VIew

November 1, 2020

```
[27]: import pandas as pd
      import numpy as np
      import seaborn as sns
      import matplotlib.pyplot as plt
      import matplotlib.patches as mpatches
      import random

      import holoviews as hv
      from holoviews import opts, dim
      import plotly.graph_objects as go
```

```
[28]: hv.extension('bokeh')
```

```
[29]: entrenamiento = pd.read_csv('Entrenamieto_ECI_2020.csv')
      entrenamiento.head()
```

```
[29]:       ID    Region   Territory  Pricing, Delivery_Terms_Quote_Appr  \
      0  27761      EMEA        None                                   1
      1  27760      EMEA        None                                   0
      2  27446  Americas  NW America                                   0
      3  16808  Americas  NW America                                   1
      4  16805  Americas  NW America                                   1

         Pricing, Delivery_Terms_Approved  Bureaucratic_Code_0_Approval  \
      0                                 1                             1
      1                                 0                             0
      2                                 0                             0
      3                                 0                             1
      4                                 0                             1

         Bureaucratic_Code_0_Approved  Submitted_for_Approval    Bureaucratic_Code  \
      0                             1                       0  Bureaucratic_Code_4
      1                             0                       0  Bureaucratic_Code_4
      2                             0                       0  Bureaucratic_Code_4
      3                             0                       0  Bureaucratic_Code_5
      4                             0                       0  Bureaucratic_Code_5
```

```
  Account_Created_Date  ... Delivery_Quarter Delivery_Year  \
0            6/16/2015  ...               Q2          2016
1            6/16/2015  ...               Q1          2016
2            4/21/2015  ...               Q1          2016
3            7/27/2013  ...               Q1          2018
4            7/27/2013  ...               Q1          2018

  Actual_Delivery_Date TRF  Total_Amount_Currency Total_Amount  \
0                  NaT  10                    EUR    5272800.0
1                  NaT   0                    EUR      48230.0
2                  NaT   0                    USD      83865.6
3                  NaT  14                    USD    7421881.5
4                  NaT  25                    USD   13357192.5

  Total_Taxable_Amount_Currency Total_Taxable_Amount        Stage  \
0                           EUR            5272800.0  Closed Lost
1                           EUR              48230.0   Closed Won
2                           USD              83865.6   Closed Won
3                           USD            7421881.5  Closed Lost
4                           USD           13357192.5  Closed Lost

        Prod_Category_A
0  Prod_Category_A_None
1  Prod_Category_A_None
2  Prod_Category_A_None
3  Prod_Category_A_None
4  Prod_Category_A_None

[5 rows x 52 columns]
```

[40]:

```
Stages = entrenamiento[['Pricing, Delivery_Terms_Approved','Stage']]
Stages = Stages.groupby(['Pricing, Delivery_Terms_Approved','Stage']).
↪agg({'Stage' : ['count']})
Stages
```

[40]:

```
                                               Stage
                                               count
Pricing, Delivery_Terms_Approved Stage
0                                Closed Lost    3907
                                 Closed Won     3129
                                 Negotiation       1
                                 Proposal          9
                                 Qualification    11
1                                Closed Lost    3443
                                 Closed Won     6404
                                 Negotiation      17
                                 Proposal         26
```

[39]:
```
aprobaciones = entrenamiento[['Pricing, Delivery_Terms_Approved','Pricing,␣
 ↪Delivery_Terms_Quote_Appr']]
aprobaciones = aprobaciones.groupby(['Pricing,␣
 ↪Delivery_Terms_Approved','Pricing, Delivery_Terms_Quote_Appr']).
 ↪agg({'Pricing, Delivery_Terms_Quote_Appr' : ['count']})
aprobaciones
```

[39]:
```
                                                                     Pricing,
Delivery_Terms_Quote_Appr
                 count
Pricing, Delivery_Terms_Approved Pricing, Delivery_Terms_Quote_Appr
0                                0
3254
                                 1
3803
1                                1
9890
```

[36]:
```
lista_region = list(set(entrenamiento['Region'].to_list()))
lista_region
```

[36]:
```
['Japan', 'Middle East', 'APAC', 'EMEA', 'Americas']
```

[38]:
```
label = ['No necesita aprobacion','Necesita aprobacion','Fue aprobada','No fue␣
 ↪Aprobada','Close Won', 'Close Lost']
label = label + lista_region
label
```

[38]:
```
['No necesita aprobacion',
 'Necesita aprobacion',
 'Fue aprobada',
 'No fue Aprobada',
 'Close Won',
 'Close Lost',
 'Japan',
 'Middle East',
 'APAC',
 'EMEA',
 'Americas']
```

[48]:
```
nodes = hv.Dataset(enumerate(label), 'index', 'label')
edges = [
    (0, 2, 9890), (1, 2, 3803), (1, 3, 3254), # la opor a si fueron aprobadas o␣
 ↪no
    (2, 4, 6404), (2, 5, 3443), (3, 4, 3129), (3, 5, 3907), # de las opor a si␣
 ↪perdieron o ganaron
```

```
    (4, 6, 1793), (4, 7, 50), (4, 8, 2471), (4, 9, 1896), (4, 10, 3323), # las␣
↪Won a region
    (5, 6, 2140), (5, 7, 120), (5, 8, 2189), (5, 9, 1355), (5, 10, 1546) #las␣
↪Lost a Region
]

value_dim = hv.Dimension('Cantidad')
careers = hv.Sankey((edges, nodes), ['Desde', 'Hasta'], vdims=value_dim)

careers.opts(
    opts.Sankey(labels='label', label_position='right', title = 'Estados de las␣
↪oportunidades por region considerando permisos',fontsize = 19,width=900,␣
↪height=500, cmap='Set1',
                edge_color=dim('Hasta').str(), node_color=dim('index').str()))
```

```
[48]: :Sankey   [Desde,Hasta]   (Cantidad)
```