21 PM
tabular | mimic.lightgbm | raw | regression | ff385aea-cb55-4ace-9c49-883ac3202521 | 9/23/2020, 7:02:23 PM
Dataset Exploration
Global Importance
Explanation Exploration
Summary Importance
X value:
Index
Y value:
temp
Color:
Predicted Y
100
50
0
temp
0
1000
2000
3000
4000
5000
Index
10k
8k
6k
Predicted Y

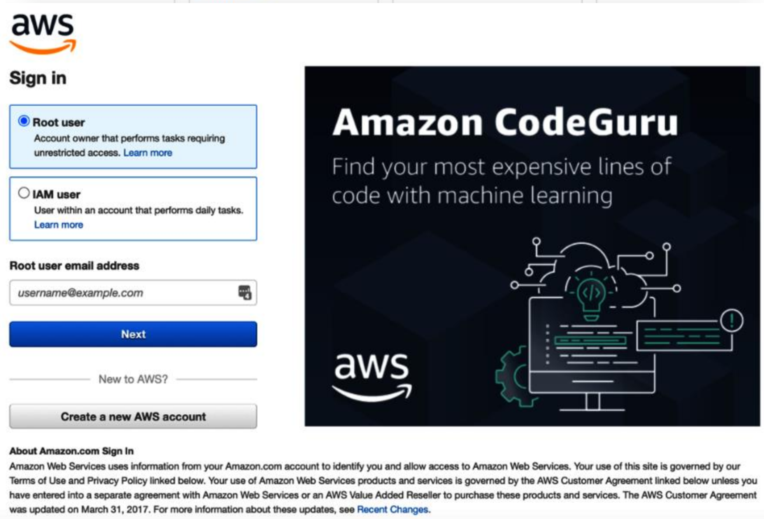
Chapter 6: Machine Learning with Amazon Web Services

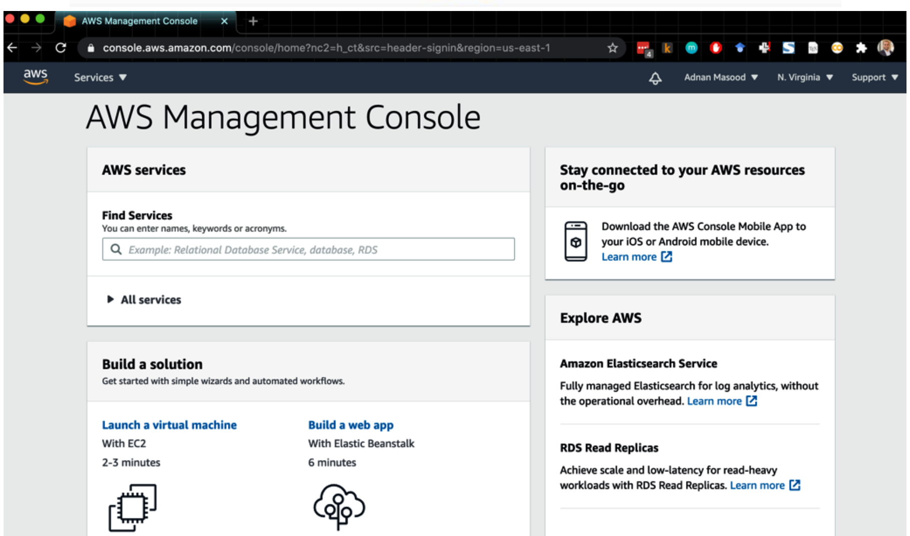


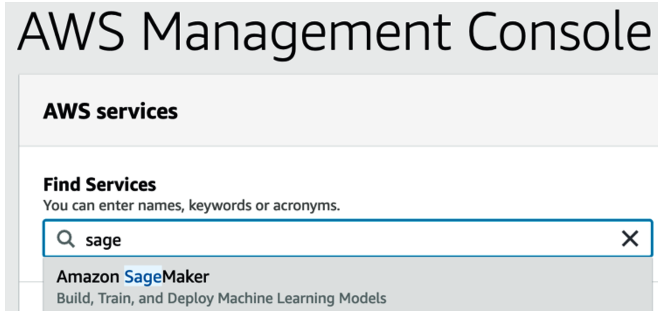


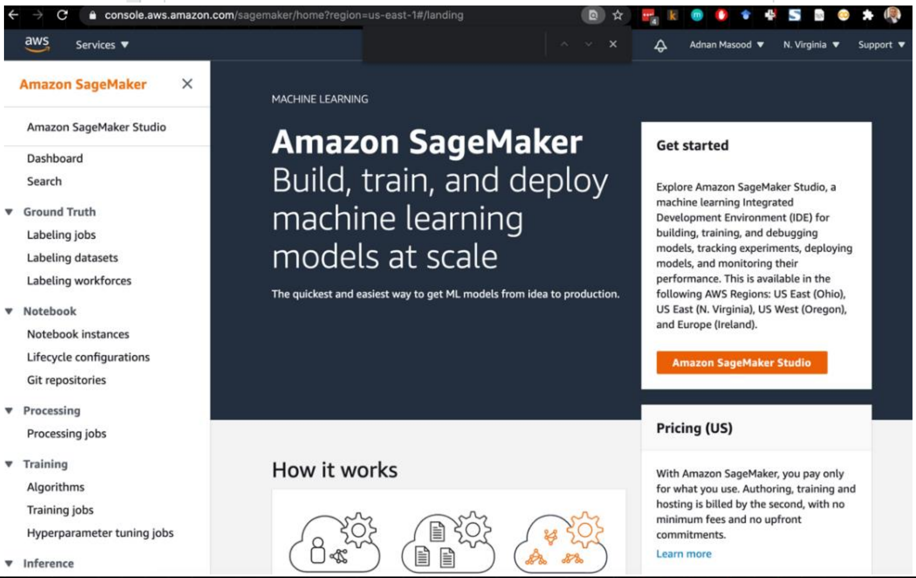

Classification
•Linear Learner
•XGBoost
•KNN
Working with Text
•BlazingText
•Supervised
•Unsupervised
Regression
•Linear Learner
•XGBoost
Computer Vision
•Image Classification
•Object Detection
•Semantic Segmentation
Recommendation
•Factorization Machines
Anomaly Detection
•Random Cut Forests
•IP Insights
Sequence Translation
•Seq2seq
Topic Modeling
•LDA
•NTM
Forecasting
•DeepAR
Clustering
•K-Means
Feature Reduction
•PCA
•Object2Vec
Prebuilt Notebooks for Common Problems
Built-In High-Performance Algorithms
One-Click Training & Deployment
Optimization – Training & Tuning of Models
Fully Managed with Auto Scaling, Health Checks, Automatic Handling of Node Failures, and Security Checks

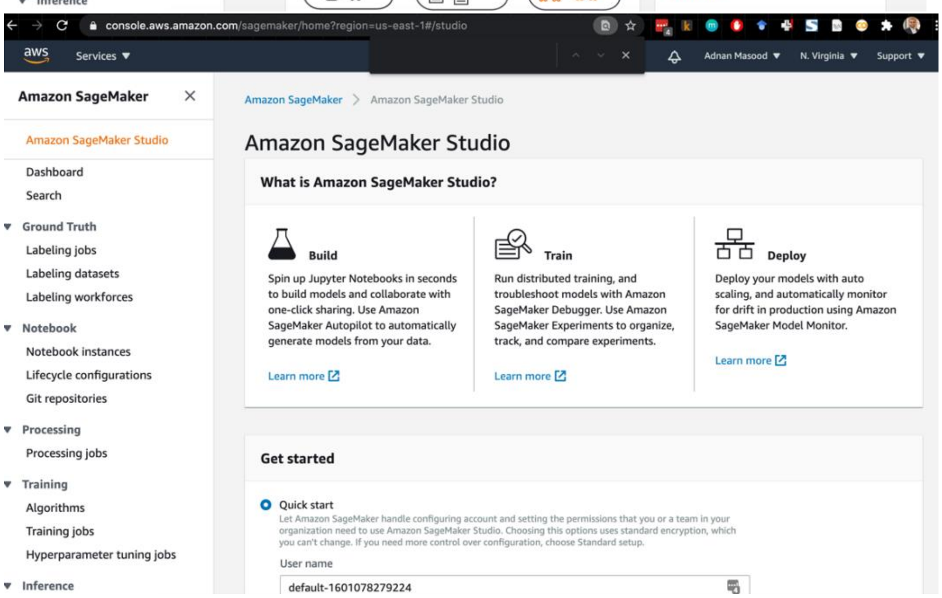

aws
Contact Sales
Support
English
My Account
Sign In to the Console
Products
Solutions
Pricing
Documentation
Learn
Partner Network
AWS Marketplace
Customer Enablement
Events
Explore More
AWS Management Console
Overview
Features
FAQs
AWS Management Console
Everything you need to access and manage the AWS cloud — in one web interface.
Log back in
Explore more from AWS
Amazon EC2 Spot Instances
Run Kubernetes, Apache Spark, and other fault-tolerant workloads for up to 90% off.
Learn more >>
Amazon EFS for AWS Lambda
Simplify file storage for serverless, and securely read, write and persist large data volumes at virtually any scale.
Learn more >>
AWS Amplify
Add authentication and data syncing with AWS Amplify in just a few lines of code.
Learn more >>
Amazon CodeGuru
Find code issues before they hit production, powered by machine learning.
Learn more >>

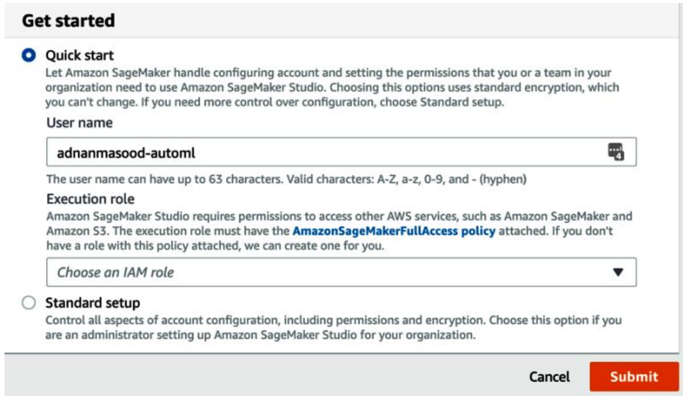

aws
Sign in
Root user
Account owner that performs tasks requiring unrestricted access. Learn more
IAM user
User within an account that performs daily tasks. Learn more
Root user email address
username@example.com
Next
New to AWS?
Create a new AWS account
Amazon CodeGuru
Find your most expensive lines of code with machine learning
aws
About Amazon.com Sign In
Amazon Web Services uses information from your Amazon.com account to identify you and allow access to Amazon Web Services. Your use of this site is governed by our Terms of Use and Privacy Policy linked below. Your use of Amazon Web Services products and services is governed by the AWS Customer Agreement linked below unless you have entered into a separate agreement with Amazon Web Services or an AWS Value Added Reseller to purchase these products and services. The AWS Customer Agreement was updated on March 31, 2017. For more information about these updates, see Recent Changes.

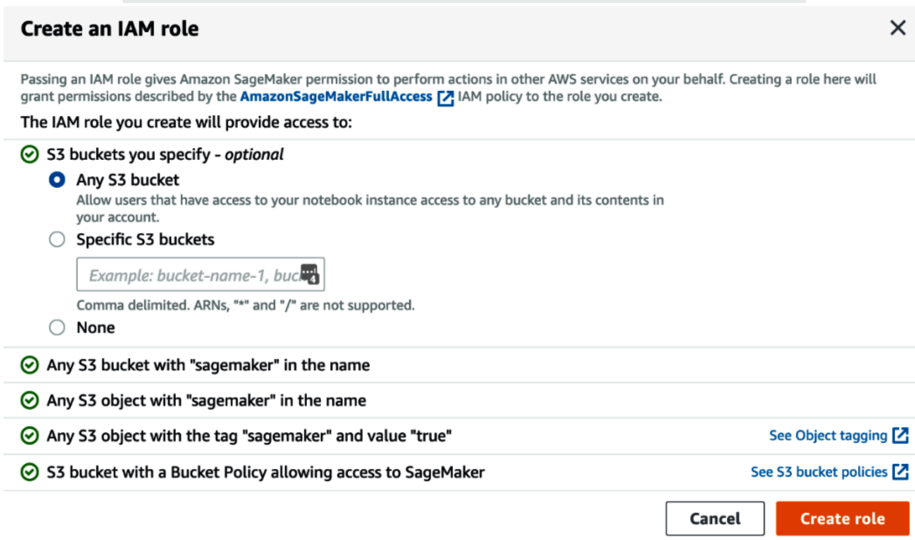

AWS Management Console
console.aws.amazon.com/console/home?nc2=h_ct&src=header-signin&region=us-east-1
aws
Services
Adnan Masood
N. Virginia
Support
AWS Management Console
AWS services
Find Services
You can enter names, keywords or acronyms.
Example: Relational Database Service, database, RDS
All services
Build a solution
Get started with simple wizards and automated workflows.
Launch a virtual machine
With EC2
2-3 minutes
Build a web app
With Elastic Beanstalk
6 minutes
Stay connected to your AWS resources on-the-go
Download the AWS Console Mobile App to your iOS or Android mobile device.
Learn more
Explore AWS
Amazon Elasticsearch Service
Fully managed Elasticsearch for log analytics, without the operational overhead. Learn more
RDS Read Replicas
Achieve scale and low-latency for read-heavy workloads with RDS Read Replicas. Learn more

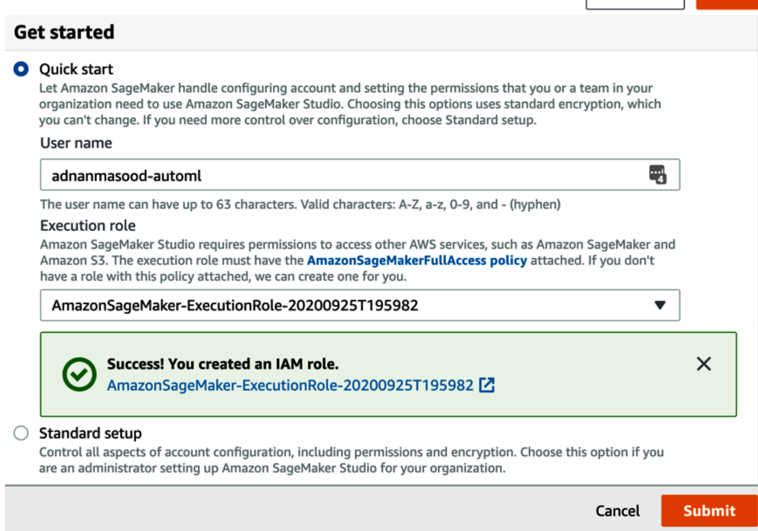

AWS Management Console
AWS services
Find Services
You can enter names, keywords or acronyms.
sage
Amazon SageMaker
Build, Train, and Deploy Machine Learning Models

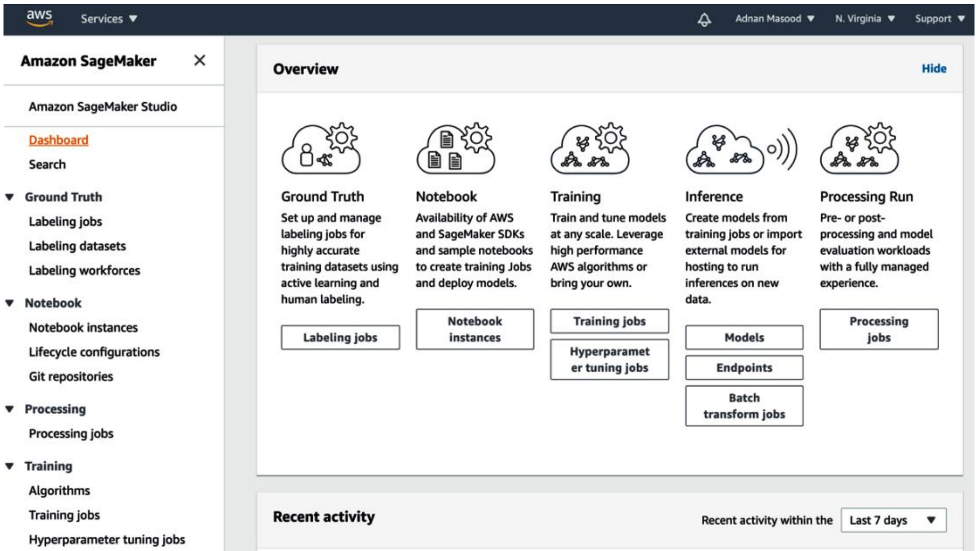

console.aws.amazon.com/sagemaker/home?region=us-east-1#/landing
Services
Adnan Masood
N. Virginia
Support
Amazon SageMaker
Amazon SageMaker Studio
Dashboard
Search
Ground Truth
Labeling jobs
Labeling datasets
Labeling workforces
Notebook
Notebook instances
Lifecycle configurations
Git repositories
Processing
Processing jobs
Training
Algorithms
Training jobs
Hyperparameter tuning jobs
Inference
MACHINE LEARNING
Amazon SageMaker
Build, train, and deploy machine learning models at scale
The quickest and easiest way to get ML models from idea to production.
Get started
Explore Amazon SageMaker Studio, a machine learning Integrated Development Environment (IDE) for building, training, and debugging models, tracking experiments, deploying models, and monitoring their performance. This is available in the following AWS Regions: US East (Ohio), US East (N. Virginia), US West (Oregon), and Europe (Ireland).
Amazon SageMaker Studio
Pricing (US)
With Amazon SageMaker, you pay only for what you use. Authoring, training and hosting is billed by the second, with no minimum fees and no upfront commitments.
Learn more
How it works

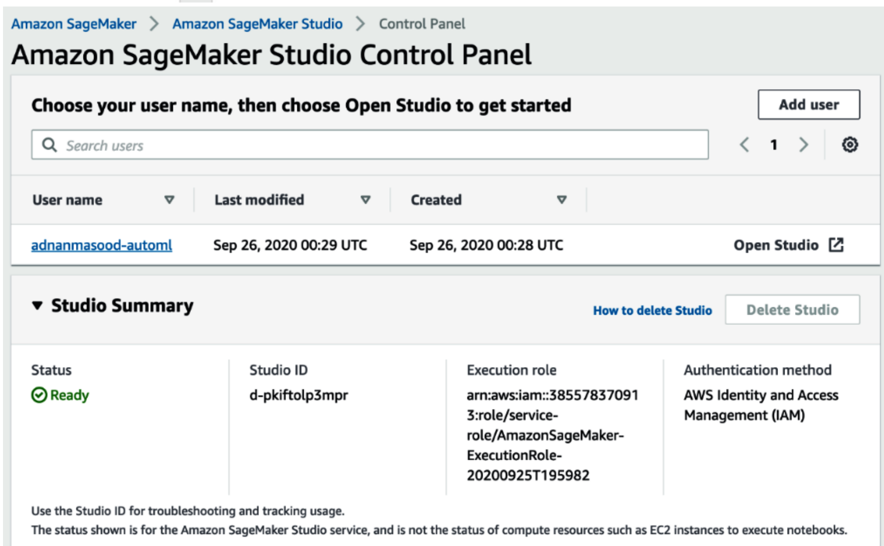

console.aws.amazon.com/sagemaker/home?region=us-east-1#/studio
Services
Adnan Masood
N. Virginia
Support
Amazon SageMaker
Amazon SageMaker Studio
Dashboard
Search
Ground Truth
Labeling jobs
Labeling datasets
Labeling workforces
Notebook
Notebook instances
Lifecycle configurations
Git repositories
Processing
Processing jobs
Training
Algorithms
Training jobs
Hyperparameter tuning jobs
Inference
Amazon SageMaker > Amazon SageMaker Studio
Amazon SageMaker Studio
What is Amazon SageMaker Studio?
Build
Spin up Jupyter Notebooks in seconds to build models and collaborate with one-click sharing. Use Amazon SageMaker Autopilot to automatically generate models from your data.
Learn more
Train
Run distributed training, and troubleshoot models with Amazon SageMaker Debugger. Use Amazon SageMaker Experiments to organize, track, and compare experiments.
Learn more
Deploy
Deploy your models with auto scaling, and automatically monitor for drift in production using Amazon SageMaker Model Monitor.
Learn more
Get started
Quick start
Let Amazon SageMaker handle configuring account and setting the permissions that you or a team in your organization need to use Amazon SageMaker Studio. Choosing this options uses standard encryption, which you can't change. If you need more control over configuration, choose Standard setup.
User name
default-1601078279224

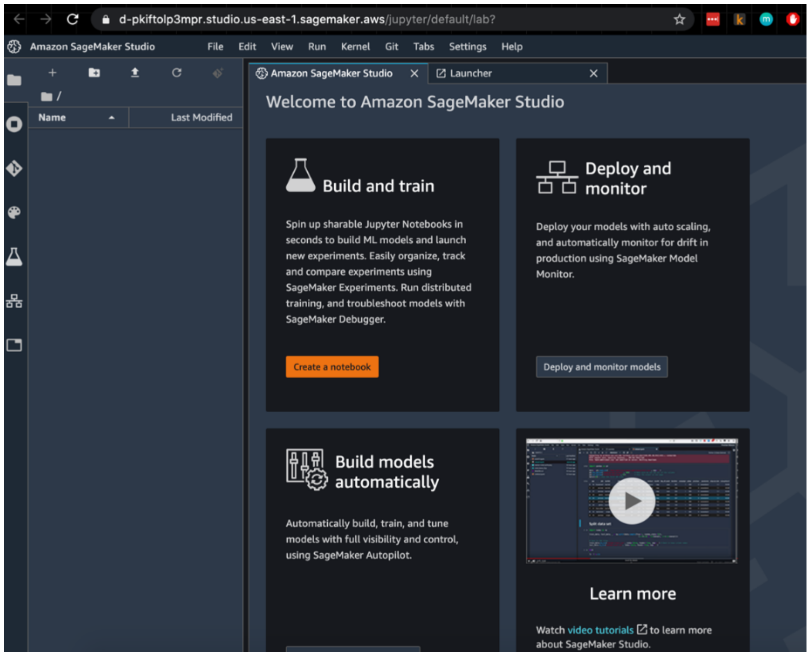

Get started
Quick start
Let Amazon SageMaker handle configuring account and setting the permissions that you or a team in your organization need to use Amazon SageMaker Studio. Choosing this options uses standard encryption, which you can't change. If you need more control over configuration, choose Standard setup.
User name
adnanmasood-automl
The user name can have up to 63 characters. Valid characters: A-Z, a-z, 0-9, and - (hyphen)
Execution role
Amazon SageMaker Studio requires permissions to access other AWS services, such as Amazon SageMaker and Amazon S3. The execution role must have the AmazonSageMakerFullAccess policy attached. If you don't have a role with this policy attached, we can create one for you.
Choose an IAM role
Standard setup
Control all aspects of account configuration, including permissions and encryption. Choose this option if you are an administrator setting up Amazon SageMaker Studio for your organization.
Cancel
Submit


Create an IAM role
Passing an IAM role gives Amazon SageMaker permission to perform actions in other AWS services on your behalf. Creating a role here will grant permissions described by the AmazonSageMakerFullAccess IAM policy to the role you create.
The IAM role you create will provide access to:
S3 buckets you specify - optional
Any S3 bucket
Allow users that have access to your notebook instance access to any bucket and its contents in your account.
Specific S3 buckets
Example: bucket-name-1, buc
Comma delimited. ARNs, "*" and "/" are not supported.
None
Any S3 bucket with "sagemaker" in the name
Any S3 object with "sagemaker" in the name
Any S3 object with the tag "sagemaker" and value "true"
See Object tagging
S3 bucket with a Bucket Policy allowing access to SageMaker
See S3 bucket policies
Cancel
Create role

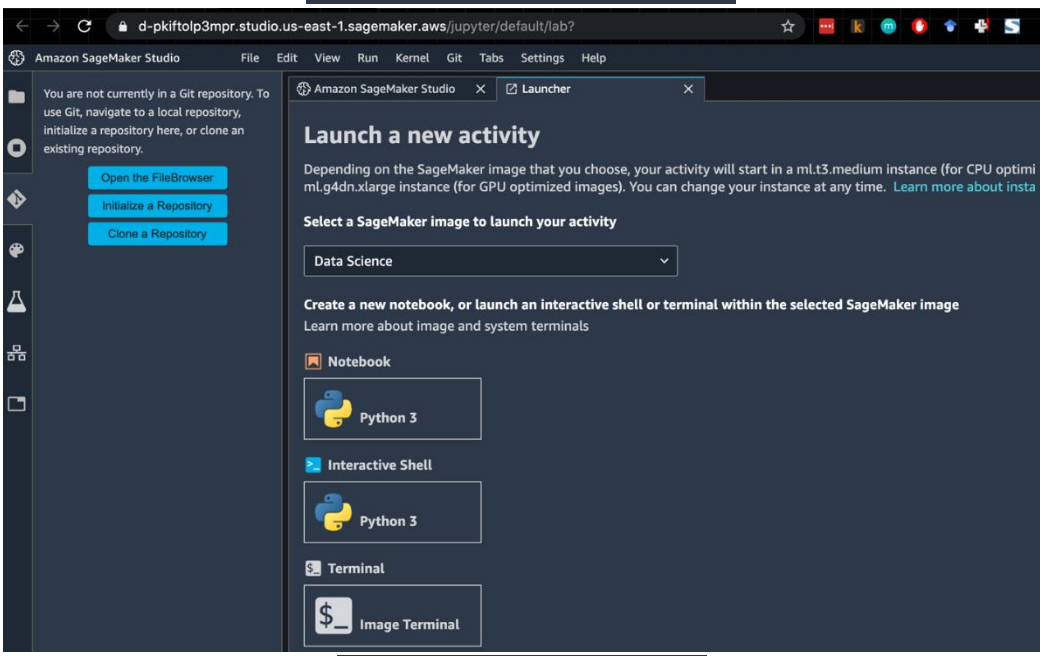

Get started
Quick start
Let Amazon SageMaker handle configuring account and setting the permissions that you or a team in your organization need to use Amazon SageMaker Studio. Choosing this options uses standard encryption, which you can't change. If you need more control over configuration, choose Standard setup.
User name
adnanmasood-automl
The user name can have up to 63 characters. Valid characters: A-Z, a-z, 0-9, and - (hyphen)
Execution role
Amazon SageMaker Studio requires permissions to access other AWS services, such as Amazon SageMaker and Amazon S3. The execution role must have the AmazonSageMakerFullAccess policy attached. If you don't have a role with this policy attached, we can create one for you.
AmazonSageMaker-ExecutionRole-20200925T195982
Success! You created an IAM role.
AmazonSageMaker-ExecutionRole-20200925T195982
Standard setup
Control all aspects of account configuration, including permissions and encryption. Choose this option if you are an administrator setting up Amazon SageMaker Studio for your organization.
Cancel
Submit

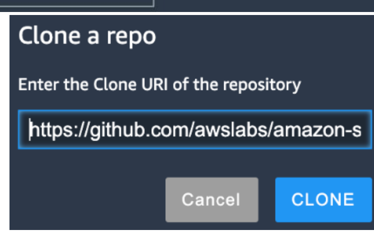

aws
Services
Adnan Masood
N. Virginia
Support
Amazon SageMaker
Amazon SageMaker Studio
Dashboard
Search
Ground Truth
Labeling jobs
Labeling datasets
Labeling workforces
Notebook
Notebook instances
Lifecycle configurations
Git repositories
Processing
Processing jobs
Training
Algorithms
Training jobs
Hyperparameter tuning jobs
Overview
Hide
Ground Truth
Set up and manage labeling jobs for highly accurate training datasets using active learning and human labeling.
Labeling jobs
Notebook
Availability of AWS and SageMaker SDKs and sample notebooks to create training Jobs and deploy models.
Notebook instances
Training
Train and tune models at any scale. Leverage high performance AWS algorithms or bring your own.
Training jobs
Hyperparameter tuning jobs
Inference
Create models from training jobs or import external models for hosting to run inferences on new data.
Models
Endpoints
Batch transform jobs
Processing Run
Pre- or post-processing and model evaluation workloads with a fully managed experience.
Processing jobs
Recent activity
Recent activity within the
Last 7 days

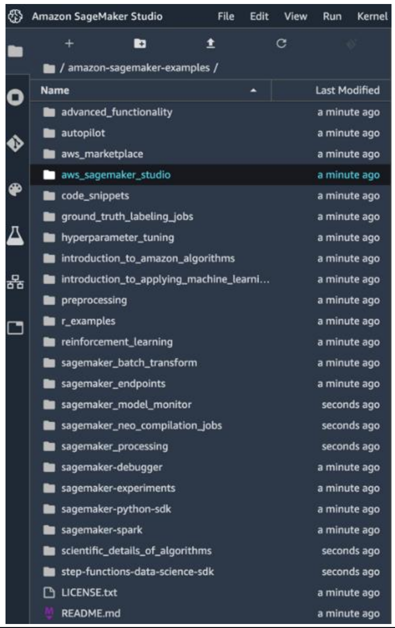

Amazon SageMaker > Amazon SageMaker Studio > Control Panel
Amazon SageMaker Studio Control Panel
Choose your user name, then choose Open Studio to get started
Add user
Search users
1
User name
Last modified
Created
adnanmasood-automl
Sep 26, 2020 00:29 UTC
Sep 26, 2020 00:28 UTC
Open Studio
Studio Summary
How to delete Studio
Delete Studio
Status
Ready
Studio ID
d-pkiftolp3mpr
Execution role
arn:aws:iam::385578370913:role/service-role/AmazonSageMaker-ExecutionRole-20200925T195982
Authentication method
AWS Identity and Access Management (IAM)
Use the Studio ID for troubleshooting and tracking usage.
The status shown is for the Amazon SageMaker Studio service, and is not the status of compute resources such as EC2 instances to execute notebooks.

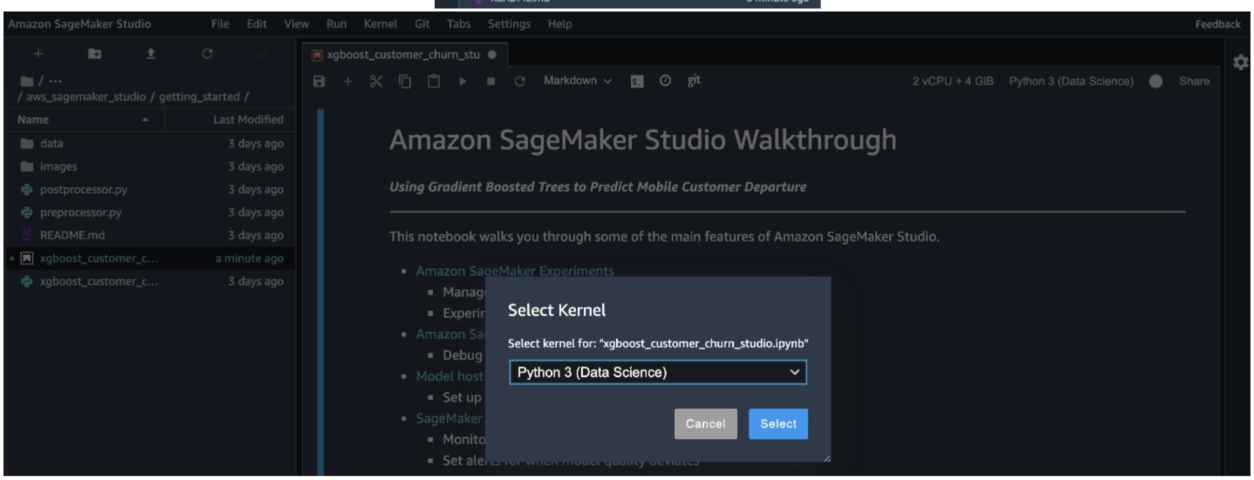

d-pkiftolp3mpr.studio.us-east-1.sagemaker.aws/jupyter/default/lab?
Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
Name
Last Modified
Amazon SageMaker Studio
Launcher
Welcome to Amazon SageMaker Studio
Build and train
Spin up sharable Jupyter Notebooks in seconds to build ML models and launch new experiments. Easily organize, track and compare experiments using SageMaker Experiments. Run distributed training, and troubleshoot models with SageMaker Debugger.
Create a notebook
Deploy and monitor
Deploy your models with auto scaling, and automatically monitor for drift in production using SageMaker Model Monitor.
Deploy and monitor models
Build models automatically
Automatically build, train, and tune models with full visibility and control, using SageMaker Autopilot.
Learn more
Watch video tutorials to learn more about SageMaker Studio.

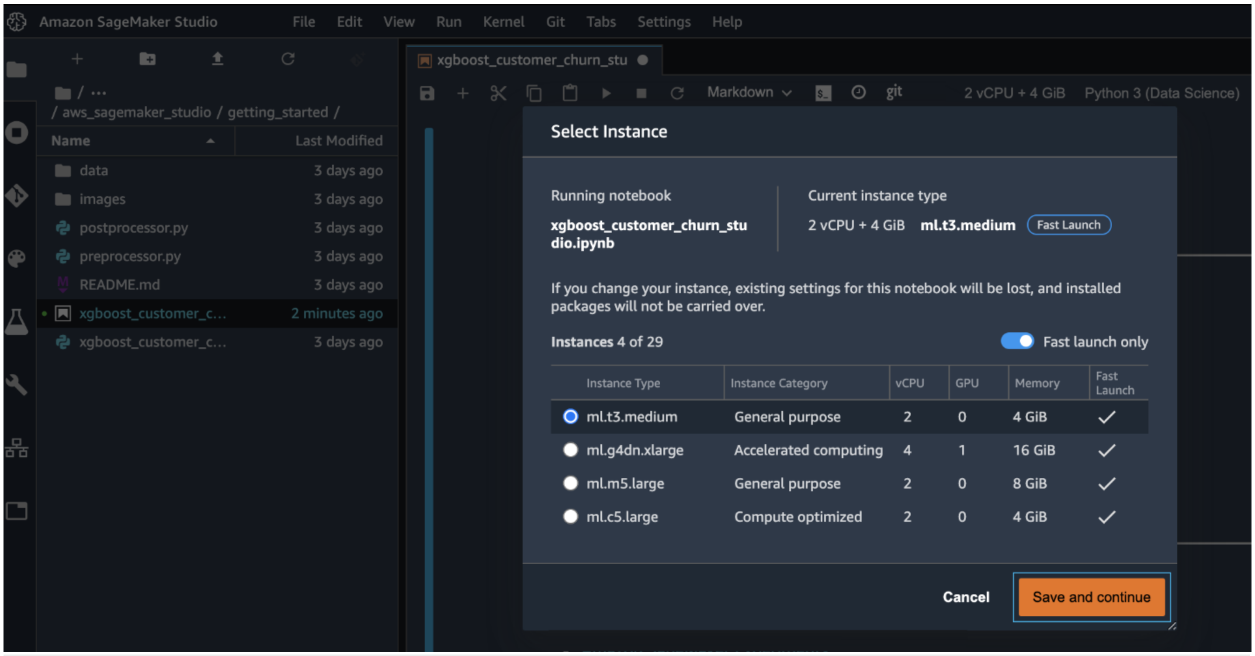

Amazon SageMaker Studio
Loading the JupyterServer application default...


d-pkiftolp3mpr.studio.us-east-1.sagemaker.aws/jupyter/default/lab?
Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
You are not currently in a Git repository. To use Git, navigate to a local repository, initialize a repository here, or clone an existing repository.
Open the FileBrowser
Initialize a Repository
Clone a Repository
Amazon SageMaker Studio
Launcher
Launch a new activity
Depending on the SageMaker image that you choose, your activity will start in a ml.t3.medium instance (for CPU optimi
ml.g4dn.xlarge instance (for GPU optimized images). You can change your instance at any time. Learn more about insta
Select a SageMaker image to launch your activity
Data Science
Create a new notebook, or launch an interactive shell or terminal within the selected SageMaker image
Learn more about image and system terminals
Notebook
Python 3
Interactive Shell
Python 3
Terminal
Image Terminal

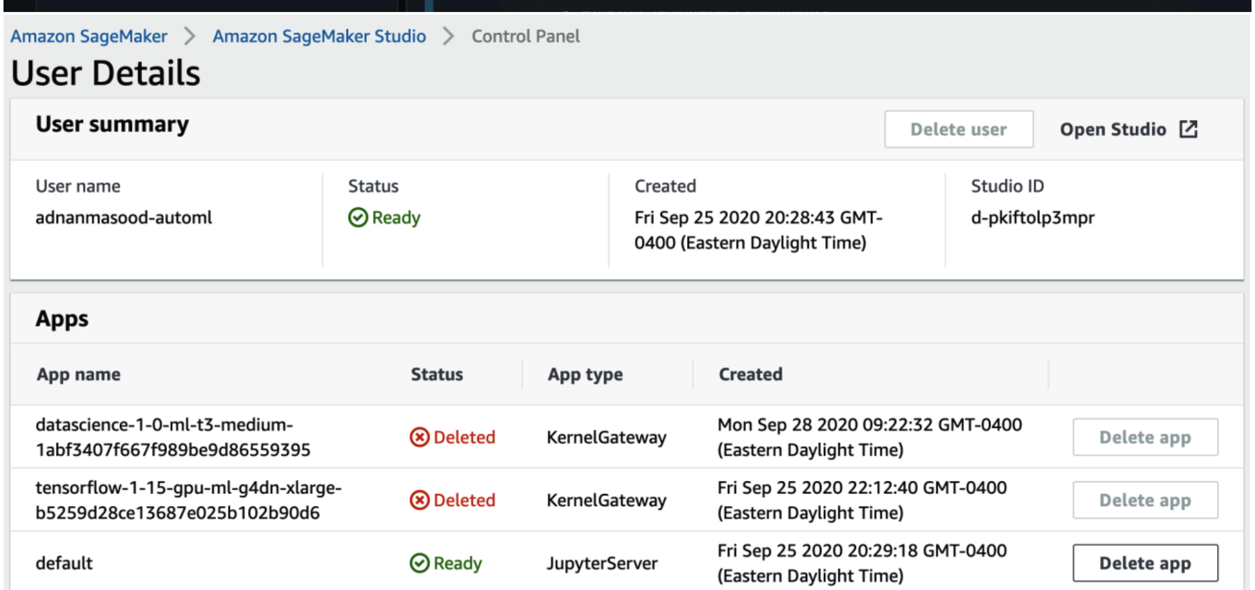

Clone a repo
Enter the Clone URI of the repository
https://github.com/awslabs/amazon-s
Cancel
CLONE


Amazon SageMaker Studio
File
Edit
View
Run
Kernel
/ amazon-sagemaker-examples /
Name
Last Modified
advanced_functionality
autopilot
aws_marketplace
aws_sagemaker_studio
code_snippets
ground_truth_labeling_jobs
hyperparameter_tuning
introduction_to_amazon_algorithms
introduction_to_applying_machine_learni...
preprocessing
r_examples
reinforcement_learning
sagemaker_batch_transform
sagemaker_endpoints
sagemaker_model_monitor
sagemaker_neo_compilation_jobs
sagemaker_processing
sagemaker-debugger
sagemaker-experiments
sagemaker-python-sdk
sagemaker-spark
scientific_details_of_algorithms
step-functions-data-science-sdk
LICENSE.txt
README.md
a minute ago
seconds ago

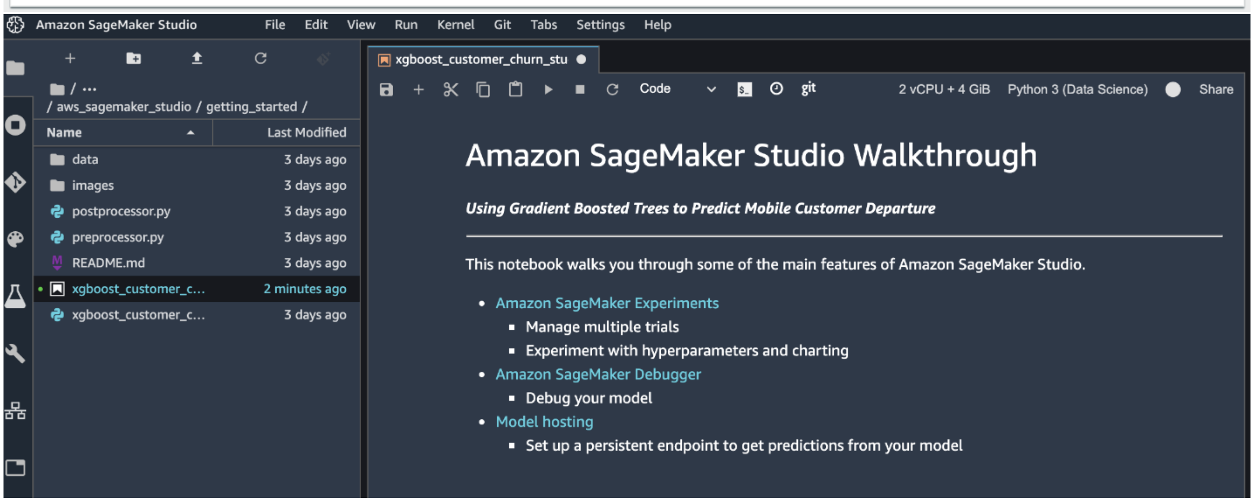

Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
Feedback
/ aws_sagemaker_studio / getting_started /
Name
Last Modified
data
images
postprocessor.py
preprocessor.py
README.md
xgboost_customer_c...
3 days ago
a minute ago
xgboost_customer_churn_stu
Markdown
git
2 vCPU + 4 GiB
Python 3 (Data Science)
Share
Amazon SageMaker Studio Walkthrough
Using Gradient Boosted Trees to Predict Mobile Customer Departure
This notebook walks you through some of the main features of Amazon SageMaker Studio.
Amazon SageMaker Experiments
Select Kernel
Select kernel for: "xgboost_customer_churn_studio.ipynb"
Python 3 (Data Science)
Cancel
Select



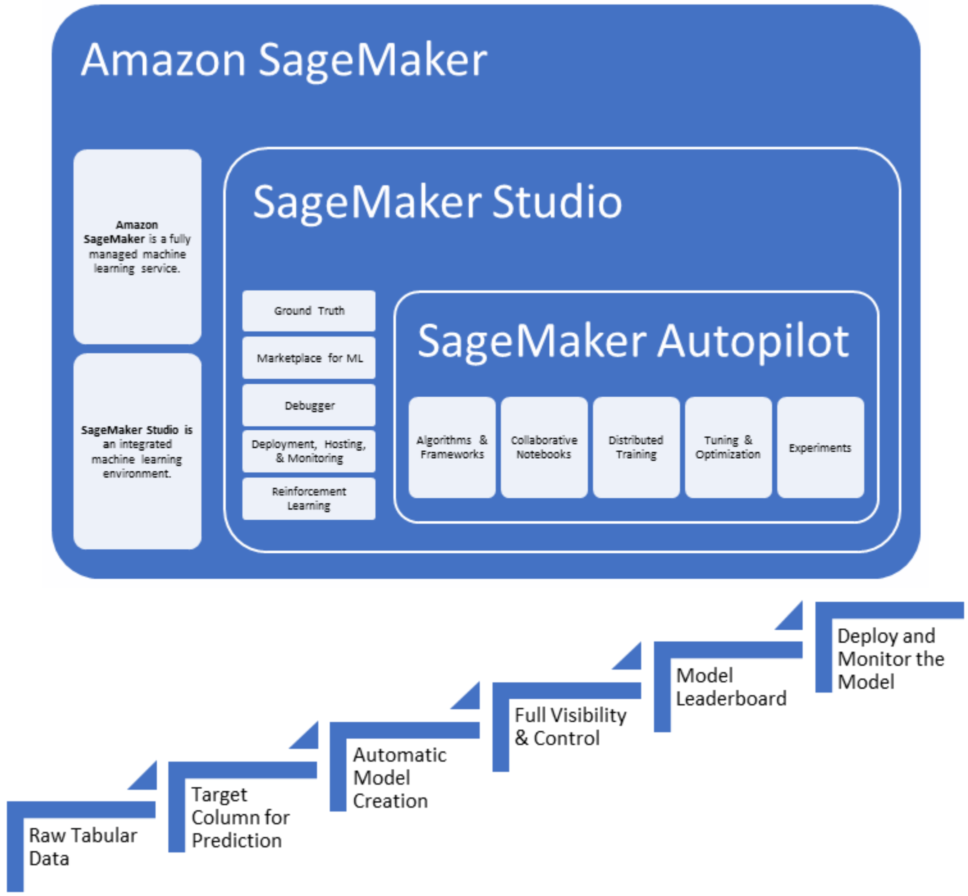
Amazon SageMaker Studio
File Edit View Run Kernel Git Tabs Settings Help
/ aws_sagemaker_studio / getting_started /
Name
Last Modified
data 3 days ago
images 3 days ago
postprocessor.py 3 days ago
preprocessor.py 3 days ago
README.md 3 days ago
xgboost_customer_c... 2 minutes ago
xgboost_customer_c... 3 days ago
xgboost_customer_churn_stu
Markdown
2 vCPU + 4 GiB Python 3 (Data Science)
Select Instance
Running notebook
xgboost_customer_churn_studio.ipynb
Current instance type
2 vCPU + 4 GiB ml.t3.medium Fast Launch
If you change your instance, existing settings for this notebook will be lost, and installed packages will not be carried over.
Instances 4 of 29
Fast launch only
Instance Type Instance Category vCPU GPU Memory Fast Launch
ml.t3.medium General purpose 2 0 4 GiB
ml.g4dn.xlarge Accelerated computing 4 1 16 GiB
ml.m5.large General purpose 2 0 8 GiB
ml.c5.large Compute optimized 2 0 4 GiB
Cancel
Save and continue

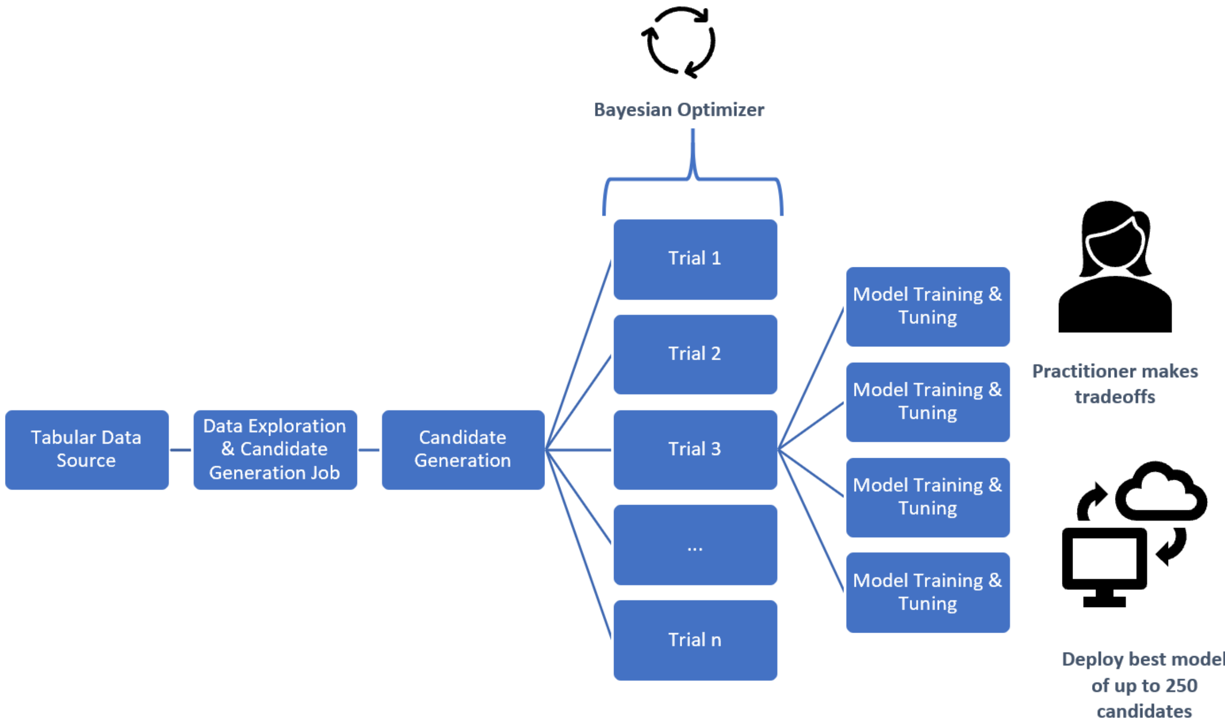


Amazon SageMaker > Amazon SageMaker Studio > Control Panel
User Details
User summary
Delete user
Open Studio
User name
adnanmasood-automl
Status
Ready
Created
Fri Sep 25 2020 20:28:43 GMT-0400 (Eastern Daylight Time)
Studio ID
d-pkiftolp3mpr
Apps
App name Status App type Created
datascience-1-0-ml-t3-medium-1abf3407f667f989be9d86559395 Deleted KernelGateway Mon Sep 28 2020 09:22:32 GMT-0400 (Eastern Daylight Time) Delete app
tensorflow-1-15-gpu-ml-g4dn-xlarge-b5259d28ce13687e025b102b90d6 Deleted KernelGateway Fri Sep 25 2020 22:12:40 GMT-0400 (Eastern Daylight Time) Delete app
default Ready JupyterServer Fri Sep 25 2020 20:29:18 GMT-0400 (Eastern Daylight Time) Delete app



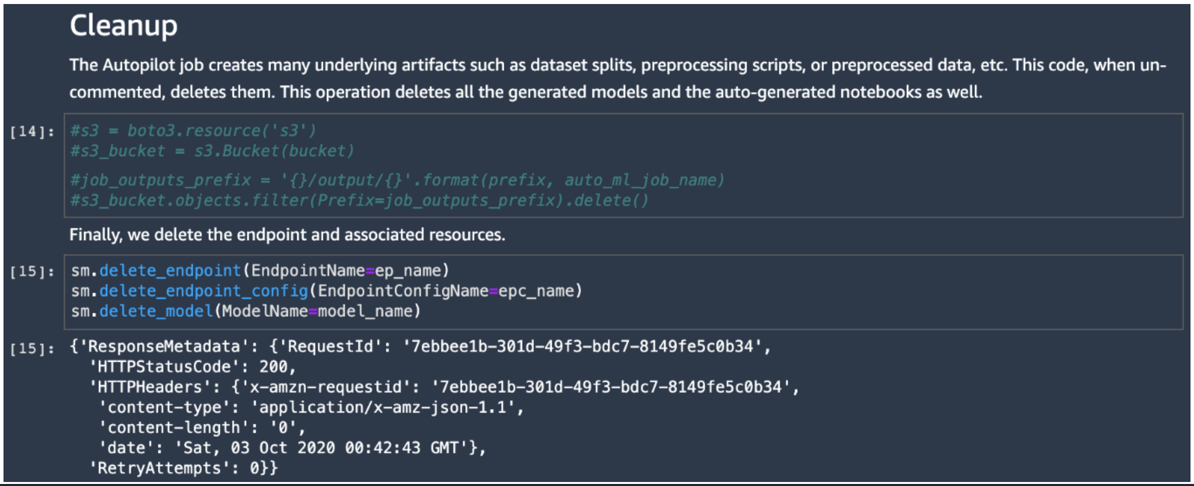
Amazon SageMaker Studio
File Edit View Run Kernel Git Tabs Settings Help
/ aws_sagemaker_studio / getting_started /
Name
Last Modified
data 3 days ago
images 3 days ago
postprocessor.py 3 days ago
preprocessor.py 3 days ago
README.md 3 days ago
xgboost_customer_c... 2 minutes ago
xgboost_customer_c... 3 days ago
xgboost_customer_churn_stu
Code
2 vCPU + 4 GiB Python 3 (Data Science) Share
Amazon SageMaker Studio Walkthrough
Using Gradient Boosted Trees to Predict Mobile Customer Departure
This notebook walks you through some of the main features of Amazon SageMaker Studio.
Amazon SageMaker Experiments
Manage multiple trials
Experiment with hyperparameters and charting
Amazon SageMaker Debugger
Debug your model
Model hosting
Set up a persistent endpoint to get predictions from your model

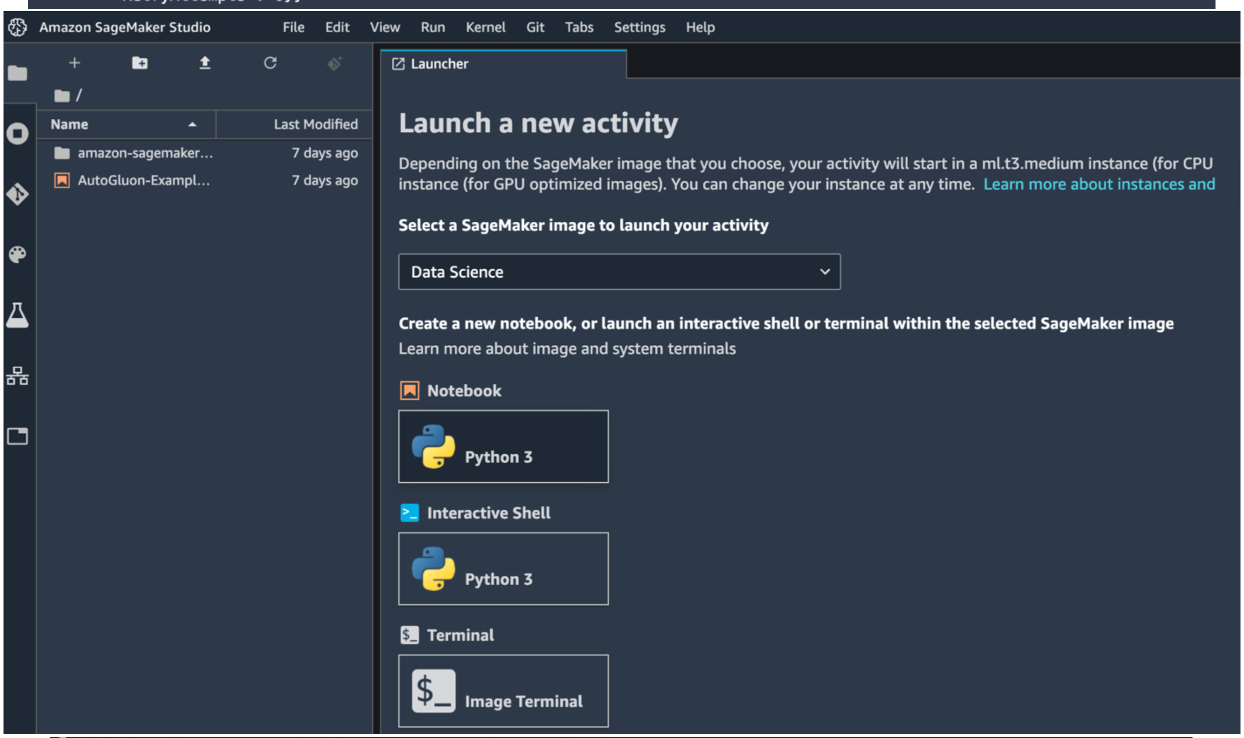

Amazon SageMaker
Amazon SageMaker is a fully managed machine learning service.
SageMaker Studio is an integrated machine learning environment.
SageMaker Studio
Ground Truth
Marketplace for ML
Debugger
Deployment, Hosting, & Monitoring
Reinforcement Learning
SageMaker Autopilot
Algorithms & Frameworks
Collaborative Notebooks
Distributed Training
Tuning & Optimization
Experiments
Raw Tabular Data
Target Column for Prediction
Automatic Model Creation
Full Visibility & Control
Model Leaderboard
Deploy and Monitor the Model

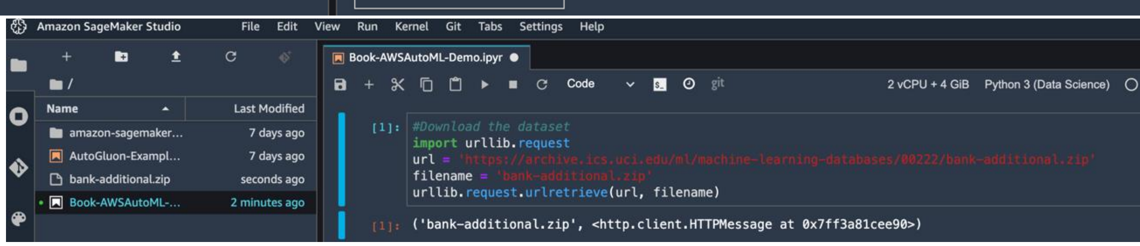


#	Model	Accuracy	Latency	Model Size
1	churn-xgboost-1756-013-33398f0	95%	450 ms	9.1 MB
2	churn-xgboost-1756-014-53facc2	93%	200 ms	4.8 MB
3	churn-xgboost-1756-015-58bc692	92%	200 ms	4.3 MB
4	churn-linear-1756-016-db54598	91%	50 ms	1.3 MB
5	churn-xgboost-1756-017-af8d756	91%	190 ms	4.2 MB

Chapter 7: Doing Automated Machine Learning with Amazon SageMaker Autopilot







Bank Marketing Data Set

Download: Data Folder, Data Set Description

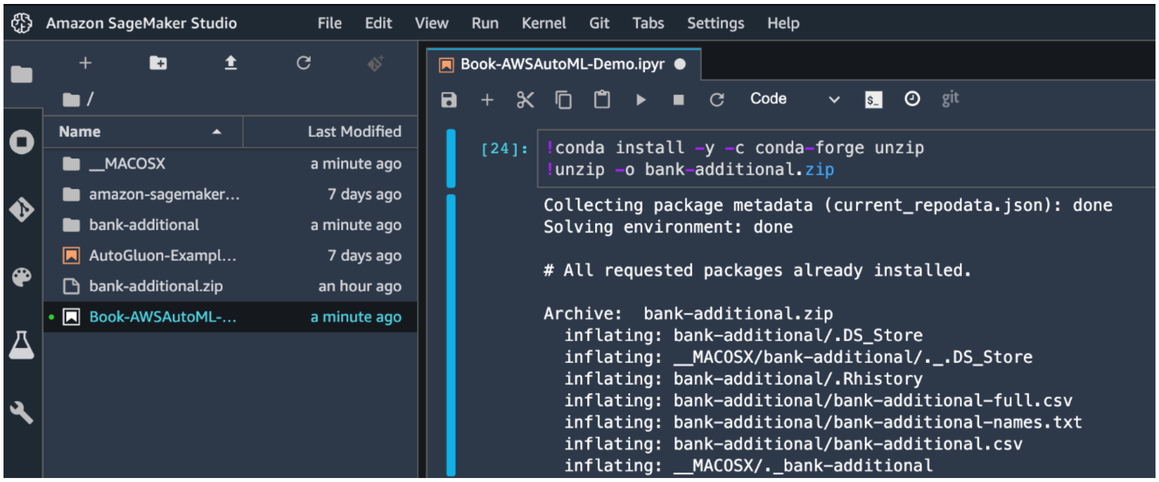
Abstract: The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe a term deposit (variable y).

Data Set Characteristics:	Multivariate	**Number of Instances:**	45211	**Area:**	Business
Attribute Characteristics:	Real	**Number of Attributes:**	17	**Date Donated**	2012-02-14
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	1285737

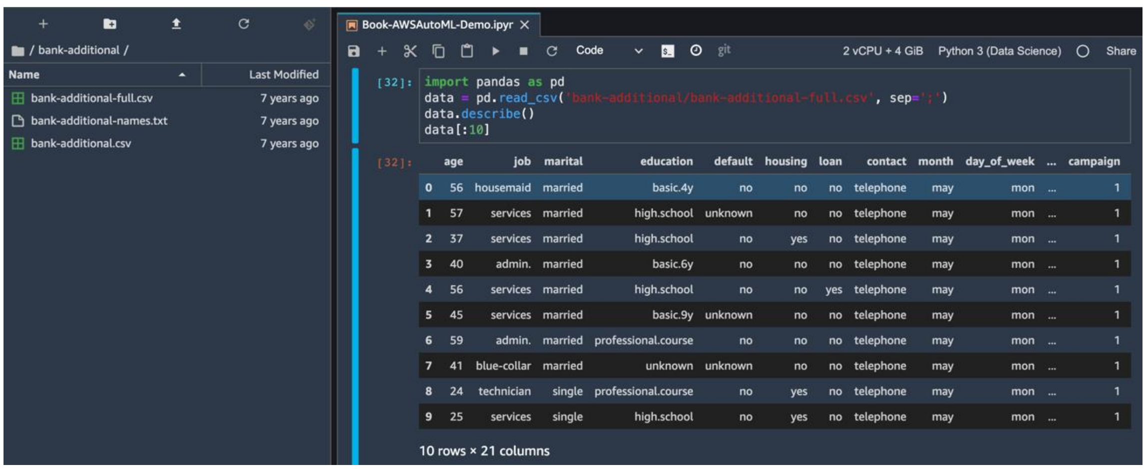
Attribute Information:

Input variables:
bank client data:
1 - age (numeric)
2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')
3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)
4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')
5 - default: has credit in default? (categorical: 'no','yes','unknown')
6 - housing: has housing loan? (categorical: 'no','yes','unknown')
7 - loan: has personal loan? (categorical: 'no','yes','unknown')
related with the last contact of the current campaign:
8 - contact: contact communication type (categorical: 'cellular','telephone')
9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
10 - day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')
11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.
other attributes:
12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)
14 - previous: number of contacts performed before this campaign and for this client (numeric)
15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')
social and economic context attributes
16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)
17 - cons.price.idx: consumer price index - monthly indicator (numeric)
18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)
19 - euribor3m: euribor 3 month rate - daily indicator (numeric)
20 - nr.employed: number of employees - quarterly indicator (numeric)

Output variable (desired target):
21 - y - has the client subscribed a term deposit? (binary: 'yes','no')



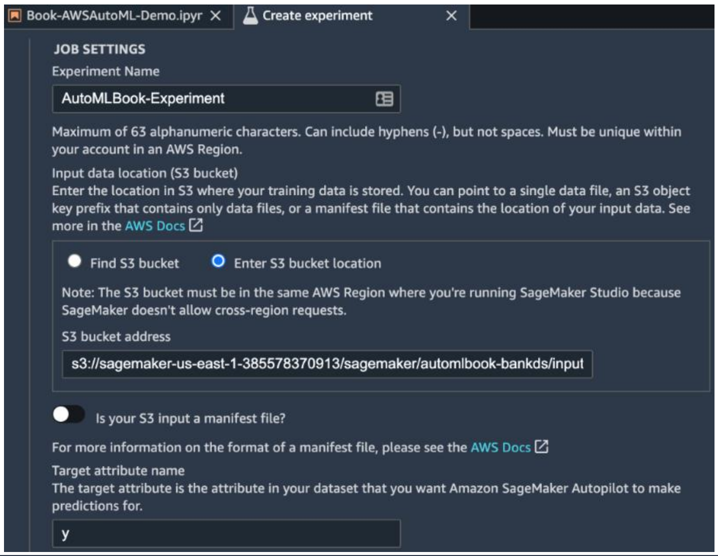
Amazon SageMaker Studio
File Edit View Run Kernel Git Tabs Settings Help
Book-AWSAutoML-Demo.ipyr
Code
git
Name
Last Modified
__MACOSX a minute ago
amazon-sagemaker… 7 days ago
bank-additional a minute ago
AutoGluon-Exampl… 7 days ago
bank-additional.zip an hour ago
Book-AWSAutoML-… a minute ago
[24]:
!conda install -y -c conda-forge unzip
!unzip -o bank-additional.zip
Collecting package metadata (current_repodata.json): done
Solving environment: done
All requested packages already installed.
Archive: bank-additional.zip
inflating: bank-additional/.DS_Store
inflating: __MACOSX/bank-additional/._.DS_Store
inflating: bank-additional/.Rhistory
inflating: bank-additional/bank-additional-full.csv
inflating: bank-additional/bank-additional-names.txt
inflating: bank-additional/bank-additional.csv
inflating: __MACOSX/._bank-additional

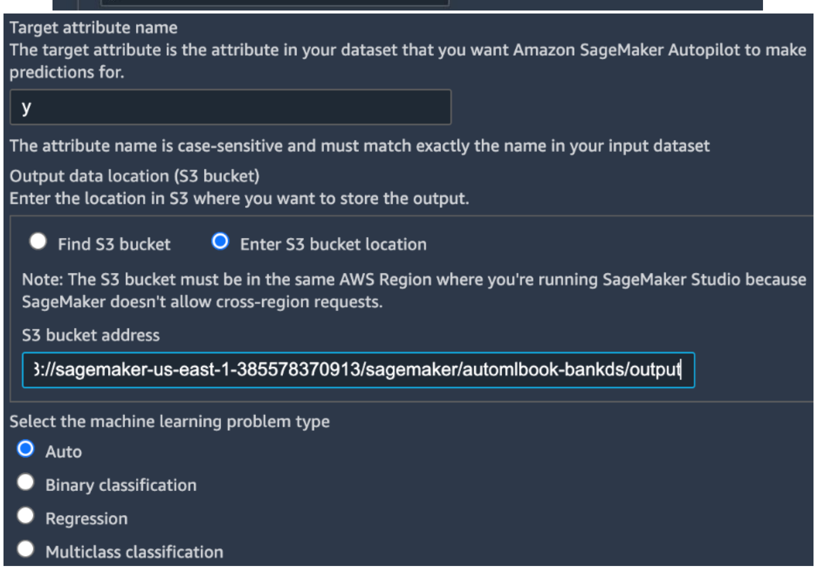


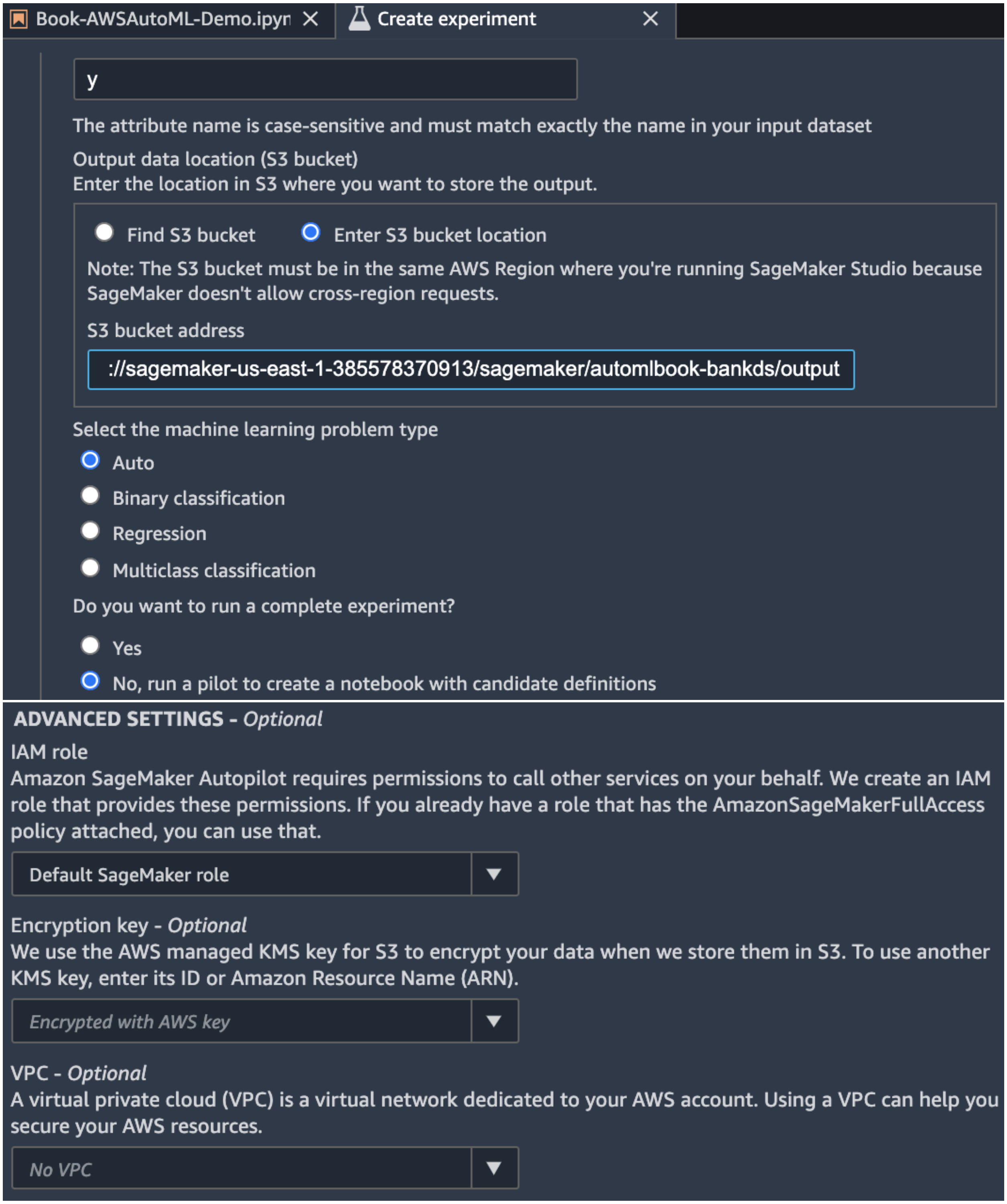
/ bank-additional /
Name
Last Modified
bank-additional-full.csv 7 years ago
bank-additional-names.txt 7 years ago
bank-additional.csv 7 years ago
Book-AWSAutoML-Demo.ipyr
Code
git
2 vCPU + 4 GiB
Python 3 (Data Science)
Share
[32]:
import pandas as pd
data = pd.read_csv('bank-additional/bank-additional-full.csv', sep=';')
data.describe()
data[:10]
[32]:
age job marital education default housing loan contact month day_of_week ... campaign
0 56 housemaid married basic.4y no no no telephone may mon ... 1
1 57 services married high.school unknown no no telephone may mon ... 1
2 37 services married high.school no yes no telephone may mon ... 1
3 40 admin. married basic.6y no no no telephone may mon ... 1
4 56 services married high.school no no yes telephone may mon ... 1
5 45 services married basic.9y unknown no no telephone may mon ... 1
6 59 admin. married professional.course no no no telephone may mon ... 1
7 41 blue-collar married unknown unknown no no telephone may mon ... 1
8 24 technician single professional.course no yes no telephone may mon ... 1
9 25 services single high.school no yes no telephone may mon ... 1
10 rows × 21 columns

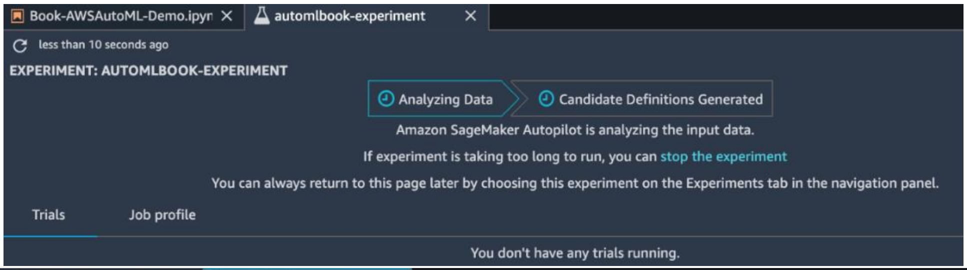
automl-test.csv seconds ago
automl-train.csv seconds ago
bank-additional.zip an hour ago
Book-AWSAutoML-Demo.ipynb seconds ago
[36]:
import numpy as np
train_data, test_data, _ = np.split(data.sample(frac=1, random_state=123),
[int(0.95 * len(data)), int(len(data))])
train_data.to_csv('automl-train.csv', index=False, header=True, sep=',')
test_data.to_csv('automl-test.csv', index=False, header=True, sep=',')

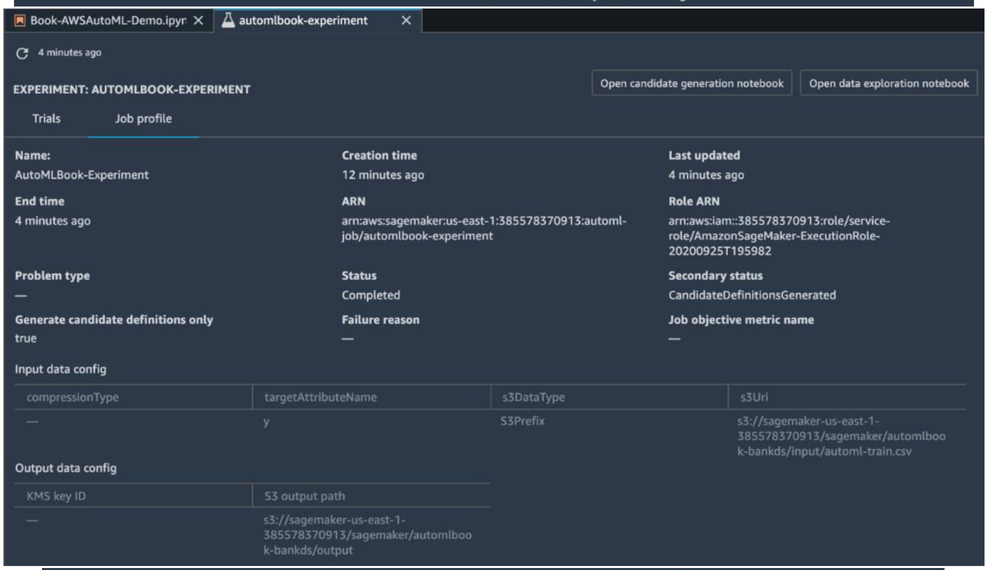
```
[37]: import sagemaker
      prefix = 'sagemaker/automlbook-bankds/input'
      sess   = sagemaker.Session()
      uri = sess.upload_data(path="automl-train.csv", key_prefix=prefix)
      print(uri)

      s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/input/automl-train.csv
```

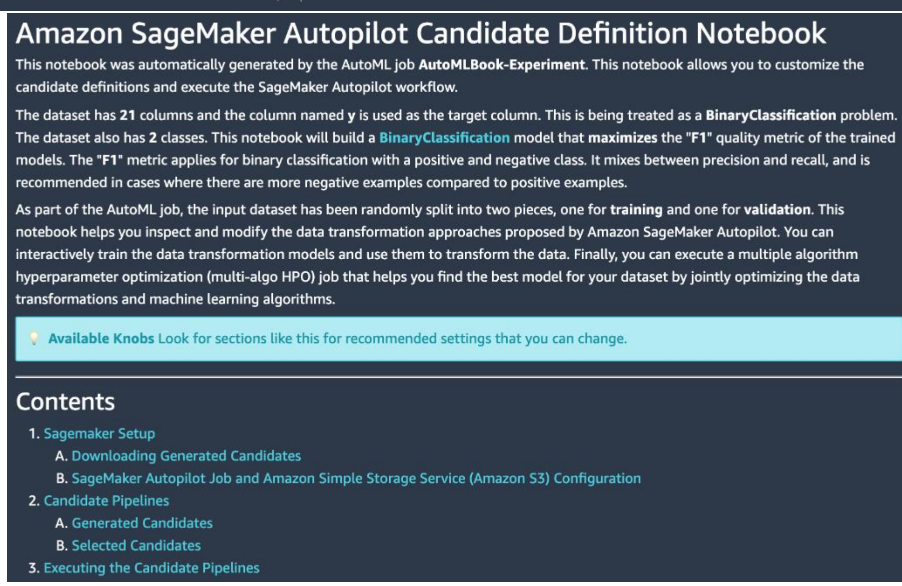





Book-AWSAutoML-Demo.ipyn
Create experiment
y
The attribute name is case-sensitive and must match exactly the name in your input dataset
Output data location (S3 bucket)
Enter the location in S3 where you want to store the output.
Find S3 bucket
Enter S3 bucket location
Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.
S3 bucket address
://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/output
Select the machine learning problem type
Auto
Binary classification
Regression
Multiclass classification
Do you want to run a complete experiment?
Yes
No, run a pilot to create a notebook with candidate definitions
ADVANCED SETTINGS - Optional
IAM role
Amazon SageMaker Autopilot requires permissions to call other services on your behalf. We create an IAM role that provides these permissions. If you already have a role that has the AmazonSageMakerFullAccess policy attached, you can use that.
Default SageMaker role
Encryption key - Optional
We use the AWS managed KMS key for S3 to encrypt your data when we store them in S3. To use another KMS key, enter its ID or Amazon Resource Name (ARN).
Encrypted with AWS key
VPC - Optional
A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. Using a VPC can help you secure your AWS resources.
No VPC


Book-AWSAutoML-Demo.ipyn
automlbook-experiment
less than 10 seconds ago
EXPERIMENT: AUTOMLBOOK-EXPERIMENT
Analyzing Data
Candidate Definitions Generated
Amazon SageMaker Autopilot is analyzing the input data.
If experiment is taking too long to run, you can stop the experiment
You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.
Trials
Job profile
You don't have any trials running.


Book-AWSAutoML-Demo.ipyn
automlbook-experiment
4 minutes ago
EXPERIMENT: AUTOMLBOOK-EXPERIMENT
Open candidate generation notebook
Open data exploration notebook
Trials
Job profile
Name:
AutoMLBook-Experiment
Creation time
12 minutes ago
Last updated
4 minutes ago
End time
4 minutes ago
ARN
arn:aws:sagemaker:us-east-1:385578370913:automl-job/automlbook-experiment
Role ARN
arn:aws:iam::385578370913:role/service-role/AmazonSageMaker-ExecutionRole-20200925T195982
Problem type
—
Status
Completed
Secondary status
CandidateDefinitionsGenerated
Generate candidate definitions only
true
Failure reason
—
Job objective metric name
—
Input data config
compressionType
targetAttributeName
s3DataType
s3Uri
—
y
S3Prefix
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/input/automl-train.csv
Output data config
KMS key ID
S3 output path
—
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/output


Amazon SageMaker Autopilot Candidate Definition Notebook
This notebook was automatically generated by the AutoML job AutoMLBook-Experiment. This notebook allows you to customize the candidate definitions and execute the SageMaker Autopilot workflow.
The dataset has 21 columns and the column named y is used as the target column. This is being treated as a BinaryClassification problem. The dataset also has 2 classes. This notebook will build a BinaryClassification model that maximizes the "F1" quality metric of the trained models. The "F1" metric applies for binary classification with a positive and negative class. It mixes between precision and recall, and is recommended in cases where there are more negative examples compared to positive examples.
As part of the AutoML job, the input dataset has been randomly split into two pieces, one for training and one for validation. This notebook helps you inspect and modify the data transformation approaches proposed by Amazon SageMaker Autopilot. You can interactively train the data transformation models and use them to transform the data. Finally, you can execute a multiple algorithm hyperparameter optimization (multi-algo HPO) job that helps you find the best model for your dataset by jointly optimizing the data transformations and machine learning algorithms.
Available Knobs Look for sections like this for recommended settings that you can change.
Contents
1. Sagemaker Setup
A. Downloading Generated Candidates
B. SageMaker Autopilot Job and Amazon Simple Storage Service (Amazon S3) Configuration
2. Candidate Pipelines
A. Generated Candidates
B. Selected Candidates
3. Executing the Candidate Pipelines

Book-AWSAutoML-Demo.ipyn | automlbook-experiment | SageMakerAutopilotDataExpl

Read-only mode

Import notebook

Amazon SageMaker Autopilot Data Exploration

This report provides insights about the dataset you provided as input to the AutoML job. It was automatically generated by the AutoML training job: **AutoMLBook-Experiment**.

As part of the AutoML job, the input dataset was randomly split into two pieces, one for **training** and one for **validation**. The training dataset was randomly sampled, and metrics were computed for each of the columns. This notebook provides these metrics so that you can:

1. Understand how the job analyzed features to select the candidate pipelines.
2. Modify and improve the generated AutoML pipelines using knowledge that you have about the dataset.

We read `39128` rows from the training dataset. The dataset has `21` columns and the column named `y` is used as the target column. This is identified as a `BinaryClassification` problem. Here are 2 examples of labels: `['yes', 'no']`.

Suggested Action Items

- Look for sections like this for recommended actions that you can take.

Contents

Descriptive Statistics

For each of the numerical input features, several descriptive statistics are computed from the data sample.

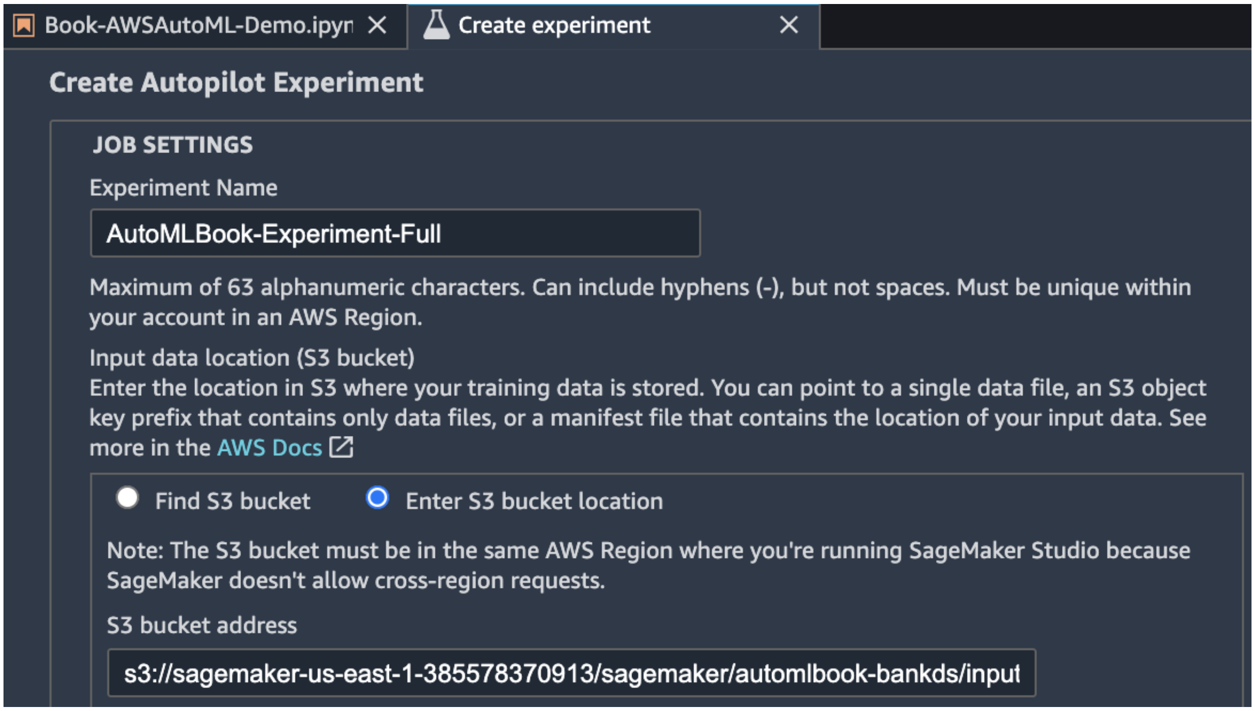
SageMaker Autopilot may treat numerical features as `Categorical` if the number of unique entries is sufficiently low. For `Numerical` features, we may apply numerical transformations such as normalization, log and quantile transforms, and binning to manage outlier values and difference in feature scales.

We found **10 of the 21** columns contained at least one numerical value. The table below shows the **10** columns which have the largest percentage of numerical values.

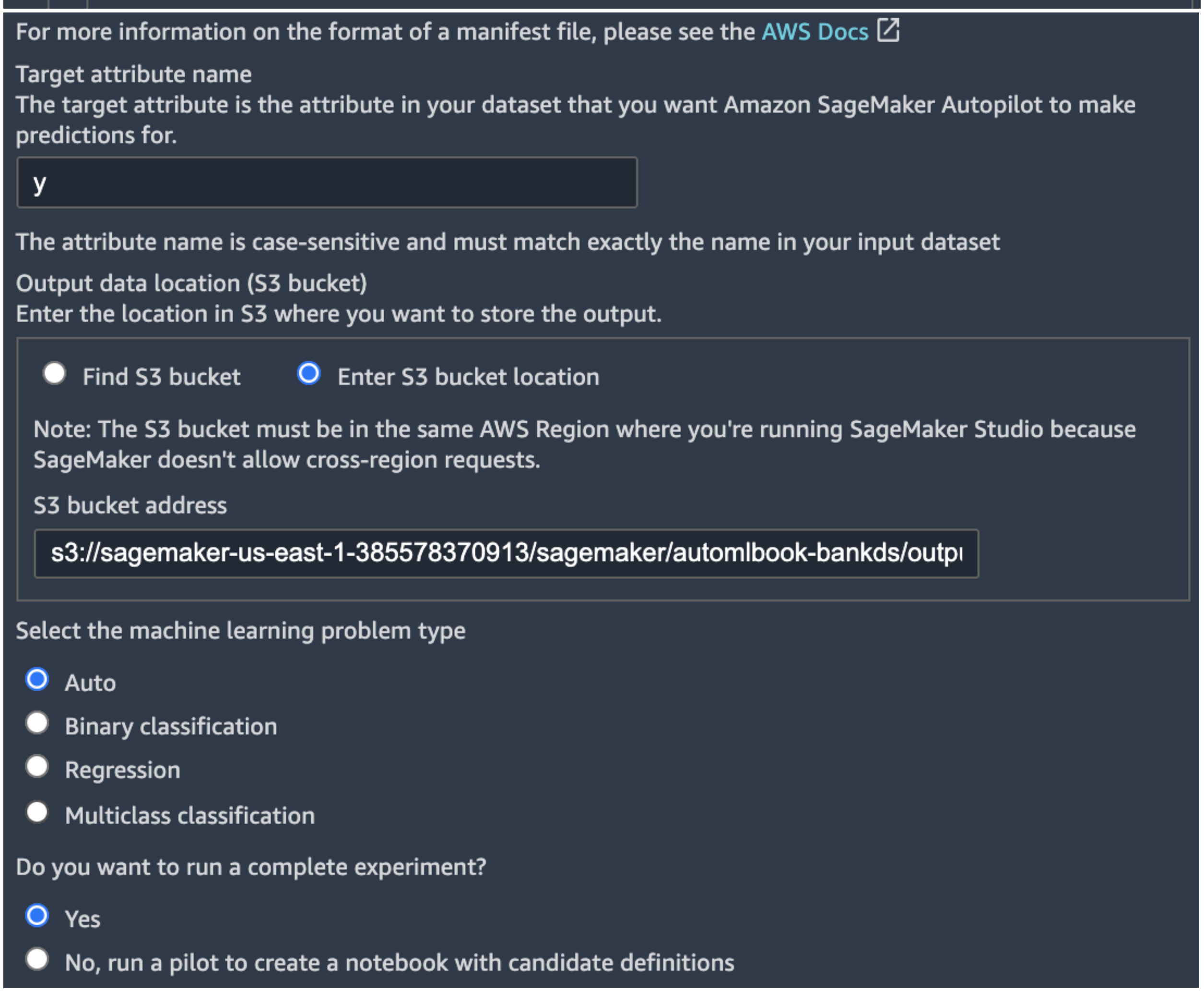
Suggested Action Items

- Investigate the origin of the data field. Are some values non-finite (e.g. infinity, nan)? Are they missing or is it an error in data input?
- Missing and extreme values may indicate a bug in the data collection process. Verify the numerical descriptions align with expectations. For example, use domain knowledge to check that the range of values for a feature meets with expectations.

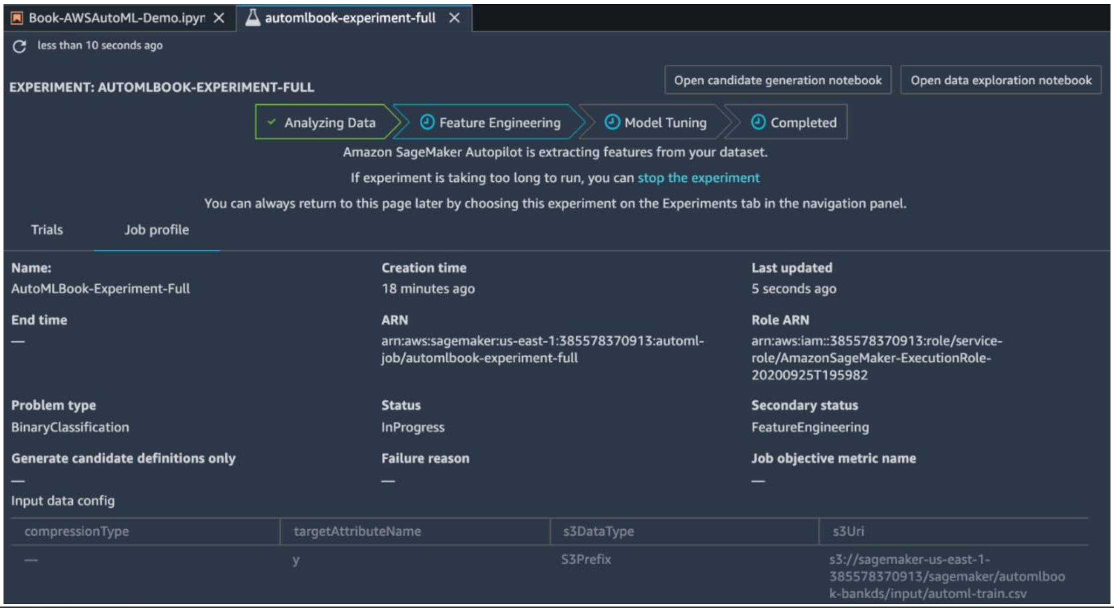
	% of Numerical Values	Mean	Median	Min	Max
age	100.0%	40.0096	38.0	17.0	98.0
duration	100.0%	258.631	178.0	0.0	4918.0
campaign	100.0%	2.57031	2.0	1.0	56.0
pdays	100.0%	962.305	999.0	0.0	999.0
previous	100.0%	0.173099	0.0	0.0	7.0
emp.var.rate	100.0%	0.0813279	1.1	-3.4	1.4
cons.price.idx	100.0%	93.5751	93.837	92.201	94.767
cons.conf.idx	100.0%	-40.5078	-41.8	-50.8	-26.9
euribor3m	100.0%	3.62068	4.857	0.634	5.045
nr.employed	100.0%	5167.03	5191.0	4963.6	5228.1

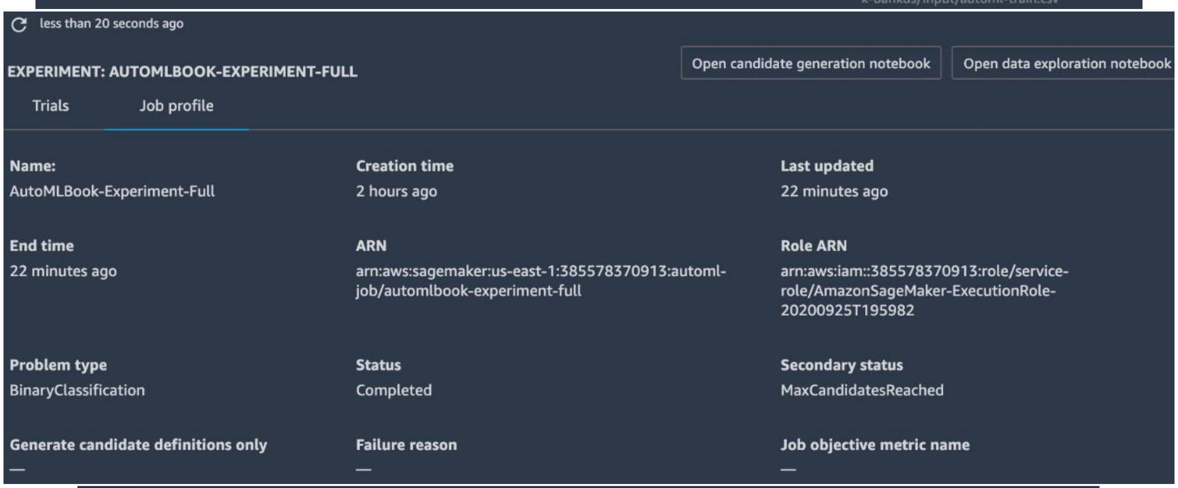


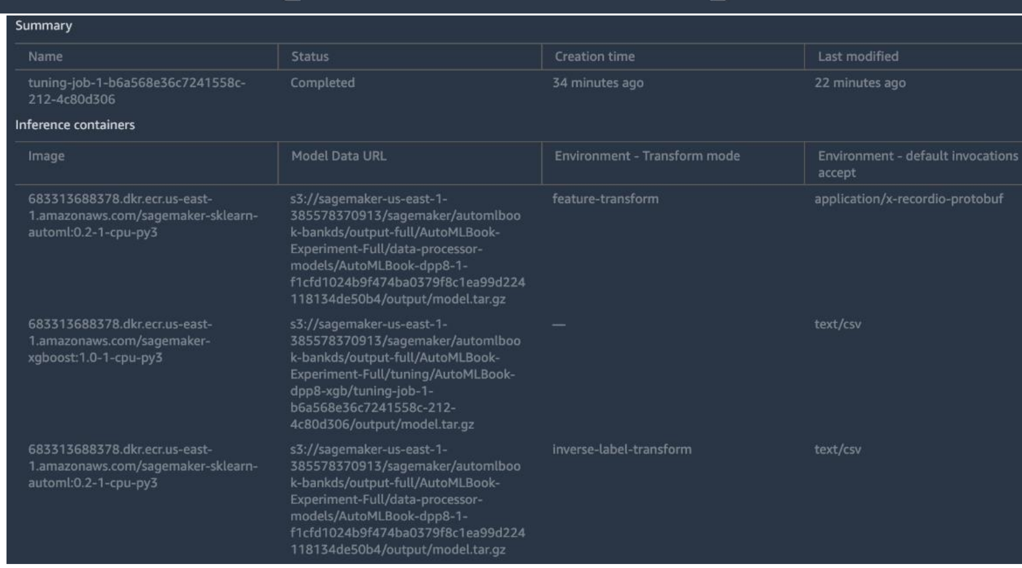
Book-AWSAutoML-Demo.ipyn
Create experiment
Create Autopilot Experiment
JOB SETTINGS
Experiment Name
AutoMLBook-Experiment-Full
Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.
Input data location (S3 bucket)
Enter the location in S3 where your training data is stored. You can point to a single data file, an S3 object key prefix that contains only data files, or a manifest file that contains the location of your input data. See more in the AWS Docs
Find S3 bucket
Enter S3 bucket location
Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.
S3 bucket address
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/input

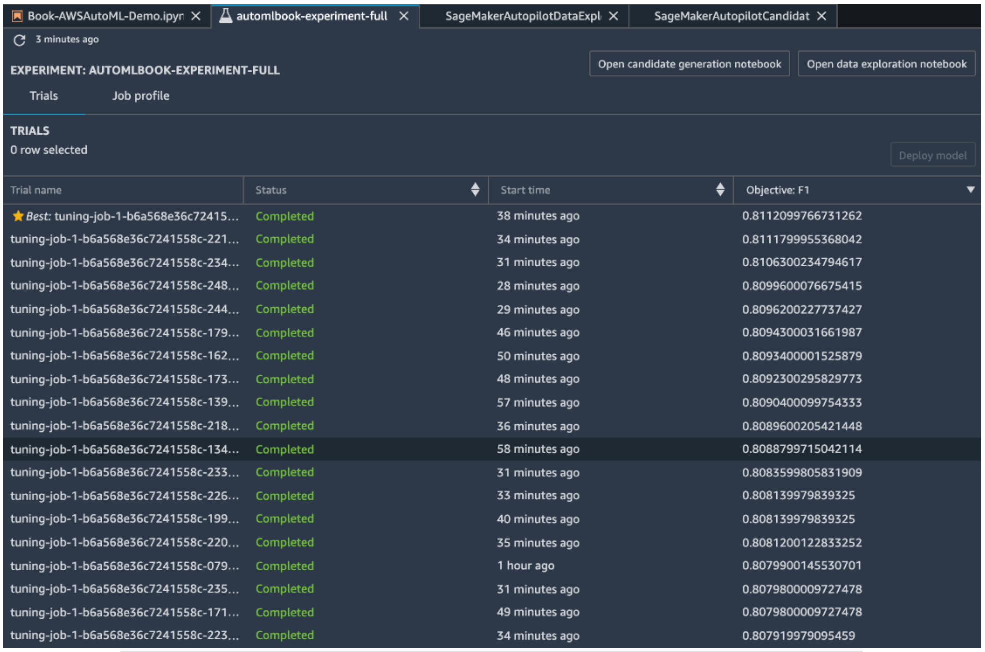


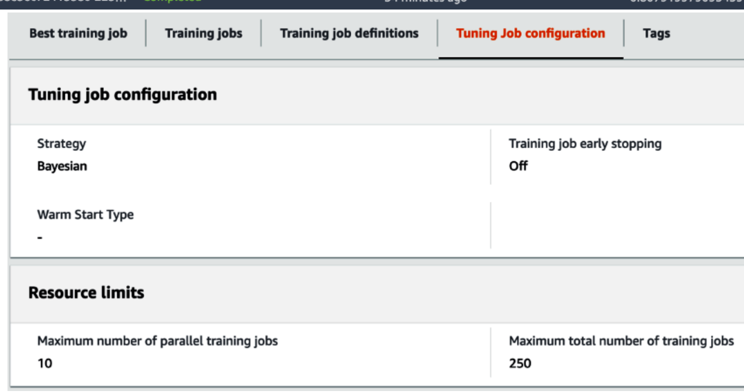
For more information on the format of a manifest file, please see the AWS Docs
Target attribute name
The target attribute is the attribute in your dataset that you want Amazon SageMaker Autopilot to make predictions for.
y
The attribute name is case-sensitive and must match exactly the name in your input dataset
Output data location (S3 bucket)
Enter the location in S3 where you want to store the output.
Find S3 bucket
Enter S3 bucket location
Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.
S3 bucket address
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/outpu
Select the machine learning problem type
Auto
Binary classification
Regression
Multiclass classification
Do you want to run a complete experiment?
Yes
No, run a pilot to create a notebook with candidate definitions

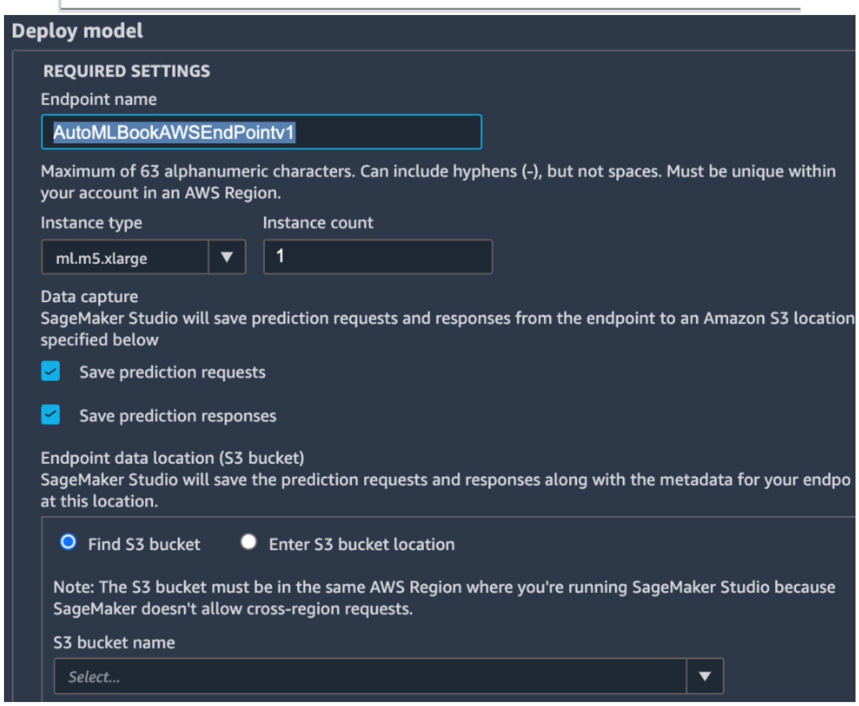

Book-AWSAutoML-Demo.ipyr
automlbook-experiment-full
less than 10 seconds ago
EXPERIMENT: AUTOMLBOOK-EXPERIMENT-FULL
Open candidate generation notebook
Open data exploration notebook
Analyzing Data
Feature Engineering
Model Tuning
Completed
Amazon SageMaker Autopilot is extracting features from your dataset.
If experiment is taking too long to run, you can stop the experiment
You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.
Trials
Job profile
Name:
AutoMLBook-Experiment-Full
Creation time
18 minutes ago
Last updated
5 seconds ago
End time
—
ARN
arn:aws:sagemaker:us-east-1:385578370913:automl-job/automlbook-experiment-full
Role ARN
arn:aws:iam::385578370913:role/service-role/AmazonSageMaker-ExecutionRole-20200925T195982
Problem type
BinaryClassification
Status
InProgress
Secondary status
FeatureEngineering
Generate candidate definitions only
—
Failure reason
—
Job objective metric name
—
Input data config
compressionType
targetAttributeName
s3DataType
s3Uri
—
y
S3Prefix
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/input/automl-train.csv

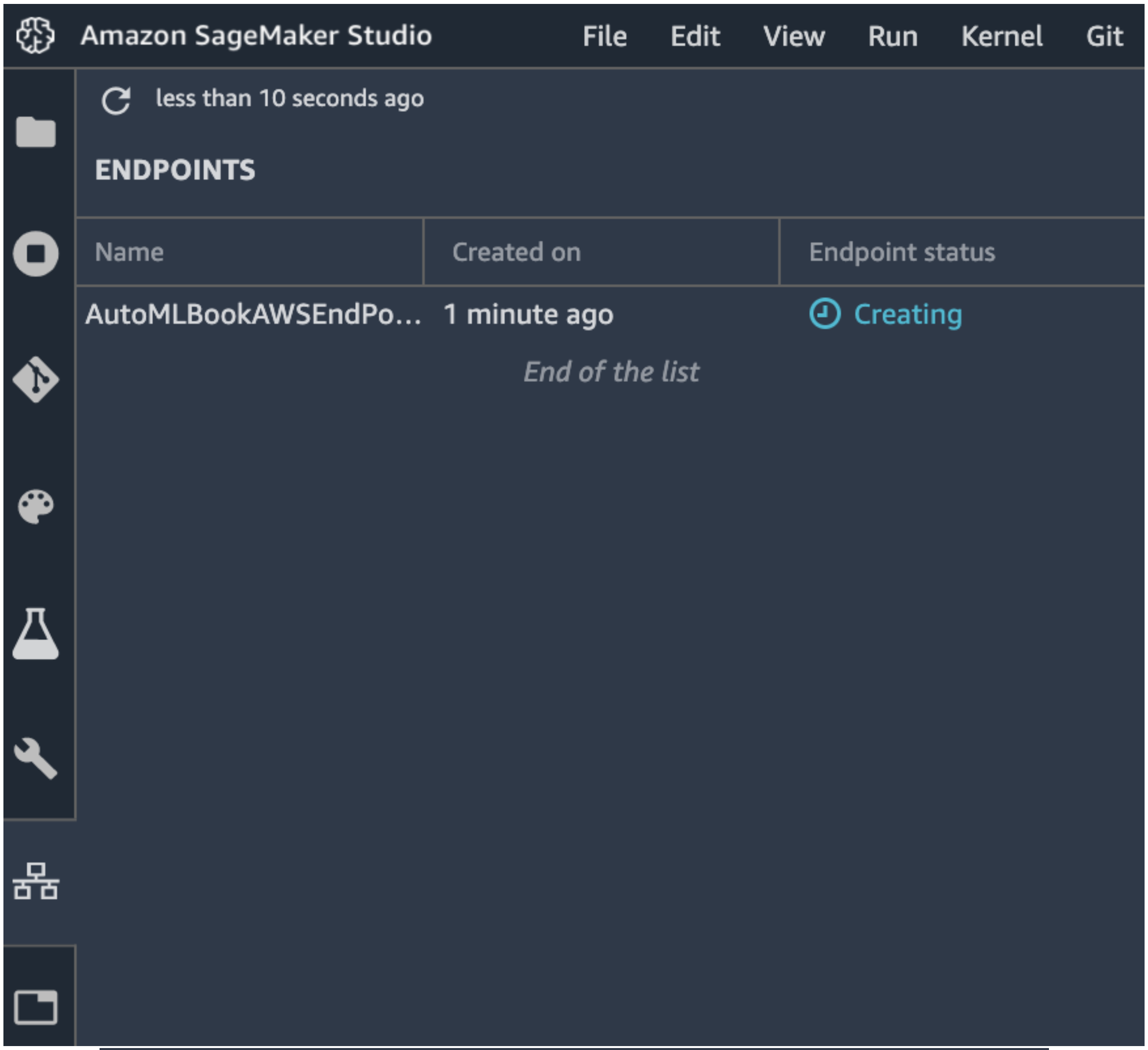

less than 20 seconds ago
EXPERIMENT: AUTOMLBOOK-EXPERIMENT-FULL
Open candidate generation notebook
Open data exploration notebook
Trials
Job profile
Name:
AutoMLBook-Experiment-Full
Creation time
2 hours ago
Last updated
22 minutes ago
End time
22 minutes ago
ARN
arn:aws:sagemaker:us-east-1:385578370913:automl-job/automlbook-experiment-full
Role ARN
arn:aws:iam::385578370913:role/service-role/AmazonSageMaker-ExecutionRole-20200925T195982
Problem type
BinaryClassification
Status
Completed
Secondary status
MaxCandidatesReached
Generate candidate definitions only
—
Failure reason
—
Job objective metric name
—

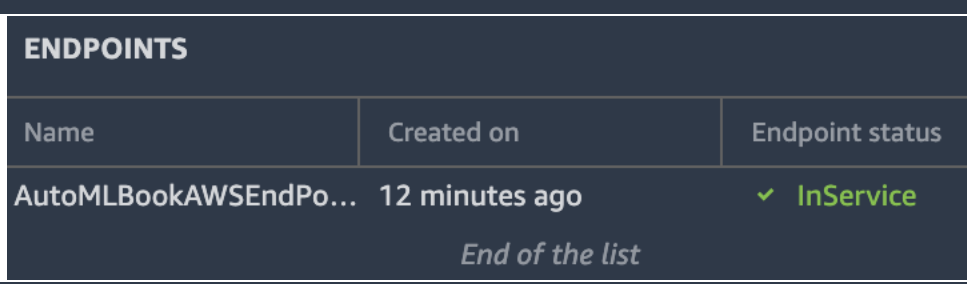

Summary
Name
Status
Creation time
Last modified
tuning-job-1-b6a568e36c7241558c-212-4c80d306
Completed
34 minutes ago
22 minutes ago
Inference containers
Image
Model Data URL
Environment - Transform mode
Environment - default invocations accept
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:0.2-1-cpu-py3
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/output-full/AutoMLBook-Experiment-Full/data-processor-models/AutoMLBook-dpp8-1-f1cfd1024b9f474ba0379f8c1ea99d224118134de50b4/output/model.tar.gz
feature-transform
application/x-recordio-protobuf
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.0-1-cpu-py3
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/output-full/AutoMLBook-Experiment-Full/tuning/AutoMLBook-dpp8-xgb/tuning-job-1-b6a568e36c7241558c-212-4c80d306/output/model.tar.gz
—
text/csv
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:0.2-1-cpu-py3
s3://sagemaker-us-east-1-385578370913/sagemaker/automlbook-bankds/output-full/AutoMLBook-Experiment-Full/data-processor-models/AutoMLBook-dpp8-1-f1cfd1024b9f474ba0379f8c1ea99d224118134de50b4/output/model.tar.gz
inverse-label-transform
text/csv

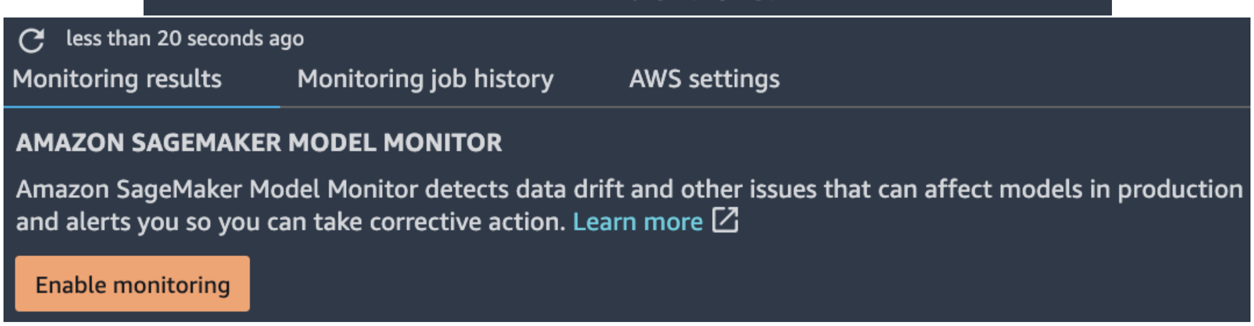

Book-AWSAutoML-Demo.ipyn
automlbook-experiment-full
SageMakerAutopilotDataExpl
SageMakerAutopilotCandidat
3 minutes ago
EXPERIMENT: AUTOMLBOOK-EXPERIMENT-FULL
Open candidate generation notebook
Open data exploration notebook
Trials
Job profile
TRIALS
0 row selected
Deploy model
Trial name
Status
Start time
Objective: F1
Best: tuning-job-1-b6a568e36c72415... Completed 38 minutes ago 0.8112099766731262
tuning-job-1-b6a568e36c7241558c-221... Completed 34 minutes ago 0.811179995368042
tuning-job-1-b6a568e36c7241558c-234... Completed 31 minutes ago 0.8106300234794617
tuning-job-1-b6a568e36c7241558c-248... Completed 28 minutes ago 0.8099600076675415
tuning-job-1-b6a568e36c7241558c-244... Completed 29 minutes ago 0.8096200227737427
tuning-job-1-b6a568e36c7241558c-179... Completed 46 minutes ago 0.8094300031661987
tuning-job-1-b6a568e36c7241558c-162... Completed 50 minutes ago 0.8093400001525879
tuning-job-1-b6a568e36c7241558c-173... Completed 48 minutes ago 0.8092300295829773
tuning-job-1-b6a568e36c7241558c-139... Completed 57 minutes ago 0.8090400099754333
tuning-job-1-b6a568e36c7241558c-218... Completed 36 minutes ago 0.8089600205421448
tuning-job-1-b6a568e36c7241558c-134... Completed 58 minutes ago 0.8088799715042114
tuning-job-1-b6a568e36c7241558c-233... Completed 31 minutes ago 0.8083599805831909
tuning-job-1-b6a568e36c7241558c-226... Completed 33 minutes ago 0.808139979839325
tuning-job-1-b6a568e36c7241558c-199... Completed 40 minutes ago 0.808139979839325
tuning-job-1-b6a568e36c7241558c-220... Completed 35 minutes ago 0.8081200122833252
tuning-job-1-b6a568e36c7241558c-079... Completed 1 hour ago 0.8079900145530701
tuning-job-1-b6a568e36c7241558c-235... Completed 31 minutes ago 0.8079800009727478
tuning-job-1-b6a568e36c7241558c-171... Completed 49 minutes ago 0.8079800009727478
tuning-job-1-b6a568e36c7241558c-223... Completed 34 minutes ago 0.807919979095459


Best training job
Training jobs
Training job definitions
Tuning Job configuration
Tags
Tuning job configuration
Strategy
Bayesian
Training job early stopping
Off
Warm Start Type
-
Resource limits
Maximum number of parallel training jobs
10
Maximum total number of training jobs
250


Deploy model
REQUIRED SETTINGS
Endpoint name
AutoMLBookAWSEndPointv1
Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.
Instance type
Instance count
ml.m5.xlarge
1
Data capture
SageMaker Studio will save prediction requests and responses from the endpoint to an Amazon S3 location specified below
Save prediction requests
Save prediction responses
Endpoint data location (S3 bucket)
SageMaker Studio will save the prediction requests and responses along with the metadata for your endpo at this location.
Find S3 bucket
Enter S3 bucket location
Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.
S3 bucket name
Select...


Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
less than 10 seconds ago
ENDPOINTS
Name
Created on
Endpoint status
AutoMLBookAWSEndPo…
1 minute ago
Creating
End of the list


ENDPOINTS
Name
Created on
Endpoint status
AutoMLBookAWSEndPo…
12 minutes ago
InService
End of the list


less than 20 seconds ago
Monitoring results
Monitoring job history
AWS settings
AMAZON SAGEMAKER MODEL MONITOR
Amazon SageMaker Model Monitor detects data drift and other issues that can affect models in production and alerts you so you can take corrective action. Learn more
Enable monitoring

Enable Amazon SageMaker Model Monitor

Amazon SageMaker provides the ability to monitor machine learning models in production and detect deviations in data quality in comparison to a baseline dataset (e.g. training data set). This notebook walks you through enabling data capture and setting up continous monitoring for an existing Endpoint.

This Notebook helps with the following:

- Update your existing SageMaker Endpoint to enable Model Monitoring
- Analyze the training dataset to generate a baseline constraint
- Setup a MonitoringSchedule for monitoring deviations from the specified baseline

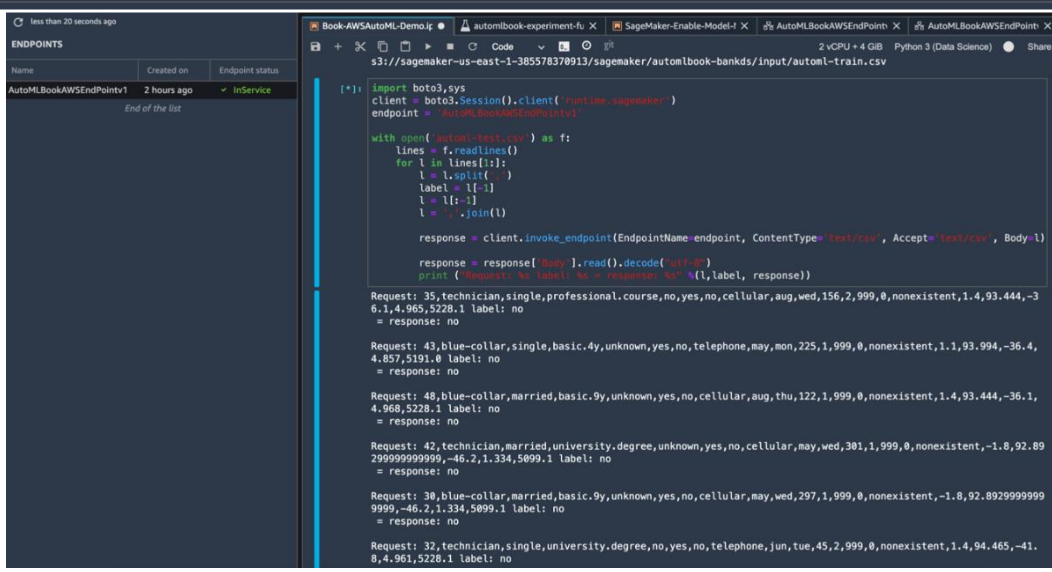
---

Step 1: Enable real-time inference data capture

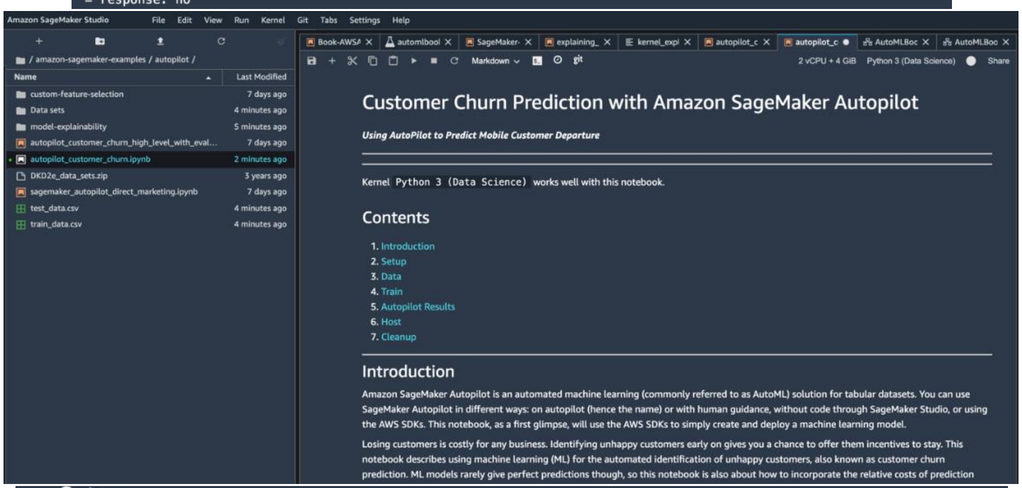
To enable data capture for monitoring the model data quality, you specify the new capture option called `DataCaptureConfig`. You can capture the request payload, the response payload or both with this configuration. The capture config applies to all variants. Please provide the Endpoint name in the following cell:

[ ]:

```
# Please fill in the following for enabling data capture
endpoint_name = 'FILL-IN-HERE-YOUR-ENDPOINT-NAME'
s3_capture_upload_path = 'FILL-IN-HERE-YOUR-S3-BUCKET-PREFIX-HERE' #example: s3://bucket-name/path/to/endpoint-data

#####
## IMPORTANT
##
## Please make sure to add the "s3:PutObject" permission to the "role' you provided in the SageMaker Model
## behind this Endpoint. Otherwise, Endpoint data capture will not work.
##
#####
```



```
Request: 32,technician,single,university.degree,no,yes,no,telephone,jun,tue,45,2,999,0,nonexistent,1.4,94.465,-41.
8,4.961,5228.1 label: no
 = response: no

Request: 46,blue-collar,married,unknown,no,no,yes,cellular,jul,thu,36,1,999,0,nonexistent,1.4,93.91799999999999,-4
2.7,4.962,5228.1 label: no
 = response: no

Request: 29,admin.,single,university.degree,no,yes,yes,cellular,nov,fri,1222,2,999,0,nonexistent,-0.1,93.2,-42.0,4.
021,5195.8 label: yes
 = response: yes

Request: 24,blue-collar,single,basic.4y,no,yes,yes,cellular,jul,wed,132,1,999,0,nonexistent,1.4,93.91799999999999,-
42.7,4.963,5228.1 label: no
 = response: no

Request: 23,entrepreneur,married,professional.course,no,no,no,cellular,jul,tue,58,1,999,0,nonexistent,1.4,93.917999
99999999,-42.7,4.962,5228.1 label: no
 = response: no

Request: 45,management,single,basic.9y,no,yes,no,telephone,jun,thu,69,1,999,0,nonexistent,1.4,94.465,-41.8,4.961,52
28.1 label: no
 = response: no

Request: 38,admin.,married,university.degree,no,no,no,cellular,oct,wed,180,2,999,1,failure,-3.4,92.431,-26.9,0.74,5
017.5 label: no
 = response: yes

Request: 58,services,married,high.school,no,yes,no,cellular,jul,fri,72,30,999,0,nonexistent,1.4,93.91799999999999,-
42.7,4.962,5228.1 label: no
 = response: no
```



Setup

This notebook was created and tested on an ml.m4.xlarge notebook instance.

Let's start by specifying:

- The S3 bucket and prefix that you want to use for training and model data. This should be within the same region as the Notebook Instance, training, and hosting.
- The IAM role arn used to give training and hosting access to your data. See the documentation for how to create these. Note, if more than one role is required for notebook instances, training, and/or hosting, please replace the boto regexp with a the appropriate full IAM role arn string(s).

[1]:
```python
import sagemaker
import boto3
from sagemaker import get_execution_role

region = boto3.Session().region_name

session = sagemaker.Session()

# You can modify the following to use a bucket of your choosing
bucket = session.default_bucket()
prefix = 'sagemaker/DEMO-autopilot-churn'

role = get_execution_role()

# This is the client we will use to interact with SageMaker AutoPilot
sm = boto3.Session().client(service_name='sagemaker',region_name=region)
```

Data

Mobile operators have historical records on which customers ultimately ended up churning and which continued using the service. We can use this historical information to construct an ML model of one mobile operator's churn using a process called training. After training the model, we can pass the profile information of an arbitrary customer (the same profile information that we used to train the model) to the model, and have the model predict whether this customer is going to churn. Of course, we expect the model to make mistakes–after all, predicting the future is tricky business! But I'll also show how to deal with prediction errors.

The dataset we use is publicly available and was mentioned in the book Discovering Knowledge in Data by Daniel T. Larose. It is attributed by the author to the University of California Irvine Repository of Machine Learning Datasets. Let's download and read that dataset in now:

[3]:
```
!apt-get install unzip
!wget http://dataminingconsultant.com/DKD2e_data_sets.zip
!unzip -o DKD2e_data_sets.zip
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 19 not upgraded.
Need to get 172 kB of archives.
After this operation, 580 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian buster/main amd64 unzip amd64 6.0-23+deb10u1 [172 kB]
Fetched 172 kB in 0s (10.8 MB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package unzip.
(Reading database ... 16492 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-23+deb10u1_amd64.deb ...
Unpacking unzip (6.0-23+deb10u1) ...
Setting up unzip (6.0-23+deb10u1) ...
Processing triggers for mime-support (3.62) ...
--2020-10-03 00:04:29--  http://dataminingconsultant.com/DKD2e_data_sets.zip
Resolving dataminingconsultant.com (dataminingconsultant.com)... 160.153.91.162
Connecting to dataminingconsultant.com (dataminingconsultant.com)|160.153.91.162|:80... connected.
```

[4]:
```
churn = pd.read_csv('./Data sets/churn.txt')
pd.set_option('display.max_columns', 500)
churn
```

[4]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	Eve Mins	Eve Calls	Eve Charge	Night Mins	Night Calls	Night Charge	Intl Mins	Intl Calls	Intl Charge	CustS C
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	197.4	99	16.78	244.7	91	11.01	10.0	3	2.70	
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	195.5	103	16.62	254.4	103	11.45	13.7	3	3.70	
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	121.2	110	10.30	162.6	104	7.32	12.2	5	3.29	
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	61.9	88	5.26	196.9	89	8.86	6.6	7	1.78	
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	148.3	122	12.61	186.9	121	8.41	10.1	3	2.73	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
3328	AZ	192	415	414-4276	no	yes	36	156.2	77	26.55	215.5	126	18.32	279.1	83	12.56	9.9	6	2.67	
3329	WV	68	415	370-3271	no	no	0	231.1	57	39.29	153.4	55	13.04	191.3	123	8.61	9.6	4	2.59	
3330	RI	28	510	328-8230	no	no	0	180.8	109	30.74	288.8	58	24.55	191.9	91	8.64	14.1	6	3.81	
3331	CT	184	510	364-6381	yes	no	0	213.8	105	36.35	159.6	84	13.57	139.2	137	6.26	5.0	10	1.35	
3332	TN	74	415	400-4344	no	yes	25	234.4	113	39.85	265.9	82	22.60	241.4	77	10.86	13.7	4	3.70	

3333 rows × 21 columns

By modern standards, it's a relatively small dataset, with only 3,333 records, where each record uses 21 attributes to describe the profile of a customer of an unknown US mobile operator. The attributes are:

Reserve some data for calling inference on the model

Divide the data into training and testing splits. The training split is used by SageMaker Autopilot. The testing split is reserved to perform inference using the suggested model.

```
[5]: train_data = churn.sample(frac=0.8,random_state=200)
     test_data = churn.drop(train_data.index)
     test_data_no_target = test_data.drop(columns=['Churn?'])
```

Now we'll upload these files to S3.

```
[6]: train_file = 'train_data.csv';
     train_data.to_csv(train_file, index=False, header=True)
     train_data_s3_path = session.upload_data(path=train_file, key_prefix=prefix + "/train")
     print('Train data uploaded to: ' + train_data_s3_path)

     test_file = 'test_data.csv';
     test_data_no_target.to_csv(test_file, index=False, header=False)
     test_data_s3_path = session.upload_data(path=test_file, key_prefix=prefix + "/test")
     print('Test data uploaded to: ' + test_data_s3_path)

     Train data uploaded to: s3://sagemaker-us-east-1-385578370913/sagemaker/DEMO-autopilot-churn/train/train_data.csv
     Test data uploaded to: s3://sagemaker-us-east-1-385578370913/sagemaker/DEMO-autopilot-churn/test/test_data.csv
```

Setting up the SageMaker Autopilot Job

After uploading the dataset to Amazon S3, you can invoke Autopilot to find the best ML pipeline to train a model on this dataset.

The required inputs for invoking a Autopilot job are:

- Amazon S3 location for input dataset and for all output artifacts
- Name of the column of the dataset you want to predict (`Churn?` in this case)
- An IAM role

Currently Autopilot supports only tabular datasets in CSV format. Either all files should have a header row, or the first file of the dataset, when sorted in alphabetical/lexical order by name, is expected to have a header row.

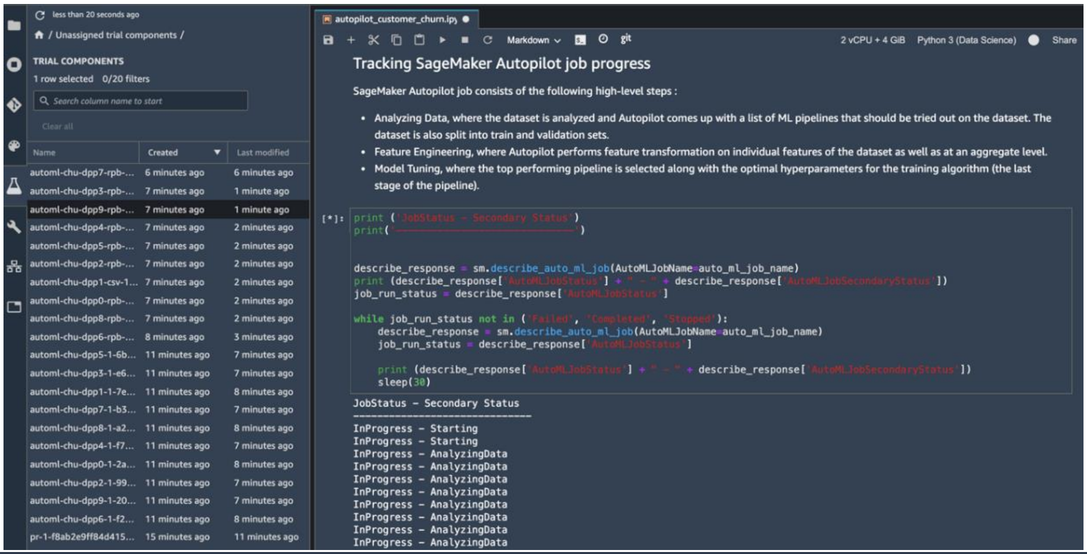
```
[7]: input_data_config = [{
           'DataSource': {
             'S3DataSource': {
               'S3DataType': 'S3Prefix',
               'S3Uri': 's3://{}/{}/train'.format(bucket,prefix)
             }
           },
           'TargetAttributeName': 'Churn?'
         }
       ]

     output_data_config = {
         'S3OutputPath': 's3://{}/{}/output'.format(bucket,prefix)
       }
```

You can also specify the type of problem you want to solve with your dataset (`Regression, MulticlassClassification, BinaryClassification`). In case you are not sure, SageMaker Autopilot will infer the problem type based on statistics of the target column (the column you want to predict).

Launching the SageMaker Autopilot Job

You can now launch the Autopilot job by calling the `create_auto_ml_job` API. We limit the number of candidates to 20 so that the job finishes in a few minutes.

```
[8]: from time import gmtime, strftime, sleep
     timestamp_suffix = strftime('%d-%H-%M-%S', gmtime())

     auto_ml_job_name = 'automl-churn-' + timestamp_suffix
     print('AutoMLJobName: ' + auto_ml_job_name)

     sm.create_auto_ml_job(AutoMLJobName=auto_ml_job_name,
                           InputDataConfig=input_data_config,
                           OutputDataConfig=output_data_config,
                           AutoMLJobConfig={'CompletionCriteria':
                                                {'MaxCandidates': 20}
                                               },
                           RoleArn=role)

     AutoMLJobName: automl-churn-03-00-04-31

[8]: {'AutoMLJobArn': 'arn:aws:sagemaker:us-east-1:385578370913:automl-job/automl-churn-03-00-04-31',
      'ResponseMetadata': {'RequestId': '50a2c4c1-f90c-4f28-a669-560c3d8f4254',
       'HTTPStatusCode': 200,
       'HTTPHeaders': {'x-amzn-requestid': '50a2c4c1-f90c-4f28-a669-560c3d8f4254',
        'content-type': 'application/x-amz-json-1.1',
        'content-length': '95',
        'date': 'Sat, 03 Oct 2020 00:04:32 GMT'},
       'RetryAttempts': 0}}
```



Tracking SageMaker Autopilot job progress

SageMaker Autopilot job consists of the following high-level steps :

- Analyzing Data, where the dataset is analyzed and Autopilot comes up with a list of ML pipelines that should be tried out on the dataset. The dataset is also split into train and validation sets.
- Feature Engineering, where Autopilot performs feature transformation on individual features of the dataset as well as at an aggregate level.
- Model Tuning, where the top performing pipeline is selected along with the optimal hyperparameters for the training algorithm (the last stage of the pipeline).

[*]:

```
print ('JobStatus - Secondary Status')
print('------------------------------')

describe_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
print (describe_response['AutoMLJobStatus'] + " - " + describe_response['AutoMLJobSecondaryStatus'])
job_run_status = describe_response['AutoMLJobStatus']

while job_run_status not in ('Failed', 'Completed', 'Stopped'):
    describe_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    job_run_status = describe_response['AutoMLJobStatus']

    print (describe_response['AutoMLJobStatus'] + " - " + describe_response['AutoMLJobSecondaryStatus'])
    sleep(30)
```

```
JobStatus - Secondary Status
------------------------------
InProgress - Starting
InProgress - Starting
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
InProgress - AnalyzingData
```

```
InProgress - ModelTuning
InProgress - ModelTuning
InProgress - ModelTuning
InProgress - ModelTuning
Completed - MaxCandidatesReached
```

Results

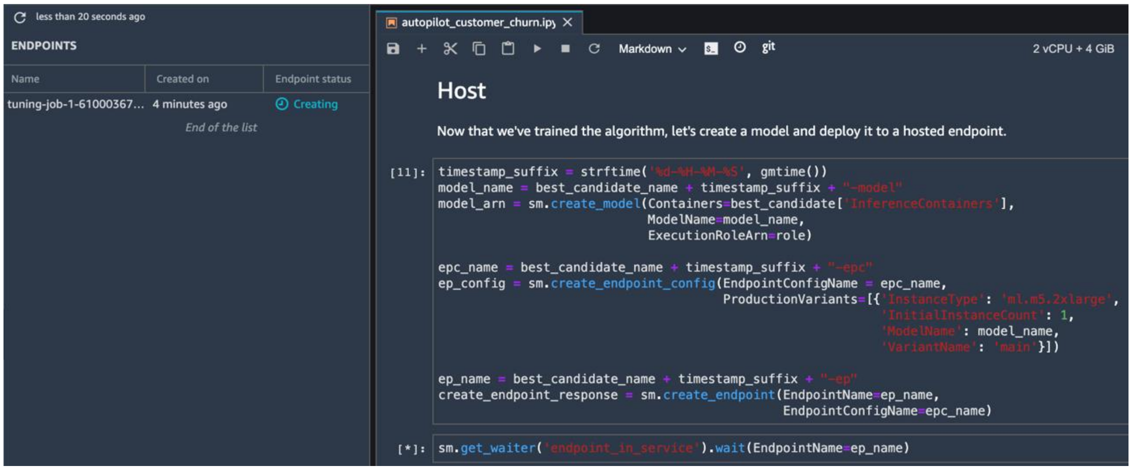
Now use the describe_auto_ml_job API to look up the best candidate selected by the SageMaker Autopilot job.

0]:
```python
best_candidate = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)['BestCandidate']
best_candidate_name = best_candidate['CandidateName']
print(best_candidate)
print('\n')
print("CandidateName: " + best_candidate_name)
print("FinalAutoMLJobObjectiveMetricName: " + best_candidate['FinalAutoMLJobObjectiveMetric']['MetricName'])
print("FinalAutoMLJobObjectiveMetricValue: " + str(best_candidate['FinalAutoMLJobObjectiveMetric']['Value']))
```


```
{'CandidateName': 'tuning-job-1-61000367db764868a7-020-2e4499ff', 'FinalAutoMLJobObjectiveMetric': {'MetricName': 'valida
tion:f1', 'Value': 0.923229992389679}, 'ObjectiveStatus': 'Succeeded', 'CandidateSteps': [{'CandidateStepType': 'AWS::Sag
eMaker::ProcessingJob', 'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:385578370913:processing-job/db-1-823a0a699a494f8
58351af33214ee54957bd65fb089f455d878abe698b', 'CandidateStepName': 'db-1-823a0a699a494f858351af33214ee54957bd65fb089f455d
878abe698b'}, {'CandidateStepType': 'AWS::SageMaker::TrainingJob', 'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:38557
8370913:training-job/automl-chu-dpp9-1-2017d334a7da4432961a41b9c8b8127e178053fc51a04', 'CandidateStepName': 'automl-chu-d
pp9-1-2017d334a7da4432961a41b9c8b8127e178053fc51a04'}, {'CandidateStepType': 'AWS::SageMaker::TransformJob', 'CandidateSt
epArn': 'arn:aws:sagemaker:us-east-1:385578370913:transform-job/automl-chu-dpp9-rpb-1-3156254e873d445c98a900c5439b6fcaecc
a2702e', 'CandidateStepName': 'automl-chu-dpp9-rpb-1-3156254e873d445c98a900c5439b6fcaecca2702e'}, {'CandidateStepType':
'AWS::SageMaker::TrainingJob', 'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:385578370913:training-job/tuning-job-1-61
000367db764868a7-020-2e4499ff', 'CandidateStepName': 'tuning-job-1-61000367db764868a7-020-2e4499ff'}], 'CandidateStatus':
'Completed', 'InferenceContainers': [{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:0.2
```


```
CandidateName: tuning-job-1-61000367db764868a7-020-2e4499ff
FinalAutoMLJobObjectiveMetricName: validation:f1
FinalAutoMLJobObjectiveMetricValue: 0.923229992389679
```

Due to some randomness in the algorithms involved, different runs will provide slightly different results, but accuracy will be around or above 93%, which is a good result.




```
[12]: sm.get_waiter('endpoint_in_service').wait(EndpointName=ep_name)
```

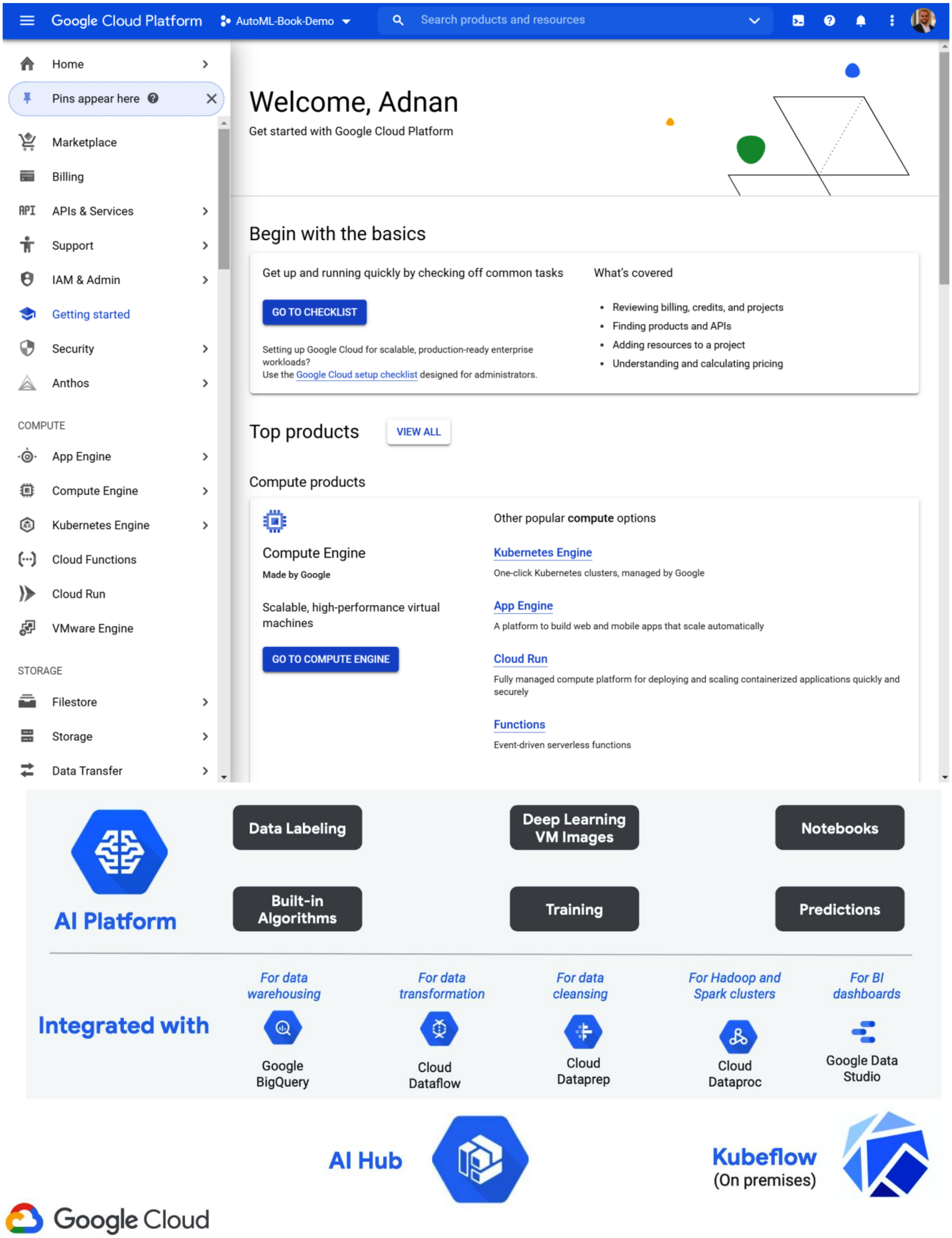
Evaluate

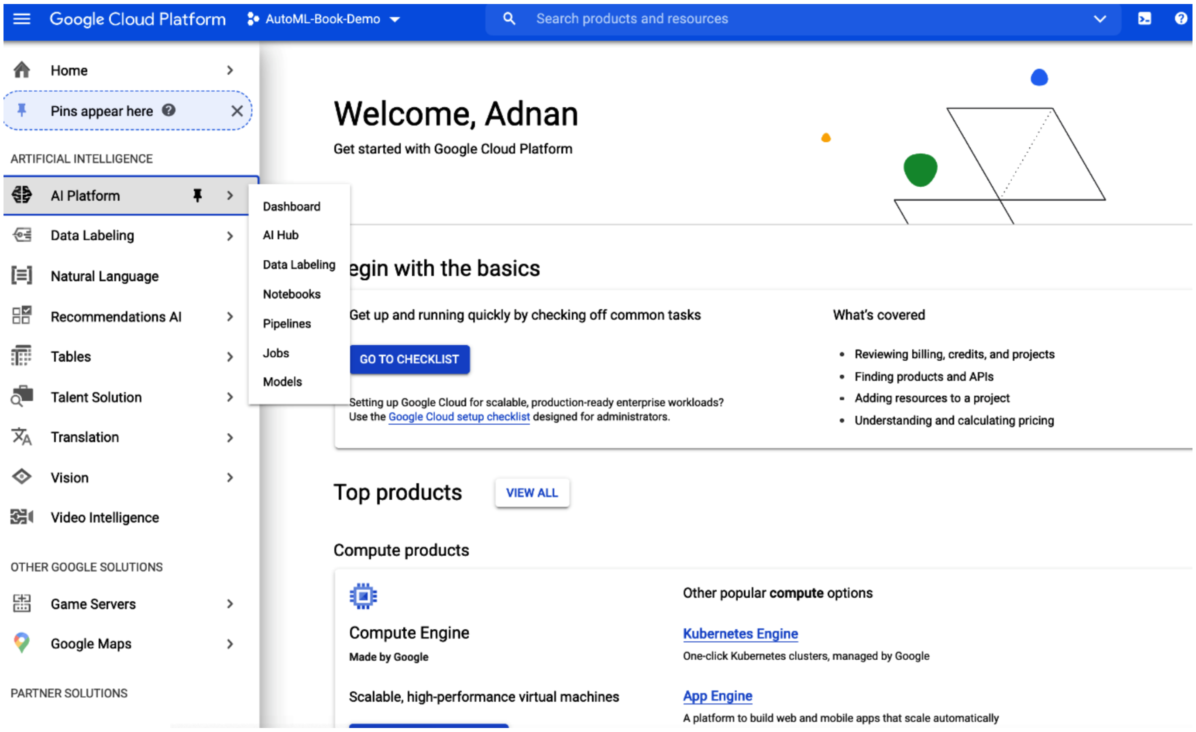
Now that we have a hosted endpoint running, we can make real-time predictions from our model very easily, simply by making an http POST request. But first, we'll need to setup serializers and deserializers for passing our `test_data` NumPy arrays to the model behind the endpoint.

```
[13]: from io import StringIO
from sagemaker.predictor import RealTimePredictor
from sagemaker.content_types import CONTENT_TYPE_CSV

predictor = RealTimePredictor(
    endpoint=ep_name,
    sagemaker_session=session,
    content_type=CONTENT_TYPE_CSV,
    accept=CONTENT_TYPE_CSV)

# Remove the target column from the test data
test_data_inference = test_data.drop('Churn?', axis=1)

# Obtain predictions from SageMaker endpoint
prediction = predictor.predict(test_data_inference.to_csv(sep=',', header=False, index=False)).decode('utf-8')

# Load prediction in pandas and compare to ground truth
prediction_df = pd.read_csv(StringIO(prediction), header=None)
accuracy = (test_data.reset_index()['Churn?'] == prediction_df[0]).sum() / len(test_data_inference)
print('Accuracy: {}'.format(accuracy))
```

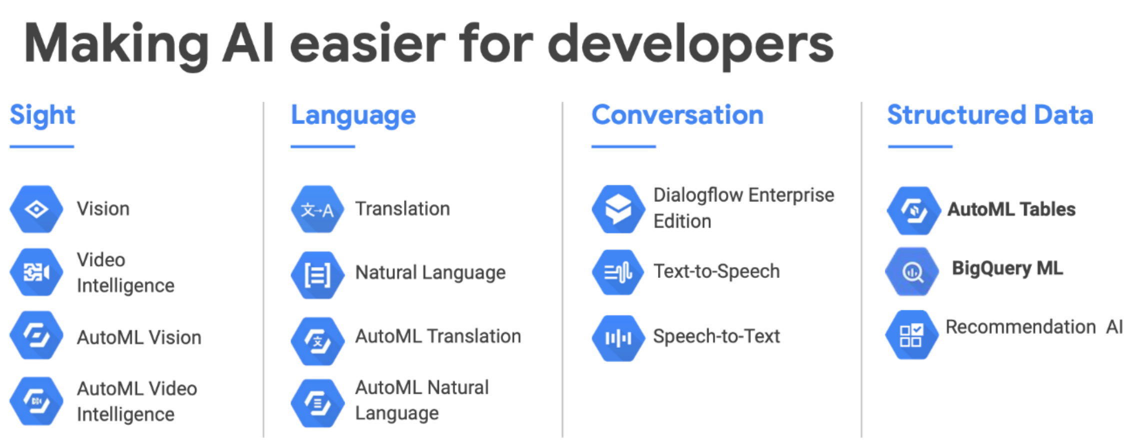
```
Accuracy: 0.9685157421289355
```

Chapter 8: Machine Learning with Google Cloud Platform

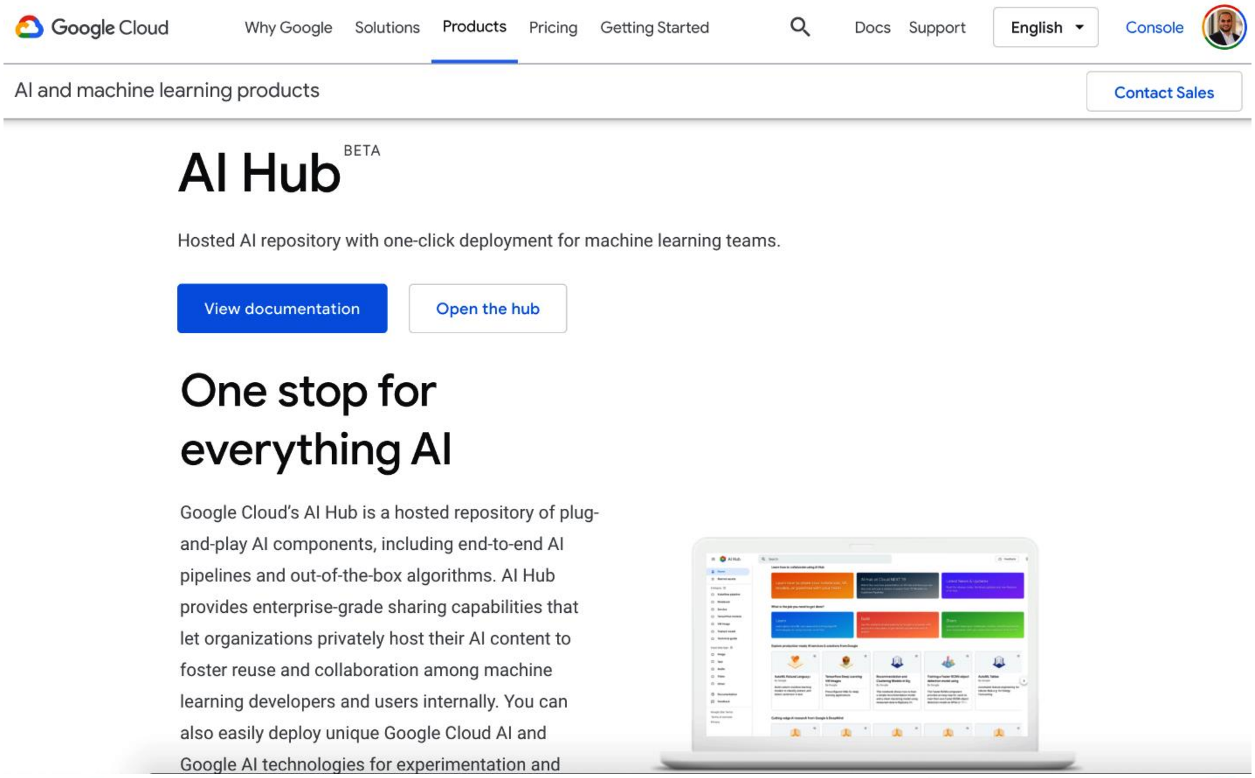


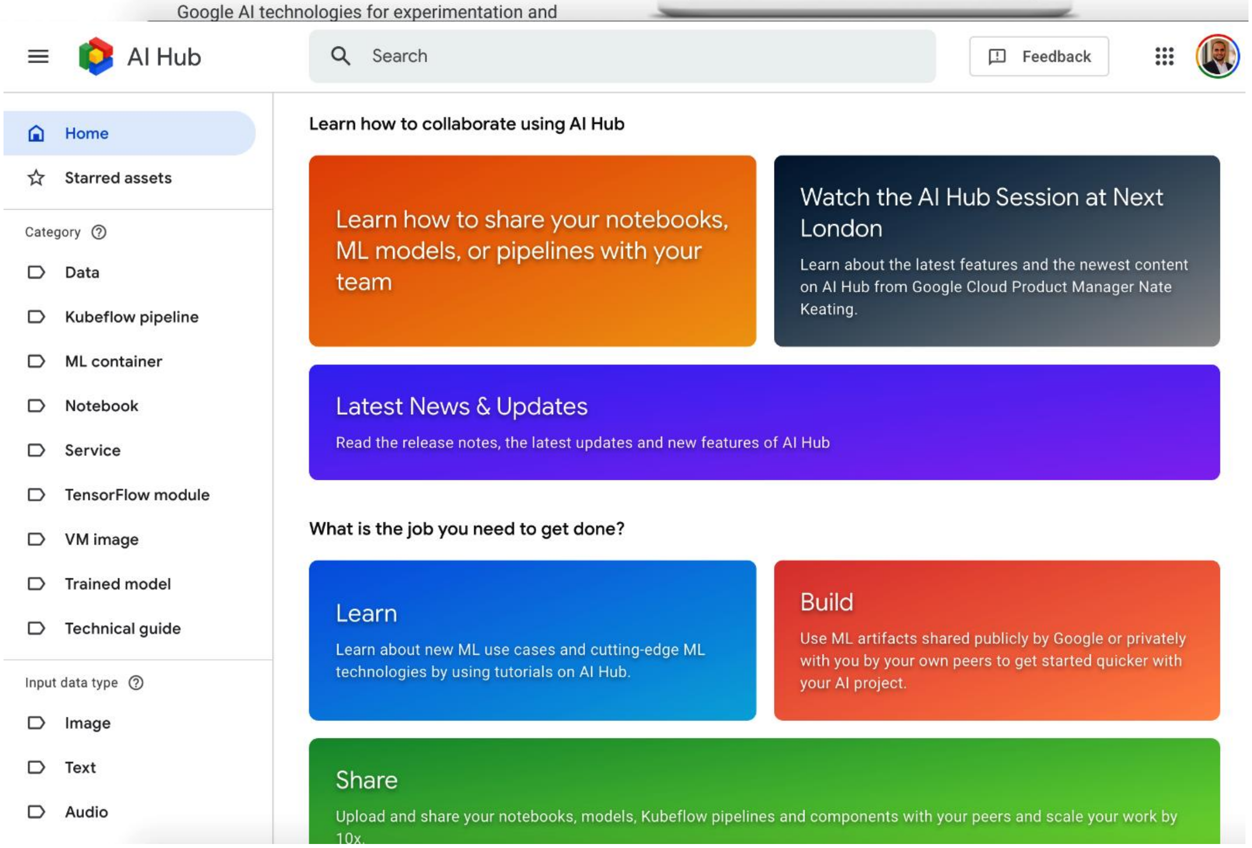


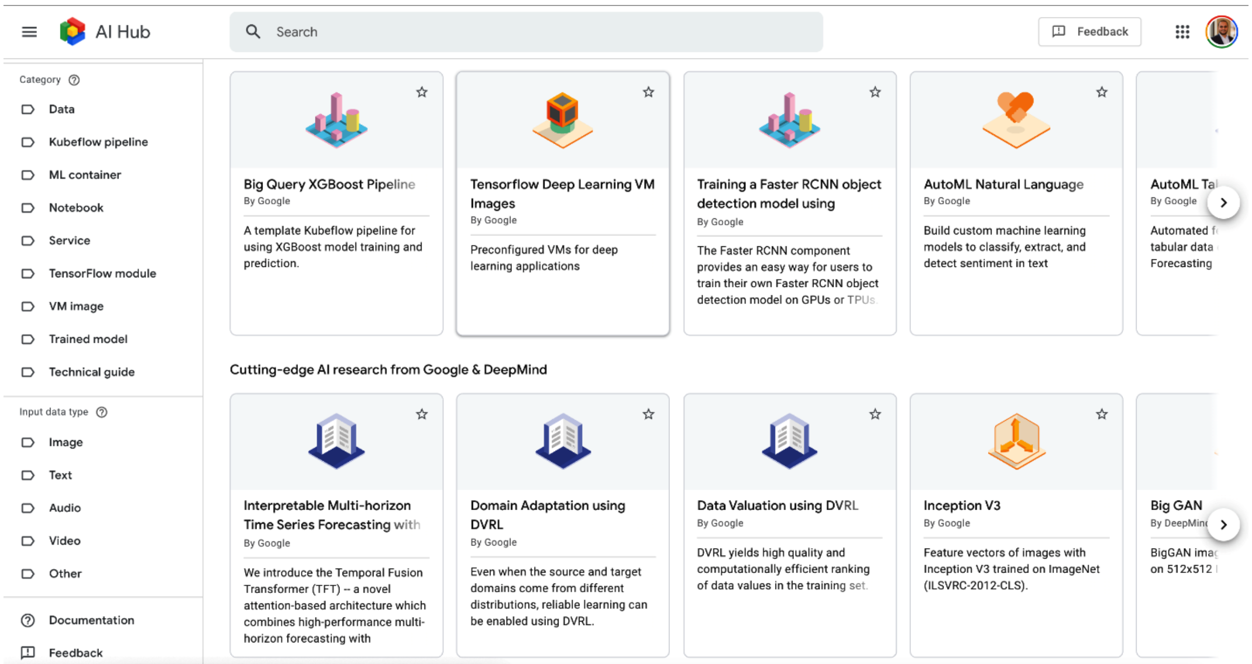
Google Cloud Platform
AutoML-Book-Demo
Search products and resources
Home
Pins appear here
ARTIFICIAL INTELLIGENCE
AI Platform
Data Labeling
Natural Language
Recommendations AI
Tables
Talent Solution
Translation
Vision
Video Intelligence
OTHER GOOGLE SOLUTIONS
Game Servers
Google Maps
PARTNER SOLUTIONS
Dashboard
AI Hub
Data Labeling
Notebooks
Pipelines
Jobs
Models
Welcome, Adnan
Get started with Google Cloud Platform
egin with the basics
Get up and running quickly by checking off common tasks
GO TO CHECKLIST
Setting up Google Cloud for scalable, production-ready enterprise workloads?
Use the Google Cloud setup checklist designed for administrators.
What's covered
Reviewing billing, credits, and projects
Finding products and APIs
Adding resources to a project
Understanding and calculating pricing
Top products
VIEW ALL
Compute products
Compute Engine
Made by Google
Scalable, high-performance virtual machines
Other popular compute options
Kubernetes Engine
One-click Kubernetes clusters, managed by Google
App Engine
A platform to build web and mobile apps that scale automatically

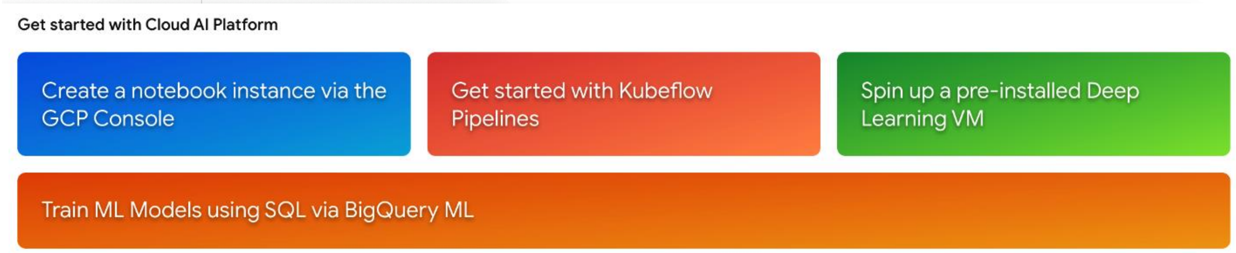


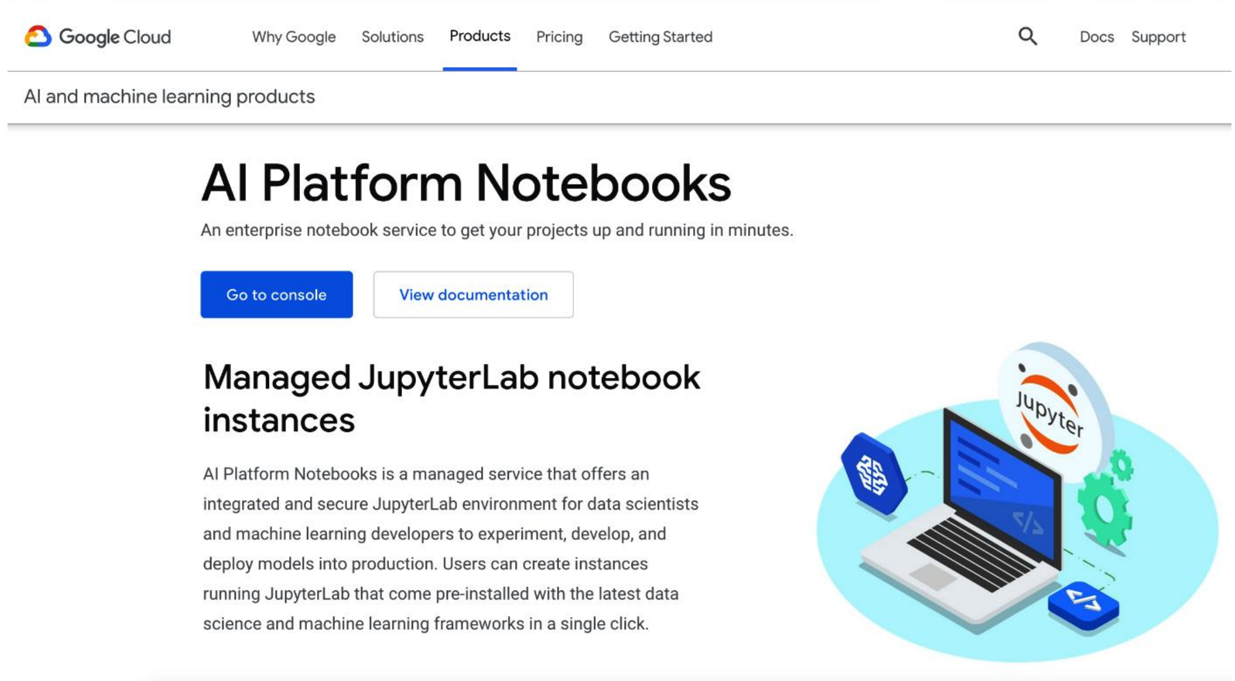
Making AI easier for developers
Sight
Vision
Video Intelligence
AutoML Vision
AutoML Video Intelligence
Language
Translation
Natural Language
AutoML Translation
AutoML Natural Language
Conversation
Dialogflow Enterprise Edition
Text-to-Speech
Speech-to-Text
Structured Data
AutoML Tables
BigQuery ML
Recommendation AI

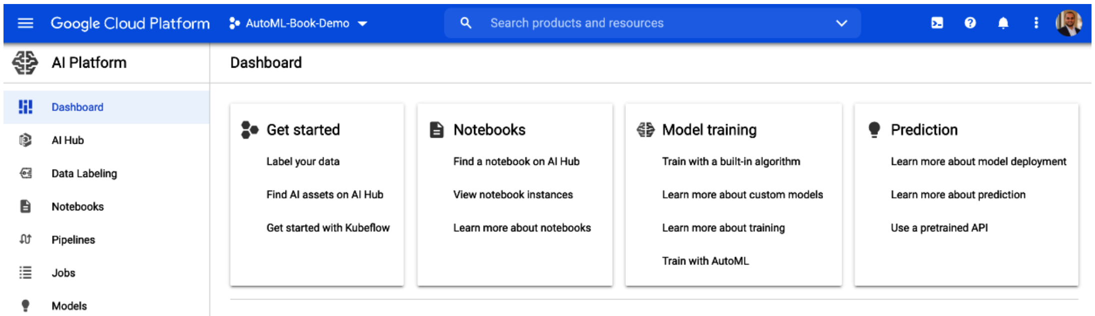

Google Cloud
Why Google
Solutions
Products
Pricing
Getting Started
Docs
Support
English
Console
AI and machine learning products
Contact Sales
AI Hub BETA
Hosted AI repository with one-click deployment for machine learning teams.
View documentation
Open the hub
One stop for everything AI
Google Cloud's AI Hub is a hosted repository of plug-and-play AI components, including end-to-end AI pipelines and out-of-the-box algorithms. AI Hub provides enterprise-grade sharing capabilities that let organizations privately host their AI content to foster reuse and collaboration among machine learning developers and users internally. You can also easily deploy unique Google Cloud AI and Google AI technologies for experimentation and

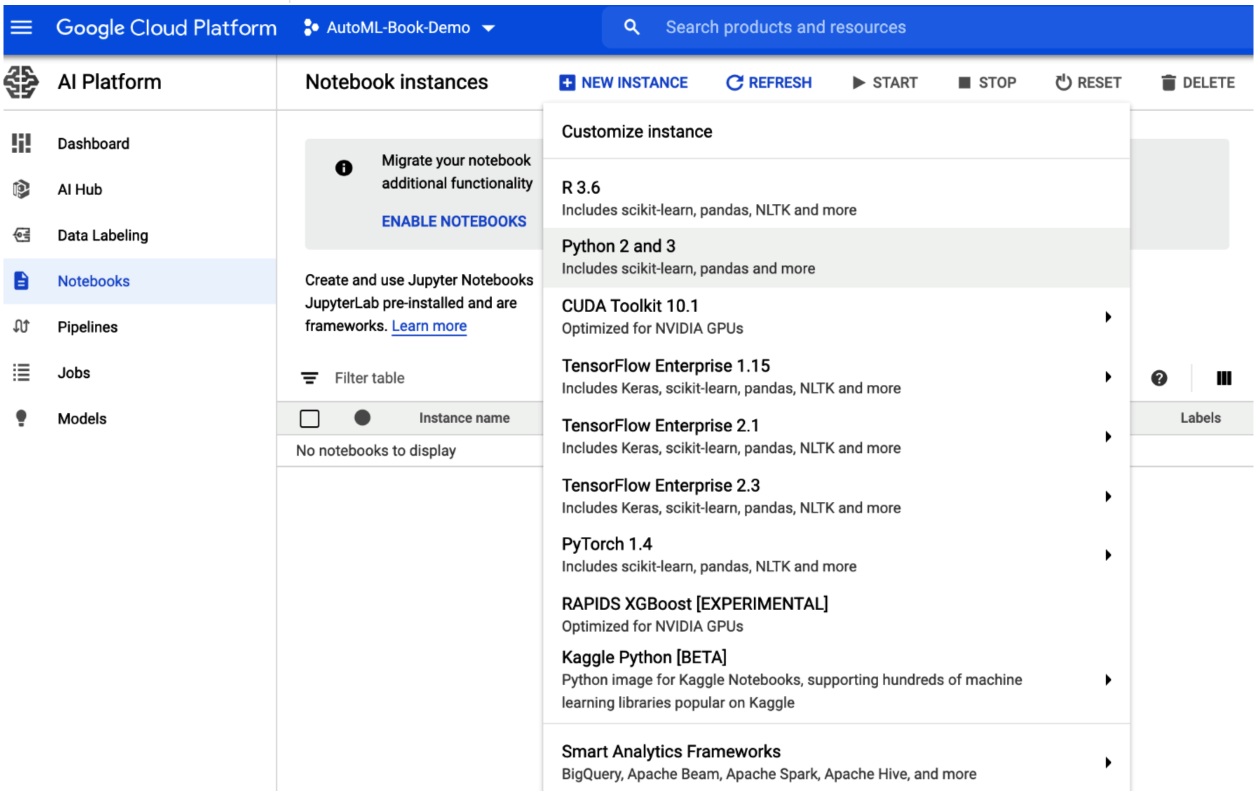

AI Hub
Search
Feedback
Home
Starred assets
Category
Data
Kubeflow pipeline
ML container
Notebook
Service
TensorFlow module
VM image
Trained model
Technical guide
Input data type
Image
Text
Audio
Learn how to collaborate using AI Hub
Learn how to share your notebooks, ML models, or pipelines with your team
Watch the AI Hub Session at Next London
Learn about the latest features and the newest content on AI Hub from Google Cloud Product Manager Nate Keating.
Latest News & Updates
Read the release notes, the latest updates and new features of AI Hub
What is the job you need to get done?
Learn
Learn about new ML use cases and cutting-edge ML technologies by using tutorials on AI Hub.
Build
Use ML artifacts shared publicly by Google or privately with you by your own peers to get started quicker with your AI project.
Share
Upload and share your notebooks, models, Kubeflow pipelines and components with your peers and scale your work by 10x.

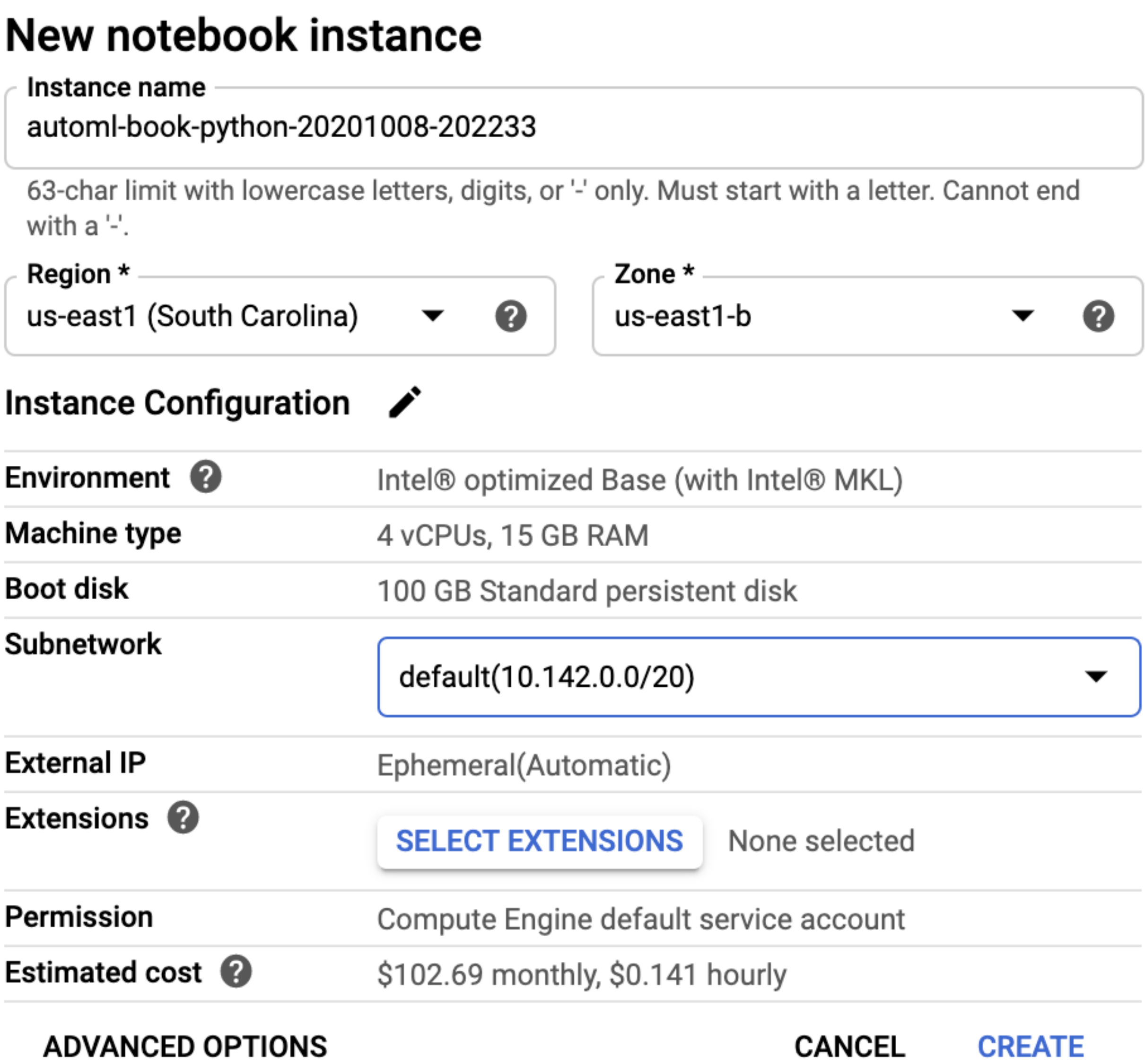

AI Hub
Search
Feedback
Category
Data
Kubeflow pipeline
ML container
Notebook
Service
TensorFlow module
VM image
Trained model
Technical guide
Input data type
Image
Text
Audio
Video
Other
Documentation
Feedback
Big Query XGBoost Pipeline
By Google
A template Kubeflow pipeline for using XGBoost model training and prediction.
Tensorflow Deep Learning VM Images
By Google
Preconfigured VMs for deep learning applications
Training a Faster RCNN object detection model using
By Google
The Faster RCNN component provides an easy way for users to train their own Faster RCNN object detection model on GPUs or TPUs.
AutoML Natural Language
By Google
Build custom machine learning models to classify, extract, and detect sentiment in text
AutoML Ta
By Google
Automated
tabular data
Forecasting
Cutting-edge AI research from Google & DeepMind
Interpretable Multi-horizon Time Series Forecasting with
By Google
We introduce the Temporal Fusion Transformer (TFT) -- a novel attention-based architecture which combines high-performance multi-horizon forecasting with
Domain Adaptation using DVRL
By Google
Even when the source and target domains come from different distributions, reliable learning can be enabled using DVRL.
Data Valuation using DVRL
By Google
DVRL yields high quality and computationally efficient ranking of data values in the training set.
Inception V3
By Google
Feature vectors of images with Inception V3 trained on ImageNet (ILSVRC-2012-CLS).
Big GAN
By DeepMin
BigGAN imag
on 512x512

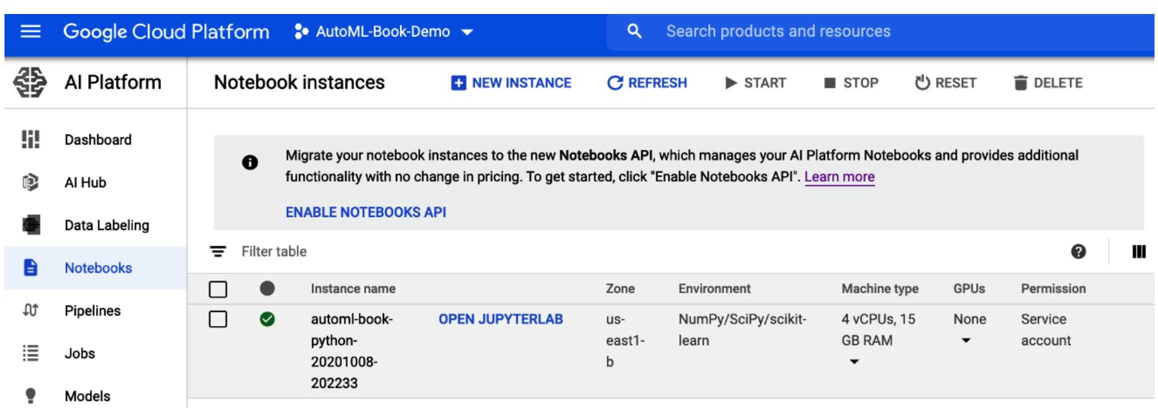

Get started with Cloud AI Platform
Create a notebook instance via the GCP Console
Get started with Kubeflow Pipelines
Spin up a pre-installed Deep Learning VM
Train ML Models using SQL via BigQuery ML

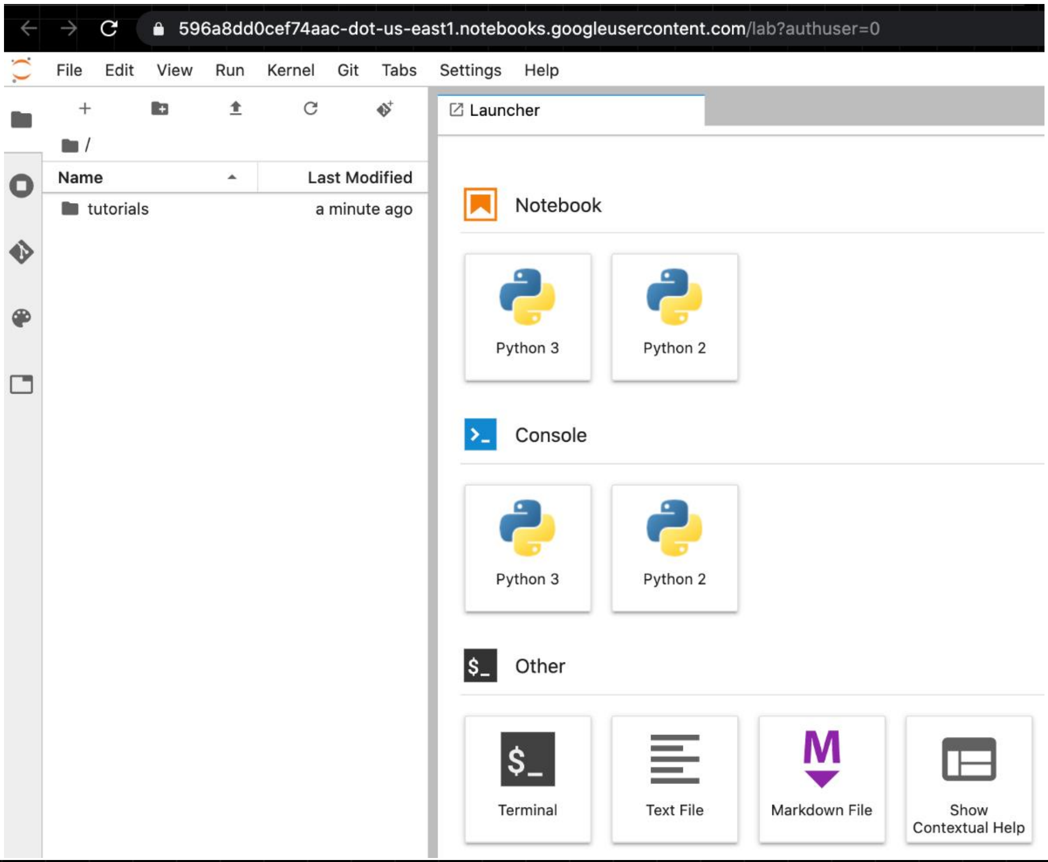

Google Cloud
Why Google
Solutions
Products
Pricing
Getting Started
Docs
Support
AI and machine learning products
AI Platform Notebooks
An enterprise notebook service to get your projects up and running in minutes.
Go to console
View documentation
Managed JupyterLab notebook instances
AI Platform Notebooks is a managed service that offers an integrated and secure JupyterLab environment for data scientists and machine learning developers to experiment, develop, and deploy models into production. Users can create instances running JupyterLab that come pre-installed with the latest data science and machine learning frameworks in a single click.
Jupyter


Google Cloud Platform
AutoML-Book-Demo
Search products and resources
AI Platform
Dashboard
Dashboard
AI Hub
Data Labeling
Notebooks
Pipelines
Jobs
Models
Get started
Label your data
Find AI assets on AI Hub
Get started with Kubeflow
Notebooks
Find a notebook on AI Hub
View notebook instances
Learn more about notebooks
Model training
Train with a built-in algorithm
Learn more about custom models
Learn more about training
Train with AutoML
Prediction
Learn more about model deployment
Learn more about prediction
Use a pretrained API

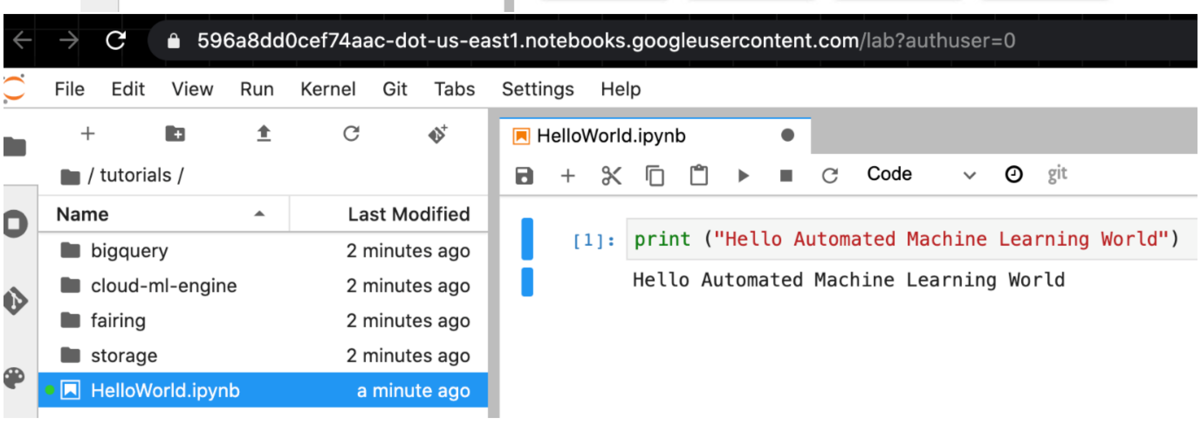

Google Cloud Platform
AutoML-Book-Demo
Search products and resources
AI Platform
Notebook instances
NEW INSTANCE
REFRESH
START
STOP
RESET
DELETE
Dashboard
AI Hub
Data Labeling
Notebooks
Pipelines
Jobs
Models
Migrate your notebook
additional functionality
ENABLE NOTEBOOKS
Create and use Jupyter Notebooks
JupyterLab pre-installed and are
frameworks. Learn more
Filter table
Instance name
Labels
No notebooks to display
Customize instance
R 3.6
Includes scikit-learn, pandas, NLTK and more
Python 2 and 3
Includes scikit-learn, pandas and more
CUDA Toolkit 10.1
Optimized for NVIDIA GPUs
TensorFlow Enterprise 1.15
Includes Keras, scikit-learn, pandas, NLTK and more
TensorFlow Enterprise 2.1
Includes Keras, scikit-learn, pandas, NLTK and more
TensorFlow Enterprise 2.3
Includes Keras, scikit-learn, pandas, NLTK and more
PyTorch 1.4
Includes scikit-learn, pandas, NLTK and more
RAPIDS XGBoost [EXPERIMENTAL]
Optimized for NVIDIA GPUs
Kaggle Python [BETA]
Python image for Kaggle Notebooks, supporting hundreds of machine learning libraries popular on Kaggle
Smart Analytics Frameworks
BigQuery, Apache Beam, Apache Spark, Apache Hive, and more


New notebook instance
Instance name
automl-book-python-20201008-202233
63-char limit with lowercase letters, digits, or '-' only. Must start with a letter. Cannot end with a '-'.
Region *
us-east1 (South Carolina)
Zone *
us-east1-b
Instance Configuration
Environment
Intel® optimized Base (with Intel® MKL)
Machine type
4 vCPUs, 15 GB RAM
Boot disk
100 GB Standard persistent disk
Subnetwork
default(10.142.0.0/20)
External IP
Ephemeral(Automatic)
Extensions
SELECT EXTENSIONS
None selected
Permission
Compute Engine default service account
Estimated cost
$102.69 monthly, $0.141 hourly
ADVANCED OPTIONS
CANCEL
CREATE


Google Cloud Platform
AutoML-Book-Demo
Search products and resources
AI Platform
Notebook instances
NEW INSTANCE
REFRESH
START
STOP
RESET
DELETE
Dashboard
AI Hub
Data Labeling
Notebooks
Pipelines
Jobs
Models
Migrate your notebook instances to the new Notebooks API, which manages your AI Platform Notebooks and provides additional functionality with no change in pricing. To get started, click "Enable Notebooks API". Learn more
ENABLE NOTEBOOKS API
Filter table
Instance name
Zone
Environment
Machine type
GPUs
Permission
automl-book-python-20201008-202233
OPEN JUPYTERLAB
us-east1-b
NumPy/SciPy/scikit-learn
4 vCPUs, 15 GB RAM
None
Service account

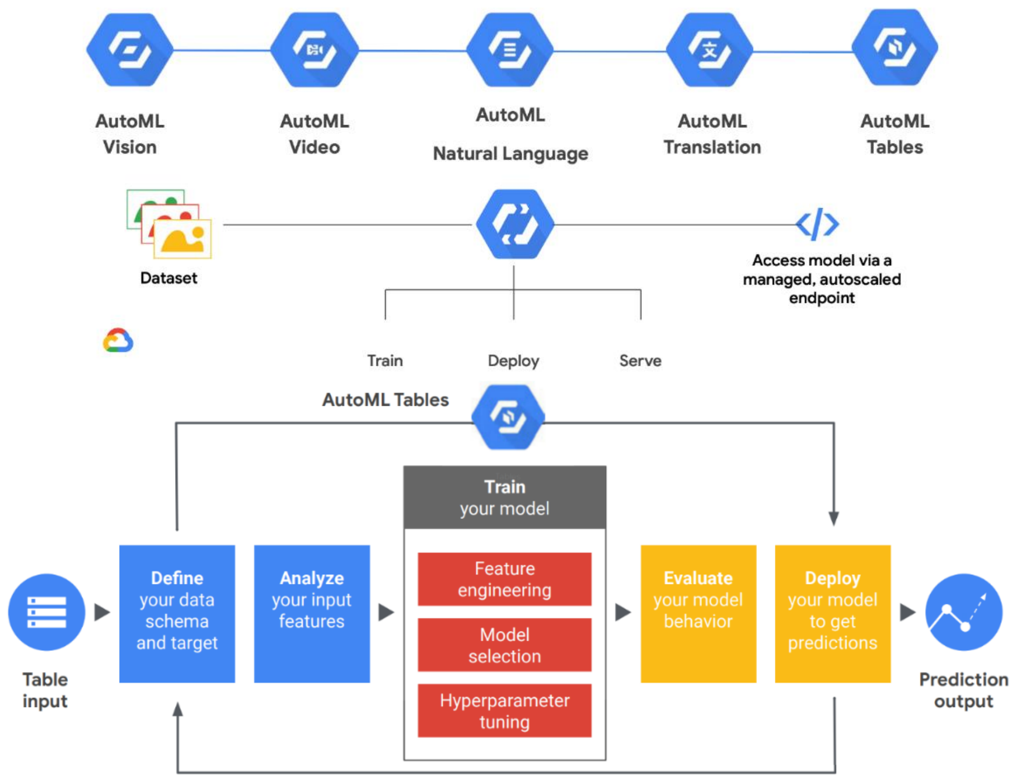


596a8dd0cef74aac-dot-us-east1.notebooks.googleusercontent.com/lab?authuser=0
File Edit View Run Kernel Git Tabs Settings Help
Launcher
Name
Last Modified
tutorials
a minute ago
Notebook
Python 3
Python 2
Console
Python 3
Python 2
Other
Terminal
Text File
Markdown File
Show Contextual Help

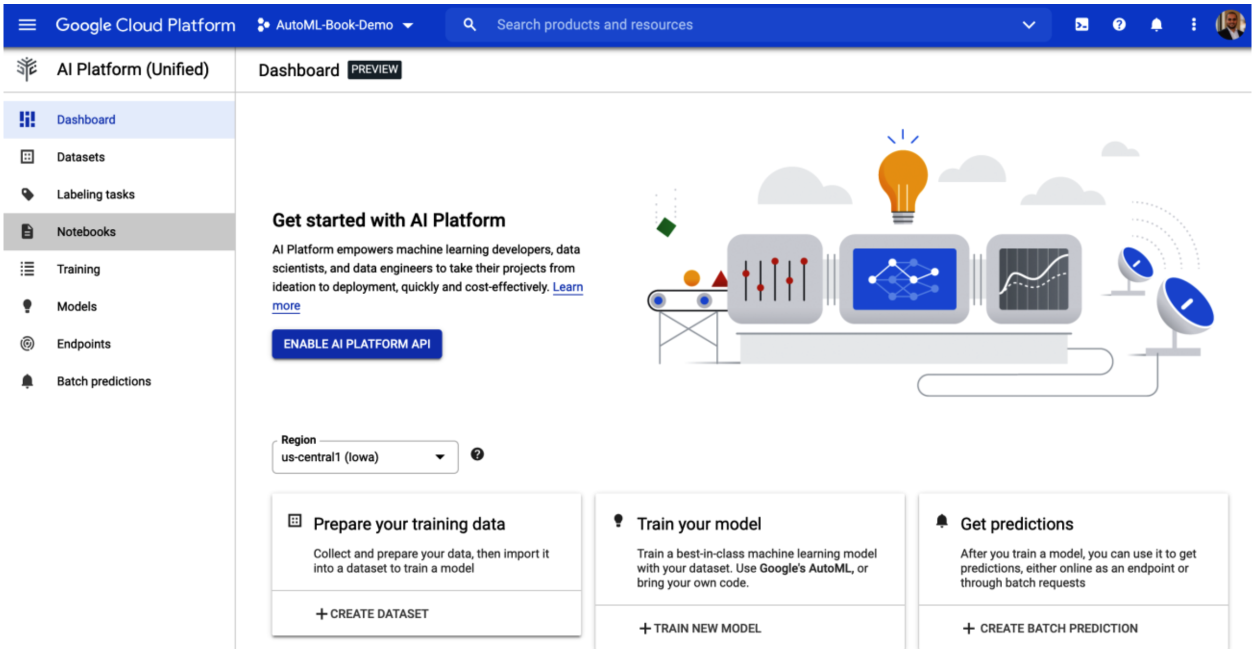


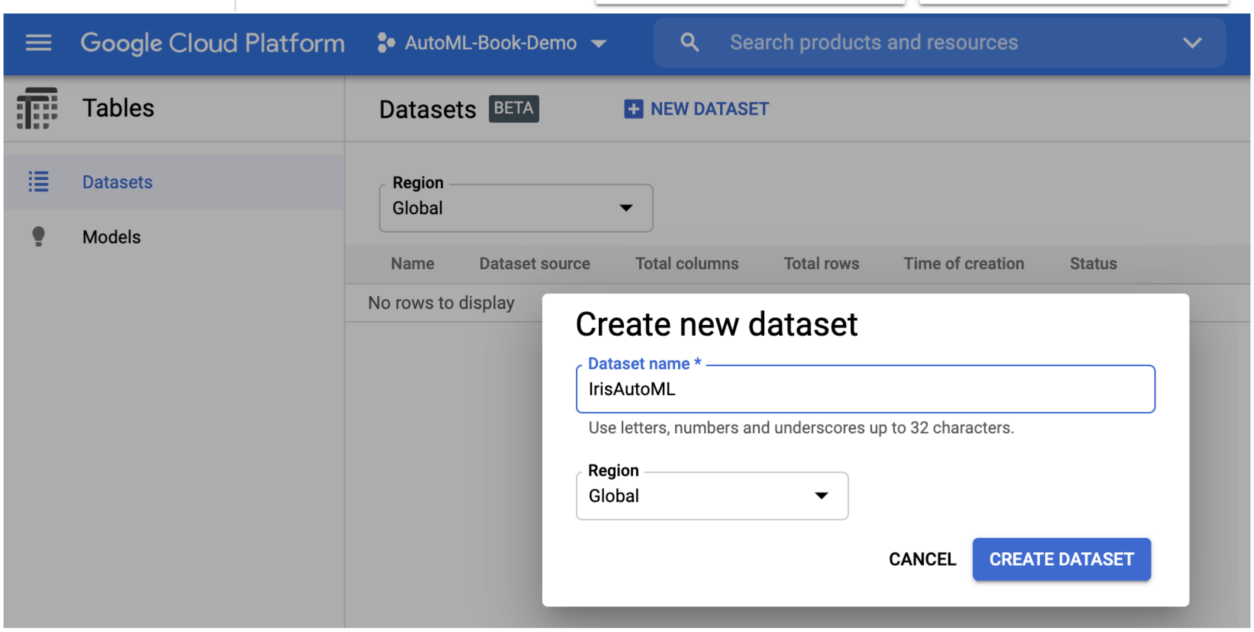
596a8dd0cef74aac-dot-us-east1.notebooks.googleusercontent.com/lab?authuser=0
File Edit View Run Kernel Git Tabs Settings Help
/ tutorials /
Name
Last Modified
bigquery
2 minutes ago
cloud-ml-engine
2 minutes ago
fairing
2 minutes ago
storage
2 minutes ago
HelloWorld.ipynb
a minute ago
HelloWorld.ipynb
Code
git
[1]: print ("Hello Automated Machine Learning World")
Hello Automated Machine Learning World

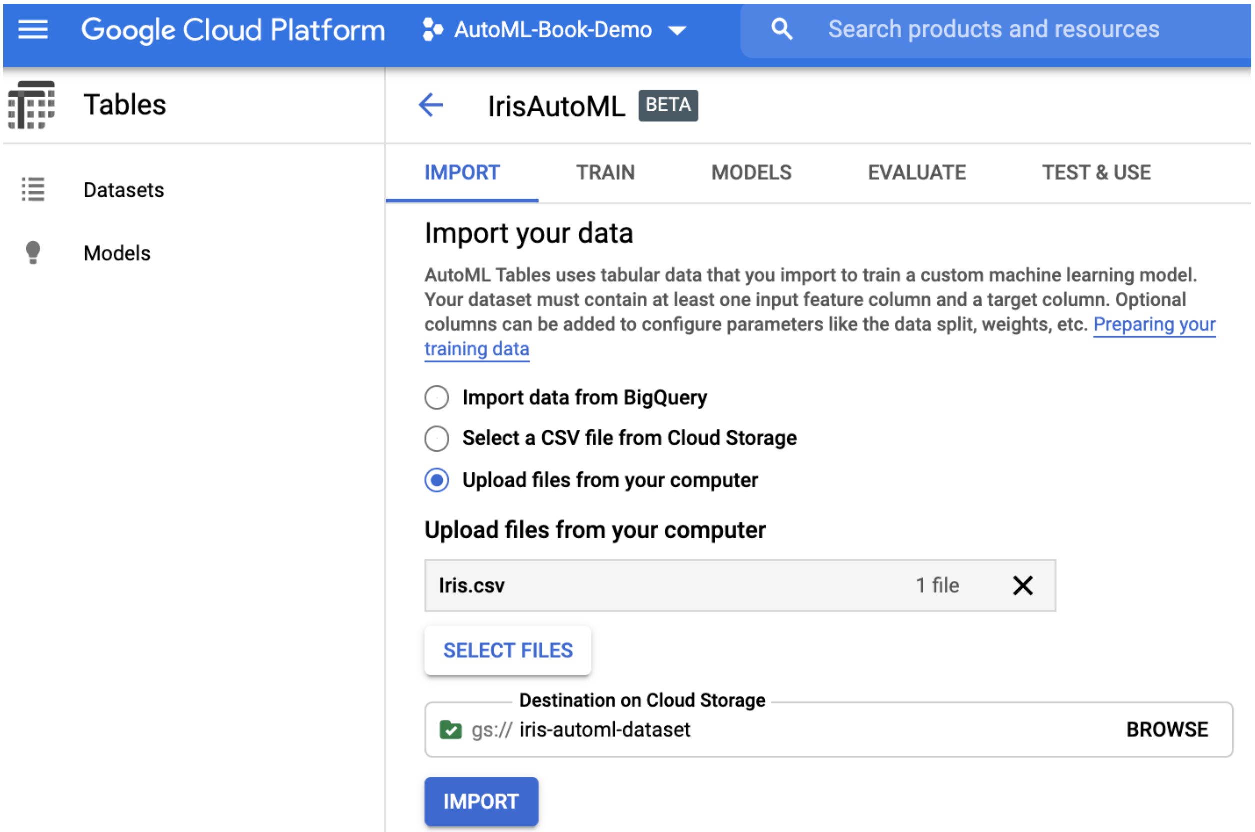
AutoML products

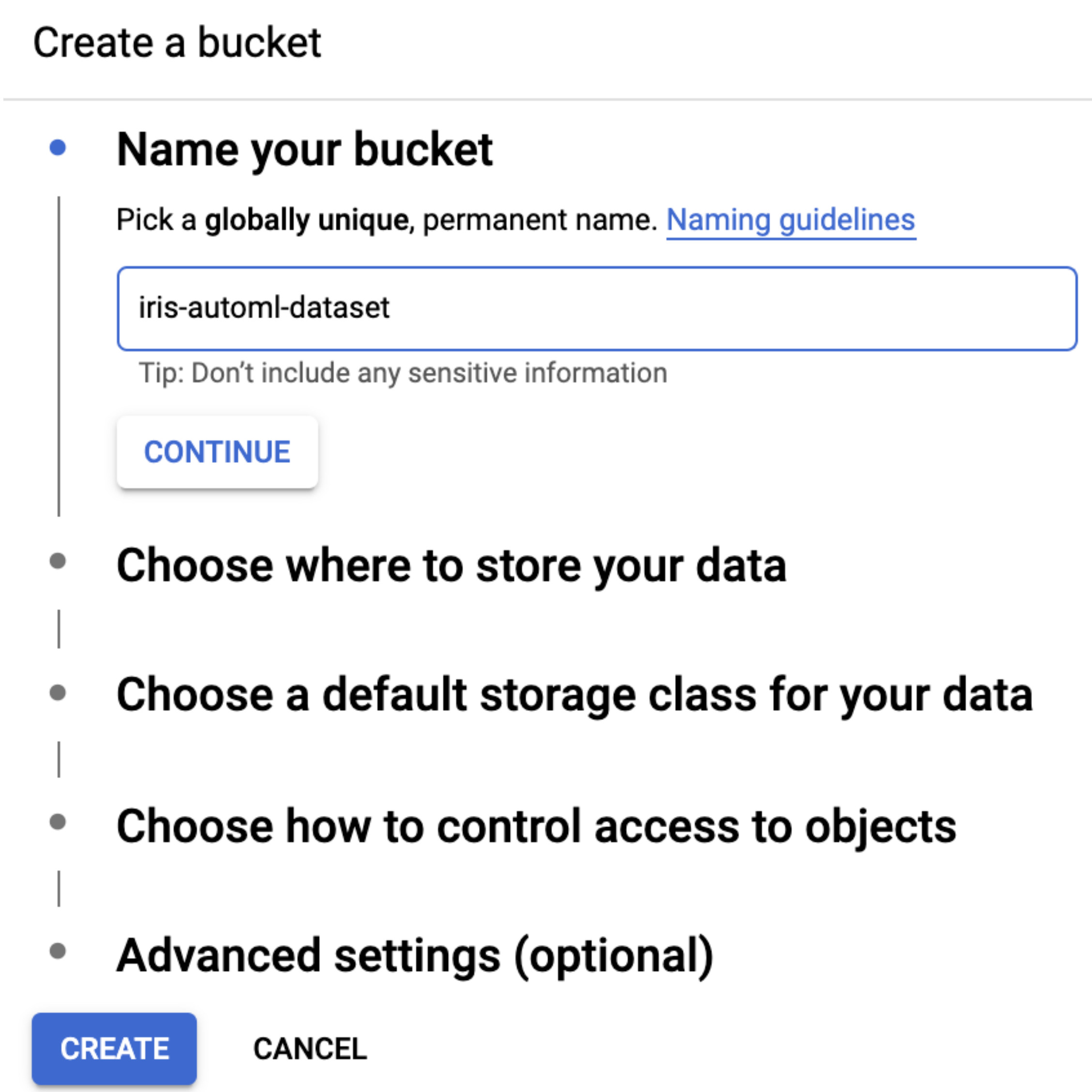


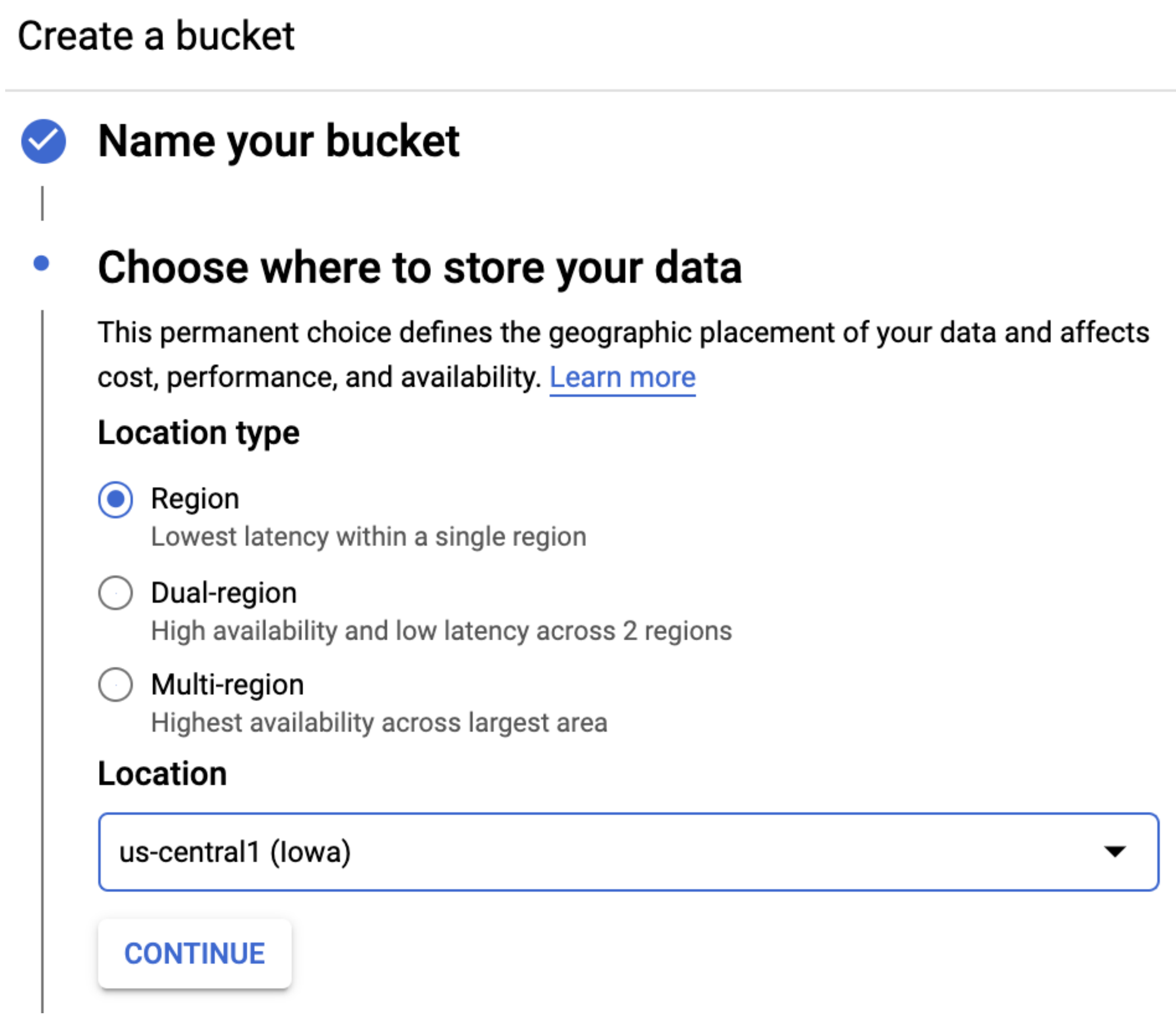
Chapter 9: Automated Machine Learning with GCP Cloud AutoML






Google Cloud Platform
AutoML-Book-Demo
Search products and resources
Tables
Datasets
Models
IrisAutoML
BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Import your data
AutoML Tables uses tabular data that you import to train a custom machine learning model. Your dataset must contain at least one input feature column and a target column. Optional columns can be added to configure parameters like the data split, weights, etc. Preparing your training data
Import data from BigQuery
Select a CSV file from Cloud Storage
Upload files from your computer
Upload files from your computer
Iris.csv
1 file
SELECT FILES
Destination on Cloud Storage
gs://
iris-automl-dataset
BROWSE
IMPORT


Create a bucket
Name your bucket
Pick a globally unique, permanent name. Naming guidelines
iris-automl-dataset
Tip: Don't include any sensitive information
CONTINUE
Choose where to store your data
Choose a default storage class for your data
Choose how to control access to objects
Advanced settings (optional)
CREATE
CANCEL


Create a bucket
Name your bucket
Choose where to store your data
This permanent choice defines the geographic placement of your data and affects cost, performance, and availability. Learn more
Location type
Region
Lowest latency within a single region
Dual-region
High availability and low latency across 2 regions
Multi-region
Highest availability across largest area
Location
us-central1 (Iowa)
CONTINUE

Create a bucket

Name your bucket

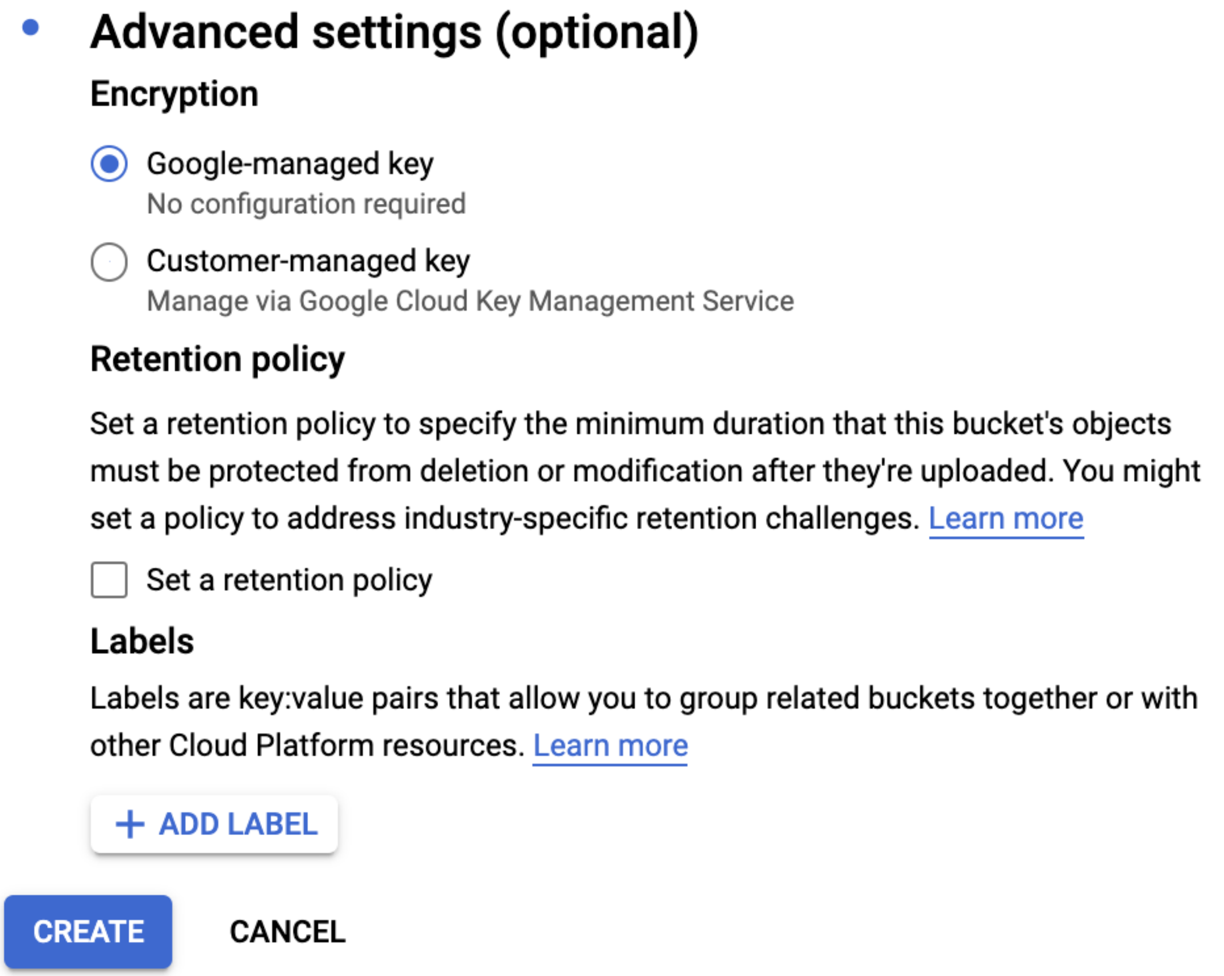
Choose where to store your data

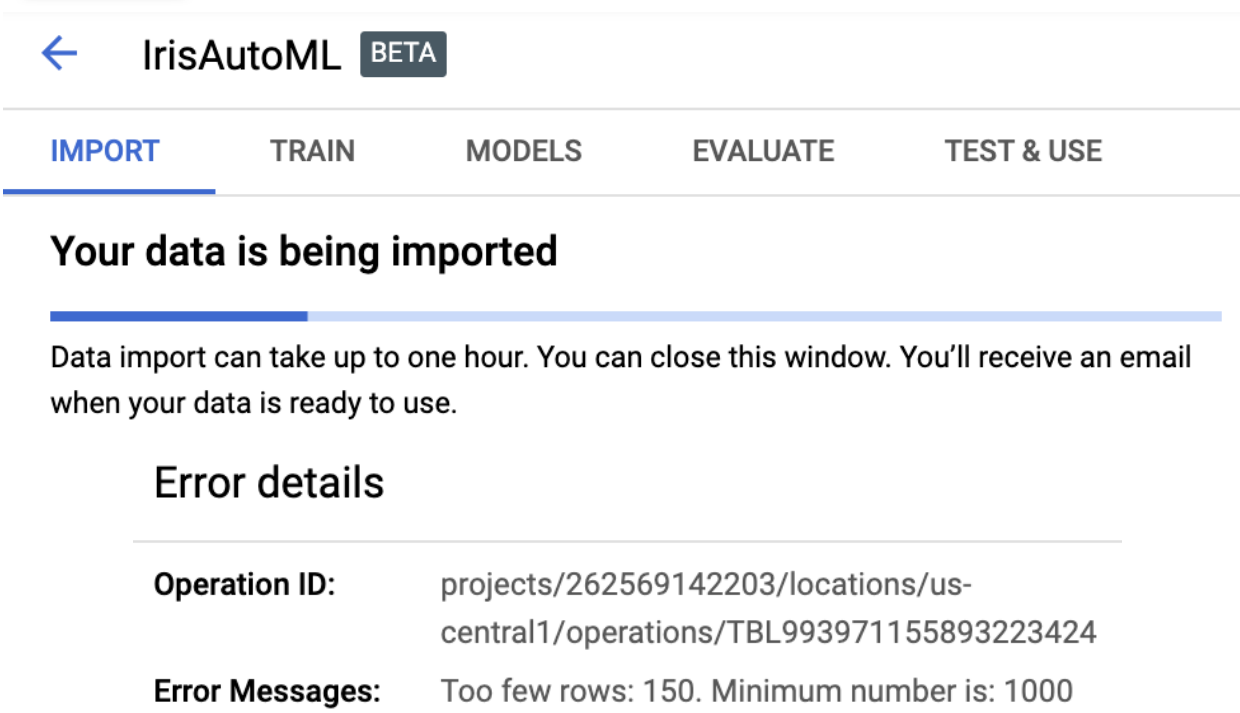
Choose a default storage class for your data

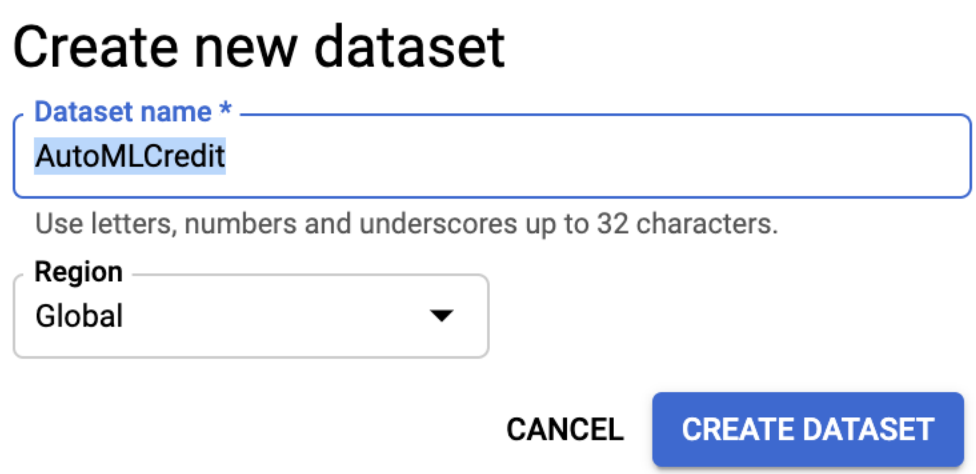
A storage class sets costs for storage, retrieval, and operations. Pick a default storage class based on how long you plan to store your data and how often it will be accessed. Learn more

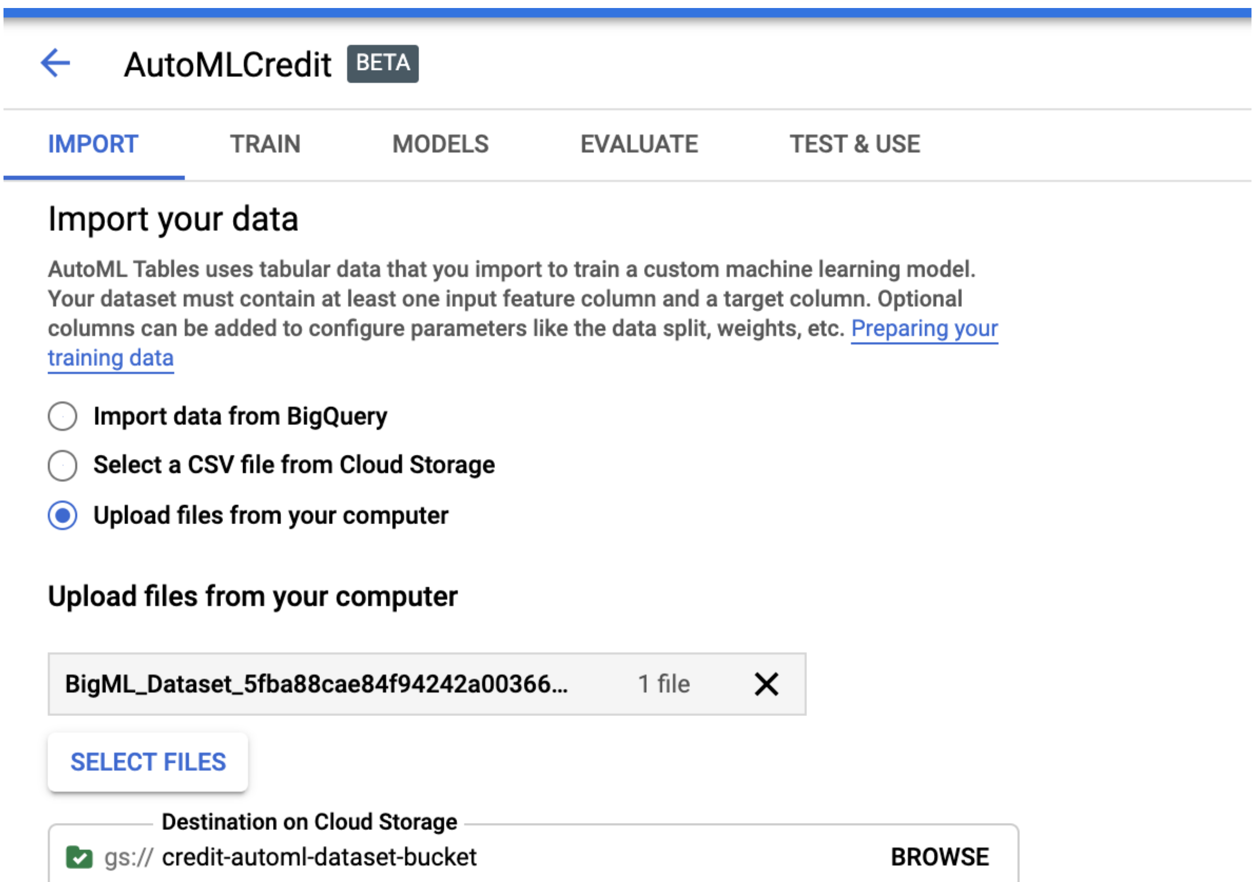
- (●) Standard
  Best for short-term storage and frequently accessed data
- ( ) Nearline
  Best for backups and data accessed less than once a month
- ( ) Coldline
  Best for disaster recovery and data accessed less than once a quarter
- ( ) Archive
  Best for long-term digital preservation of data accessed less than once a year

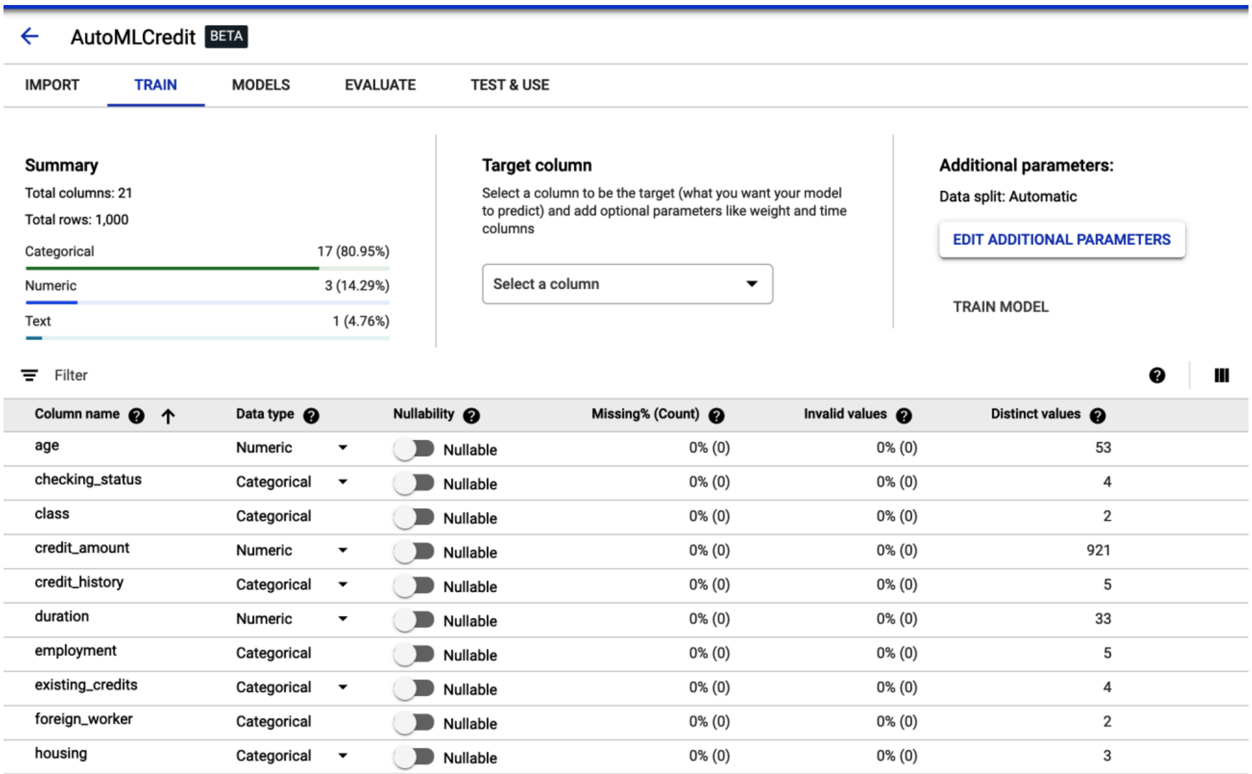
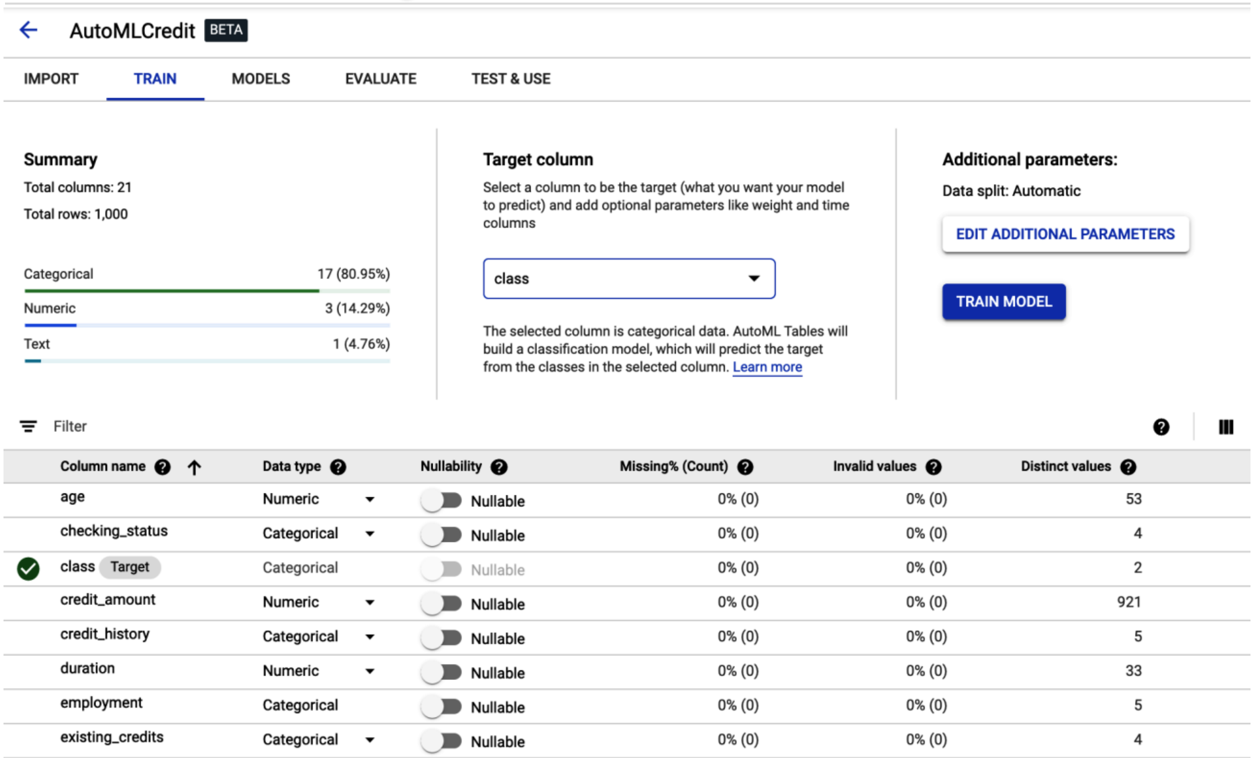
CONTINUE


Advanced settings (optional)
Encryption
Google-managed key
No configuration required
Customer-managed key
Manage via Google Cloud Key Management Service
Retention policy
Set a retention policy to specify the minimum duration that this bucket's objects must be protected from deletion or modification after they're uploaded. You might set a policy to address industry-specific retention challenges. Learn more
Set a retention policy
Labels
Labels are key:value pairs that allow you to group related buckets together or with other Cloud Platform resources. Learn more
+ ADD LABEL
CREATE
CANCEL


IrisAutoML
BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Your data is being imported
Data import can take up to one hour. You can close this window. You'll receive an email when your data is ready to use.
Error details
Operation ID:
projects/262569142203/locations/us-central1/operations/TBL993971155893223424
Error Messages:
Too few rows: 150. Minimum number is: 1000


Create new dataset
Dataset name *
AutoMLCredit
Use letters, numbers and underscores up to 32 characters.
Region
Global
CANCEL
CREATE DATASET


AutoMLCredit
BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Import your data
AutoML Tables uses tabular data that you import to train a custom machine learning model. Your dataset must contain at least one input feature column and a target column. Optional columns can be added to configure parameters like the data split, weights, etc. Preparing your training data
Import data from BigQuery
Select a CSV file from Cloud Storage
Upload files from your computer
Upload files from your computer
BigML_Dataset_5fba88cae84f94242a00366...
1 file
SELECT FILES
Destination on Cloud Storage
gs:// credit-automl-dataset-bucket
BROWSE


AutoMLCredit BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Summary
Total columns: 21
Total rows: 1,000
Categorical 17 (80.95%)
Numeric 3 (14.29%)
Text 1 (4.76%)
Target column
Select a column to be the target (what you want your model to predict) and add optional parameters like weight and time columns
Select a column
Additional parameters:
Data split: Automatic
EDIT ADDITIONAL PARAMETERS
TRAIN MODEL
Filter
Column name | Data type | Nullability | Missing% (Count) | Invalid values | Distinct values
age | Numeric | Nullable | 0% (0) | 0% (0) | 53
checking_status | Categorical | Nullable | 0% (0) | 0% (0) | 4
class | Categorical | Nullable | 0% (0) | 0% (0) | 2
credit_amount | Numeric | Nullable | 0% (0) | 0% (0) | 921
credit_history | Categorical | Nullable | 0% (0) | 0% (0) | 5
duration | Numeric | Nullable | 0% (0) | 0% (0) | 33
employment | Categorical | Nullable | 0% (0) | 0% (0) | 5
existing_credits | Categorical | Nullable | 0% (0) | 0% (0) | 4
foreign_worker | Categorical | Nullable | 0% (0) | 0% (0) | 2
housing | Categorical | Nullable | 0% (0) | 0% (0) | 3

AutoMLCredit BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Summary
Total columns: 21
Total rows: 1,000
Categorical 17 (80.95%)
Numeric 3 (14.29%)
Text 1 (4.76%)
Target column
Select a column to be the target (what you want your model to predict) and add optional parameters like weight and time columns
class
The selected column is categorical data. AutoML Tables will build a classification model, which will predict the target from the classes in the selected column. Learn more
Additional parameters:
Data split: Automatic
EDIT ADDITIONAL PARAMETERS
TRAIN MODEL
Filter
Column name | Data type | Nullability | Missing% (Count) | Invalid values | Distinct values
age | Numeric | Nullable | 0% (0) | 0% (0) | 53
checking_status | Categorical | Nullable | 0% (0) | 0% (0) | 4
class Target | Categorical | Nullable | 0% (0) | 0% (0) | 2
credit_amount | Numeric | Nullable | 0% (0) | 0% (0) | 921
credit_history | Categorical | Nullable | 0% (0) | 0% (0) | 5
duration | Numeric | Nullable | 0% (0) | 0% (0) | 33
employment | Categorical | Nullable | 0% (0) | 0% (0) | 5
existing_credits | Categorical | Nullable | 0% (0) | 0% (0) | 4

Train your model

Model name *

AutoMLCredit_20201122111615

Training budget

Enter a number between 1 and 72 for the maximum number of node hours to spend training your model. If your model stops improving before then, AutoML Tables will stop training and you'll only be charged for the actual node hours used. Training budget doesn't include setup, preprocessing, and tear down. These steps usually don't exceed one hour total and you won't be charged for that time. Training pricing guide

Budget *

5 maximum node hours

Input feature selection

By default, all other columns in your dataset will be used as input features for training (excluding target, weight, and split columns).

20 feature columns *

All columns selected

Summary

Model type: Binary classification model

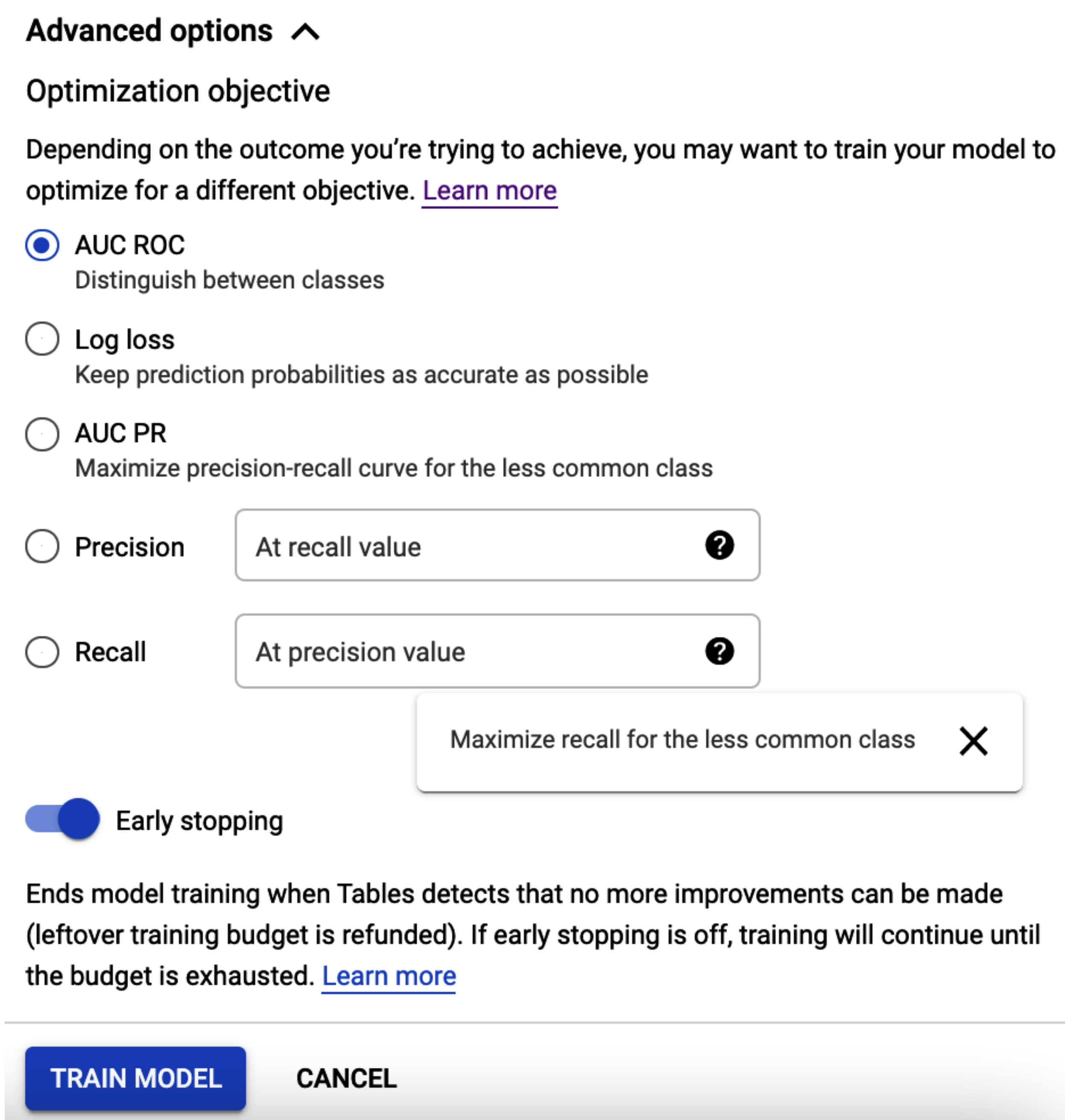
Data split: Automatic

Target: class

Input features: 20 features

Rows: 1,000 rows

Rows	Suggested training time
Less than 100,000	1-3 hours
100,000 - 1,000,000	1-6 hours
1,000,000 - 10,000,000	1-12 hours
More than 10,000,000	3 - 24 hours


Advanced options
Optimization objective
Depending on the outcome you're trying to achieve, you may want to train your model to optimize for a different objective. Learn more
AUC ROC
Distinguish between classes
Log loss
Keep prediction probabilities as accurate as possible
AUC PR
Maximize precision-recall curve for the less common class
Precision
At recall value
Recall
At precision value
Maximize recall for the less common class
Early stopping
Ends model training when Tables detects that no more improvements can be made (leftover training budget is refunded). If early stopping is off, training will continue until the budget is exhausted. Learn more
TRAIN MODEL
CANCEL

AutoMLCredit BETA

IMPORT | TRAIN | MODELS | EVALUATE | TEST & USE

Models

AutoMLCredit_20201122111853

Training may take several hours. This includes node training time as well as infrastructure set up and tear down, which you aren't charged for.

You will be emailed once training completes.

Infrastructure setting up

CANCEL

Binary classification model

AutoMLCredit_20201122022635

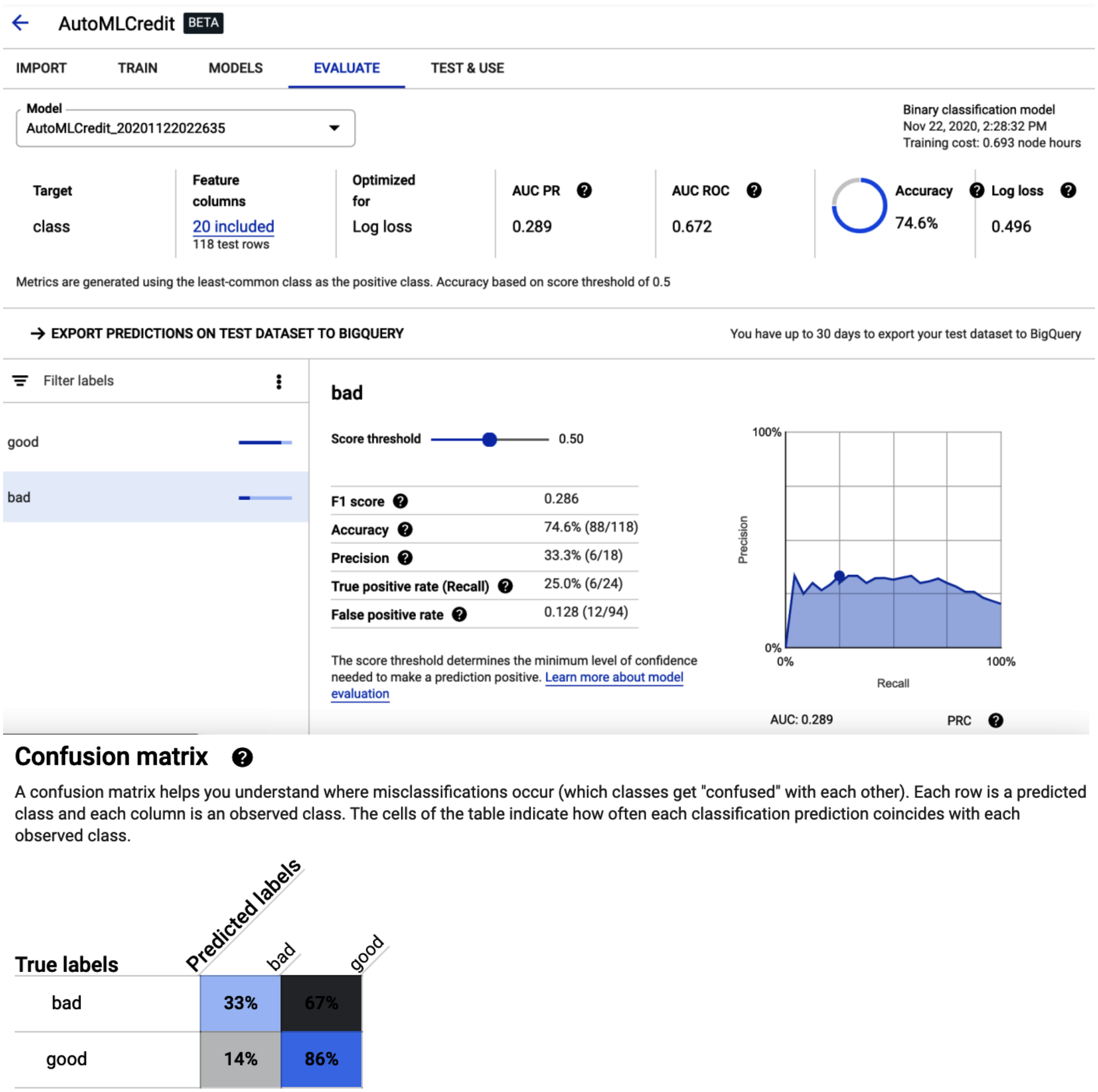
AUC PR

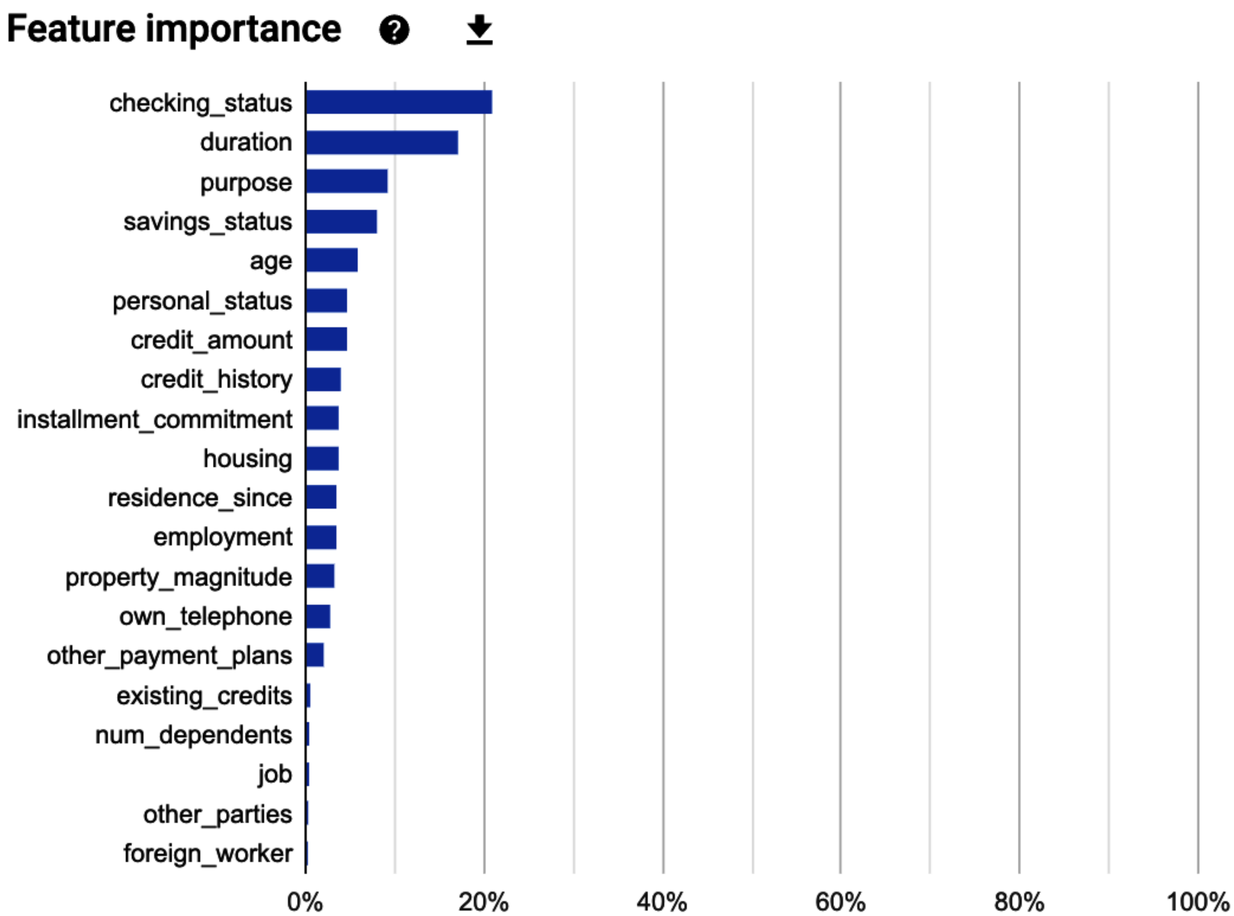
0.289

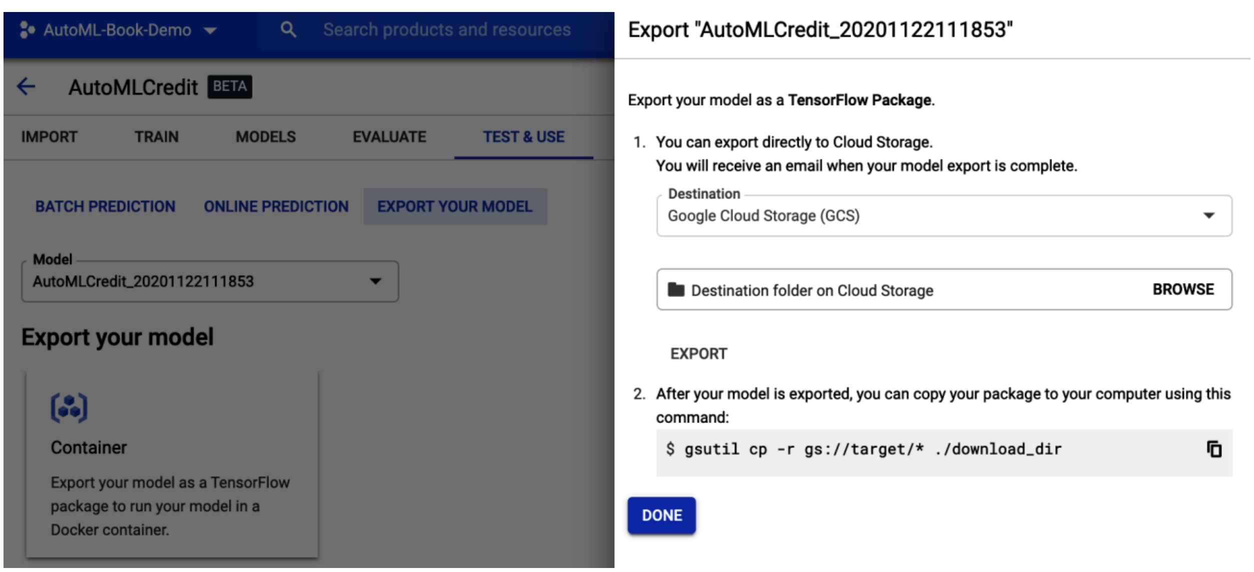
AUC ROC	0.672
Accuracy	74.58%
Log loss	0.496

Metrics are generated based on the less common label being the positive class.
Accuracy is based on a score threshold of 0.5

Model ID	TBL7217873822907629568
Created on	Nov 22, 2020, 2:28:32 PM
Target	class
Feature columns	20 included
Test rows	118
Optimization objective	Log loss
Training cost	0.693 node hours
Model hyperparameters	Model Trials
Status	Not deployed


AutoMLCredit BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
Model
AutoMLCredit_20201122022635
Binary classification model
Nov 22, 2020, 2:28:32 PM
Training cost: 0.693 node hours
Target
class
Feature columns
20 included
118 test rows
Optimized for
Log loss
AUC PR
0.289
AUC ROC
0.672
Accuracy
74.6%
Log loss
0.496
Metrics are generated using the least-common class as the positive class. Accuracy based on score threshold of 0.5
EXPORT PREDICTIONS ON TEST DATASET TO BIGQUERY
You have up to 30 days to export your test dataset to BigQuery
Filter labels
good
bad
bad
Score threshold
0.50
F1 score 0.286
Accuracy 74.6% (88/118)
Precision 33.3% (6/18)
True positive rate (Recall) 25.0% (6/24)
False positive rate 0.128 (12/94)
The score threshold determines the minimum level of confidence needed to make a prediction positive. Learn more about model evaluation
100%
0%
Precision
0%
100%
Recall
AUC: 0.289
PRC
Confusion matrix
A confusion matrix helps you understand where misclassifications occur (which classes get "confused" with each other). Each row is a predicted class and each column is an observed class. The cells of the table indicate how often each classification prediction coincides with each observed class.
True labels
Predicted labels
bad
good
bad
33%
67%
good
14%
86%


Feature importance
checking_status
duration
purpose
savings_status
age
personal_status
credit_amount
credit_history
installment_commitment
housing
residence_since
employment
property_magnitude
own_telephone
other_payment_plans
existing_credits
num_dependents
job
other_parties
foreign_worker
0%
20%
40%
60%
80%
100%


AutoML-Book-Demo
Search products and resources
AutoMLCredit BETA
IMPORT
TRAIN
MODELS
EVALUATE
TEST & USE
BATCH PREDICTION
ONLINE PREDICTION
EXPORT YOUR MODEL
Model
AutoMLCredit_20201122111853
Export your model
Container
Export your model as a TensorFlow package to run your model in a Docker container.
Export "AutoMLCredit_20201122111853"
Export your model as a TensorFlow Package.
1. You can export directly to Cloud Storage.
You will receive an email when your model export is complete.
Destination
Google Cloud Storage (GCS)
Destination folder on Cloud Storage
BROWSE
EXPORT
2. After your model is exported, you can copy your package to your computer using this command:
$ gsutil cp -r gs://target/* ./download_dir
DONE

BATCH PREDICTION | ONLINE PREDICTION | EXPORT YOUR MODEL

Model
AutoMLCredit_20201122111853

To use online prediction, deploy your model to the cloud. Deployment takes 10-15 minutes. Once your model is deployed, charges are per hour and depend on model size and number of machines used. (Your model is 760.590 MB) Learn more

DEPLOY MODEL

Online prediction

FEATURE COLUMN VIEW

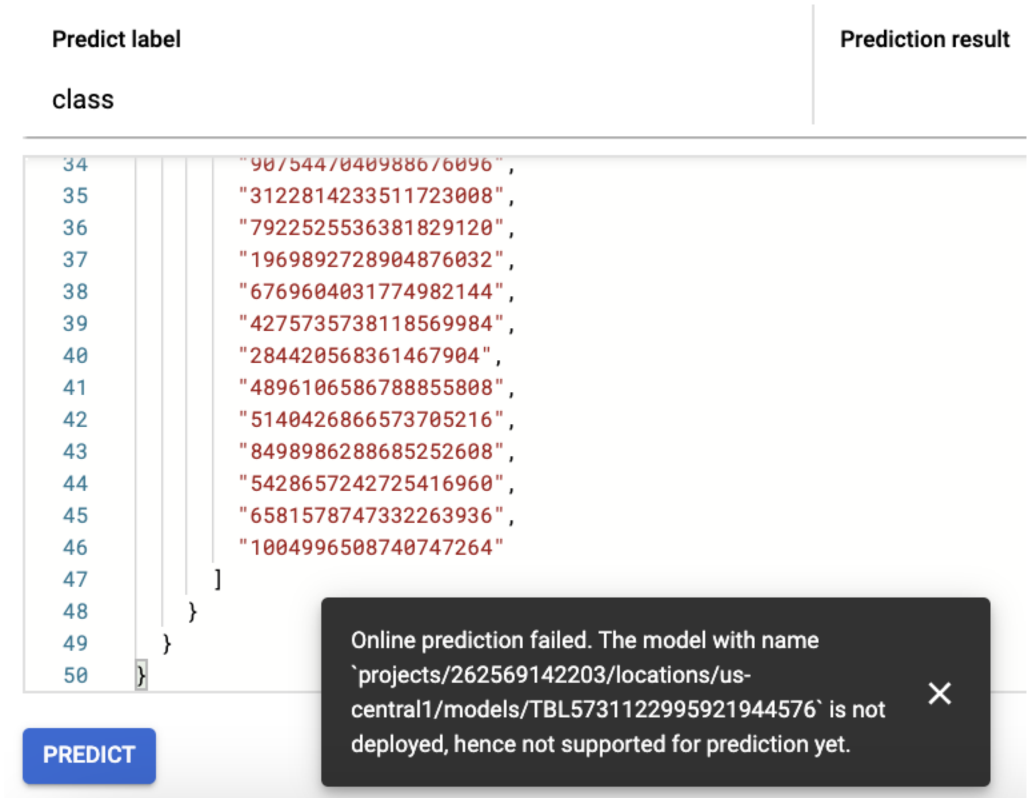
Online prediction deploys your model so you can send real-time REST requests to it. Online prediction is useful for time-sensitive predictions (for example, in response to an application request). Learn more

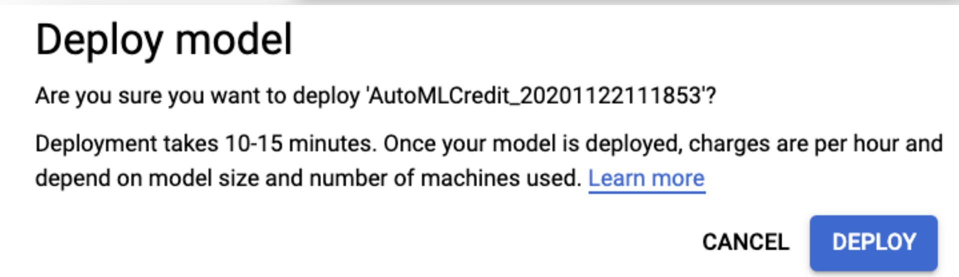
Online prediction pricing is based on the size of your model and the length of time your model is deployed. View pricing guide
Your model's endpoints are available as a JSON object. You can execute a query using the command line interface (CLI). Switch to JSON CODE VIEW to get a JSON request. Learn more

Predict label	Prediction result
class	

```
{
  "payload": {
    "row": {
      "values": [
        "48",
        "male single",
        "no known property",
        "none",
        "2",
        "10127",
        "1",
        "for free",
        "new car",
        "no checking",
        "bank",
        "2",
```

PREDICT


Predict label
class
Prediction result
"3122814233511723008",
"7922525536381829120",
"1969892728904876032",
"6769604031774982144",
"4275735738118569984",
"2844205683614679040",
"4896106586788855808",
"5140426866573705216",
"8498986288685252608",
"5428657242725416960",
"6581578747332263936",
"1004996508740747264"
]
}
}
}
PREDICT
Online prediction failed. The model with name `projects/262569142203/locations/us-central1/models/TBL5731122995921944576` is not deployed, hence not supported for prediction yet.


Deploy model
Are you sure you want to deploy 'AutoMLCredit_20201122111853'?
Deployment takes 10-15 minutes. Once your model is deployed, charges are per hour and depend on model size and number of machines used. Learn more
CANCEL
DEPLOY

Your model was deployed and is available for online prediction requests. Your model size is 760.590 MB. Learn more

REMOVE DEPLOYMENT

Online prediction

FEATURE COLUMN VIEW

Online prediction deploys your model so you can send real-time REST requests to it. Online prediction is useful for time-sensitive predictions (for example, in response to an application request). Learn more

Online prediction pricing is based on the size of your model and the length of time your model is deployed. View pricing guide
Your model's endpoints are available as a JSON object. You can execute a query using the command line interface (CLI). Switch to JSON CODE VIEW to get a JSON request. Learn more

Predict label

class

Prediction result


```
          "9075447040988676096",
          "3122814233511723008",
          "7922525536381829120",
          "1969892728904876032",
          "6769604031774982144",
          "4275735738118569984",
          "284420568361467904",
          "4896106586788855808",
          "5140426866573705216",
          "8498986288685252608",
          "5428657242725416960",
          "6581578747332263936",
          "1004996508740747264"
        ]
      }
    }
}
```


PREDICT

Online prediction

FEATURE COLUMN VIEW

Online prediction deploys your model so you can send real-time REST requests to it. Online prediction is useful for time-sensitive predictions (for example, in response to an application request). Learn more

Online prediction pricing is based on the size of your model and the length of time your model is deployed. View pricing guide
Your model's endpoints are available as a JSON object. You can execute a query using the command line interface (CLI). Switch to JSON CODE VIEW to get a JSON request. Learn more

Predict label

class

Prediction result

good

Confidence score: 0.661

bad

Confidence score: 0.339


```
    "row": {
      "values": [
        "48",
        "48",
        "male single",
        "no known property",
        "none",
        "2",
        "10127",
        "1",
        "for free",
        "new car",
        "no checking",
        "bank",
        "2",
        "1<=X<4",
        "none",
```


PREDICT

Predict label	Prediction result
class	Baseline prediction value: 0.162 **good** Confidence score: 0.462 **bad** Confidence score: 0.538

Feature column name	Column ID	Data type	Status ↓	Value	Local feature importance
age	1004996508740747264	Numeric	Required	18	0.105
checking_status	7922525536381829120	Categorical	Required	no checking	0.000
credit_amount	4463761022561288192	Numeric	Required	10127	0.008
credit_history	5428657242725416960	Categorical	Required	critical/other existing credit	-0.048
duration	8887421756545957888	Numeric	Required	60	0.225
employment	4275735738118569984	Categorical	Required	1<=X<4	0.000
existing_credits	6581578747332263936	Categorical	Required	1	0.000
foreign_worker	5140426866573705216	Categorical	Required	yes	0.000
housing	9075447040988676096	Categorical	Required	for free	0.038
installment_commitment	5616682527168135168	Categorical	Required	1	-0.056

Rows per page: 10 ▾ 1 – 10 of 20

☑ Generate feature importance

PREDICT RESET

Create new dataset

Dataset name *

AutoMLIncome

Use letters, numbers and underscores up to 32 characters.

Region

Global

European Union

CANCEL CREATE DATASET

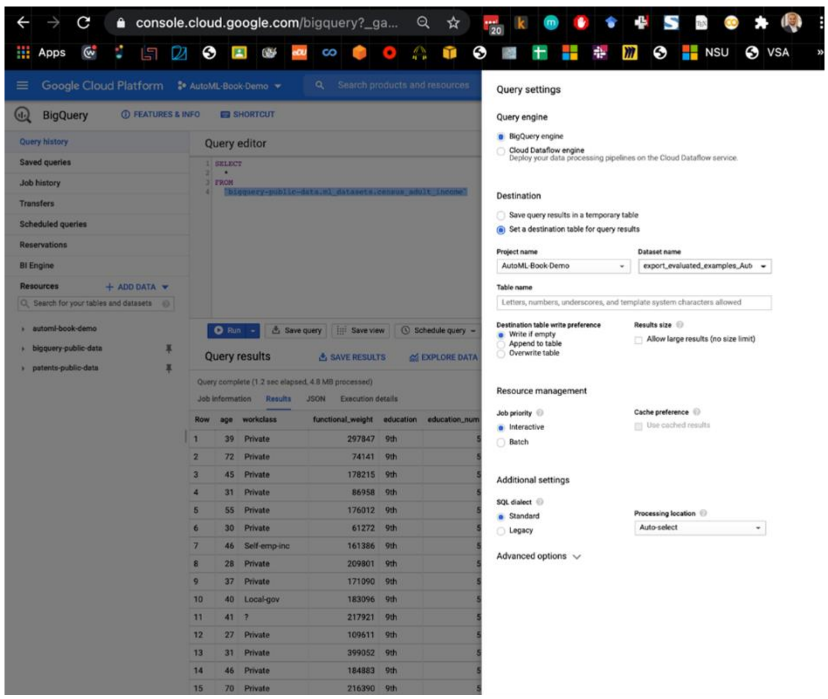
Add data to your dataset

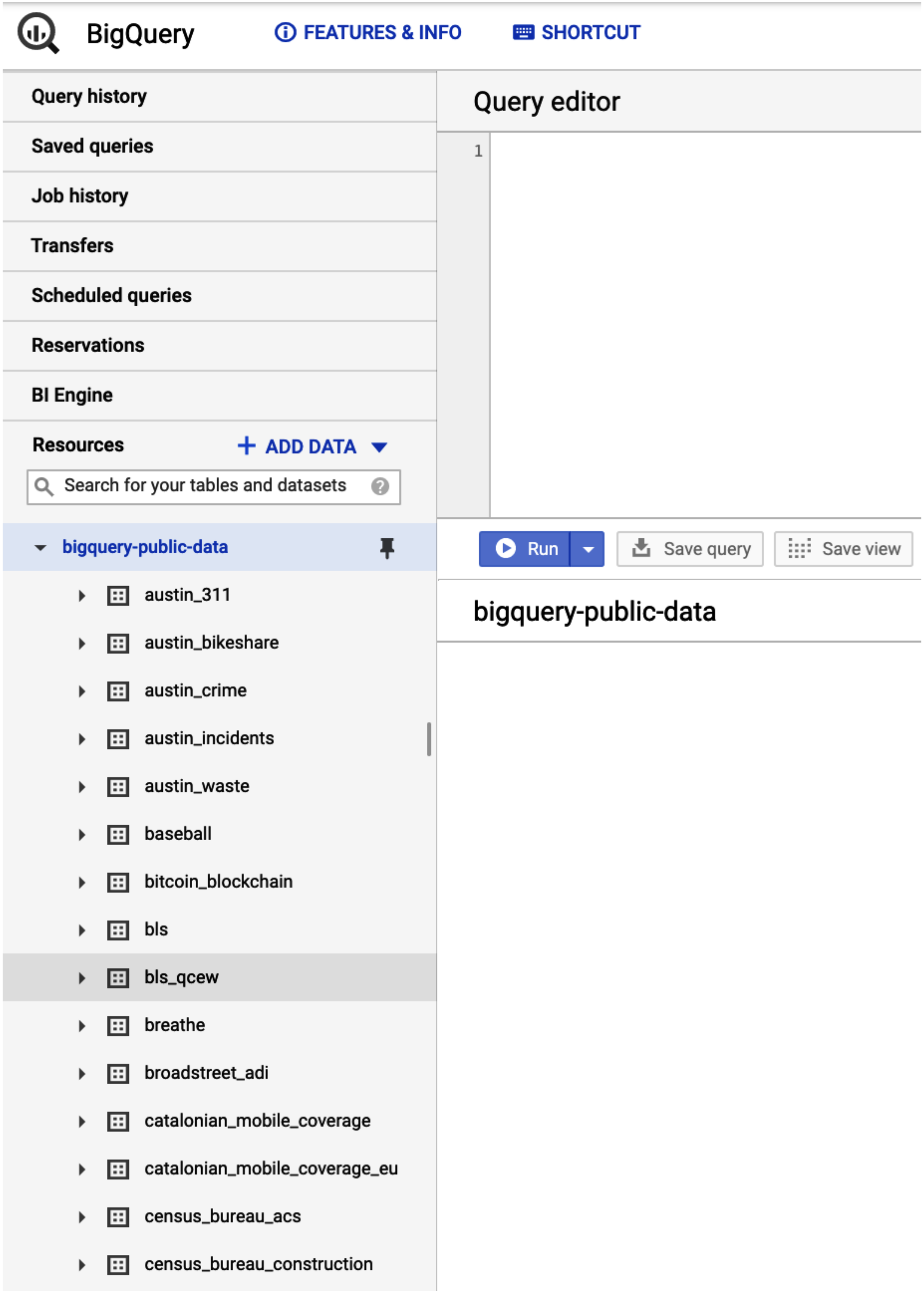
Before you begin, read the data guide to learn how to prepare your data. Then choose a data source:

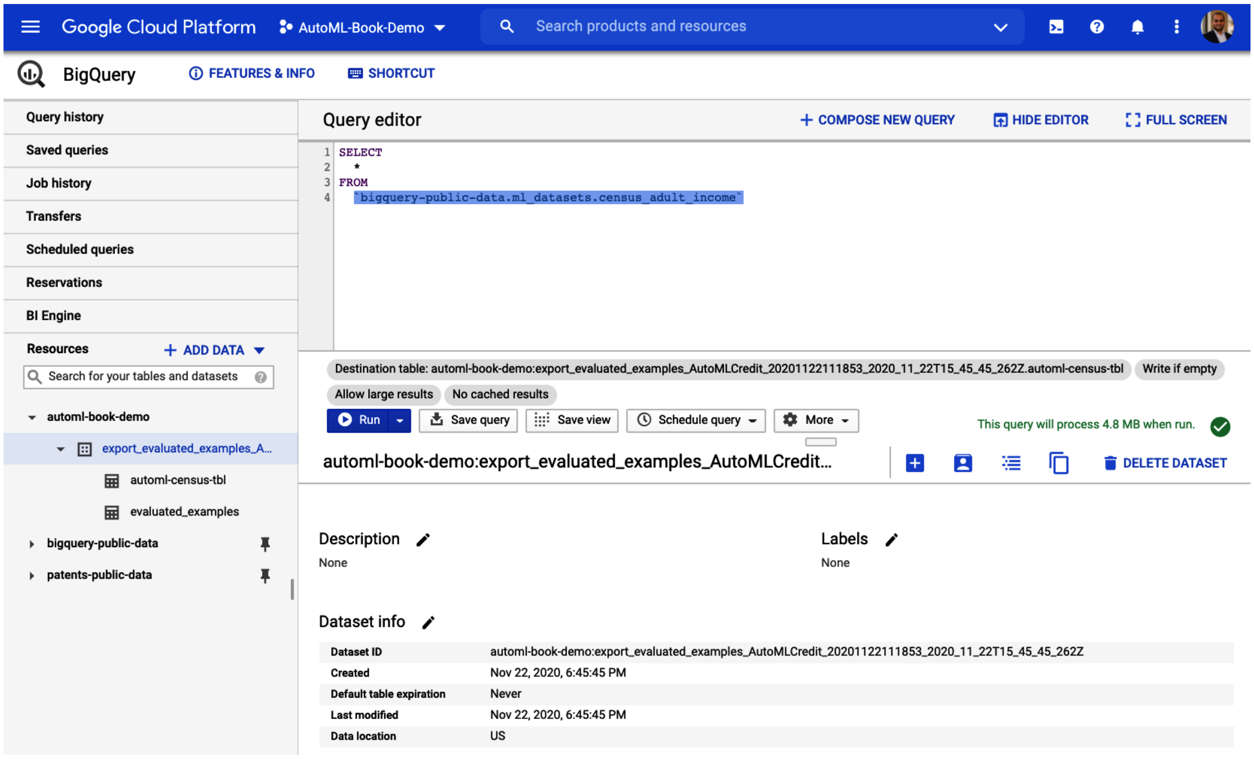
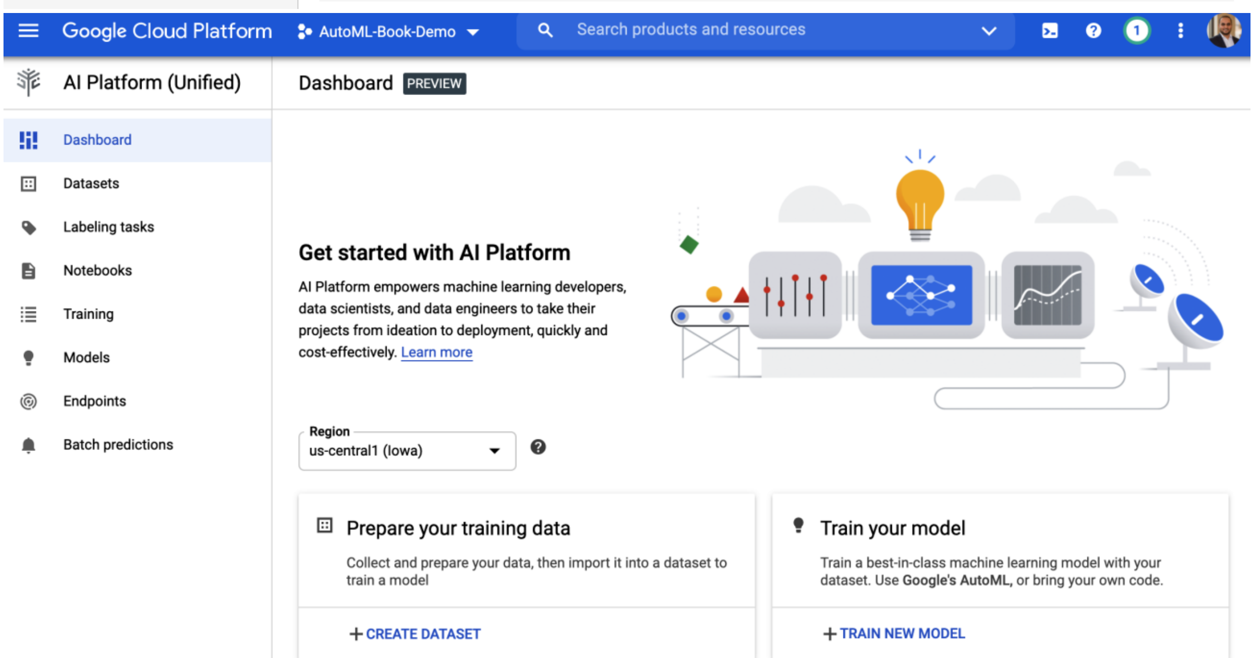
- **CSV file**: Can be uploaded from your computer or on Cloud Storage. Learn more
- **Bigquery**: Select a table or view from BigQuery. Learn more

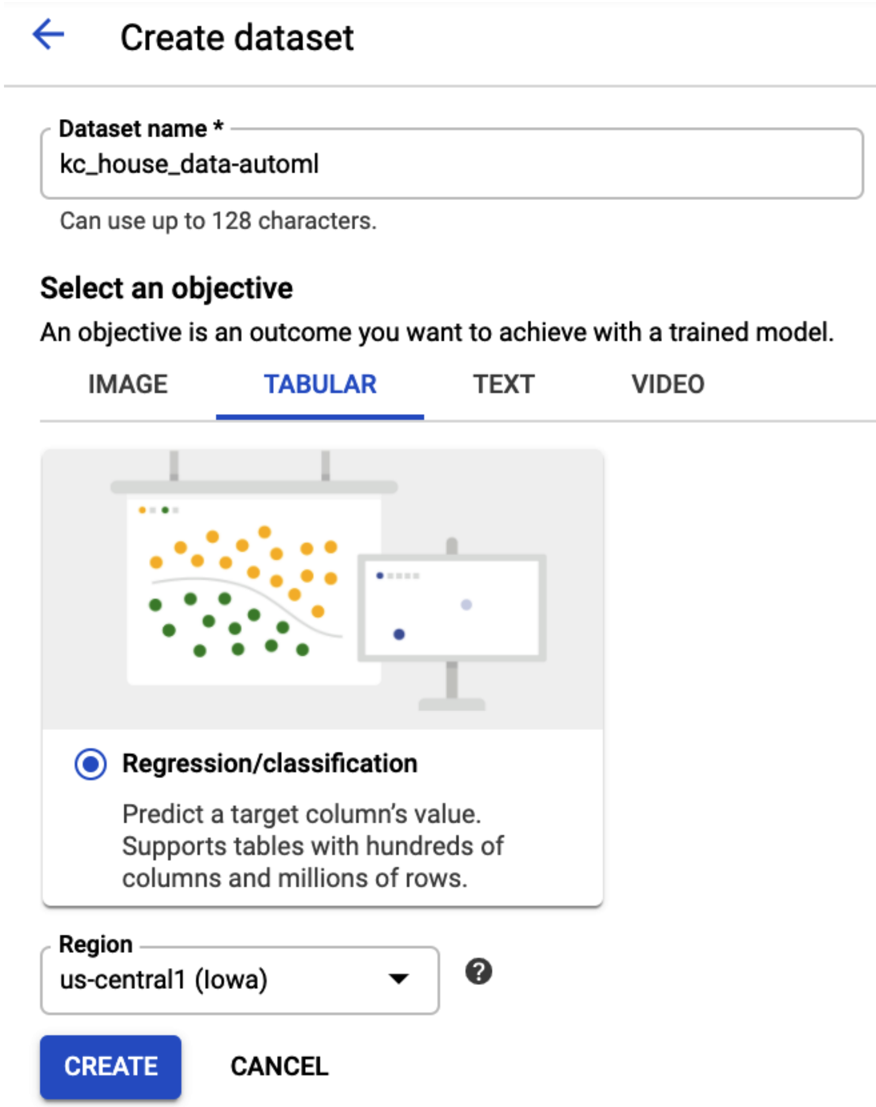
Select a data source

- ○ Upload CSV files from your computer
- ○ Select CSV files from Cloud Storage
- ◉ Select a table or view from BigQuery




BigQuery
FEATURES & INFO
SHORTCUT
Query history
Saved queries
Job history
Transfers
Scheduled queries
Reservations
BI Engine
Resources
ADD DATA
Search for your tables and datasets
bigquery-public-data
austin_311
austin_bikeshare
austin_crime
austin_incidents
austin_waste
baseball
bitcoin_blockchain
bls
bls_qcew
breathe
broadstreet_adi
catalonian_mobile_coverage
catalonian_mobile_coverage_eu
census_bureau_acs
census_bureau_construction
Query editor
1
Run
Save query
Save view
bigquery-public-data


Google Cloud Platform
AutoML-Book-Demo
Search products and resources
BigQuery
FEATURES & INFO
SHORTCUT
Query history
Saved queries
Job history
Transfers
Scheduled queries
Reservations
BI Engine
Resources
ADD DATA
Search for your tables and datasets
automl-book-demo
export_evaluated_examples_A...
automl-census-tbl
evaluated_examples
bigquery-public-data
patents-public-data
Query editor
COMPOSE NEW QUERY
HIDE EDITOR
FULL SCREEN
SELECT
*
FROM
`bigquery-public-data.ml_datasets.census_adult_income`
Destination table: automl-book-demo:export_evaluated_examples_AutoMLCredit_20201122111853_2020_11_22T15_45_45_262Z.automl-census-tbl
Write if empty
Allow large results
No cached results
Run
Save query
Save view
Schedule query
More
This query will process 4.8 MB when run.
automl-book-demo:export_evaluated_examples_AutoMLCredit...
DELETE DATASET
Description
None
Labels
None
Dataset info
Dataset ID
automl-book-demo:export_evaluated_examples_AutoMLCredit_20201122111853_2020_11_22T15_45_45_262Z
Created
Nov 22, 2020, 6:45:45 PM
Default table expiration
Never
Last modified
Nov 22, 2020, 6:45:45 PM
Data location
US

Google Cloud Platform
AutoML-Book-Demo
Search products and resources
AI Platform (Unified)
Dashboard
PREVIEW
Dashboard
Datasets
Labeling tasks
Notebooks
Training
Models
Endpoints
Batch predictions
Get started with AI Platform
AI Platform empowers machine learning developers, data scientists, and data engineers to take their projects from ideation to deployment, quickly and cost-effectively. Learn more
Region
us-central1 (Iowa)
Prepare your training data
Collect and prepare your data, then import it into a dataset to train a model
CREATE DATASET
Train your model
Train a best-in-class machine learning model with your dataset. Use Google's AutoML, or bring your own code.
TRAIN NEW MODEL


Create dataset
Dataset name *
kc_house_data-automl
Can use up to 128 characters.
Select an objective
An objective is an outcome you want to achieve with a trained model.
IMAGE
TABULAR
TEXT
VIDEO
Regression/classification
Predict a target column's value. Supports tables with hundreds of columns and millions of rows.
Region
us-central1 (Iowa)
CREATE
CANCEL

kc_house_data-automl

SOURCE ANALYZE

Add data to your dataset

Before you begin, read the data guide to learn how to prepare your data. Then choose a data source:

- **CSV file**: Can be uploaded from your computer or on Cloud Storage. Learn more
- **Bigquery**: Select a table or view from BigQuery. Learn more

Select a data source

- (●) Upload CSV files from your computer
- ( ) Select CSV files from Cloud Storage
- ( ) Select a table or view from BigQuery

Upload CSV files from your computer

Add up to 500 CSV files per upload. The files will be stored in a new Cloud Storage bucket (charges apply). Data from multiple files will be referenced as one dataset.

kc_house_data.csv 1 file ✕

SELECT FILES

Select a Cloud Storage path

Choose where your uploaded CSV files will be stored (charges apply)

Cloud Storage path

gs:// automl-zillow-pricing-ds BROWSE

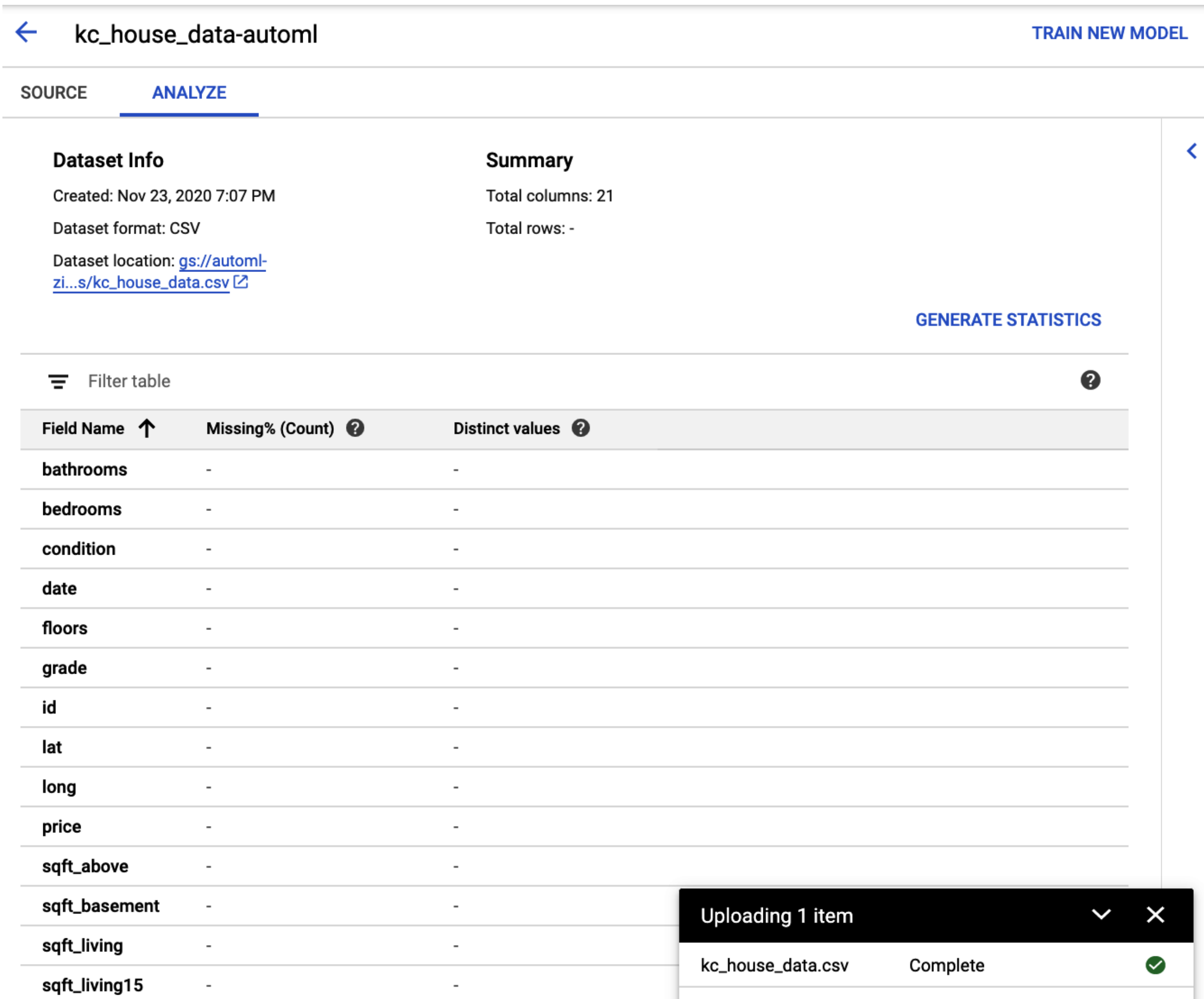
What happens next?

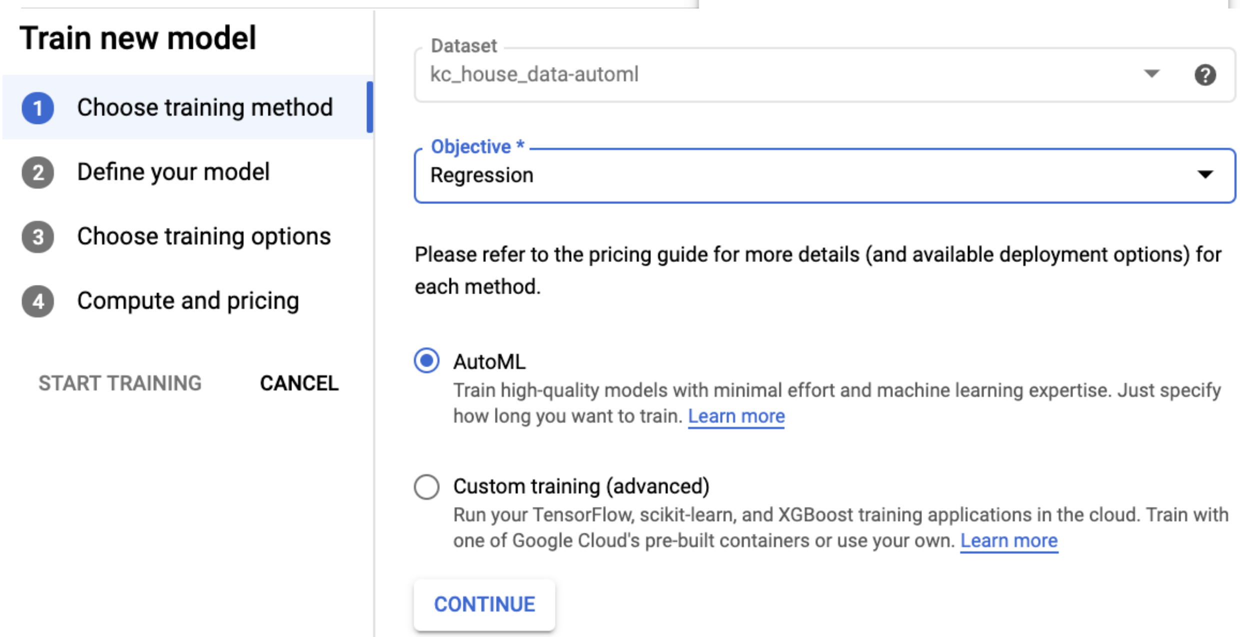
The CSV file data will be uploaded to Cloud Storage and associated with your dataset. Making changes to the referenced CSV files will affect the dataset before training.

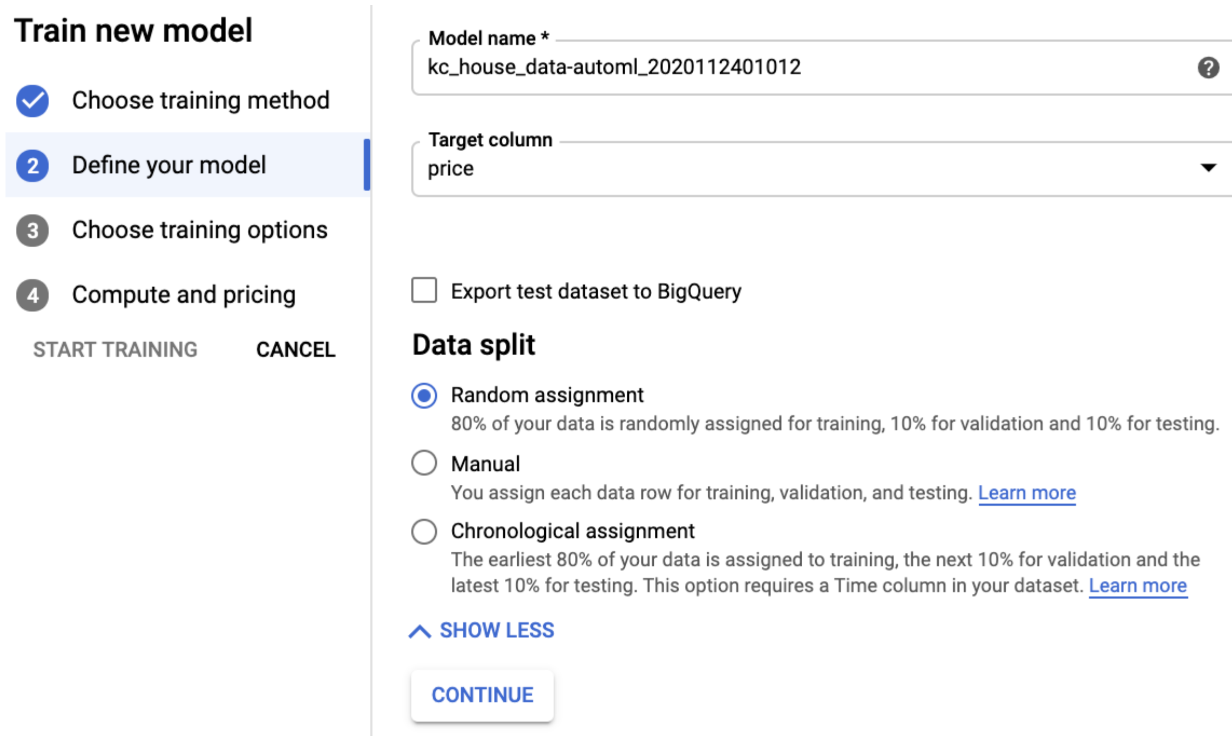
CONTINUE

You can build two model types with tabular data. The model type is automatically chosen based on the data type of your target column.

- **Regression models** predict a numeric value. For example, predicting home prices or consumer spending.
- **Classification models** predict a category from a fixed number of categories. Examples include predicting whether an email is spam or not, or classes a student might be interested in attending.


kc_house_data-automl
TRAIN NEW MODEL
SOURCE
ANALYZE
Dataset Info
Created: Nov 23, 2020 7:07 PM
Dataset format: CSV
Dataset location: gs://automl-zi...s/kc_house_data.csv
Summary
Total columns: 21
Total rows: -
GENERATE STATISTICS
Filter table
Field Name
Missing% (Count)
Distinct values
bathrooms - -
bedrooms - -
condition - -
date - -
floors - -
grade - -
id - -
lat - -
long - -
price - -
sqft_above - -
sqft_basement - -
sqft_living - -
sqft_living15 - -
Uploading 1 item
kc_house_data.csv
Complete


Train new model
1 Choose training method
2 Define your model
3 Choose training options
4 Compute and pricing
START TRAINING
CANCEL
Dataset
kc_house_data-automl
Objective *
Regression
Please refer to the pricing guide for more details (and available deployment options) for each method.
AutoML
Train high-quality models with minimal effort and machine learning expertise. Just specify how long you want to train. Learn more
Custom training (advanced)
Run your TensorFlow, scikit-learn, and XGBoost training applications in the cloud. Train with one of Google Cloud's pre-built containers or use your own. Learn more
CONTINUE


Train new model
Choose training method
2 Define your model
3 Choose training options
4 Compute and pricing
START TRAINING
CANCEL
Model name *
kc_house_data-automl_2020112401012
Target column
price
Export test dataset to BigQuery
Data split
Random assignment
80% of your data is randomly assigned for training, 10% for validation and 10% for testing.
Manual
You assign each data row for training, validation, and testing. Learn more
Chronological assignment
The earliest 80% of your data is assigned to training, the next 10% for validation and the latest 10% for testing. This option requires a Time column in your dataset. Learn more
SHOW LESS
CONTINUE

Train new model

- ✓ Choose training method
- ✓ Define your model
- 3 Choose training options
- 4 Compute and pricing

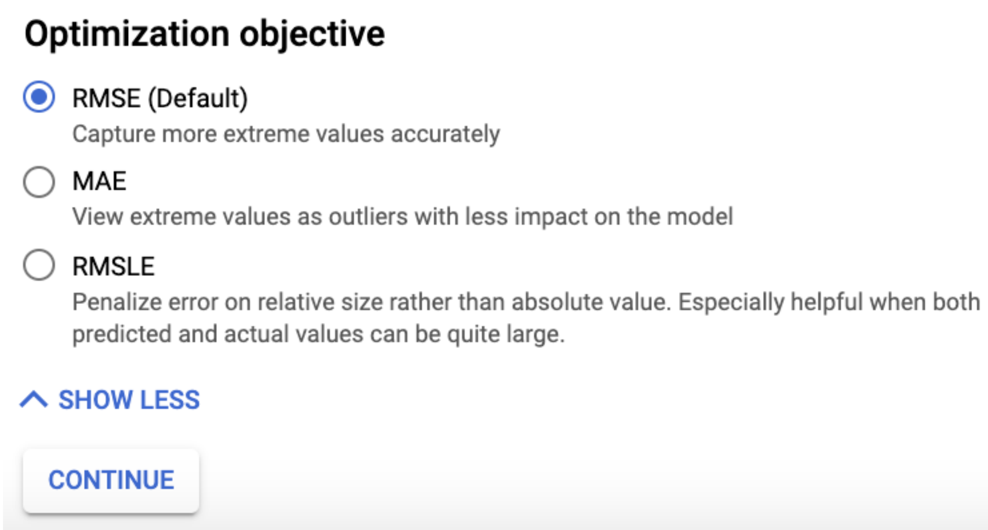
START TRAINING CANCEL

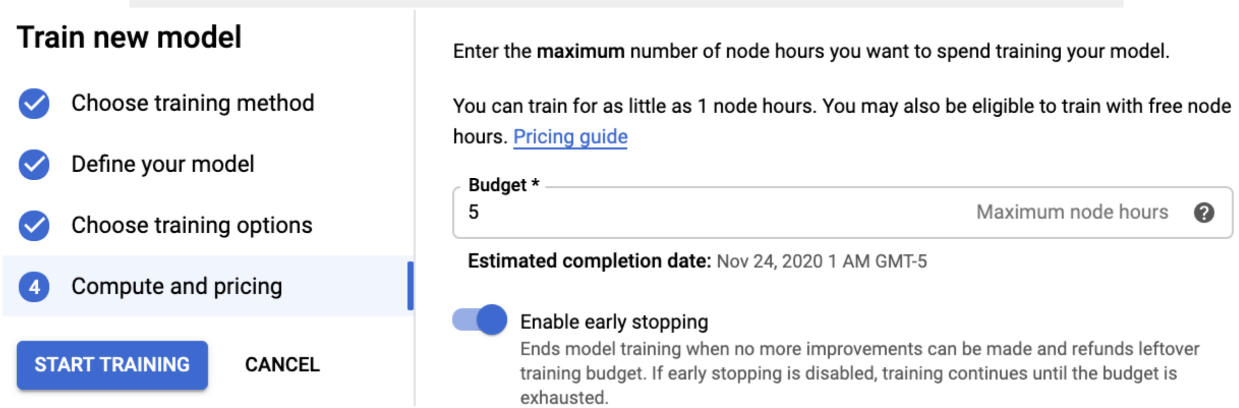
GENERATE STATISTICS

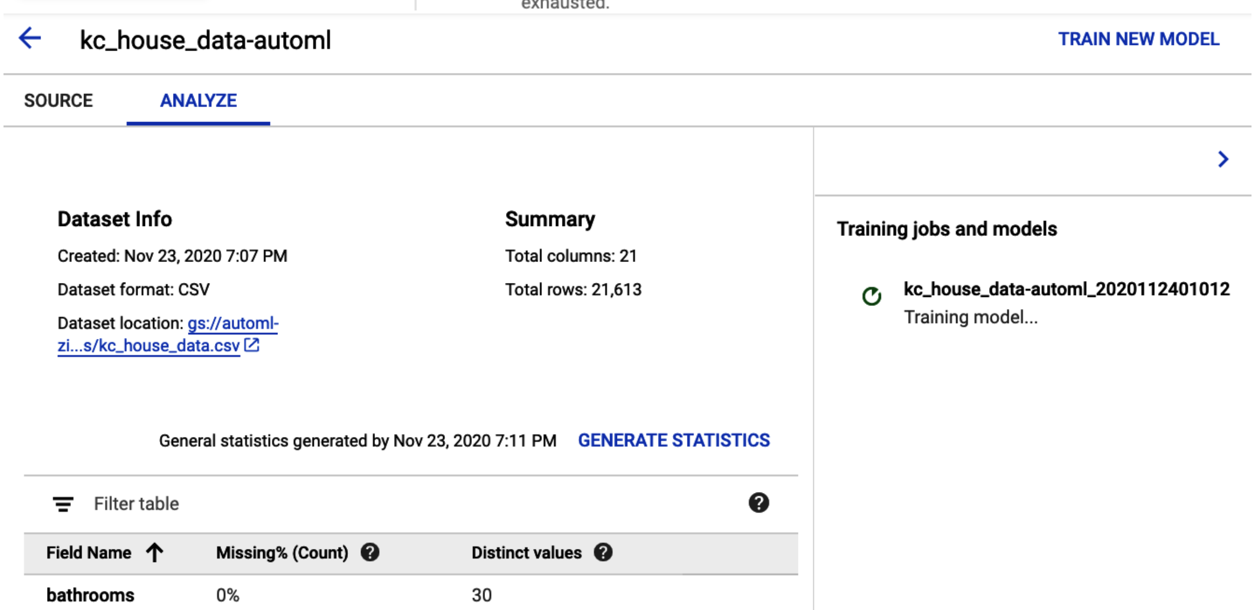
Filter table

Field Name	Transformation	Missing% (Count)	
bathrooms	Auto	-	-
bedrooms	Auto	-	-
condition	Auto	-	-
date	Auto	-	-
floors	Auto	-	-
grade	Auto	-	-
id	Auto	-	-
lat	Auto	-	-
long	Auto	-	-
price Target		-	-
sqft_above	Auto	-	-
sqft_basement	Auto	-	-
sqft_living	Auto	-	-
sqft_living15	Auto	-	-
sqft_lot	Auto	-	-
sqft_lot15	Auto	-	-
view	Auto	-	-
waterfront	Auto	-	-
yr_built	Auto	-	-
yr_renovated	Auto	-	-
zipcode	Auto	-	-

Rows per page: 50 1 – 21 of 21


Optimization objective
RMSE (Default)
Capture more extreme values accurately
MAE
View extreme values as outliers with less impact on the model
RMSLE
Penalize error on relative size rather than absolute value. Especially helpful when both predicted and actual values can be quite large.
SHOW LESS
CONTINUE


Train new model
Choose training method
Define your model
Choose training options
4 Compute and pricing
START TRAINING
CANCEL
Enter the maximum number of node hours you want to spend training your model.
You can train for as little as 1 node hours. You may also be eligible to train with free node hours. Pricing guide
Budget *
5
Maximum node hours
Estimated completion date: Nov 24, 2020 1 AM GMT-5
Enable early stopping
Ends model training when no more improvements can be made and refunds leftover training budget. If early stopping is disabled, training continues until the budget is exhausted.


kc_house_data-automl
TRAIN NEW MODEL
SOURCE
ANALYZE
Dataset Info
Created: Nov 23, 2020 7:07 PM
Dataset format: CSV
Dataset location: gs://automl-zi...s/kc_house_data.csv
Summary
Total columns: 21
Total rows: 21,613
Training jobs and models
kc_house_data-automl_2020112401012
Training model...
General statistics generated by Nov 23, 2020 7:11 PM
GENERATE STATISTICS
Filter table
Field Name
Missing% (Count)
Distinct values
bathrooms
0%
30

kc_house_data-automl_2020112401012 VIEW TRAINING DATASET

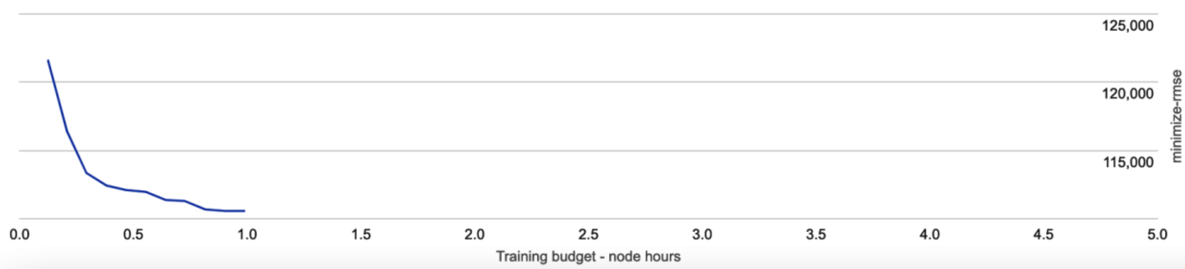
Training pipeline was completed on Nov 23, 2020, 8:48:52 PM.

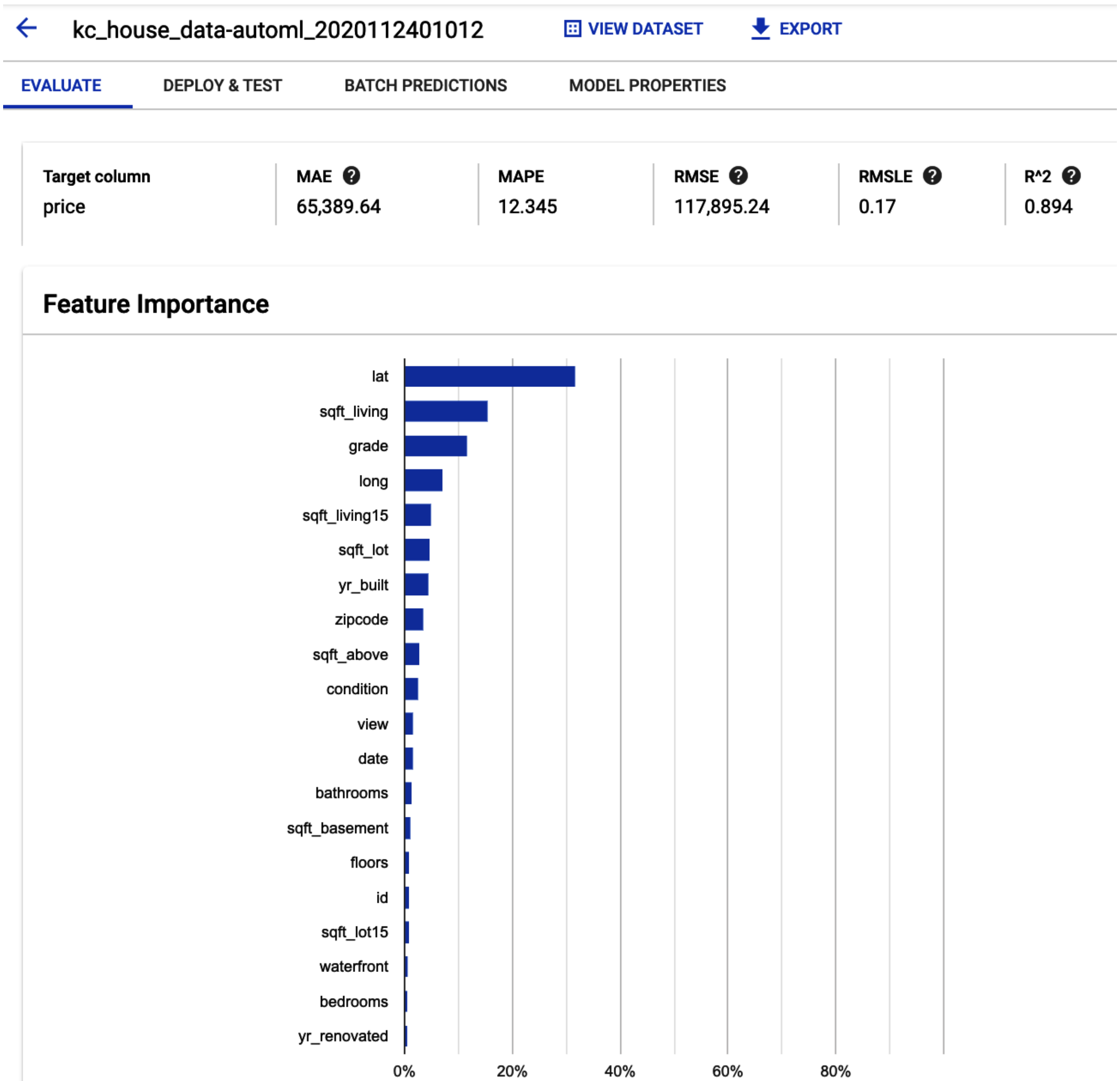
Status	Succeeded
Training pipeline ID	4457402546817859584
Created	Nov 23, 2020, 7:13:34 PM
Start time	Nov 23, 2020, 7:15:04 PM
Elapsed time	1 hr 35 min
Region	us-central1

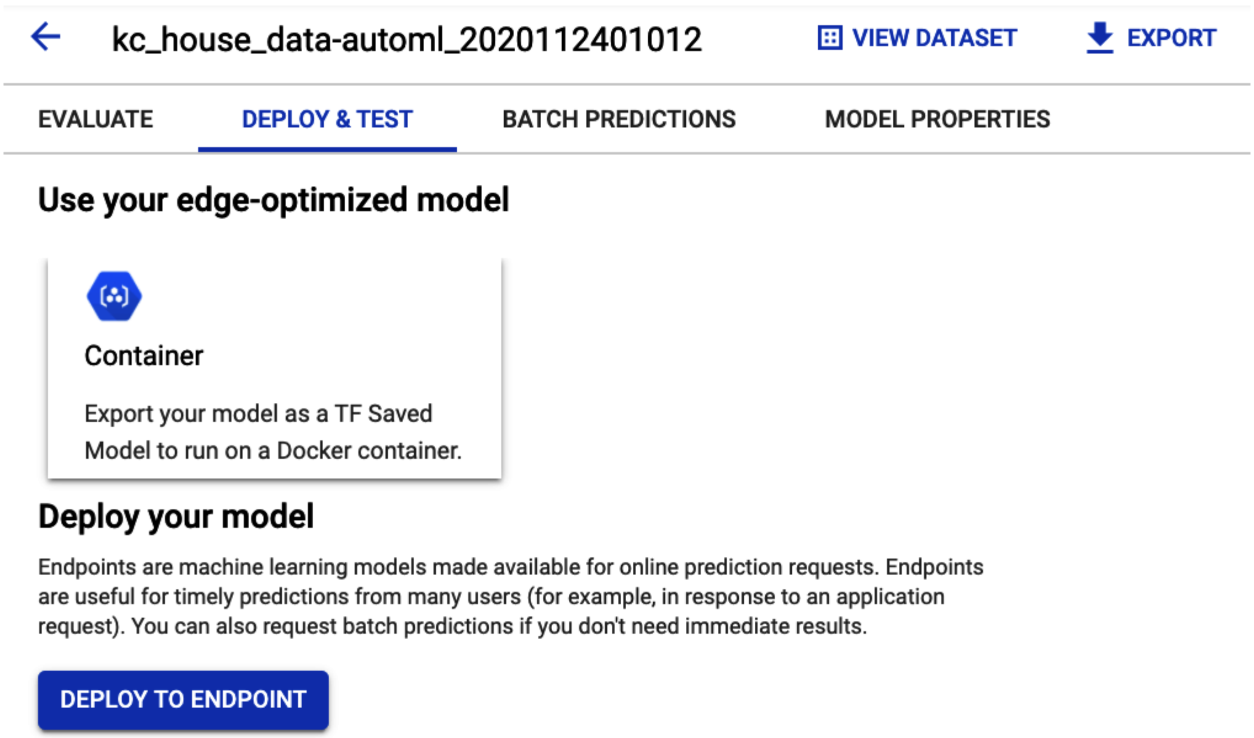
Dataset	kc_house_data-automl
Target column	price
Data split	Randomly assigned (80/10/10)
Transformation options	View details

Algorithm	AutoML
Objective	Tabular regression
Optimized for	RMSE
Training stage	Model post processing

Training performance




kc_house_data-automl_2020112401012
VIEW DATASET
EXPORT
EVALUATE
DEPLOY & TEST
BATCH PREDICTIONS
MODEL PROPERTIES
Target column
price
MAE
65,389.64
MAPE
12.345
RMSE
117,895.24
RMSLE
0.17
R^2
0.894
Feature Importance
lat
sqft_living
grade
long
sqft_living15
sqft_lot
yr_built
zipcode
sqft_above
condition
view
date
bathrooms
sqft_basement
floors
id
sqft_lot15
waterfront
bedrooms
yr_renovated
0%
20%
40%
60%
80%


kc_house_data-automl_2020112401012
VIEW DATASET
EXPORT
EVALUATE
DEPLOY & TEST
BATCH PREDICTIONS
MODEL PROPERTIES
Use your edge-optimized model
Container
Export your model as a TF Saved Model to run on a Docker container.
Deploy your model
Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.
DEPLOY TO ENDPOINT

Chapter 10: AutoML in the Enterprise