In [1]:
```
#2020-01-15
#Raw codes of each functions are stored in my hjin85 account in GT Ent
erprise Github
#CSE6250 homework project 1
#The purpose of this project is to
#1. pre-process publicly available patient survival data
#2. train machine using scikit-learn package
#3. predict patient outcome
#4. Assess and validate the credibility of the algorithm.

import event_statistics
```

In [7]:
```
event_statistics.main()
#Summary of statistics
#Eventn count, Encounter count and Record length
```

```
Time to compute event count metrics: 0.08703780174255371s
(1, 8635, 982.014, 1, 12627, 498.118)
Time to compute encounter count metrics: 1.2906949520111084s
(1, 203, 23.038, 1, 391, 15.452)
Time to compute record length metrics: 0.21472501754760742s
(0, 1972, 127.532, 0, 2914, 159.2)
```

In [8]:
```
import etl
```

In [9]:
```
etl.main()
#Feature construciton
#Converting raw data into SVMlight format
```

```
Calculating index date...
Filtering events...
Aggregating events...
Re-mapping feature ids...
Building feature tuples...
Saving in svm format...
```

In [10]:
```
import models_partb
```

In [11]:
```
models_partb.main()
#Machine Learning algorithm based model creation. Logistic Regression,
SVM and Decesion Tree
```

```
--------------------------------------------------------------------------------
Classifier: Logistic Regression
Accuracy: 0.9545454545454546
AUC: 0.9454047619047619
Precision: 0.9869281045751634
Recall: 0.8988095238095238
F1-score: 0.9408099688473521
--------------------------------------------------------------------------------


--------------------------------------------------------------------------------
Classifier: SVM
Accuracy: 0.9940191387559809
AUC: 0.9945119047619048
Precision: 0.9882005899705014
Recall: 0.9970238095238095
F1-score: 0.9925925925925925
--------------------------------------------------------------------------------


--------------------------------------------------------------------------------
Classifier: Decision Tree
Accuracy: 0.7763157894736842
AUC: 0.7475952380952382
Precision: 0.792156862745098
Recall: 0.6011904761904762
F1-score: 0.6835871404399323
--------------------------------------------------------------------------------
```

In [12]:
```
import models_partc
```

In [13]:
```
models_partc.main()
#Evaluate the model in a seperate dataset
```

```
_____________________________________________________
Classifier: Logistic Regression
Accuracy: 0.7380952380952381
AUC: 0.7375
Precision: 0.6804123711340206
Recall: 0.7333333333333333
F1-score: 0.7058823529411765
_____________________________________________________


_____________________________________________________
Classifier: SVM
Accuracy: 0.7380952380952381
AUC: 0.7388888888888889
Precision: 0.6767676767676768
Recall: 0.7444444444444445
F1-score: 0.708994708994709
_____________________________________________________


_____________________________________________________
Classifier: Decision Tree
Accuracy: 0.6714285714285714
AUC: 0.6569444444444444
Precision: 0.6329113924050633
Recall: 0.5555555555555556
F1-score: 0.591715976331361
_____________________________________________________
```

In [14]:
```
import cross
```

In [15]:
```
cross.main()
#Model validation using K-fold and randomized K-fold
```

```
Classifier: Logistic Regression__________
Average Accuracy in KFold CV: 0.6646706586826348
Average AUC in KFold CV: 0.7089067999394432
Average Accuracy in Randomised CV: 0.7142857142857143
Average AUC in Randomised CV: 0.7308461500509049
```

In [17]:
```
import my_model
```

In [18]:
```
my_model.main()
#Creating My Model, modeified from previous default test to improve pr
ediciton power.
```

```
Classifier: Decision Tree Regressor__________
Average Accuracy in KFold CV: 0.7065868263473054
Average AUC in KFold CV: 0.6826458532395303
Average Accuracy in Randomised CV: 0.6746411483253588
Average AUC in Randomised CV: 0.6628297504864005
```