automated-feature-engineering_tutorial

September 22, 2018

1 Tutorial - Using Featuretools to Predict Missed Appointments

source: https://github.com/Featuretools/predict-appointment-noshow/blob/master/Tutorial.ipynb

```
In [69]: import utils
         import numpy as np
         import pandas as pd
         import featuretools as ft
         print('Featuretools version {}'.format(ft.__version__))

         from IPython.core.interactiveshell import InteractiveShell
         InteractiveShell.ast_node_interactivity = "all"

         %matplotlib inline
```

```
Featuretools version 0.3.0
```

```
In [70]: data = utils.load_data("./data/KaggleV2-May-2016.csv")
         data.head(3)
```

```
110527 Appointments, 14 Columns
Appointments: 110527
Schedule times: 103549
Patients: 62299
Neighborhoods: 81
```

```
Out[70]:                      patient_id  appointment_id gender      scheduled_time  \
         AppointmentID
         5642903        29872499824296.00         5642903      F 2016-04-29 18:38:08
         5642503       558997776694438.00         5642503      M 2016-04-29 16:08:27
         5642549         4262962299951.00         5642549      F 2016-04-29 16:19:04

                           appointment_day  age     neighborhood  scholarship  \
         AppointmentID
         5642903       2016-04-29 23:59:59   62  JARDIM DA PENHA            0
```

```
5642503       2016-04-29 23:59:59   56  JARDIM DA PENHA            0
5642549       2016-04-29 23:59:59   62    MATA DA PRAIA            0

               hypertension  diabetes  alcoholism  handicap  sms_received  \
AppointmentID
5642903                   1         0           0         0             0
5642503                   0         0           0         0             0
5642549                   0         0           0         0             0

               no_show
AppointmentID
5642903          False
5642503          False
5642549          False
```

2 Structure the Data

```
In [71]: # List the semantic type for each column
         import featuretools.variable_types as vtypes


         variable_types = {'gender': vtypes.Categorical,
                           'patient_id': vtypes.Categorical,
                           'age': vtypes.Ordinal,
                           'scholarship': vtypes.Boolean,
                           'hypertension': vtypes.Boolean,
                           'diabetes': vtypes.Boolean,
                           'alcoholism': vtypes.Boolean,
                           'handicap': vtypes.Boolean,
                           'no_show': vtypes.Boolean,
                           'sms_received': vtypes.Boolean}

In [72]: # Make an entity named 'appointments' which stores dataset metadata with the datafram
         es = ft.EntitySet('Appointments')
         es = es.entity_from_dataframe(entity_id="appointments",
                                       dataframe=data,
                                       index='appointment_id',
                                       time_index='scheduled_time',
                                       secondary_time_index={'appointment_day': ['no_show', 'sm
                                       variable_types=variable_types)

         es['appointments']

Out[72]: Entity: appointments
           Variables:
             appointment_id (dtype: index)
             scheduled_time (dtype: datetime_time_index)
             appointment_day (dtype: datetime)
```

```
    neighborhood (dtype: categorical)
    gender (dtype: categorical)
    patient_id (dtype: categorical)
    age (dtype: ordinal)
    scholarship (dtype: boolean)
    hypertension (dtype: boolean)
    diabetes (dtype: boolean)
    alcoholism (dtype: boolean)
    handicap (dtype: boolean)
    no_show (dtype: boolean)
    sms_received (dtype: boolean)
  Shape:
    (Rows: 110527, Columns: 14)
```

```
In [73]: es
```

```
Out[73]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 14]
           Relationships:
             No relationships
```

The time index and secondary time index notate what time the data is recorded. By doing that, we can avoid using data from the future while creating features. Since the label is in the dataframe, we either need to specify a time index or drop the column entirely.

Finally, we build new entities from our existing one using normalize_entity. We take unique values from patient, age, neighborhood and gender and make a new Entity for each whose rows are the unique values. To do that we only need to specify where we start (appointments), the name of the new entity (e.g. patients) and what the index should be (e.g. patient_id). Having those additional Entities and Relationships tells the algorithm about reasonable groupings which allows for some neat aggregations.

2.1 appointments & patients

```
In [74]: # Make a patients entity with patient-specific variables
         es.normalize_entity('appointments', 'patients', 'patient_id',
                             additional_variables=['scholarship',
                                                   'hypertension',
                                                   'diabetes',
                                                   'alcoholism',
                                                   'handicap'])

         es
```

```
Out[74]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 9]
             patients [Rows: 62299, Columns: 7]
           Relationships:
             appointments.patient_id -> patients.patient_id
```

```
Out[74]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 9]
             patients [Rows: 62299, Columns: 7]
           Relationships:
             appointments.patient_id -> patients.patient_id
```

```
In [75]: es['appointments'].df.head()
         es['patients'].df.head()
```

```
Out[75]:          appointment_id      scheduled_time     appointment_day  \
         5030230         5030230 2015-11-10 07:13:56 2016-05-04 23:59:59
         5122866         5122866 2015-12-03 08:17:28 2016-05-02 23:59:59
         5134197         5134197 2015-12-07 10:40:59 2016-06-03 23:59:59
         5134220         5134220 2015-12-07 10:42:42 2016-06-03 23:59:59
         5134223         5134223 2015-12-07 10:43:01 2016-06-03 23:59:59

                    neighborhood gender         patient_id  age  no_show  sms_received
         5030230     RESISTÊNCIA      F 832256398961987.00   51    False             1
         5122866      VILA RUBIM      M  91637474953513.00   34     True             1
         5134197  SÃO CRISTÓVÃO       F   1216586867796.00   27     True             1
         5134220         MARUÍPE      F  31899595421534.00   48    False             1
         5134223  SÃO CRISTÓVÃO       F   9582232334148.00   80    False             1
```

```
Out[75]:                            patient_id  scholarship  hypertension  diabetes  \
         832256398961987.00 832256398961987.00            0             0         0
         91637474953513.00   91637474953513.00            0             1         0
         1216586867796.00     1216586867796.00            1             0         0
         31899595421534.00   31899595421534.00            0             1         1
         9582232334148.00     9582232334148.00            0             1         1

                             alcoholism  handicap first_appointments_time
         832256398961987.00           0         0     2015-11-10 07:13:56
         91637474953513.00            0         0     2015-12-03 08:17:28
         1216586867796.00             0         0     2015-12-07 10:40:59
         31899595421534.00            0         0     2015-12-07 10:42:42
         9582232334148.00             0         0     2015-12-07 10:43:01
```

2.2 appointments & patients & locations

```
In [76]: # Make locations, ages and genders
         es.normalize_entity('appointments', 'locations', 'neighborhood',
                             make_time_index=False)
         es
```

```
Out[76]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 9]
             patients [Rows: 62299, Columns: 7]
```

```
    locations [Rows: 81, Columns: 1]
  Relationships:
    appointments.patient_id -> patients.patient_id
    appointments.neighborhood -> locations.neighborhood
```

```
Out[76]: Entityset: Appointments
  Entities:
    appointments [Rows: 110527, Columns: 9]
    patients [Rows: 62299, Columns: 7]
    locations [Rows: 81, Columns: 1]
  Relationships:
    appointments.patient_id -> patients.patient_id
    appointments.neighborhood -> locations.neighborhood
```

```
In [77]: es['appointments'].df.head()
         es['patients'].df.head()
         es['locations'].df.head()
```

```
Out[77]:          appointment_id      scheduled_time     appointment_day  \
         5030230         5030230 2015-11-10 07:13:56 2016-05-04 23:59:59
         5122866         5122866 2015-12-03 08:17:28 2016-05-02 23:59:59
         5134197         5134197 2015-12-07 10:40:59 2016-06-03 23:59:59
         5134220         5134220 2015-12-07 10:42:42 2016-06-03 23:59:59
         5134223         5134223 2015-12-07 10:43:01 2016-06-03 23:59:59

                    neighborhood gender         patient_id  age  no_show  sms_received
         5030230     RESISTÊNCIA      F 832256398961987.00   51    False             1
         5122866      VILA RUBIM      M  91637474953513.00   34     True             1
         5134197  SÃO CRISTÓVÃO       F   1216586867796.00   27     True             1
         5134220         MARUÍPE      F  31899595421534.00   48    False             1
         5134223  SÃO CRISTÓVÃO       F   9582232334148.00   80    False             1
```

```
Out[77]:                            patient_id  scholarship  hypertension  diabetes  \
         832256398961987.00 832256398961987.00            0             0         0
         91637474953513.00   91637474953513.00            0             1         0
         1216586867796.00     1216586867796.00            1             0         0
         31899595421534.00   31899595421534.00            0             1         1
         9582232334148.00     9582232334148.00            0             1         1

                             alcoholism  handicap first_appointments_time
         832256398961987.00           0         0     2015-11-10 07:13:56
         91637474953513.00            0         0     2015-12-03 08:17:28
         1216586867796.00             0         0     2015-12-07 10:40:59
         31899595421534.00            0         0     2015-12-07 10:42:42
         9582232334148.00             0         0     2015-12-07 10:43:01
```

```
Out[77]:                             neighborhood
         AEROPORTO                     AEROPORTO
         ANDORINHAS                   ANDORINHAS
```

```
ANTÔNIO HONÓRIO          ANTÔNIO HONÓRIO
ARIOVALDO FAVALESSA  ARIOVALDO FAVALESSA
BARRO VERMELHO            BARRO VERMELHO
```

2.3 appointments & patients & locations & ages

```
In [78]: es.normalize_entity('appointments', 'ages', 'age',
                             make_time_index=False)
         es
```

```
Out[78]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 9]
             patients [Rows: 62299, Columns: 7]
             locations [Rows: 81, Columns: 1]
             ages [Rows: 104, Columns: 1]
           Relationships:
             appointments.patient_id -> patients.patient_id
             appointments.neighborhood -> locations.neighborhood
             appointments.age -> ages.age
```

```
Out[78]: Entityset: Appointments
           Entities:
             appointments [Rows: 110527, Columns: 9]
             patients [Rows: 62299, Columns: 7]
             locations [Rows: 81, Columns: 1]
             ages [Rows: 104, Columns: 1]
           Relationships:
             appointments.patient_id -> patients.patient_id
             appointments.neighborhood -> locations.neighborhood
             appointments.age -> ages.age
```

```
In [79]: es['appointments'].df.head()
         es['patients'].df.head()
         es['locations'].df.head()
         es['ages'].df.head()
```

```
Out[79]:          appointment_id      scheduled_time     appointment_day  \
         5030230         5030230 2015-11-10 07:13:56 2016-05-04 23:59:59
         5122866         5122866 2015-12-03 08:17:28 2016-05-02 23:59:59
         5134197         5134197 2015-12-07 10:40:59 2016-06-03 23:59:59
         5134220         5134220 2015-12-07 10:42:42 2016-06-03 23:59:59
         5134223         5134223 2015-12-07 10:43:01 2016-06-03 23:59:59

                    neighborhood gender         patient_id  age  no_show  sms_received
         5030230    RESISTÊNCIA      F 832256398961987.00   51    False             1
         5122866      VILA RUBIM     M  91637474953513.00   34     True             1
         5134197  SÃO CRISTÓVÃO      F   1216586867796.00   27     True             1
         5134220         MARUÍPE     F  31899595421534.00   48    False             1
         5134223  SÃO CRISTÓVÃO      F   9582232334148.00   80    False             1
```

Out[79]:

	patient_id	scholarship	hypertension	diabetes	alcoholism	handicap	first_appointments_time
832256398961987.00	832256398961987.00	0	0	0	0	0	2015-11-10 07:13:56
91637474953513.00	91637474953513.00	0	1	0	0	0	2015-12-03 08:17:28
1216586867796.00	1216586867796.00	1	0	0	0	0	2015-12-07 10:40:59
31899595421534.00	31899595421534.00	0	1	1	0	0	2015-12-07 10:42:42
9582232334148.00	9582232334148.00	0	1	1	0	0	2015-12-07 10:43:01

Out[79]:

	neighborhood
AEROPORTO	AEROPORTO
ANDORINHAS	ANDORINHAS
ANTÔNIO HONÓRIO	ANTÔNIO HONÓRIO
ARIOVALDO FAVALESSA	ARIOVALDO FAVALESSA
BARRO VERMELHO	BARRO VERMELHO

Out[79]:

	age
-1	-1
0	0
1	1
2	2
3	3

2.4 appointments & patients & locations & ages & genders

In [80]:
```
es.normalize_entity('appointments', 'genders', 'gender',
                    make_time_index=False)
es
```

Out[80]:
```
Entityset: Appointments
  Entities:
    appointments [Rows: 110527, Columns: 9]
    patients [Rows: 62299, Columns: 7]
    locations [Rows: 81, Columns: 1]
    ages [Rows: 104, Columns: 1]
    genders [Rows: 2, Columns: 1]
  Relationships:
    appointments.patient_id -> patients.patient_id
    appointments.neighborhood -> locations.neighborhood
    appointments.age -> ages.age
    appointments.gender -> genders.gender
```

Out[80]:
```
Entityset: Appointments
  Entities:
```

```
    appointments [Rows: 110527, Columns: 9]
    patients [Rows: 62299, Columns: 7]
    locations [Rows: 81, Columns: 1]
    ages [Rows: 104, Columns: 1]
    genders [Rows: 2, Columns: 1]
  Relationships:
    appointments.patient_id -> patients.patient_id
    appointments.neighborhood -> locations.neighborhood
    appointments.age -> ages.age
    appointments.gender -> genders.gender
```

In [81]:
```
es['appointments'].df.head()
es['patients'].df.head()
es['locations'].df.head()
es['ages'].df.head()
es['genders'].df.head()
```

Out[81]:

	appointment_id	scheduled_time	appointment_day	\
5030230	5030230	2015-11-10 07:13:56	2016-05-04 23:59:59	
5122866	5122866	2015-12-03 08:17:28	2016-05-02 23:59:59	
5134197	5134197	2015-12-07 10:40:59	2016-06-03 23:59:59	
5134220	5134220	2015-12-07 10:42:42	2016-06-03 23:59:59	
5134223	5134223	2015-12-07 10:43:01	2016-06-03 23:59:59	

	neighborhood	gender	patient_id	age	no_show	sms_received
5030230	RESISTÊNCIA	F	832256398961987.00	51	False	1
5122866	VILA RUBIM	M	91637474953513.00	34	True	1
5134197	SÃO CRISTÓVÃO	F	1216586867796.00	27	True	1
5134220	MARUÍPE	F	31899595421534.00	48	False	1
5134223	SÃO CRISTÓVÃO	F	9582232334148.00	80	False	1

Out[81]:

	patient_id	scholarship	hypertension	diabetes	\
832256398961987.00	832256398961987.00	0	0	0	
91637474953513.00	91637474953513.00	0	1	0	
1216586867796.00	1216586867796.00	1	0	0	
31899595421534.00	31899595421534.00	0	1	1	
9582232334148.00	9582232334148.00	0	1	1	

	alcoholism	handicap	first_appointments_time
832256398961987.00	0	0	2015-11-10 07:13:56
91637474953513.00	0	0	2015-12-03 08:17:28
1216586867796.00	0	0	2015-12-07 10:40:59
31899595421534.00	0	0	2015-12-07 10:42:42
9582232334148.00	0	0	2015-12-07 10:43:01

Out[81]:

	neighborhood
AEROPORTO	AEROPORTO
ANDORINHAS	ANDORINHAS
ANTÔNIO HONÓRIO	ANTÔNIO HONÓRIO

```
ARIOVALDO FAVALESSA  ARIOVALDO FAVALESSA
BARRO VERMELHO            BARRO VERMELHO
```

Out[81]:
```
     age
-1   -1
 0    0
 1    1
 2    2
 3    3
```

Out[81]:
```
  gender
F      F
M      M
```

2.5 Generating Features with Deep Feature Synthesis

With our data structued in an EntitySet, we can immediately build features across our entity and relationships with Deep Feature Synthesis (DFS). As an example, the feature locations.PERCENT_TRUE(no_show) will calculate percentage of patients of at this location that haven't shown up in the past.

This is where the time indices get used. We set the cutoff_time for each row to be when the patient schedules the appointment. That means that DFS, while building features, will only use the data that is known as the appointment is made. In particular, it won't use the label to create features.

In [82]:
```
# Take the index and the appointment time to use as a cutoff time
cutoff_times = es['appointments'].df[['appointment_id', 'scheduled_time', 'no_show']]
```

In [83]:
```
es['appointments'].df.shape
es['appointments'].df.head()
cutoff_times.shape
cutoff_times.head()
```

Out[83]:
```
(110527, 9)
```

Out[83]:
```
          appointment_id      scheduled_time     appointment_day  \
5030230         5030230 2015-11-10 07:13:56 2016-05-04 23:59:59
5122866         5122866 2015-12-03 08:17:28 2016-05-02 23:59:59
5134197         5134197 2015-12-07 10:40:59 2016-06-03 23:59:59
5134220         5134220 2015-12-07 10:42:42 2016-06-03 23:59:59
5134223         5134223 2015-12-07 10:43:01 2016-06-03 23:59:59

                neighborhood gender         patient_id  age  no_show  sms_received
5030230    RESISTÊNCIA      F 832256398961987.00   51    False             1
5122866     VILA RUBIM      M  91637474953513.00   34     True             1
5134197  SÃO CRISTÓVÃO      F   1216586867796.00   27     True             1
5134220        MARUÍPE      F  31899595421534.00   48    False             1
5134223  SÃO CRISTÓVÃO      F   9582232334148.00   80    False             1
```

Out[83]:
```
(110527, 3)
```

```
Out[83]:          appointment_id      scheduled_time  no_show
          5030230         5030230 2015-11-10 07:13:56    False
          5122866         5122866 2015-12-03 08:17:28     True
          5134197         5134197 2015-12-07 10:40:59     True
          5134220         5134220 2015-12-07 10:42:42    False
          5134223         5134223 2015-12-07 10:43:01    False

In [84]: # Rename columns to avoid confusion
         cutoff_times.rename(columns = {'scheduled_time': 'cutoff_time',
                                        'no_show': 'label'},
                             inplace = True)

In [85]: cutoff_times.head()

Out[85]:          appointment_id         cutoff_time  label
          5030230         5030230 2015-11-10 07:13:56  False
          5122866         5122866 2015-12-03 08:17:28   True
          5134197         5134197 2015-12-07 10:40:59   True
          5134220         5134220 2015-12-07 10:42:42  False
          5134223         5134223 2015-12-07 10:43:01  False

In [86]: # Generate features using the constructed entityset
         fm, features = ft.dfs(entityset=es,
                               target_entity='appointments',
                               agg_primitives=['count', 'percent_true'],
                               trans_primitives=['weekend', 'weekday', 'day', 'month', 'year']
                               max_depth=3,
                               approximate='6h',
                               cutoff_time=cutoff_times[20000:],
                               verbose=True)

Built 38 features
Elapsed: 02:43 | Remaining: 00:00 | Progress: 100%|| Calculated: 11/11 chunks


In [87]: fm.head()

Out[87]:                  label    neighborhood gender        patient_id  age  \
         appointment_id
         5623805          False    SANTA MARTHA      F    45329232236.00   18
         5623811          False    SANTA MARTHA      M  7756471622192.00   27
         5623814          False  JARDIM CAMBURI      M  7267524297161.00   20
         5623815          False   SANTO ANTÔNIO      F 77628457333817.00   33
         5623817          False       REPÚBLICA      F  2986374988373.00   74

                         WEEKEND(scheduled_time)  WEEKEND(appointment_day)  \
         appointment_id
         5623805                           False                     False
         5623811                           False                     False
```

```
5623814                          False                     False
5623815                          False                     False
5623817                          False                     False

               WEEKDAY(scheduled_time)  WEEKDAY(appointment_day)  \
appointment_id
5623805                              1                         1
5623811                              1                         3
5623814                              1                         2
5623815                              1                         0
5623817                              1                         3

               DAY(scheduled_time)  DAY(appointment_day)  \
appointment_id
5623805                         26                   31
5623811                         26                    5
5623814                         26                   11
5623815                         26                   16
5623817                         26                    5

               MONTH(scheduled_time)  MONTH(appointment_day)  \
appointment_id
5623805                            4                       5
5623811                            4                       5
5623814                            4                       5
5623815                            4                       5
5623817                            4                       5

               YEAR(scheduled_time)  YEAR(appointment_day)  \
appointment_id
5623805                        2016                   2016
5623811                        2016                   2016
5623814                        2016                   2016
5623815                        2016                   2016
5623817                        2016                   2016

               patients.COUNT(appointments)  \
appointment_id
5623805                                   0
5623811                                   0
5623814                                   0
5623815                                   0
5623817                                   0

               patients.PERCENT_TRUE(appointments.no_show)  \
appointment_id
5623805                                               0.00
5623811                                               0.00
```

```
5623814                                                0.00
5623815                                                0.00
5623817                                                0.00

               patients.PERCENT_TRUE(appointments.sms_received)  \
appointment_id
5623805                                                     0.00
5623811                                                     0.00
5623814                                                     0.00
5623815                                                     0.00
5623817                                                     0.00

               patients.WEEKDAY(first_appointments_time)  \
appointment_id
5623805                                                   1
5623811                                                   1
5623814                                                   1
5623815                                                   1
5623817                                                   1

               patients.DAY(first_appointments_time)  \
appointment_id
5623805                                           26
5623811                                           26
5623814                                           26
5623815                                           26
5623817                                           26

               patients.MONTH(first_appointments_time)  \
appointment_id
5623805                                              4
5623811                                              4
5623814                                              4
5623815                                              4
5623817                                              4

               patients.YEAR(first_appointments_time)  \
appointment_id
5623805                                         2016
5623811                                         2016
5623814                                         2016
5623815                                         2016
5623817                                         2016

               locations.COUNT(appointments)  \
appointment_id
5623805                              457.00
5623811                              457.00
```

```
5623814                              2272.00
5623815                               279.00
5623817                                61.00

               locations.PERCENT_TRUE(appointments.no_show)  \
appointment_id
5623805                                                0.00
5623811                                                0.00
5623814                                                0.00
5623815                                                0.00
5623817                                                0.00

               locations.PERCENT_TRUE(appointments.sms_received)  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

               ages.COUNT(appointments)  \
appointment_id
5623805                          212.00
5623811                          223.00
5623814                          249.00
5623815                          296.00
5623817                          129.00

               ages.PERCENT_TRUE(appointments.no_show)  \
appointment_id
5623805                                         0.00
5623811                                         0.00
5623814                                         0.00
5623815                                         0.00
5623817                                         0.00

               ages.PERCENT_TRUE(appointments.sms_received)  \
appointment_id
5623805                                                0.00
5623811                                                0.00
5623814                                                0.00
5623815                                                0.00
5623817                                                0.00

               genders.COUNT(appointments)  \
appointment_id
5623805                             13105
5623811                              6438
```

```
5623814                              6438
5623815                             13105
5623817                             13105

                genders.PERCENT_TRUE(appointments.no_show)  \
appointment_id
5623805                                              0.00
5623811                                              0.00
5623814                                              0.00
5623815                                              0.00
5623817                                              0.00

                genders.PERCENT_TRUE(appointments.sms_received)  \
appointment_id
5623805                                                  0.00
5623811                                                  0.00
5623814                                                  0.00
5623815                                                  0.00
5623817                                                  0.00

                patients.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                patients.PERCENT_TRUE(appointments.WEEKEND(appointment_day))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                locations.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                locations.PERCENT_TRUE(appointments.WEEKEND(appointment_day))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
```

```
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                ages.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                ages.PERCENT_TRUE(appointments.WEEKEND(appointment_day))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                genders.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))  \
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00

                genders.PERCENT_TRUE(appointments.WEEKEND(appointment_day))
appointment_id
5623805                                                    0.00
5623811                                                    0.00
5623814                                                    0.00
5623815                                                    0.00
5623817                                                    0.00
```

```
In [88]: features
```

```
Out[88]: [<Feature: neighborhood>,
          <Feature: gender>,
          <Feature: patient_id>,
          <Feature: age>,
          <Feature: WEEKEND(scheduled_time)>,
          <Feature: WEEKEND(appointment_day)>,
          <Feature: WEEKDAY(scheduled_time)>,
          <Feature: WEEKDAY(appointment_day)>,
          <Feature: DAY(scheduled_time)>,
          <Feature: DAY(appointment_day)>,
```

```
<Feature: MONTH(scheduled_time)>,
<Feature: MONTH(appointment_day)>,
<Feature: YEAR(scheduled_time)>,
<Feature: YEAR(appointment_day)>,
<Feature: patients.COUNT(appointments)>,
<Feature: patients.PERCENT_TRUE(appointments.no_show)>,
<Feature: patients.PERCENT_TRUE(appointments.sms_received)>,
<Feature: patients.WEEKDAY(first_appointments_time)>,
<Feature: patients.DAY(first_appointments_time)>,
<Feature: patients.MONTH(first_appointments_time)>,
<Feature: patients.YEAR(first_appointments_time)>,
<Feature: locations.COUNT(appointments)>,
<Feature: locations.PERCENT_TRUE(appointments.no_show)>,
<Feature: locations.PERCENT_TRUE(appointments.sms_received)>,
<Feature: ages.COUNT(appointments)>,
<Feature: ages.PERCENT_TRUE(appointments.no_show)>,
<Feature: ages.PERCENT_TRUE(appointments.sms_received)>,
<Feature: genders.COUNT(appointments)>,
<Feature: genders.PERCENT_TRUE(appointments.no_show)>,
<Feature: genders.PERCENT_TRUE(appointments.sms_received)>,
<Feature: patients.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))>,
<Feature: patients.PERCENT_TRUE(appointments.WEEKEND(appointment_day))>,
<Feature: locations.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))>,
<Feature: locations.PERCENT_TRUE(appointments.WEEKEND(appointment_day))>,
<Feature: ages.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))>,
<Feature: ages.PERCENT_TRUE(appointments.WEEKEND(appointment_day))>,
<Feature: genders.PERCENT_TRUE(appointments.WEEKEND(scheduled_time))>,
<Feature: genders.PERCENT_TRUE(appointments.WEEKEND(appointment_day))>]
```

```
In [89]: fm.shape

Out[89]: (90527, 39)
```

We have applied and stacked primitives like MONTH, WEEKDAY and PERCENT_TRUE to build features accross all the Entities in our EntitySet.

Feel free to fork this kernel and modify the parameters. By doing so, you can get very different feature matrices. Here's a short overview of the keywords used:

- **target_entity** is the entity for which we're building features. It would be equally easy to make a feature matrix for the locations entity
- **agg_primitives** and trans_primitives are lists of which primitives will be used while constructing features. The full list can be found by running ft.list_primitives()
- **max_depth=3** says to stack up to 3 primitives deep.
- **approximate='3h'** rounds cutoff times into blocks that are 3 hours long for faster computation
- **cutoff_time** is a dataframe that says when to calculate each row
- **verbose=True** makes the progress bar

2.6 Machine Learning

We can put the created feature matrix directly into sklearn. Similar to the other kernels, we do not do a good job predicting no-shows. With one unshuffled train test split, our roc_auc_score is roughly .5 with similar scores for F1 and K-first.

```
In [90]: from sklearn.model_selection import train_test_split
         from sklearn.ensemble import RandomForestClassifier
         from sklearn.metrics import accuracy_score, roc_auc_score
```

```
In [97]: %%time

         X = fm.copy().fillna(0)
         label = X.pop('label')
         X = X.drop(['patient_id', 'neighborhood', 'gender'], axis=1)

         X_train, X_test, y_train, y_test = train_test_split(X, label, test_size=0.30, shuffle=

         clf = RandomForestClassifier(n_estimators=150)
         clf.fit(X_train, y_train)
         probs = clf.predict_proba(X_test)
         print('AUC score of {:.3f}'.format(roc_auc_score(y_test, probs[:,1])))
```

```
AUC score of 0.531
Wall time: 13.6 s
```

max_depth=3: AUC score of 0.531

```
In [92]: feature_imps = [(imp, X.columns[i]) for i, imp in enumerate(clf.feature_importances_)]
         feature_imps.sort()
         feature_imps.reverse()
         print('Random Forest Feature Importances:')
         for i, f in enumerate(feature_imps[0:8]):
             print('{}: {} [{:.3f}]'.format(i + 1, f[1], f[0]/feature_imps[0][0]))
```

```
Random Forest Feature Importances:
1: locations.COUNT(appointments) [1.000]
2: ages.COUNT(appointments) [0.903]
3: age [0.891]
4: DAY(appointment_day) [0.874]
5: locations.PERCENT_TRUE(appointments.no_show) [0.630]
6: locations.PERCENT_TRUE(appointments.sms_received) [0.625]
7: ages.PERCENT_TRUE(appointments.no_show) [0.618]
8: ages.PERCENT_TRUE(appointments.sms_received) [0.609]
```

```
In [93]: p1 = utils.plot_roc_auc(y_test, probs)
         p2 = utils.plot_f1(y_test, probs, 1000)
         # p3 = utils.plot_kfirst(y_test, probs, 300)
```

```
D:\Programs\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:1135: UndefinedMetri
  'precision', 'predicted', average, warn_for)
D:\Programs\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:1135: UndefinedMetri
  'precision', 'predicted', average, warn_for)
```

```
In [94]: from bokeh.io import show
         from bokeh.layouts import gridplot
         show(gridplot([p1, p2], ncols=1))
```

```
In [95]: p4 = utils.plot_locations(fm)
         p5 = utils.plot_noshow_by_loc(fm)
         p6 = utils.plot_ages(fm)
         p7 = utils.plot_noshow_by_age(X)
```

```
In [96]: show(gridplot([p4, p6, p5, p7], ncols=2))
```

```
In [ ]:
```