

# 03\_Split\_Final

April 14, 2023

## 1 ADS-508-01-SP23 Team 8: Final Project

## 2 Split and Preprocess ABT

Much of the code is modified from Fregly, C., & Barth, A. (2021). Data science on AWS: Implementing end-to-end, continuous AI and machine learning pipelines. O'Reilly.

### 2.1 Install missing dependencies

`PyAthena` is a Python DB API 2.0 (PEP 249) compliant client for Amazon Athena.

```
[2]: !pip install --disable-pip-version-check -q PyAthena==2.1.0
!pip install missingno
```

WARNING: The directory '/root/.cache/pip' or its parent directory is not owned or is not writable by the current user. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you should use sudo's -H flag.

WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead:

<https://pip.pypa.io/warnings/venv>

WARNING: The directory '/root/.cache/pip' or its parent directory is not owned or is not writable by the current user. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you should use sudo's -H flag.

Requirement already satisfied: missingno in /opt/conda/lib/python3.7/site-packages (0.5.2)

Requirement already satisfied: matplotlib in /opt/conda/lib/python3.7/site-packages (from missingno) (3.1.3)

Requirement already satisfied: scipy in /opt/conda/lib/python3.7/site-packages (from missingno) (1.4.1)

Requirement already satisfied: numpy in /opt/conda/lib/python3.7/site-packages

(from missingno) (1.21.6)  
Requirement already satisfied: seaborn in /opt/conda/lib/python3.7/site-packages (from missingno) (0.10.0)  
Requirement already satisfied: cycycler>=0.10 in /opt/conda/lib/python3.7/site-packages (from matplotlib->missingno) (0.10.0)  
Requirement already satisfied: kiwisolver>=1.0.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib->missingno) (1.1.0)  
Requirement already satisfied: python-dateutil>=2.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib->missingno) (2.8.2)  
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib->missingno) (2.4.6)  
Requirement already satisfied: pandas>=0.22.0 in /opt/conda/lib/python3.7/site-packages (from seaborn->missingno) (1.3.5)  
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from cycycler>=0.10->matplotlib->missingno) (1.14.0)  
Requirement already satisfied: setuptools in /opt/conda/lib/python3.7/site-packages (from kiwisolver>=1.0.1->matplotlib->missingno) (59.3.0)  
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.22.0->seaborn->missingno) (2019.3)  
**WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: <https://pip.pypa.io/warnings/venv>**

## 2.2 Globally import libraries

```
[3]: import boto3
from botocore.client import ClientError
import sagemaker
import pandas as pd
import numpy as np
from pyathena import connect
from IPython.core.display import display, HTML
import missingno as msno
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import make_pipeline, Pipeline
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.model_selection import train_test_split, cross_val_score, GridSearchCV
from sklearn.feature_selection import VarianceThreshold
import datetime as dt
from io import BytesIO

%matplotlib inline
```

## 2.3 Instantiate AWS SageMaker and S3 sessions

```
[4]: session = boto3.session.Session()
region = session.region_name
sagemaker_session = sagemaker.Session()
def_bucket = sagemaker_session.default_bucket()
bucket = 'sagemaker-us-east-ads508-sp23-t8'

s3 = boto3.Session().client(service_name="s3",
                             region_name=region)
```

```
[5]: print(f"Default bucket: {def_bucket}")
print(f"Public T8 bucket: {bucket}")
```

Default bucket: sagemaker-us-east-1-657724983756  
Public T8 bucket: sagemaker-us-east-ads508-sp23-t8

## 2.4 Pass in ABT from CSV

```
[6]: s3_abt_csv_path = f"s3://{def_bucket}/team_8_data/abt/abt_encoded_df01.csv"
abt_encoded_df01 = pd.read_csv(s3_abt_csv_path)
```

### 2.4.1 Perform train/test split

```
[7]: y01 = ['childpoverty']
abt_encoded_y01_vc01 = abt_encoded_df01[y01].to_numpy()
print(abt_encoded_y01_vc01.shape)
display(abt_encoded_y01_vc01[0:11])
abt_encoded_x01_df01 = abt_encoded_df01.drop(y01, axis=1)
print(abt_encoded_x01_df01.shape)
display(abt_encoded_x01_df01.head(11))

'''Citation for stratification by multiple cols:
https://stackoverflow.com/questions/45516424/
↳sklearn-train-test-split-on-pandas-stratify-by-multiple-columns'''
abt_encoded_x01_df01['boroughs'] = abt_encoded_x01_df01['borough_bronx'].
↳astype(int).astype(str) + abt_encoded_x01_df01['borough_brooklyn'].
↳astype(int).astype(str) + abt_encoded_x01_df01['borough_manhattan'].
↳astype(int).astype(str) + abt_encoded_x01_df01['borough_queens'].astype(int).
↳astype(str) + abt_encoded_x01_df01['borough_staten island'].astype(int).
↳astype(str)
display(abt_encoded_x01_df01.head(5))
train_x01, test_x01, train_y01, test_y01 =
↳train_test_split(abt_encoded_x01_df01,
                    abt_encoded_y01_vc01,
                    test_size=.2,
```

```

↳stratify=abt_encoded_x01_df01[['boroughs']],
                                                                    shuffle=True,
                                                                    random_state=1699)

train_x01 = train_x01.drop(['boroughs', 'poverty'], axis=1)
test_x01 = test_x01.drop(['boroughs', 'poverty'], axis=1)

print(f'{train_x01.shape}')
print(f'{train_y01.shape}')
print(f'\n{test_x01.shape}')
print(f'{test_y01.shape}')

```

(31605, 1)

```

array([[20.7],
       [23.6],
       [35.9],
       [31.5],
       [67.7],
       [68.3],
       [ 0. ],
       [62.4],
       [64.9],
       [63.1],
       [ 6.5]])

```

(31605, 49)

	borough_bronx	borough_brooklyn	borough_manhattan	borough_queens	\
0	1.0	0.0	0.0	0.0	
1	1.0	0.0	0.0	0.0	
2	1.0	0.0	0.0	0.0	
3	1.0	0.0	0.0	0.0	
4	1.0	0.0	0.0	0.0	
5	1.0	0.0	0.0	0.0	
6	1.0	0.0	0.0	0.0	
7	1.0	0.0	0.0	0.0	
8	1.0	0.0	0.0	0.0	
9	1.0	0.0	0.0	0.0	
10	1.0	0.0	0.0	0.0	

  

	borough_statens_island	relative_data_year_-4	relative_data_year_-3	\
0	0.0	1.0	0.0	
1	0.0	1.0	0.0	
2	0.0	1.0	0.0	
3	0.0	1.0	0.0	
4	0.0	1.0	0.0	
5	0.0	1.0	0.0	

6	0.0	1.0	0.0
7	0.0	1.0	0.0
8	0.0	1.0	0.0
9	0.0	1.0	0.0
10	0.0	1.0	0.0

	relative_data_year_-2	relative_data_year_-1	relative_data_year_0	...	\
0	0.0	0.0	0.0	...	
1	0.0	0.0	0.0	...	
2	0.0	0.0	0.0	...	
3	0.0	0.0	0.0	...	
4	0.0	0.0	0.0	...	
5	0.0	0.0	0.0	...	
6	0.0	0.0	0.0	...	
7	0.0	0.0	0.0	...	
8	0.0	0.0	0.0	...	
9	0.0	0.0	0.0	...	
10	0.0	0.0	0.0	...	

	walk	othertransp	workathome	meancommute	employed	privatework	\
0	2.9	0.0	0.0	43.0	2308.0	80.8	
1	1.4	0.5	2.1	45.0	2675.0	71.7	
2	8.6	1.6	1.7	38.8	2120.0	75.0	
3	3.0	2.4	6.2	45.4	1083.0	76.8	
4	4.3	1.0	0.0	46.0	2508.0	71.0	
5	14.0	1.5	4.1	42.7	1191.0	74.2	
6	0.0	0.0	0.0	NaN	113.0	62.8	
7	17.7	1.8	2.7	35.5	1691.0	85.1	
8	18.0	0.0	1.6	42.8	1102.0	86.9	
9	7.1	0.7	0.5	44.0	1559.0	75.0	
10	2.0	0.6	2.7	47.3	2394.0	61.9	

	publicwork	selfemployed	familywork	unemployment
0	16.2	2.9	0.0	7.7
1	25.3	2.5	0.6	9.5
2	21.3	3.8	0.0	8.7
3	15.5	7.7	0.0	19.2
4	21.3	7.7	0.0	17.2
5	16.1	9.7	0.0	18.9
6	37.2	0.0	0.0	0.0
7	8.3	6.1	0.5	9.4
8	8.5	4.5	0.0	15.2
9	14.0	11.0	0.0	10.6
10	37.4	0.6	0.0	12.8

[11 rows x 49 columns]

borough_bronx	borough_brooklyn	borough_manhattan	borough_queens	\
---------------	------------------	-------------------	----------------	---

0	1.0	0.0	0.0	0.0
1	1.0	0.0	0.0	0.0
2	1.0	0.0	0.0	0.0
3	1.0	0.0	0.0	0.0
4	1.0	0.0	0.0	0.0

  

	borough_statens_island	relative_data_year_-4	relative_data_year_-3	\
0	0.0	1.0	0.0	
1	0.0	1.0	0.0	
2	0.0	1.0	0.0	
3	0.0	1.0	0.0	
4	0.0	1.0	0.0	

  

	relative_data_year_-2	relative_data_year_-1	relative_data_year_0	...	\
0	0.0	0.0	0.0	...	
1	0.0	0.0	0.0	...	
2	0.0	0.0	0.0	...	
3	0.0	0.0	0.0	...	
4	0.0	0.0	0.0	...	

  

	othertransp	workathome	meancommute	employed	privatework	publicwork	\
0	0.0	0.0	43.0	2308.0	80.8	16.2	
1	0.5	2.1	45.0	2675.0	71.7	25.3	
2	1.6	1.7	38.8	2120.0	75.0	21.3	
3	2.4	6.2	45.4	1083.0	76.8	15.5	
4	1.0	0.0	46.0	2508.0	71.0	21.3	

  

	selfemployed	familywork	unemployment	boroughs
0	2.9	0.0	7.7	10000
1	2.5	0.6	9.5	10000
2	3.8	0.0	8.7	10000
3	7.7	0.0	19.2	10000
4	7.7	0.0	17.2	10000

[5 rows x 50 columns]

(25284, 48)

(25284, 1)

(6321, 48)

(6321, 1)

### Examine features with near zero variances

```
[8]: # Review near-zero variance (NZV) features for possible removal
train_x01_nzv_fit = VarianceThreshold(.025).fit(train_x01)
train_x01_nzv_vc01 = train_x01_nzv_fit.transform(train_x01)

# Get the names of the selected features
```

```

train_x01_nzv_fit_select_features = train_x01.columns[train_x01_nzv_fit.
    ↪get_support()]

train_x01_nzv_df01 = pd.DataFrame(train_x01_nzv_vc01,
                                   columns=train_x01_nzv_fit_select_features)

display(train_x01_nzv_df01.head(5))
print(f'NZV transformed matrix dimensions = {train_x01_nzv_df01.shape}')

print(f'\n{train_x01.shape[1] - train_x01_nzv_df01.shape[1]} near zero variance_
    ↪features were eliminated')

print(train_x01.columns)
print(train_x01_nzv_df01.columns)

```

	borough_bronx	borough_brooklyn	borough_manhattan	borough_queens	\
0	0.0	0.0	0.0	1.0	
1	0.0	0.0	0.0	1.0	
2	0.0	0.0	0.0	1.0	
3	0.0	1.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	

  

	borough_staten	island	relative_data_year_-4	relative_data_year_-3	\
0		0.0	1.0	0.0	
1		0.0	1.0	0.0	
2		0.0	0.0	1.0	
3		0.0	1.0	0.0	
4		1.0	0.0	0.0	

  

	relative_data_year_-2	relative_data_year_-1	relative_data_year_0	...	\
0	0.0	0.0	0.0	...	
1	0.0	0.0	0.0	...	
2	0.0	0.0	0.0	...	
3	0.0	0.0	0.0	...	
4	1.0	0.0	0.0	...	

  

	walk	othertransp	workathome	meancommute	employed	privatework	\
0	4.6	0.6	3.9	42.9	3293.0	81.3	
1	7.2	2.4	3.4	30.4	3872.0	86.0	
2	3.7	0.4	3.1	43.0	2014.0	78.6	
3	4.1	7.0	6.8	40.4	1782.0	82.7	
4	0.6	1.2	0.8	42.0	801.0	76.3	

  

	publicwork	selfemployed	familywork	unemployment
0	13.2	5.6	0.0	3.4
1	5.9	8.1	0.0	6.5
2	13.5	7.9	0.0	6.7

3	9.2	8.1	0.0	7.8
4	20.1	3.6	0.0	8.9

[5 rows x 48 columns]

NZV transformed matrix dimensions = (25284, 48)

0 near zero variance features were eliminated

```
Index(['borough_bronx', 'borough_brooklyn', 'borough_manhattan',
      'borough_queens', 'borough_staten island', 'relative_data_year_-4',
      'relative_data_year_-3', 'relative_data_year_-2',
      'relative_data_year_-1', 'relative_data_year_0',
      'complaint_type_FELONY', 'complaint_type_MISDEMEANOR',
      'complaint_type_VIOLATION', 'annual_evictions_x_borough',
      'annual_complaint_counts', 'annual_grad_n', 'annual_dropped_out_n',
      'totalpop', 'men', 'women', 'hispanic', 'white', 'black', 'native',
      'asian', 'citizen', 'income', 'incomeerr', 'incomepercap',
      'incomepercaperr', 'professional', 'service', 'office', 'construction',
      'production', 'drive', 'carpool', 'transit', 'walk', 'othertransp',
      'workathome', 'meancommute', 'employed', 'privatework', 'publicwork',
      'selfemployed', 'familywork', 'unemployment'],
      dtype='object')
Index(['borough_bronx', 'borough_brooklyn', 'borough_manhattan',
      'borough_queens', 'borough_staten island', 'relative_data_year_-4',
      'relative_data_year_-3', 'relative_data_year_-2',
      'relative_data_year_-1', 'relative_data_year_0',
      'complaint_type_FELONY', 'complaint_type_MISDEMEANOR',
      'complaint_type_VIOLATION', 'annual_evictions_x_borough',
      'annual_complaint_counts', 'annual_grad_n', 'annual_dropped_out_n',
      'totalpop', 'men', 'women', 'hispanic', 'white', 'black', 'native',
      'asian', 'citizen', 'income', 'incomeerr', 'incomepercap',
      'incomepercaperr', 'professional', 'service', 'office', 'construction',
      'production', 'drive', 'carpool', 'transit', 'walk', 'othertransp',
      'workathome', 'meancommute', 'employed', 'privatework', 'publicwork',
      'selfemployed', 'familywork', 'unemployment'],
      dtype='object')
```

## 2.4.2 Save ABT to S3

```
[9]: print(f'{train_x01.head(5)}')
      print(f'{train_y01[0:5]}')
      print(f'\n{test_x01.head(5)}')
      print(f'{test_y01[0:5]}')

s3_train_x01_csv_path = f"s3://{def_bucket}/team_8_data/modeling_data/training/
➔train_x01.csv"
train_x01.to_csv(s3_train_x01_csv_path,
                  index=False,
```



```

        header=True)

s3_test_x01_csv_path = f"s3://{def_bucket}/team_8_data/modeling_data/testing/
↳test_x01.csv"
test_x01.to_csv(s3_test_x01_csv_path,
                index=False,
                header=True)

```

	borough_bronx	borough_brooklyn	borough_manhattan	borough_queens	\
20821	0.0	0.0	0.0	1.0	
20374	0.0	0.0	0.0	1.0	
23796	0.0	0.0	0.0	1.0	
6080	0.0	1.0	0.0	0.0	
30916	0.0	0.0	0.0	0.0	

	borough_staten	island	relative_data_year_-4	relative_data_year_-3	\
20821		0.0	1.0	0.0	
20374		0.0	1.0	0.0	
23796		0.0	0.0	1.0	
6080		0.0	1.0	0.0	
30916		1.0	0.0	0.0	

	relative_data_year_-2	relative_data_year_-1	relative_data_year_0	\
20821	0.0	0.0	0.0	
20374	0.0	0.0	0.0	
23796	0.0	0.0	0.0	
6080	0.0	0.0	0.0	
30916	1.0	0.0	0.0	

	...	walk	othertransp	workathome	meancommute	employed	privatework	\
20821	...	4.6	0.6	3.9	42.9	3293.0	81.3	
20374	...	7.2	2.4	3.4	30.4	3872.0	86.0	
23796	...	3.7	0.4	3.1	43.0	2014.0	78.6	
6080	...	4.1	7.0	6.8	40.4	1782.0	82.7	
30916	...	0.6	1.2	0.8	42.0	801.0	76.3	

	publicwork	selfemployed	familywork	unemployment
20821	13.2	5.6	0.0	3.4
20374	5.9	8.1	0.0	6.5
23796	13.5	7.9	0.0	6.7
6080	9.2	8.1	0.0	7.8
30916	20.1	3.6	0.0	8.9

[5 rows x 48 columns]

[[ 5.1]

[ 1.8]

[33.4]

[25. ]

```
[18.4]]
```

```
      borough_bronx  borough_brooklyn  borough_manhattan  borough_queens  \
3566              1.0              0.0              0.0              0.0
19683              0.0              0.0              1.0              0.0
8603               0.0              1.0              0.0              0.0
19957              0.0              0.0              1.0              0.0
9723               0.0              1.0              0.0              0.0

      borough_staten_island  relative_data_year_-4  relative_data_year_-3  \
3566                    0.0                    0.0                    0.0
19683                    0.0                    0.0                    0.0
8603                     0.0                    0.0                    1.0
19957                    0.0                    0.0                    0.0
9723                     0.0                    0.0                    0.0

      relative_data_year_-2  relative_data_year_-1  relative_data_year_0  \
3566                    0.0                    1.0                    0.0
19683                    0.0                    0.0                    1.0
8603                     0.0                    0.0                    0.0
19957                    0.0                    0.0                    1.0
9723                     1.0                    0.0                    0.0

      ...  walk  othertransp  workathome  meancommute  employed  privatework  \
3566  ...  18.6          2.1          1.3          36.1    1567.0          75.0
19683  ...  29.7          9.4          5.7          24.5    2029.0          91.2
8603   ...   4.5          1.1          1.8          46.8    2261.0          77.8
19957  ...  15.5          3.2          5.7          32.2    3135.0          85.9
9723   ...   4.7          1.6          3.6          46.5    1830.0          73.6

      publicwork  selfemployed  familywork  unemployment
3566          18.4           6.5          0.0           15.4
19683           2.3           6.5          0.0           3.8
8603          15.7           6.5          0.0          14.0
19957           8.3           5.8          0.0           2.6
9723          23.7           2.7          0.0          12.8
```

```
[5 rows x 48 columns]
```

```
[[38.7]
```

```
[ 0.9]
```

```
[36.6]
```

```
[ 0. ]
```

```
[14.8]]
```

```
[10]: # Define the S3 object key
train_y01_s3_key = 'team_8_data/modeling_data/training/train_y01.npy'
```

```

# Save the numpy array to S3
with BytesIO() as data:
    np.save(data, train_y01)
    data.seek(0)
    s3.upload_fileobj(data, def_bucket, train_y01_s3_key)

# Confirm that the numpy array was saved to S3
train_y01_response = s3.list_objects(Bucket=def_bucket,
                                     Prefix=train_y01_s3_key)
print(train_y01_response)

# Define the S3 object key
test_y01_s3_key = 'team_8_data/modeling_data/testing/test_y01.npy'

# Save the numpy array to S3
with BytesIO() as data:
    np.save(data, test_y01)
    data.seek(0)
    s3.upload_fileobj(data, def_bucket, test_y01_s3_key)

# Confirm that the numpy array was saved to S3
test_y01_response = s3.list_objects(Bucket=def_bucket,
                                     Prefix=test_y01_s3_key)
print(test_y01_response)

```

```

{'ResponseMetadata': {'RequestId': 'BX6Q8H5DSQWE7YWD', 'HostId':
'L72U9CI7UAiDGjIlqH6y+OINDcJ2jkKtOZHAiHBj7zSNrmBjMqMgOaB4u2Ew/72ZfvQycDX7Y4I=',
'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amz-id-2':
'L72U9CI7UAiDGjIlqH6y+OINDcJ2jkKtOZHAiHBj7zSNrmBjMqMgOaB4u2Ew/72ZfvQycDX7Y4I=',
'x-amz-request-id': 'BX6Q8H5DSQWE7YWD', 'date': 'Thu, 13 Apr 2023 17:09:46 GMT',
'x-amz-bucket-region': 'us-east-1', 'content-type': 'application/xml',
'transfer-encoding': 'chunked', 'server': 'AmazonS3'}, 'RetryAttempts': 0},
'IsTruncated': False, 'Marker': '', 'Contents': [{'Key':
'team_8_data/modeling_data/training/train_y01.npy', 'LastModified':
datetime.datetime(2023, 4, 13, 17, 9, 46, tzinfo=tzlocal()), 'ETag':
'"fec93318dc01a0851bfb63e52dcdfd6a"', 'Size': 202400, 'StorageClass':
'STANDARD', 'Owner': {'DisplayName': 'awslabsc0w5192702t1672660495', 'ID':
'6205bdaa014eec2453ddb24fb65480c671e128d98965891a2b0bf2ba5e5cced6'}}], 'Name':
'sagemaker-us-east-1-657724983756', 'Prefix':
'team_8_data/modeling_data/training/train_y01.npy', 'MaxKeys': 1000,
'EncodingType': 'url'}
{'ResponseMetadata': {'RequestId': 'BX6GTM5FTDVZ9TEC', 'HostId':
'NOBWLcj9aYvRJmskDXBbYHlD7AB/18YVwYB7ZKshzLusq+TcaZhjplr4qA8uW8qICNj2V3AB7b0=',
'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amz-id-2':
'NOBWLcj9aYvRJmskDXBbYHlD7AB/18YVwYB7ZKshzLusq+TcaZhjplr4qA8uW8qICNj2V3AB7b0=',
'x-amz-request-id': 'BX6GTM5FTDVZ9TEC', 'date': 'Thu, 13 Apr 2023 17:09:46 GMT',
'x-amz-bucket-region': 'us-east-1', 'content-type': 'application/xml',
'transfer-encoding': 'chunked', 'server': 'AmazonS3'}, 'RetryAttempts': 0},

```

```
'IsTruncated': False, 'Marker': '', 'Contents': [{'Key':
'team_8_data/modeling_data/testing/test_y01.npy', 'LastModified':
datetime.datetime(2023, 4, 13, 17, 9, 46, tzinfo=tzlocal()), 'ETag':
'"47711adfc2e66766a7a8efe248848df6"', 'Size': 50696, 'StorageClass': 'STANDARD',
'Owner': {'DisplayName': 'awslabsc0w5192702t1672660495', 'ID':
'6205bdaa014eec2453ddb24fb65480c671e128d98965891a2b0bf2ba5e5cced6'}}], 'Name':
'sagemaker-us-east-1-657724983756', 'Prefix':
'team_8_data/modeling_data/testing/test_y01.npy', 'MaxKeys': 1000,
'EncodingType': 'url'}
```

## 2.5 Release Resources

```
[11]: %%html

<p><b>Shutting down your kernel for this notebook to release resources.</b></p>
<button class="sm-command-button" data-commandlinker-command="kernelmenu:
↪shutdown" style="display:none;">Shutdown Kernel</button>

<script>
try {
    els = document.getElementsByClassName("sm-command-button");
    els[0].click();
}
catch(err) {
    // NoOp
}
</script>
```

<IPython.core.display.HTML object>

```
[12]: %%javascript

try {
    Jupyter.notebook.save_checkpoint();
    Jupyter.notebook.session.delete();
}
catch(err) {
    // NoOp
}
```

<IPython.core.display.Javascript object>