Old Dataset(Staircase)

Importing Jury data csv file to data and renaming the columns ¶

In [1]:

```
# coding: utf-8
import pandas as pd
import numpy as np

data = pd.read_csv('jury_data.csv', encoding= 'ISO-8859-1', skiprows=[0,2])
data.rename(columns={"Was defendant Mesa Management negligent?": "Mesa_Negligen
t",
                         "Was Mesa Management's negligence a substantial factor
 in causing harm to  Mackenzie Dunn?":"Liability",
                         "What are the total damages that you find that MacKenzi
e Dunn sufferered?":"damages" ,
                         "What is your sex?": "gender",
                         "Please write your answer to the preceding damages ques
tion in words (quality check).":"damages_word",
                         "What percentage of responsibility for Mackenzie Dunn's
 injuries was each party responsible for? (Answers should add up to 100%) - Mesa
 Management Co":"Mesa_reponsible_percentage",
                         "Path":"Scenario",
                         "Was MacKenzie Dunn negligent?":"Dunn_negligent",
                         "Unnamed: 63":"perc_calc",
                     "Which of the following best describes your total household
 income?":"Income"
                         },inplace=True)
#data['mm_perc'] = np.where(data['Mesa_reponsible_percentage']>=1, data['perc_ca
lc'],data['Mesa_reponsible_percentage'])
data['mm_perc']=data['Mesa_reponsible_percentage']
req_data = pd.DataFrame(data[["Mesa_Negligent","damages","Liability",
                 "gender",
                 "damages_word",
                 "Scenario","Dunn_negligent","perc_calc","Start Date","End Date"
,"mm_perc","Income"]])


req_data['Liability'] = req_data['Liability'].map({'Yes': 1, 'No': 0})


print(req_data.columns)
```

```
Index(['Mesa_Negligent', 'damages', 'Liability', 'gender', 'damages_
word',
       'Scenario', 'Dunn_negligent', 'perc_calc', 'Start Date', 'End
Date',
       'mm_perc', 'Income'],
      dtype='object')
```

In [2]:

```
data['mm_perc'].unique()
```

Out[2]:

```
array([        nan,  0.65      , 65.        ,  0.6       ,  0.1
  ,
        0.5       ,  0.7       ,  0.85      ,  0.45      ,  0.
  ,
        0.75      ,  0.51      ,  0.9       ,  0.3       ,  0.2
  ,
        0.4       ,  0.49526316])
```

Replacing Null values and changing the data types of required columns

In [3]:

```
print(pd.isnull(req_data).any())
print(pd.isnull(req_data['Scenario']).any())
req_data = req_data[np.isfinite(data['Scenario'])]
print(pd.isnull(req_data['Scenario']).any())
req_data['damages'].fillna(0,inplace=True)
req_data['damages_word'].fillna(0,inplace=True)
req_data['mm_perc'].fillna(1,inplace=True)
req_data['perc_calc'].fillna(0,inplace=True)
#Dropping the last two rows which has null values
#data[pd.isnull(data['Path'])]
#data['Path']=data.Path.dropna(inplace= True)
#data[pd.isnull(data['Path'])]
print(pd.isnull(req_data).any())




#Changing data types of columns
req_data['End Date'] = pd.to_datetime(data['End Date'])
req_data['Start Date'] = pd.to_datetime(data['Start Date'])
req_data['Scenario']= req_data.Scenario.astype(int)
req_data['Liability']= req_data.Liability.astype(int)
req_data.dtypes

# Getting the id of the column
data.columns.get_loc("Liability")
```

```
Mesa_Negligent     True
damages            True
Liability          True
gender             True
damages_word       True
Scenario           True
Dunn_negligent     True
perc_calc          True
Start Date        False
End Date          False
mm_perc            True
Income             True
dtype: bool
True
False
Mesa_Negligent    False
damages           False
Liability         False
gender            False
damages_word      False
Scenario          False
Dunn_negligent     True
perc_calc         False
Start Date        False
End Date          False
mm_perc           False
Income            False
dtype: bool
```

Out[3]:

```
55
```

Cleaning Damages and perc_calc column

In [4]:

```
print(req_data.isnull().any())
req_data['damages'] = req_data['damages'].str.replace(',', '')
req_data['perc_calc'] = req_data['perc_calc'].str.replace('$', '')
req_data['perc_calc'] = req_data['perc_calc'].str.replace(',', '')
req_data['perc_calc'] = req_data['perc_calc'].str.replace('-', '')
req_data['perc_calc'] = req_data['perc_calc'].str.replace("  ", '')

#req_data['mm_perc'] = req_data['mm_perc'].str.replace("$", '')
#req_data['mm_perc'] = req_data['mm_perc'].str.replace(",", '')
#req_data['mm_perc'] = req_data['mm_perc'].str.replace("  ", '')
#req_data.damages=pd.to_numeric(req_data['damages'].str.replace(',', ''))
#req_data.perc_calc=pd.to_numeric(req_data.perc_calc)

req_data.damages=pd.to_numeric(req_data['damages'])
req_data.perc_calc=pd.to_numeric(req_data.perc_calc)
req_data['damages'].fillna(0,inplace=True)
req_data['mm_perc'].fillna(1,inplace=True)
req_data['perc_calc'].fillna(0,inplace=True)
#print(req_data.damages)
print(req_data.isnull().any())

print(req_data[pd.isnull(req_data['Dunn_negligent'])])
```

```
Mesa_Negligent    False
damages           False
Liability         False
gender            False
damages_word      False
Scenario          False
Dunn_negligent     True
perc_calc         False
Start Date        False
End Date          False
mm_perc           False
Income            False
dtype: bool
Mesa_Negligent    False
damages           False
Liability         False
gender            False
damages_word      False
Scenario          False
Dunn_negligent     True
perc_calc         False
Start Date        False
End Date          False
mm_perc           False
Income            False
dtype: bool
    Mesa_Negligent  damages  Liability  gender damages_word  Scenari
o  \
0               No      0.0          0  Female            0
1
1               No      0.0          0    Male            0
1
2               No      0.0          0    Male            0
1
3               No      0.0          0    Male            0
1
4               No      0.0          0    Male            0
1
5               No      0.0          0    Male            0
1
6               No      0.0          0    Male            0
1
7               No      0.0          0  Female            0
1
8               No      0.0          0    Male            0
1
9               No      0.0          0  Female            0
1
10              No      0.0          0    Male            0
1
11              No      0.0          0  Female            0
1
12              No      0.0          0    Male            0
1
13              No      0.0          0  Female            0
1
14              No      0.0          0    Male            0
1
15              No      0.0          0    Male            0
1
16              No      0.0          0    Male            0
```

```
1
17               No      0.0          0    Male            0
1
18               No      0.0          0    Male            0
1
19               No      0.0          0    Male            0
1
20               No      0.0          0    Male            0
1
21               No      0.0          0  Female            0
1
22               No      0.0          0  Female            0
1
23               No      0.0          0  Female            0
1
24               No      0.0          0    Male            0
1
25               No      0.0          0    Male            0
1
26               No      0.0          0    Male            0
1
27               No      0.0          0    Male            0
1
28               No      0.0          0  Female            0
1
29               No      0.0          0    Male            0
1
..              ...      ...        ...     ...          ...
...
397             Yes      0.0          0  Female            0
2
398             Yes      0.0          0    Male            0
2
399             Yes      0.0          0  Female            0
2
400             Yes      0.0          0  Female            0
2
401             Yes      0.0          0  Female            0
2
402             Yes      0.0          0  Female            0
2
403             Yes      0.0          0  Female            0
2
404             Yes      0.0          0  Female            0
2
405             Yes      0.0          0  Female            0
2
406             Yes      0.0          0  Female            0
2
407             Yes      0.0          0    Male            0
3
408             Yes      0.0          0  Female            0
3
409             Yes      0.0          0    Male            0
3
410             Yes      0.0          0  Female            0
4
411             Yes      0.0          0    Male            0
4
412             Yes      0.0          0    Male            0
4
```

```
413            Yes      0.0          0    Male           0
4
414            Yes      0.0          0    Male           0
4
415            Yes      0.0          0    Male           0
4
416            Yes      0.0          0    Male           0
4
417            Yes      0.0          0  Female           0
5
418            Yes      0.0          0    Male           0
5
419            Yes      0.0          0  Female           0
5
420            Yes      0.0          0  Female           0
5
421            Yes      0.0          0    Male           0
5
422            Yes      0.0          0  Female           0
5
423            Yes      0.0          0  Female           0
5
424            Yes      0.0          0  Female           0
5
425            Yes      0.0          0  Female           0
5
426            Yes      0.0          0    Male           0
5

    Dunn_negligent  perc_calc          Start Date            End Dat
e  \
0              NaN        0.0 2017-09-29 13:58:00 2017-09-29 14:16:0
0
1              NaN        0.0 2017-09-29 14:00:00 2017-09-29 14:18:0
0
2              NaN        0.0 2017-09-29 13:57:00 2017-09-29 14:19:0
0
3              NaN        0.0 2017-09-29 14:01:00 2017-09-29 14:21:0
0
4              NaN        0.0 2017-09-29 14:04:00 2017-09-29 14:23:0
0
5              NaN        0.0 2017-09-29 14:06:00 2017-09-29 14:25:0
0
6              NaN        0.0 2017-09-29 14:06:00 2017-09-29 14:26:0
0
7              NaN        0.0 2017-09-29 14:16:00 2017-09-29 14:35:0
0
8              NaN        0.0 2017-09-29 14:27:00 2017-09-29 14:49:0
0
9              NaN        0.0 2017-09-29 14:44:00 2017-09-29 15:02:0
0
10             NaN        0.0 2017-09-29 15:09:00 2017-09-29 15:27:0
0
11             NaN        0.0 2017-09-29 15:55:00 2017-09-29 16:16:0
0
12             NaN        0.0 2017-09-29 18:28:00 2017-09-29 18:51:0
0
13             NaN        0.0 2017-09-29 19:54:00 2017-09-29 20:14:0
0
14             NaN        0.0 2017-09-29 20:56:00 2017-09-29 21:16:0
0
```

```
15              NaN        0.0 2017-09-30 11:47:00 2017-09-30 12:04:0
0
16              NaN        0.0 2017-09-30 12:56:00 2017-09-30 13:13:0
0
17              NaN        0.0 2017-09-30 14:15:00 2017-09-30 14:26:0
0
18              NaN        0.0 2017-09-30 16:19:00 2017-09-30 16:38:0
0
19              NaN        0.0 2017-10-01 21:22:00 2017-10-01 21:40:0
0
20              NaN        0.0 2017-10-01 21:47:00 2017-10-01 22:07:0
0
21              NaN        0.0 2017-10-02 02:43:00 2017-10-02 03:02:0
0
22              NaN        0.0 2017-10-02 06:47:00 2017-10-02 07:11:0
0
23              NaN        0.0 2017-10-02 07:12:00 2017-10-02 07:31:0
0
24              NaN        0.0 2017-10-02 07:21:00 2017-10-02 07:38:0
0
25              NaN        0.0 2017-10-02 07:57:00 2017-10-02 08:15:0
0
26              NaN        0.0 2017-10-02 08:31:00 2017-10-02 08:50:0
0
27              NaN        0.0 2017-10-02 09:01:00 2017-10-02 09:19:0
0
28              NaN        0.0 2017-10-02 09:20:00 2017-10-02 09:43:0
0
29              NaN        0.0 2017-10-02 10:41:00 2017-10-02 10:59:0
0
..              ...        ...                 ...
...
397             NaN        0.0 2017-09-29 14:00:00 2017-09-29 14:19:0
0
398             NaN        0.0 2017-09-29 18:28:00 2017-09-29 18:48:0
0
399             NaN        0.0 2017-09-30 11:49:00 2017-09-30 12:10:0
0
400             NaN        0.0 2017-09-30 14:52:00 2017-09-30 15:12:0
0
401             NaN        0.0 2017-10-01 10:25:00 2017-10-01 10:42:0
0
402             NaN        0.0 2017-10-03 13:07:00 2017-10-03 13:26:0
0
403             NaN        0.0 2017-10-05 18:01:00 2017-10-05 18:20:0
0
404             NaN        0.0 2017-10-05 18:16:00 2017-10-05 18:35:0
0
405             NaN        0.0 2017-10-05 18:17:00 2017-10-05 18:40:0
0
406             NaN        0.0 2017-10-05 18:20:00 2017-10-05 18:49:0
0
407             NaN        0.0 2017-09-30 20:10:00 2017-09-30 20:30:0
0
408             NaN        0.0 2017-10-05 17:45:00 2017-10-05 18:12:0
0
409             NaN        0.0 2017-10-05 18:03:00 2017-10-05 18:25:0
0
410             NaN        0.0 2017-09-29 14:14:00 2017-09-29 14:44:0
0
411             NaN        0.0 2017-09-29 18:35:00 2017-09-29 18:58:0
```

```
0
412            NaN        0.0 2017-10-01 04:32:00 2017-10-01 04:54:0
0
413            NaN        0.0 2017-10-02 09:26:00 2017-10-02 09:48:0
0
414            NaN        0.0 2017-10-05 17:32:00 2017-10-05 17:52:0
0
415            NaN        0.0 2017-10-05 17:33:00 2017-10-05 17:54:0
0
416            NaN        0.0 2017-10-05 17:39:00 2017-10-05 18:02:0
0
417            NaN        0.0 2017-09-30 17:52:00 2017-09-30 18:11:0
0
418            NaN        0.0 2017-10-03 07:48:00 2017-10-03 08:08:0
0
419            NaN        0.0 2017-10-04 10:56:00 2017-10-04 11:17:0
0
420            NaN        0.0 2017-10-05 17:25:00 2017-10-05 17:47:0
0
421            NaN        0.0 2017-10-05 17:29:00 2017-10-05 17:50:0
0
422            NaN        0.0 2017-10-05 17:25:00 2017-10-05 17:52:0
0
423            NaN        0.0 2017-10-05 17:32:00 2017-10-05 17:55:0
0
424            NaN        0.0 2017-10-05 17:32:00 2017-10-05 17:57:0
0
425            NaN        0.0 2017-10-05 17:58:00 2017-10-05 18:19:0
0
426            NaN        0.0 2017-10-05 18:10:00 2017-10-05 18:30:0
0

     mm_perc                Income
0        1.0    $10,000 to $29,999
1        1.0    $50,000 to $99,999
2        1.0    $30,000 to $49,999
3        1.0    $10,000 to $29,999
4        1.0    $30,000 to $49,999
5        1.0  $100,000 to $199,999
6        1.0    $10,000 to $29,999
7        1.0    $50,000 to $99,999
8        1.0  $100,000 to $199,999
9        1.0  $100,000 to $199,999
10       1.0     Less than $10,000
11       1.0    $50,000 to $99,999
12       1.0    $50,000 to $99,999
13       1.0    $50,000 to $99,999
14       1.0    $10,000 to $29,999
15       1.0  $100,000 to $199,999
16       1.0    $50,000 to $99,999
17       1.0    $50,000 to $99,999
18       1.0    $10,000 to $29,999
19       1.0    $50,000 to $99,999
20       1.0    $10,000 to $29,999
21       1.0    $50,000 to $99,999
22       1.0    $30,000 to $49,999
23       1.0    $50,000 to $99,999
24       1.0    $50,000 to $99,999
25       1.0    $30,000 to $49,999
26       1.0    $30,000 to $49,999
27       1.0     Less than $10,000
```

```
28       1.0  $100,000 to $199,999
29       1.0    $50,000 to $99,999
..       ...                   ...
397      1.0     Less than $10,000
398      1.0    $10,000 to $29,999
399      1.0    $30,000 to $49,999
400      1.0    $30,000 to $49,999
401      1.0    $10,000 to $29,999
402      1.0  $100,000 to $199,999
403      1.0    $50,000 to $99,999
404      1.0    $50,000 to $99,999
405      1.0  $100,000 to $199,999
406      1.0    $50,000 to $99,999
407      1.0  $100,000 to $199,999
408      1.0    $10,000 to $29,999
409      1.0    $30,000 to $49,999
410      1.0  $100,000 to $199,999
411      1.0    $10,000 to $29,999
412      1.0    $30,000 to $49,999
413      1.0    $30,000 to $49,999
414      1.0      $200,000 or more
415      1.0    $50,000 to $99,999
416      1.0    $10,000 to $29,999
417      1.0    $10,000 to $29,999
418      1.0  $100,000 to $199,999
419      1.0    $10,000 to $29,999
420      1.0    $50,000 to $99,999
421      1.0    $10,000 to $29,999
422      1.0    $10,000 to $29,999
423      1.0  $100,000 to $199,999
424      1.0    $50,000 to $99,999
425      1.0    $10,000 to $29,999
426      1.0    $30,000 to $49,999

[427 rows x 12 columns]
```

In [5]:

```
req_data['Income'] = req_data['Income'].str.replace("  ", '')
req_data['Income'] = req_data['Income'].str.replace('$', '')
req_data['Income'] = req_data['Income'].str.replace(',', '')
```

In [6]:

```
req_data['Income'].unique()
```

Out[6]:

```
array(['10000 to 29999', '50000 to 99999', '30000 to 49999',
       '100000 to 199999', 'Less than 10000', '200000 or more'],
      dtype=object)
```

In [7]:

```
req_data['Income']=req_data['Income'].replace(['Less than 10000','10000 to 2999
9','50000 to 99999','30000 to 49999','100000 to 199999',
                           '200000 or more','nan'],['1','2','3','4','5','6','0'
])
```

In [8]:

```
req_data['Income'].unique()
```

Out[8]:

```
array(['2', '3', '4', '5', '1', '6'], dtype=object)
```

Calculating Discounted Damages

In [9]:

```
req_data['Discounted_damages']=req_data.damages*pd.to_numeric(req_data.mm_perc)
```

Replacing above 500k to 500k

In [10]:

```
req_data.loc[req_data.Discounted_damages>500000.0].shape
req_data.Discounted_damages=np.where(((req_data.Discounted_damages>500000.0)),50
0000,req_data.Discounted_damages)
```

In [11]:

```
req_data.loc[req_data.Discounted_damages>500000.0].shape
```

Out[11]:

```
(0, 13)
```

EDA

Calculating Discounted Damages stats for case expected value

In [12]:

```
req_data['winrate_percentage']=req_data.Liability
req_data['Discounted_damages_mean']=req_data['Discounted_damages']
req_data['Discounted_damages_median']=req_data['Discounted_damages']
req_data['Discounted_damages_sd']=req_data['Discounted_damages']

winrate_damages_expected=req_data.groupby('Scenario').aggregate(
    {'winrate_percentage': np.mean, 'Discounted_damages_mean': np.mean,'Discount
ed_damages_median':np.median,'Discounted_damages_sd':np.std})

winrate_damages_expected['winrate_percentage']=winrate_damages_expected['winrate
_percentage']*100.0
winrate_damages_expected
```

Out[12]:

	winrate_percentage	Discounted_damages_mean	Discounted_damages_me
Scenario			
1	47.263682	97124.378109	0.0
2	48.730964	98857.868020	0.0
3	68.817204	130739.247312	110000.0
4	60.294118	108101.715686	100000.0
5	57.591623	92761.780105	80000.0

Calculating Discounted Damages stats when plaintiff wins

In [21]:

```
#req_data['winrate_percentage']=np.mean(req_data.Juror_Response)
#print(req_data)
req_data['mm_perc'].fillna(1,inplace=True)

req_data['Discounted_damages_mean1']=req_data['Discounted_damages']
req_data['Discounted_damages_median1']=req_data['Discounted_damages']
req_data['Discounted_damages_sd1']=req_data['Discounted_damages']
#print(req_data.mm_perc)

winrate_damages_plaintiffwin=req_data.loc[(req_data['Mesa_Negligent']=='Yes') &
(req_data['Liability']==1)].groupby('Scenario').aggregate({'Discounted_damages_m
ean1': np.mean,'Discounted_damages_median1':np.median,'Discounted_damages_sd1':n
p.std})

#Here Liability is "Is substantial factor for causing plaintiff injuries"
winrate_damages_plaintiffwin
```

Out[21]:

	Discounted_damages_mean1	**Discounted_damages_median1**	**Discounted_**
Scenario			
1	205494.736842	180000.0	111371.80147
2	202864.583333	180000.0	108895.76085
3	189980.468750	180000.0	102566.32964
4	179290.650407	180000.0	77900.192120
5	161068.181818	150000.0	81512.690442

Plotting graph for Liability vs Path

In [14]:

```
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns

sns.factorplot(x='Scenario', y='damages', kind='box',data=req_data)
```

Out[14]:

```
<seaborn.axisgrid.FacetGrid at 0x109903b70>
```

```
In [15]:
```

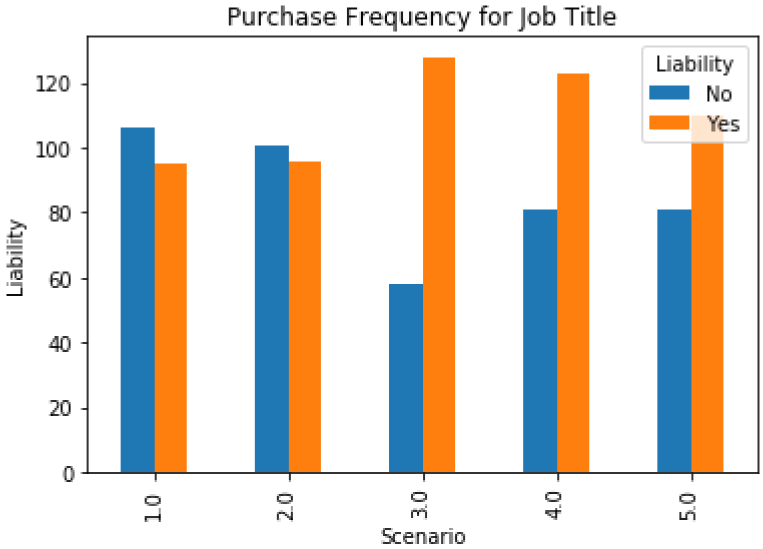
```
pd.crosstab(data.Scenario,data.Liability).plot(kind='bar')
plt.title('Purchase Frequency for Job Title')
plt.xlabel('Scenario')
plt.ylabel('Liability')
plt.savefig('Juror Response per each Scenario')



a = req_data['Scenario']
b = req_data['Liability']
pd.crosstab(a,b)
```

```
Out[15]:
```

Liability	**0**	**1**
Scenario		
1	106	95
2	101	96
3	58	128
4	81	123
5	81	110



From the graph we can see that

- Winrate increases when subsequent remedial measures introduced at scenario3.
- Winrate decreases slightly when limited jury instruction presented at Scenario4.
- Similarly winrate and damages decreases more when more explanation on limiting jury instruction provided at Scenario5.But not decreased as in Scenario2(before limiting jury instructions)

Importing data to "old_data.csv" file

In [16]:

```
req_data.to_csv("old_data.csv",sep=',')
```