Data Mining Using Python

File access required: In Colab this notebook requires first uploading files **Shop.csv** and **Movies.csv** using the *Files* feature in the left toolbar. If running the notebook on a local computer, simply ensure these files are in the same workspace as the notebook.

```
# Set-up
import csv
```

Look at CSV files: TID,item pairs

```
# Read shopping dataset from CSV file
# Create dictionary "Sitems" with key = item and value = set of transactions
# Also set variable Snumtrans = number of transactions
Sitems = {}
trans = []  # list of transactions used to set Snumtrans
with open('Shop.csv') as f:
    rows = csv.DictReader(f)
    for r in rows:
        if r['item'] not in Sitems:
            Sitems[r['item']] = {r['TID']}
        else:
            Sitems[r['item']].add(r['TID'])
        if r['TID'] not in trans:
            trans.append(r['TID'])
Snumtrans = len(trans)
print('Number of transactions:', Snumtrans)
print('Number of distinct items:', len(Sitems))
print('Item dictionary:')
Sitems
```

```
Number of transactions: 5
Number of distinct items: 5
Item dictionary:
{'milk': {'1', '2', '4', '5'},
 'eggs': {'1', '3', '4'},
 'juice': {'1', '2', '5'},
 'cookies': {'2', '5'},
 'chips': {'3', '5'}}
```

```
# Read movies dataset from CSV file
# Create dictionary "Mitems" with key = item and value = set of transactions
# Also set variable Mnumtrans = number of transactions
Mitems = {}
trans = []  # list of transactions used to set Mnumtrans
with open('Movies.csv') as f:
    rows = csv.DictReader(f)
    for r in rows:
        if r['item'] not in Mitems:
            Mitems[r['item']] = {r['TID']}
        else:
            Mitems[r['item']].add(r['TID'])
        if r['TID'] not in trans:
            trans.append(r['TID'])
Mnumtrans = len(trans)
print('Number of transactions (users):', Mnumtrans)
print('Number of distinct items (movies):', len(Mitems))
print('Item dictionary:')
Mitems.items()
```

'116822', '233054', '800', '113089', '17574', '177422', '243737', '143932', '72265', '180714', '34520', '221327', '237363',
'146114', '7550', '70922', '246994', '71268', '112880', '171801', '50188', '185316', '88642', '168845', '148467', '16138',
'179625', '30024', '67440', '214091', '232001', '46956', '186857', '70985', '228295', '77159', '130198', '121928', '35107',
'198435', '59455', '7556', '69458', '43402', '126382', '20439', '196785', '126638', '198332', '86172', '241577', '159570',
'129692', '180630', '169154', '94555', '17873', '186982', '83321', '224672', '136254', '17525', '196565', '176880', '17451',
'162227', '3967', '131967', '200446', '175118', '193074', '183699', '36445', '218962', '115786', '150148', '12966', '14658',
'147652', '3999', '121896', '115323', '93970', '149685', '75806', '13399', '104829', '234230', '52898', '157019', '27332',
'237212', '239816', '170329', '1308', '44424', '6198', '154994', '204070', '65387', '51145', '160346', '230165', '130414',
'208272', '211213', '180951', '58979', '121912', '225579', '231812', '95669', '85194', '72725', '211617', '30610', '205020',
'46770', '112697', '32458', '63031', '207966', '91830', '214028', '246553', '57305', '20660', '7353', '42036', '77397',
'117731', '92540', '10221', '206146', '179943', '100316', '2451', '158048', '192190', '151584', '78629', '21326', '94756',
'76891', '106732', '189096', '78152', '241462', '29559', '184307', '125244', '72313', '152679', '72311', '38153', '11256',
'143831', '119971', '179723', '121408', '30252', '216653', '2788', '117754', '9391', '181883', '245455', '55980', '78715',
'220272', '247354', '4208', '201265', '98657', '164259', '64050', '126642', '102107', '92853', '184040', '6199', '142634',
'41641', '180826', '122196', '40311', '49251', '165611', '4966', '216223', '222902', '49159', '194971', '134268', '208331',
'75198', '196152', '168240', '220597', '241885', '240282', '125664', '124789', '133012', '245601', '177796', '135254',
'128655', '237686', '187318', '190266', '199832', '209357', '174531', '54858', '20316', '113098', '46229', '84605', '87948',
'10088', '152735', '217853', '165357', '42553', '152646', '221712', '47351', '150186', '4588', '244116', '182961', '45340',
'126809', '226699', '124662', '76895', '1040', '87350', '99742', '40145', '182584', '131624', '25781', '6569', '12354',
'26367', '231669', '197328', '239808', '132751', '101070', '182194', '214778', '84824', '168374', '141192', '192154', '70635',
'144712', '72653', '106446', '182266', '41734', '97686', '150676', '71124', '101420', '230426', '31198', '118077', '78567',
'128', '39395', '125358', '61988', '10526', '111500', '208285', '192071', '119622', '234009', '409', '51681', '54810',
'245978', '208244', '115108', '84141', '68398', '57311', '216969', '94344', '101427', '140032', '109637', '180743', '155352',
'56105', '192921', '40611', '15030', '101773', '146075', '222511', '218549', '123700', '128203', '180358', '129638', '116803',
'27210', '4577', '196754', '90271', '197358', '39014', '71125', '33710', '12784', '101743', '16223', '219962', '117509',
'73352', '161620', '82093', '130649', '89068', '140741', '716', '32210', '127474', '52609', '151641', '192136', '200039',
'243482', '71301', '59761', '100171', '14883', '98509', '176', '181594', '3508', '180682', '67073', '50633', '5731', '235725',
'179447', '182343', '74318', '73832', '18153', '197036', '195064', '101711', '127849', '208149', '225433', '25938', '86041',
'179398', '202380', '40926', '188806', '226605', '184893', '9444', '98461', '59430', '96721', '73132', '135443', '203108',
'142355', '138662', '78095', '19464', '220810', '171061', '199715', '107460', '83074', '208473', '22260', '8215', '74343',
'207834', '156006', '201282', '178842', '79410', '6993', '15590', '72188', '126725', '4987', '35322', '175887', '138121',
'215987', '184505', '68930', '81365', '116345', '35964', '38449', '207158', '229260', '114382', '230758', '62076', '10096',
'179897', '27884', '14588', '84280', '205258', '5767', '246256', '47999', '117631', '148858', '106668', '145959', '95915',
'205045', '146607', '4904', '43142', '21717', '213657', '221353', '94911', '65892', '196710', '227554', '17175', '24416',
'197145', '60390', '53303', '4938', '208253', '198726', '169556', '15823', '104601', '174147', '214168', '178832', '211373',
'158208', '70707', '223174', '20227', '21251', '15638', '173090', '227290'}), ('Magic Mike XXL', {'105787', '49159', '43650',
'132287', '800', '231073', '86758', '177796'}), ('Teenage Mutant Ninja Turtles', {'39827', '228452', '39395', '174464',
'143342', '240694', '102107', '1040', '131624', '6569', '15108', '55096', '245978', '87907', '206452', '132751', '84141',
'60699', '216969', '132287', '92094', '23534', '114382', '7556', '72265', '173452', '169713', '32946', '92485', '105787',
'89294', '126655', '47351', '38153', '106446', '218424', '244116', '146607', '154994'}), ('Fury', {'78715', '4208', '178408',
'201265', '98657', '243356', '206486', '73352', '26191', '186857', '4231', '163775', '55096', '6199', '174580', '32210',
'181853', '64244', '152067', '41453', '109543', '200039', '71301', '49159', '194971', '176', '20439', '222011', '3508',
'59860', '36099', '120181', '67073', '159570', '50633', '169154', '241885', '66708', '195064', '19071', '135254', '49333',
'17451', '98732', '209357', '136604', '190627', '240700', '134460', '28666', '184893', '60008', '245705', '147652', '96721',
'23534', '108127', '24427', '142355', '105787', '211820', '220810', '27332', '50123', '244116', '208823', '20115', '208473',

Some new Python features

```
# Iterating through dictionaries
for i in Sitems:
    print(i)
    print(Sitems[i])
```

```
milk
{'2', '4', '1', '5'}
eggs
{'4', '1', '3'}
juice
{'2', '5', '1'}
cookies
{'2', '5'}
chips
{'5', '3'}
```

```
# Intersecting sets
# How many transactions contain both eggs and milk?
set1 = Sitems['eggs']
print('Transactions containing eggs:', set1)
set2 = Sitems['milk']
print('Transactions containing milk:', set2)
set3 = set1 & set2
print('Transactions containing both:', set3)
# print('Number of transactions containing both:', len(set3))
```

```
Transactions containing eggs: {'4', '1', '3'}
Transactions containing milk: {'2', '4', '1', '5'}
Transactions containing both: {'4', '1'}
```

Shopping dataset - frequent item-sets

Frequent item-sets of two

Print all pairs of items and the number of transactions they occur together in (see what's wrong and fix it)

```
for i1 in Sitems:
    for i2 in Sitems:
        common = len(Sitems[i1] & Sitems[i2])
        print([i1, i2, common])
```

```
['milk', 'milk', 4]
['milk', 'eggs', 2]
['milk', 'juice', 3]
['milk', 'cookies', 2]
['milk', 'chips', 1]
['eggs', 'milk', 2]
['eggs', 'eggs', 3]
['eggs', 'juice', 1]
['eggs', 'cookies', 0]
['eggs', 'chips', 1]
['juice', 'milk', 3]
['juice', 'eggs', 1]
['juice', 'juice', 3]
['juice', 'cookies', 2]
['juice', 'chips', 1]
['cookies', 'milk', 2]
['cookies', 'eggs', 0]
['cookies', 'juice', 2]
['cookies', 'cookies', 2]
['cookies', 'chips', 1]
['chips', 'milk', 1]
['chips', 'eggs', 1]
['chips', 'juice', 1]
['chips', 'cookies', 1]
['chips', 'chips', 2]
```

Now only print pairs that meet support threshold

```
support = .3
for i1 in Sitems:
    for i2 in Sitems:
      if i1 < i2:
        common = len(Sitems[i1] & Sitems[i2])
        if common/Snumtrans > support:
          print(i1, '|', i2)
```

```
eggs | milk
juice | milk
cookies | milk
cookies | juice
```

Frequent item-sets of three

```
support = .1
for i1 in Sitems:
    for i2 in Sitems:
        for i3 in Sitems:
            if i1 < i2 and i2 < i3:
                common = len(Sitems[i1] & Sitems[i2] & Sitems[i3])
                if common/Snumtrans > support:
                    print(i1, '|', i2, '|', i3)
```

```
eggs | juice | milk
cookies | juice | milk
chips | juice | milk
chips | cookies | milk
chips | cookies | juice
```

Your Turn - Movies dataset frequent item-sets

```
print(Mnumtrans, 'transactions (users)')
print(len(Mitems), 'distinct items (movies)')
```

```
1382 transactions (users)
123 distinct items (movies)
```

Mine for frequent item-sets of three and four items in the Movies dataset. Find a single support threshold where the number of frequent item-sets of three items is more than 10 but less than 20, and the number of frequent item-sets of four items is more than 0.

```
# Frequent item-sets of three
support = 0.01

count = 0
for i1 in Mitems:
    for i2 in Mitems:
        if i1 < i2:
            for i3 in Mitems:
              if i3 > i2:
                common = len(Mitems[i1] & Mitems[i2] & Mitems[i3])
                if common/Mnumtrans > support:
                  print (i1, '|', i2, '|', i3)
                  count += 1
print('Total:' , count)
```

```
Boyhood | The Fault in Our Stars | The Imitation Game
Boyhood | The Imitation Game | Wild Tales
Boyhood | Inside Out | The Fault in Our Stars
Boyhood | Inside Out | The Imitation Game
Boyhood | Inside Out | Wild Tales
Boyhood | Gone Girl | The Fault in Our Stars
Boyhood | Gone Girl | The Imitation Game
Boyhood | Gone Girl | Inside Out
Boyhood | Gone Girl | Wild Tales
Boyhood | Fury | The Fault in Our Stars
Boyhood | Fury | The Imitation Game
Boyhood | Fury | Inside Out
Boyhood | Fury | Gone Girl
Boyhood | Calvary | The Imitation Game
Boyhood | Calvary | Gone Girl
Boyhood | Calvary | Fury
Big Hero 6 | The Fault in Our Stars | The Imitation Game
Big Hero 6 | Boyhood | The Fault in Our Stars
Big Hero 6 | Boyhood | The Imitation Game
Big Hero 6 | Boyhood | Inside Out
Big Hero 6 | Boyhood | Gone Girl
Big Hero 6 | Boyhood | Fury
Big Hero 6 | Boyhood | Wild Tales
Big Hero 6 | The Imitation Game | Wild Tales
Big Hero 6 | The Imitation Game | Transformers: Age of Extinction
Big Hero 6 | Inside Out | The Fault in Our Stars
Big Hero 6 | Inside Out | The Imitation Game
Big Hero 6 | Inside Out | The Hunger Games: Mockingjay - Part 2
Big Hero 6 | Inside Out | Louis C.K.: Live at The Comedy Store
Big Hero 6 | Inside Out | Wild Tales
Big Hero 6 | Inside Out | Transformers: Age of Extinction
Big Hero 6 | Gone Girl | The Fault in Our Stars
Big Hero 6 | Gone Girl | The Imitation Game
Big Hero 6 | Gone Girl | Inside Out
Big Hero 6 | Gone Girl | Louis C.K.: Live at The Comedy Store
Big Hero 6 | Gone Girl | Wild Tales
Big Hero 6 | Gone Girl | Transformers: Age of Extinction
Big Hero 6 | Teenage Mutant Ninja Turtles | The Imitation Game
Big Hero 6 | Fury | The Fault in Our Stars
Big Hero 6 | Fury | The Imitation Game
Big Hero 6 | Fury | Inside Out
Big Hero 6 | Fury | Gone Girl
Big Hero 6 | Fury | Transformers: Age of Extinction
Big Hero 6 | The Hunger Games: Mockingjay - Part 2 | The Imitation Game
Big Hero 6 | Louis C.K.: Live at The Comedy Store | The Imitation Game
Big Hero 6 | Calvary | The Imitation Game
Big Hero 6 | Calvary | Gone Girl
Inside Out | The Fault in Our Stars | The Imitation Game
Inside Out | The Imitation Game | Wild Tales
Inside Out | The Imitation Game | Transformers: Age of Extinction
Inside Out | The Hunger Games: Mockingjay - Part 2 | The Imitation Game
```

```
Inside Out | Louis C.K.: Live at The Comedy Store | The Imitation Game
Gone Girl | The Fault in Our Stars | The Imitation Game
Gone Girl | The Imitation Game | Wild Tales
Gone Girl | The Imitation Game | Transformers: Age of Extinction
Gone Girl | Inside Out | The Fault in Our Stars
Gone Girl | Inside Out | The Imitation Game
```

```
# Frequent item-sets of four
support = 0.01

count = 0
for i1 in Mitems:
    for i2 in Mitems:
        if i1 < i2:
            for i3 in Mitems:
              if i2 < i3:
                for i4 in Mitems:
                  if i3 < i4:
                    common = len(Mitems[i1] & Mitems[i2] & Mitems[i3] & Mitems[i4])
                    if common/Mnumtrans > support:
                      print (i1, '|', i2, '|', i3, '|', i4)
                      count += 1

print('Total:' , count)
```

```
Boyhood | Inside Out | The Fault in Our Stars | The Imitation Game
Boyhood | Gone Girl | The Fault in Our Stars | The Imitation Game
Boyhood | Gone Girl | The Imitation Game | Wild Tales
Boyhood | Gone Girl | Inside Out | The Fault in Our Stars
Boyhood | Gone Girl | Inside Out | The Imitation Game
Boyhood | Fury | The Fault in Our Stars | The Imitation Game
Boyhood | Fury | Inside Out | The Imitation Game
Boyhood | Fury | Gone Girl | The Fault in Our Stars
Boyhood | Fury | Gone Girl | The Imitation Game
Boyhood | Fury | Gone Girl | Inside Out
Boyhood | Calvary | Gone Girl | The Imitation Game
Boyhood | Calvary | Fury | The Imitation Game
Boyhood | Calvary | Fury | Gone Girl
Big Hero 6 | Boyhood | The Fault in Our Stars | The Imitation Game
Big Hero 6 | Boyhood | The Imitation Game | Wild Tales
Big Hero 6 | Boyhood | Inside Out | The Fault in Our Stars
Big Hero 6 | Boyhood | Inside Out | The Imitation Game
Big Hero 6 | Boyhood | Gone Girl | The Fault in Our Stars
Big Hero 6 | Boyhood | Gone Girl | The Imitation Game
Big Hero 6 | Boyhood | Gone Girl | Inside Out
Big Hero 6 | Boyhood | Fury | The Imitation Game
Big Hero 6 | Boyhood | Fury | Inside Out
Big Hero 6 | Boyhood | Fury | Gone Girl
Big Hero 6 | Inside Out | The Fault in Our Stars | The Imitation Game
Big Hero 6 | Inside Out | The Imitation Game | Wild Tales
Big Hero 6 | Inside Out | The Hunger Games: Mockingjay - Part 2 | The Imitation Game
Big Hero 6 | Gone Girl | The Fault in Our Stars | The Imitation Game
Big Hero 6 | Gone Girl | The Imitation Game | Wild Tales
Big Hero 6 | Gone Girl | Inside Out | The Fault in Our Stars
Big Hero 6 | Gone Girl | Inside Out | The Imitation Game
Big Hero 6 | Gone Girl | Inside Out | Louis C.K.: Live at The Comedy Store
Big Hero 6 | Gone Girl | Inside Out | Wild Tales
Big Hero 6 | Gone Girl | Louis C.K.: Live at The Comedy Store | The Imitation Game
Big Hero 6 | Fury | The Imitation Game | Transformers: Age of Extinction
Big Hero 6 | Fury | Inside Out | The Imitation Game
Big Hero 6 | Fury | Gone Girl | The Imitation Game
Big Hero 6 | Fury | Gone Girl | Inside Out
Big Hero 6 | Calvary | Gone Girl | The Imitation Game
Gone Girl | Inside Out | The Fault in Our Stars | The Imitation Game
Gone Girl | Inside Out | The Imitation Game | Wild Tales
Fury | Gone Girl | The Imitation Game | Wild Tales
Fury | Gone Girl | The Imitation Game | Transformers: Age of Extinction
Fury | Gone Girl | Inside Out | The Imitation Game
Calvary | Fury | Gone Girl | The Imitation Game
Total: 44
```

Shopping dataset - association rules

Association rules with one item on the left-hand side

First compute frequent item-sets of one item, as candidate left-hand sides of assocation rules. Include the number of transactions the items occur in.

```
support = .5
frequentLHS = []
for i in Sitems:
    if len(Sitems[i])/Snumtrans > support:
        frequentLHS.append([i,len(Sitems[i])])
print(frequentLHS)
```

```
[['milk', 4], ['eggs', 3], ['juice', 3]]
```

Now find right-hand side items with sufficient confidence (see what's wrong and fix it)

```
# S -> i

confidence = .5
for lhs in frequentLHS:
    for i in Sitems:
        common = len(Sitems[lhs[0]] & Sitems[i])
        if common/lhs[1] > confidence:
            print(lhs[0], '->', i)
```

```
milk -> milk
milk -> juice
eggs -> milk
eggs -> eggs
juice -> milk
juice -> juice
juice -> cookies
```

Association rules with two items on the left-hand side

First compute frequent item-sets of two items, as candidate left-hand sides of assocation rules. Include the number of transactions the items occur in.

```
# S = [JUICE, MILK]
support = .5
frequentLHS = []
for i1 in Sitems:
    for i2 in Sitems:
        if i1 < i2:
            common = len(Sitems[i1] & Sitems[i2])
            if common/Snumtrans > support:
                frequentLHS.append([i1,i2,common])
print(frequentLHS)
```

```
[['juice', 'milk', 3]]
```

Now find right-hand side items with sufficient confidence

```
confidence = .5
for lhs in frequentLHS:
    for i in Sitems:
        if i not in lhs:
            common = len(Sitems[lhs[0]] & Sitems[lhs[1]] & Sitems[i])
            if common/lhs[2] > confidence:
                print(lhs[0], '|', lhs[1], '->', i)
```

```
juice | milk -> cookies
```

Shopping dataset - association rules with lift instead of confidence

Association rules with one item on the left-hand side

First compute frequent item-sets of one item, as candidate left-hand sides of assocation rules. Include the number of transactions the items occur in.

```
support = .5
frequentLHS = []
for i in Sitems:
    if len(Sitems[i])/Snumtrans > support:
        frequentLHS.append([i,len(Sitems[i])])
print(frequentLHS)
```

```
[['milk', 4], ['eggs', 3], ['juice', 3]]
```

Now find right-hand side items with sufficient lift

```
liftthresh = 1
for lhs in frequentLHS:
    for i in Sitems:
        if i not in lhs:
            common = len(Sitems[lhs[0]] & Sitems[i])
            lift = (common/lhs[1]) / (len(Sitems[i])/Snumtrans)
            if lift > liftthresh:
                print(lhs[0], '->', i, ' lift:', lift)
```

```
milk -> juice  lift: 1.25
milk -> cookies  lift: 1.25
juice -> milk  lift: 1.25
juice -> cookies  lift: 1.6666666666666665
```

Association rules with two items on the left-hand side

First compute frequent item-sets of two items, as candidate left-hand sides of assocation rules. Include the number of transactions the items occur in.

```
support = .5
frequentLHS = []
for i1 in Sitems:
    for i2 in Sitems:
        if i1 < i2:
            common = len(Sitems[i1] & Sitems[i2])
            if common/Snumtrans > support:
                frequentLHS.append([i1,i2,common])
print(frequentLHS)
```

```
[['juice', 'milk', 3]]
```

Now find right-hand side items with sufficient lift

```
liftthresh = 1
for lhs in frequentLHS:
    for i in Sitems:
        if i not in lhs:
            common = len(Sitems[lhs[0]] & Sitems[lhs[1]] & Sitems[i])
            lift = (common/lhs[2]) / (len(Sitems[i])/Snumtrans)
            if lift > liftthresh:
                print(lhs[0], '|', lhs[1], '->', i, ' lift:', lift)
```

```
juice | milk -> cookies  lift: 1.6666666666666665
```

Your Turn - Movies dataset association rules

Mine for association rules in the Movies dataset with three items on the left-hand side. Find support and confidence thresholds (need not be the same) so the number of association rules is more than 10 but less than 20.

```
# Association rules with three items on the left-hand side
# Hint: Make sure to include the code from the seprate cells above that
#   together implement the two steps of association rule mining
```

```
support = 0.01
frequentLHS = []
for i1 in Mitems:
  for i2 in Mitems:
    if i1 < i2:
      for i3 in Mitems:
        if i2 < i3:
          common = len(Mitems[i1] & Mitems[i2] & Mitems[i3])
          if common/Mnumtrans > support:
            frequentLHS.append([i1, i2, i3, common])
print('FrequestHLS:', frequentLHS)

confidence = 0.4
for lhs in frequentLHS:
  for i in Mitems:
    if i not in [lhs[0], lhs[1], lhs[2]]:
      common = len(Mitems[lhs[0]] & Mitems[lhs[1]] & Mitems[lhs[2]] & Mitems[i])
      if common/lhs[3] > confidence:
        print(lhs[0], ',', lhs[1], ',', lhs[2], '->', i)
```

```
Gone Girl , The Big Short , The Imitation Game -> Fury
Gone Girl , Louis C.K.: Live at The Comedy Store , The Imitation Game -> Big Hero 6
Gone Girl , Louis C.K.: Live at The Comedy Store , The Imitation Game -> Inside Out
Fury , The Fault in Our Stars , The Imitation Game -> Boyhood
Fury , The Fault in Our Stars , The Imitation Game -> Big Hero 6
Fury , The Fault in Our Stars , The Imitation Game -> Inside Out
Fury , The Fault in Our Stars , The Imitation Game -> Gone Girl
Fury , The Imitation Game , Wild Tales -> Boyhood
Fury , The Imitation Game , Wild Tales -> Big Hero 6
Fury , The Imitation Game , Wild Tales -> Inside Out
Fury , The Imitation Game , Wild Tales -> Gone Girl
Fury , The Imitation Game , Transformers: Age of Extinction -> Boyhood
Fury , The Imitation Game , Transformers: Age of Extinction -> Big Hero 6
Fury , The Imitation Game , Transformers: Age of Extinction -> Inside Out
Fury , The Imitation Game , Transformers: Age of Extinction -> Gone Girl
Fury , Inside Out , The Imitation Game -> Boyhood
Fury , Inside Out , The Imitation Game -> Big Hero 6
Fury , Inside Out , The Imitation Game -> Gone Girl
Fury , Gone Girl , The Fault in Our Stars -> Boyhood
Fury , Gone Girl , The Fault in Our Stars -> Big Hero 6
Fury , Gone Girl , The Fault in Our Stars -> The Imitation Game
Fury , Gone Girl , The Fault in Our Stars -> Inside Out
Fury , Gone Girl , The Fault in Our Stars -> Wild Tales
Fury , Gone Girl , The Imitation Game -> Boyhood
Fury , Gone Girl , The Imitation Game -> Big Hero 6
Fury , Gone Girl , The Imitation Game -> Inside Out
Fury , Gone Girl , Inside Out -> Boyhood
Fury , Gone Girl , Inside Out -> Big Hero 6
Fury , Gone Girl , Inside Out -> The Imitation Game
Fury , Gone Girl , Wild Tales -> The Fault in Our Stars
Fury , Gone Girl , Wild Tales -> Boyhood
Fury , Gone Girl , Wild Tales -> Big Hero 6
Fury , Gone Girl , Wild Tales -> The Imitation Game
Fury , Gone Girl , Wild Tales -> Inside Out
Fury , Gone Girl , Transformers: Age of Extinction -> Boyhood
Fury , Gone Girl , Transformers: Age of Extinction -> Big Hero 6
Fury , Gone Girl , Transformers: Age of Extinction -> The Imitation Game
Fury , Gone Girl , Transformers: Age of Extinction -> Inside Out
Fury , Gone Girl , Transformers: Age of Extinction -> Teenage Mutant Ninja Turtles
Big Eyes , Boyhood , The Imitation Game -> Big Hero 6
Big Eyes , Boyhood , The Imitation Game -> Inside Out
Big Eyes , Boyhood , The Imitation Game -> Gone Girl
Big Eyes , Boyhood , The Imitation Game -> Fury
Big Eyes , Boyhood , The Imitation Game -> Wild Tales
Big Eyes , Boyhood , Gone Girl -> Big Hero 6
Big Eyes , Boyhood , Gone Girl -> The Imitation Game
Big Eyes , Boyhood , Gone Girl -> Inside Out
Big Eyes , Boyhood , Gone Girl -> Fury
Big Eyes , Boyhood , Gone Girl -> Wild Tales
Big Eyes , Gone Girl , The Imitation Game -> Boyhood
Big Eyes , Gone Girl , The Imitation Game -> Big Hero 6
Big Eyes , Gone Girl , The Imitation Game -> Inside Out
Big Eyes , Gone Girl , The Imitation Game -> Fury
Calvary , Gone Girl , The Imitation Game -> Boyhood
Calvary , Gone Girl , The Imitation Game -> Big Hero 6
Calvary , Gone Girl , The Imitation Game -> Inside Out
Calvary , Gone Girl , The Imitation Game -> Fury
Calvary , Gone Girl , Inside Out -> Boyhood
```

Mine for association rules in the Movies dataset with three items on the left-hand side. Find support and lift thresholds so the number of association rules is more than 10 but less than 20. Only consider lift thresholds > 1.

```
# Association rules with three items on the left-hand side
support = 0.01
frequentLHS = []
for i1 in Mitems:
  for i2 in Mitems:
    if i1 < i2:
      for i3 in Mitems:
        if i2 < i3:
          common = len(Mitems[i1] & Mitems[i2] & Mitems[i3])
          if common / Mnumtrans >= support:
            frequentLHS.append([i1, i2, i3, common])
print('FrequestHLS:', frequentLHS)

liftthresh = 1.0
for lhs in frequentLHS:
    for i in Mitems:
        if i not in [lhs[0], lhs[1], lhs[2]]:
            common = len(Mitems[lhs[0]] & Mitems[lhs[1]] & Mitems[lhs[2]] & Mitems[i])
            lift = (common / Mnumtrans) / (len(Mitems[i]) / Mnumtrans)
            if lift >= liftthresh:
                print(lhs[0], ',', lhs[1], ',', lhs[2], '->', i, 'lift:', lift)
```

```
Fury , The Imitation Game , Transformers: Age of Extinction -> Aziz Ansari: Live at Madison Square Garden lift: 1.0
Fury , The Imitation Game , Transformers: Age of Extinction -> Exists lift: 1.0
```