RepasoParaParcial_y_PartePractica_al_final

September 19, 2017

0.1 ANOTACION:

1. Las celdas con Recomendacion, son indicaciones que pongo para que tengan en cuenta para el parcial

2. Las celdas con Repaso son puntos que deberian repasar

3. Las celdas con Ojo son puntos con los que deben tener mucho cuidado

4. POR FAVOR LEAN LOS COMENTARIOS DE LAS CELDAS ** Recomendacion **: Como deberian resolver el parcial y dividir su tiempo el tiempo en el parcial

1. Leer todo el parcial y estimar cuanto se deberian demorar en cada punto
2. Hacer exploracion rapida
3. Mirar que variables estan mas completas y limpiarlas si son necesarias
4. Organizar cada pregunta y irla respondiendo ayudada de celdas de codigo. Esto es importante para que al profesor le sea facil entender lo que se hizo
5. Impartir mas tiempo a las preguntas con mayor porcentaje del parcial

In [39]:
```
# Importando modulos
import os, sys
import numpy as np
import pandas as pd
import pandas_profiling
```

In [40]:
```
# module to do statistics
import scipy.stats  as stats

# importing modules for contingency
from scipy.stats import chi2
from scipy.stats import chi2_contingency
```

In [41]:
```
# Modulos de graficos
import matplotlib
matplotlib.style.use('ggplot')
from matplotlib import pyplot as plt
%matplotlib inline
import seaborn as sns
import bokeh
```

```
In [42]: # Funciones adicionales
         # funcion de otra persona http://stackoverflow.com/questions/26266362/how-
         def missing_values_table(df):
                 mis_val = df.isnull().sum()
                 mis_val_percent = 100 * df.isnull().sum()/len(df)
                 mis_val_table = pd.concat([mis_val, mis_val_percent], axis=1)
                 mis_val_table_ren_columns = mis_val_table.rename(
                 columns = {0 : 'Missing Values', 1 : '% of Total Values'})
                 return mis_val_table_ren_columns
```

** Recomendacion **: Hagan una jupyter notebook con sus funciones o celdas de sus propios notebooks que entiendan para hacer las cosas mas rapido y solo tener que copiar y pegar

** Recomendacion **: Para este parcial es importante el manejo del tiempo para que puedan resolver todas las preguntas

0.2 Parte 1. Importando BD y haciendo exploracion rapida

```
In [43]: # Obteniendo datos
         pathPrin= '/Users/CamilaMV/Desktop/CDAplicada/Principal.xlsx'
         # importar datos y creando dataframe
         dfPrin = pd.read_excel(pathPrin)

In [44]: # Revision rapida de las variables
         dfPrin.head()
```

Out[44]:

	Ciudad	Cliente	Localidad_Zona	Sexo	Edad	Rango_Edad	Hijos	\
0	Bogota	C_5871	Engativa	Hombre	59.0	a_Mayor_50	3.0	
1	Bogota 011	C_8093	Engativa	Hombre	52.0	a_Mayor_50	2.0	
2	Bogota	C_8387	Teusaquillo	Hombre	75.0	a_Mayor_50	4.0	
3	Bogota	C_9658	Engativa	Hombre	52.0	a_Mayor_50	1.0	
4	Bogota	C_9721	Teusaquillo	Hombre	62.0	a_Mayor_50	4.0	

	Categoria_Cliente	Activo	Paquete	Precio_Semanal_Paquete	\
0	Fiel	1	Basic	6900	
1	Nuevo	1	Basic	6900	
2	Fiel	1	Basic	6900	
3	Fiel	1	Basic	6900	
4	Fiel	1	Basic	6900	

	Cantidad_Horas_Demandadas	Costo	MedioPago	Puntos_Conexion_Cliente
0	5.2	3000.4	Presencial	2
1	3.7	2501.1	Presencial	1
2	14.3	4000.9	NaN	4
3	2.9	3500.3	Presencial	3
4	10.1	3002.7	Electronico	2

	PQR_Id	PQR_Filtro	Campaña
0	NaN	No	NO_Impacto
1	NaN	No	NO_Impacto

```
2    NaN          No  NO_Impacto
3    NaN          No  NO_Impacto
4    NaN          No  NO_Impacto
```

```
In [45]: # Revision del tamano de la base de datos
         dfPrin.shape
```

```
Out[45]: (2690, 18)
```

** Ojo **: Deteccion de tipos de variable como los importa python, pero este no es necesariamente el perfilamiento de las varaibles

```
In [46]: dfPrin.dtypes
```

```
Out[46]: Ciudad                      object
         Cliente                     object
         Localidad_Zona              object
         Sexo                        object
         Edad                       float64
         Rango_Edad                  object
         Hijos                      float64
         Categoria_Cliente           object
         Activo                       int64
         Paquete                     object
         Precio_Semanal_Paquete       int64
         Cantidad_Horas_Demandadas  float64
         Costo                      float64
         MedioPago                   object
         Puntos_Conexion_Cliente      int64
         PQR_Id                      object
         PQR_Filtro                  object
         Campaña                     object
         dtype: object
```

** Recomendacion **: Usen pandas profiling para hacer un analisis de la calidad de la BD rapido

```
In [47]: # Usando pandas profiling
         pandas_profiling.ProfileReport(dfPrin)
```

```
Out[47]: <pandas_profiling.ProfileReport at 0x1115f5cc0>
```

```
In [48]: # Mirando missing values por variable
         missing_values_table(dfPrin)
```

```
Out[48]:                          Missing Values  % of Total Values
         Ciudad                                0           0.000000
         Cliente                               0           0.000000
         Localidad_Zona                       19           0.706320
```

```
Sexo                                  750          27.881041
Edad                                  750          27.881041
Rango_Edad                            750          27.881041
Hijos                                 750          27.881041
Categoria_Cliente                       0           0.000000
Activo                                  0           0.000000
Paquete                                 0           0.000000
Precio_Semanal_Paquete                  0           0.000000
Cantidad_Horas_Demandadas               0           0.000000
Costo                                   0           0.000000
MedioPago                               3           0.111524
Puntos_Conexion_Cliente                 0           0.000000
PQR_Id                               1563          58.104089
PQR_Filtro                              0           0.000000
Campaña                               750          27.881041
```

```
In [49]: # Luego de mirar los missing values, solo explorar las variables problema
         dfPrin.Localidad_Zona.unique()
```

```
Out[49]: array(['Engativa', 'Teusaquillo', 'Suba', 'Barrios Unidos', 'Nororiental',
                'Noroccidental', 'Suroccidental', 'Suroriental',
                'Centro Occidental', 'Centro Oriental', nan], dtype=object)
```

```
In [52]: # mirando Sexo
         dfPrin.Sexo.unique()
```

```
Out[52]: array(['Hombre', 'Mujer', nan], dtype=object)
```

```
In [51]: # mirando Edad
         dfPrin.Edad.unique()
```

```
Out[51]: array([  59.,   52.,   75.,   62.,   78.,   54.,   55.,   51.,   69.,
                  65.,   42.,   58.,   38.,   39.,   35.,   49.,   34.,   31.,
                  37.,   48.,   36.,   25.,   24.,   29.,   26.,   28.,   27.,
                  22.,   23.,  220.,   71.,   80.,   74.,   53.,   68.,   56.,
                  63.,   57.,   72.,   66.,   46.,   45.,   47.,  210.,   20.,   na
```

Hay valores de 210 en la edad que se deberia cambiar

```
In [56]: # mirando Hijos
         dfPrin.Hijos.unique()
```

```
Out[56]: array([  3. ,   2. ,   4. ,   1. ,   0. ,   0.8,   0.1,  10. ,   5. ,
                  6. ,   7. ,   8. ,   nan])
```

Hay valores de 0.1 hijos

```
In [57]: dfPrin.MedioPago.unique()
```

```
Out[57]: array(['Presencial', nan, 'Electronico'], dtype=object)
```

```
In [58]: dfPrin.PQR_Id.unique()

Out[58]: array([nan, 'A30220A4', 'A30220A7', 'A302202A', 'A3022023', 'PRQA236',
                'PRQA233', 'PRQA240', 'PRQA24A', 'PRQA237', 'PRQA239', 'PRQA245',
                'PRQA252', 'PRQA256', 'PRQA255', 'PRQA263', 'PRQA267', 'A30220A5',
                'PRQA268', 'PRQA269', 'PRQA287', 'A3022022', 'A30220A6', 'PRQA232',
                'PRQA23A', 'PRQA230', 'PRQA235', 'PRQA289', 'PRQA290', 'PRQA288',
                'PRQA294', 'PRQA292', 'PRQA29A', 'PRQA293', 'PRQA295', 'A30220A0',
                'PRQA302', 'PRQA300', 'A30220A3', 'PRQA30A', 'PRQA299', 'A30220AA',
                'A3022020', 'A30220A9', 'PRQA234', 'PRQA242', 'PRQA246', 'PRQA244',
                'PRQA250', 'PRQA243', 'PRQA248', 'PRQA249', 'PRQA25A', 'PRQA253',
                'PRQA257', 'PRQA254', 'PRQA258', 'PRQA26A', 'PRQA259', 'PRQA264',
                'PRQA262', 'PRQA266', 'PRQA265', 'PRQA270', 'A30220A8', 'PRQA296',
                'PRQA297', 'PRQA298', 'A30220A2', 'PRQA238', 'PRQA247', 'PRQA260',
                'A0A4A7A8A75', 'A0A4A3A0A67', 'A0A40698A55', 'A0A40086A43',
                'A0A4A5A4A7A', 'A0A40290A47', 'A0A39474A3A', 'A0A38250A07',
                'A0A38454AAA', 'A0A38046A03', 'A0A38862AA9', 'A0A3743409A',
                'A0A37026083', 'A0A366A8075', 'A0A37842099', 'A0A36822079',
                'A0A36006063', 'A0A35802059', 'A0A34578035', 'A0A35A90047',
                'A0A3437403A', 'A0A335580A5', 'A0A34986043', 'A0A337620A9',
                'A0A333540AA', 'A0A34782039', 'A0A34A70027', 'A0A32537995',
                'A0A33966023', 'A0A32946003', 'A0A3274A999', 'A0A3233399A',
                'A0A33A50007', 'A0A32A29987', 'A0A3A3A397A', 'A0A3A925983',
                'A0A30905963', 'A0A30497955', 'A0A3070A959', 'A0A3AA09967',
                'A0A2968A939', 'A0A3029395A', 'A0A29477935', 'A0A28865923',
                'A0A282539AA', 'A0A284579A5', 'A0A2866A9A9', 'A0A28049907',
                'A0A2764A899', 'A0A27845903', 'A0A26009867', 'A0A27029887',
                'A0A2560A859', 'A0A24989847', 'A0A25397855', 'A0A262A387A',
                'A0A25A9385A', 'A0A2662A879', 'A0A264A7875', 'A0A25805863',
                'A0A24785843', 'A0A24A7383A', 'A0A2458A839', 'A0A24377835',
                'A0A23969827', 'A0A2356A8A9', 'A0A22745803', 'A0A23765823',
                'A0A23A538AA', 'A0A233578A5', 'A0A2A929787', 'A0A2A725783',
                'A0A22949807', 'A0A2254A799', 'A0A22A3379A', 'A0A22337795',
                'A0A20909767', 'A0A2A52A779', 'A0A2A3A7775', 'A0A20705763',
                'A0AA9889747', 'A0A2AAA377A', 'A0A20297755', 'A0A2009375A',
                'A0A2050A759', 'A0AA9277735', 'A0AA82577A5', 'A0AA846A7A9',
                'A0AA80537AA', 'A0AA703369A', 'A0AA744A699', 'A0AA7237695',
                'A0AA6625683', 'A0AA6829687', 'A0AA540A659', 'A0AA5605663',
                'A0AA499365A', 'A0AA4789647', 'A0AA5A97655', 'A0AA3769627',
                'A0AA336A6A9', 'A0AA397363A', 'A0AA2749607', 'A0AA3565623',
                'A0AA234A599', 'A0AA2545603', 'A0AA2A37595', 'A0AA09A357A',
                'A0AAA729587', 'A0AAA32A579', 'A0AA0709567', 'A0AAA525583',
                'A0AAA93359A', 'A0AA0505563', 'A0A0989355A', 'A0A09485543',
                'A0A0887353A', 'A0A09077535', 'A0A09689547', 'A0A0826A5A9',
                'A0A08465523', 'A0A078535AA', 'A0A08669527', 'A0A080575A5',
                'A0A0724A499', 'A0A07037495', 'A0A07445503', 'A0A0683349A',
                'A0A06425483', 'A0A07649507', 'A0A0622A479', 'A0A05405463',
                'A0A05609467', 'A0A04589447', 'A0A04997455', 'A0A0520A459',
```

```
'A0A03569427', 'A0A0479345A', 'A0A04385443', 'A0A03365423',
'A0A03977435', 'A0A0377343A', 'A0A04A8A439', 'A0A03A6A4A9',
'A0A0A937395', 'A0A02549407', 'A0A0A529387', 'A0A0AA2A379',
'A0A0A325383', 'A0A02345403', 'A0A0A73339A', 'A0A00305363',
'A0A00509367', 'A0A009A7375', 'A0099897355', 'A0098877335',
'A009867333A', 'A0096429287', 'A0097245303', 'A009704A299',
'A0096837295', 'A009602A279', 'A00956A327A', 'A0095205263',
'A009398A239', 'A009500A259', 'A009459325A', 'A0093369227',
'A0093777235', 'A009296A2A9', 'A00927572A5', 'A0092A45203',
'A009AA25A83', 'A008990AA59', 'A009092AA79', 'A0089289A47',
'A0089493A5A', 'A0089085A43', 'A0087249A07', 'A0087657AA5',
'A0086229087', 'A0087453AAA', 'A008643309A', 'A0086025083',
'A0084A89047', 'A0084597055', 'A008480A059', 'A008439305A',
'A0083577035', 'A008337303A', 'A0A40902A59', 'A0A4AA06A63',
'A0A40494A5A', 'A0A39270A27', 'A0A39882A39', 'A0A39066A23',
'A0A39678A35', 'A0A38658AA5', 'A0A37638095', 'A0A362A0067',
'A0A35598055', 'A0A3A5A7975', 'A0A29885943', 'A0A2723389A',
'A0A26825883', 'A0AA9685743', 'A0AA8869727', 'A0AA907373A',
'A0AA948A739', 'A0AA7645703', 'A0AA5809667', 'A0AA60A367A',
'A0AA3A576A5', 'A0AA4585643', 'A0AA29536AA', 'A0AAAAA7575',
'A0A0928A539', 'A0A007A337A', 'A009908A339', 'A009806A3A9',
'A0099285343', 'A00978573A5', 'A0096225283', 'A0094797255',
'A0094389247', 'A0094A85243', 'A00925532AA', 'A009357323A',
'A009A329A87', 'A00905A3A7A', 'A0090309A67', 'A008786AAA9',
'A0086637095', 'A00854A307A', 'A0085209067', 'A0085005063',
'A0083985043', 'A008378A039', 'A0083A69027', 'XLWA50A00',
'ASDA0A0A0A', 'DASDASA0A0A0', 'PQRA23', 'PARA24A234', 'PAQRA23A',
'PQR32A2', 'PQR3423', 'PQR342', 'KPQR3A4AA'], dtype=object)
```

In [59]:
```
dfPrin.Campaña.unique()
```

Out[59]:
```
array(['NO_Impacto', 'SI_Impacto', nan], dtype=object)
```

0.3 Recomendacion 0: Eliminando missing values, pero primero se debe saber donde estan y cuantos son.

0.4 Igualmente, hacer anotaciones en las celdas de su codigo que luego pueden usar para al final poner la respuesta del punto

In [62]:
```
# Generen otro dataframe que tenga un nombre facil de identificar para ust
dataNullDf = dfPrin[pd.isnull(dfPrin).any(axis=1)]
```

In [63]:
```
dataNullDf.shape
```

Out[63]:
```
(1759, 18)
```

In [64]:
```
# Filas que tienen mas de 4 Nan en la misma fila
dfErase_750 = dfPrin.ix[np.argwhere(dfPrin.isnull().sum(1)>=4).ravel()[0]:
```

In [65]:
```
dfErase_750
```

Out[65]:

	Ciudad	Cliente	Localidad_Zona	Sexo	Edad	Rango_Edad	Hijos
1940	Bogota	C_EWX1	Teusaquillo	NaN	NaN	NaN	NaN
1941	Bogota	C_EWX10	Engativa	NaN	NaN	NaN	NaN
1942	Bogota	C_EWX100	Engativa	NaN	NaN	NaN	NaN
1943	Bogota	C_EWX101	Engativa	NaN	NaN	NaN	NaN
1944	Bogota	C_EWX102	Teusaquillo	NaN	NaN	NaN	NaN
1945	Bogota	C_EWX103	Engativa	NaN	NaN	NaN	NaN
1946	Bogota	C_EWX104	Teusaquillo	NaN	NaN	NaN	NaN
1947	Bogota	C_EWX105	Engativa	NaN	NaN	NaN	NaN
1948	Bogota	C_EWX106	Teusaquillo	NaN	NaN	NaN	NaN
1949	Bogota	C_EWX107	Suba	NaN	NaN	NaN	NaN
1950	Bogota	C_EWX108	Suba	NaN	NaN	NaN	NaN
1951	Bogota	C_EWX109	Teusaquillo	NaN	NaN	NaN	NaN
1952	Bogota	C_EWX11	Engativa	NaN	NaN	NaN	NaN
1953	Bogota	C_EWX110	Teusaquillo	NaN	NaN	NaN	NaN
1954	Bogota	C_EWX111	NaN	NaN	NaN	NaN	NaN
1955	Bogota	C_EWX112	Suba	NaN	NaN	NaN	NaN
1956	Bogota	C_EWX113	Barrios Unidos	NaN	NaN	NaN	NaN
1957	Bogota	C_EWX114	Barrios Unidos	NaN	NaN	NaN	NaN
1958	Bogota	C_EWX115	Barrios Unidos	NaN	NaN	NaN	NaN
1959	Bogota	C_EWX116	Suba	NaN	NaN	NaN	NaN
1960	Bogota	C_EWX117	Teusaquillo	NaN	NaN	NaN	NaN
1961	Bogota	C_EWX118	Engativa	NaN	NaN	NaN	NaN
1962	Bogota	C_EWX119	Engativa	NaN	NaN	NaN	NaN
1963	Bogota	C_EWX12	Engativa	NaN	NaN	NaN	NaN
1964	Bogota	C_EWX120	Teusaquillo	NaN	NaN	NaN	NaN
1965	Bogota	C_EWX121	Engativa	NaN	NaN	NaN	NaN
1966	Bogota	C_EWX122	Teusaquillo	NaN	NaN	NaN	NaN
1967	Bogota	C_EWX123	Engativa	NaN	NaN	NaN	NaN
1968	Bogota	C_EWX124	Teusaquillo	NaN	NaN	NaN	NaN
1969	Bogota	C_EWX125	Suba	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...
2660	Medellin	C_MXE72	Suroriental	NaN	NaN	NaN	NaN
2661	Medellin	C_MXE73	Centro Occidental	NaN	NaN	NaN	NaN
2662	Medellin	C_MXE74	Centro Oriental	NaN	NaN	NaN	NaN
2663	Medellin	C_MXE75	Noroccidental	NaN	NaN	NaN	NaN
2664	Medellin	C_MXE76	Centro Oriental	NaN	NaN	NaN	NaN
2665	Medellin	C_MXE77	Nororiental	NaN	NaN	NaN	NaN
2666	Medellin	C_MXE78	Nororiental	NaN	NaN	NaN	NaN
2667	Medellin	C_MXE79	Noroccidental	NaN	NaN	NaN	NaN
2668	Medellin	C_MXE8	Suroccidental	NaN	NaN	NaN	NaN
2669	Medellin	C_MXE80	Suroccidental	NaN	NaN	NaN	NaN
2670	Medellin	C_MXE81	Suroriental	NaN	NaN	NaN	NaN
2671	Medellin	C_MXE82	Centro Occidental	NaN	NaN	NaN	NaN
2672	Medellin	C_MXE83	Centro Oriental	NaN	NaN	NaN	NaN
2673	Medellin	C_MXE84	Noroccidental	NaN	NaN	NaN	NaN
2674	Medellin	C_MXE85	Centro Oriental	NaN	NaN	NaN	NaN
2675	Medellin	C_MXE86	Nororiental	NaN	NaN	NaN	NaN

2676	Medellin	C_MXE87	Nororiental	NaN	NaN	NaN	NaN
2677	Medellin	C_MXE88	Noroccidental	NaN	NaN	NaN	NaN
2678	Medellin	C_MXE89	Suroccidental	NaN	NaN	NaN	NaN
2679	Medellin	C_MXE9	Suroccidental	NaN	NaN	NaN	NaN
2680	Medellin	C_MXE90	Suroriental	NaN	NaN	NaN	NaN
2681	Medellin	C_MXE91	Centro Occidental	NaN	NaN	NaN	NaN
2682	Medellin	C_MXE92	Centro Oriental	NaN	NaN	NaN	NaN
2683	Medellin	C_MXE93	Noroccidental	NaN	NaN	NaN	NaN
2684	Medellin	C_MXE94	Centro Oriental	NaN	NaN	NaN	NaN
2685	Medellin	C_MXE95	Nororiental	NaN	NaN	NaN	NaN
2686	Medellin	C_MXE96	Nororiental	NaN	NaN	NaN	NaN
2687	Medellin	C_MXE97	Noroccidental	NaN	NaN	NaN	NaN
2688	Medellin	C_MXE98	Noroccidental	NaN	NaN	NaN	NaN
2689	Medellin	C_MXE99	Noroccidental	NaN	NaN	NaN	NaN

	Categoria_Cliente	Activo	Paquete	Precio_Semanal_Paquete	\
1940	Fiel	1	Premium	27000	
1941	Flotante	1	Premium	27000	
1942	Nuevo	1	Premium	27000	
1943	Fiel	1	Premium	27000	
1944	Fiel	1	Premium	27000	
1945	Flotante	1	Premium	27000	
1946	Flotante	1	Premium	27000	
1947	Fiel	1	Premium	27000	
1948	Fiel	1	Premium	27000	
1949	Fiel	1	Premium	27000	
1950	Nuevo	1	Premium	27000	
1951	Nuevo	1	Premium	27000	
1952	Fiel	1	Basic	17200	
1953	Flotante	1	Premium	27000	
1954	Fiel	1	Premium	27000	
1955	Fiel	1	Basic	17200	
1956	Flotante	1	Basic	17200	
1957	Flotante	1	Basic	17200	
1958	Nuevo	1	Premium	27000	
1959	Nuevo	1	Premium	27000	
1960	Flotante	1	Premium	27000	
1961	Fiel	1	Premium	27000	
1962	Fiel	1	Basic	17200	
1963	Fiel	1	Basic	17200	
1964	Nuevo	1	Basic	17200	
1965	Fiel	1	Premium	27000	
1966	Flotante	1	Premium	27000	
1967	Flotante	1	Basic	17200	
1968	Flotante	1	Basic	17200	
1969	Nuevo	1	Premium	27000	
...	...	...	...	...	
2660	Nuevo	1	Premium	27000	

2661	Nuevo	1	Premium	27000
2662	Flotante	1	Basic	17200
2663	Fiel	1	Basic	17200
2664	Fiel	1	Premium	27000
2665	Flotante	1	Basic	17200
2666	Flotante	1	Basic	17200
2667	Nuevo	1	Basic	17200
2668	Fiel	1	Basic	17200
2669	Nuevo	1	Premium	27000
2670	Flotante	1	Premium	27000
2671	Fiel	1	Basic	17200
2672	Fiel	1	Premium	27000
2673	Nuevo	1	Premium	27000
2674	Fiel	1	Basic	17200
2675	Flotante	1	Basic	17200
2676	Flotante	1	Premium	27000
2677	Flotante	1	Basic	17200
2678	Nuevo	1	Basic	17200
2679	Fiel	1	Basic	17200
2680	Fiel	1	Basic	17200
2681	Fiel	1	Premium	27000
2682	Flotante	1	Premium	27000
2683	Flotante	1	Basic	17200
2684	Fiel	1	Premium	27000
2685	Fiel	1	Premium	27000
2686	Fiel	1	Basic	17200
2687	Nuevo	1	Basic	17200
2688	Nuevo	1	Premium	27000
2689	Flotante	1	Premium	27000

	Cantidad_Horas_Demandadas	Costo	MedioPago	Puntos_Conexion_Clie
1940	5.0	7775.0	Electronico	
1941	6.0	8060.0	Electronico	
1942	5.0	7775.0	Electronico	
1943	6.0	8150.0	Electronico	
1944	4.0	7760.0	Electronico	
1945	5.0	8000.0	Electronico	
1946	6.0	7790.0	Electronico	
1947	4.0	8120.0	Presencial	
1948	5.0	7775.0	Electronico	
1949	6.0	8420.0	Presencial	
1950	4.0	8180.0	Electronico	
1951	5.0	8075.0	Electronico	
1952	5.0	7170.0	Presencial	
1953	6.0	7790.0	Electronico	
1954	4.0	9080.0	Electronico	
1955	8.0	7200.0	Electronico	
1956	5.0	7245.0	Electronico	

1957	4.0	7200.0	Presencial
1958	5.0	7775.0	Presencial
1959	6.0	7880.0	Electronico
1960	7.0	7805.0	Electronico
1961	4.0	8120.0	Electronico
1962	5.0	7095.0	Electronico
1963	6.0	7350.0	Electronico
1964	6.0	7080.0	Electronico
1965	7.0	7805.0	Electronico
1966	8.0	8300.0	Presencial
1967	4.0	7080.0	Electronico
1968	5.0	7095.0	Presencial
1969	6.0	7790.0	Electronico
...	...	...	...
2660	18.0	8240.0	Presencial
2661	6.0	7970.0	Presencial
2662	5.0	7170.0	Electronico
2663	6.0	7530.0	Electronico
2664	4.0	7820.0	Electronico
2665	6.0	7170.0	Electronico
2666	5.0	7320.0	Electronico
2667	6.0	6990.0	Electronico
2668	4.0	7020.0	Presencial
2669	4.0	8060.0	Presencial
2670	5.0	7850.0	Electronico
2671	6.0	7260.0	Presencial
2672	4.0	8000.0	Electronico
2673	5.0	7850.0	Electronico
2674	6.0	7530.0	Electronico
2675	4.0	7020.0	Electronico
2676	5.0	7775.0	Presencial
2677	6.0	7440.0	Presencial
2678	4.0	7140.0	Presencial
2679	5.0	7320.0	Electronico
2680	5.0	7020.0	Electronico
2681	6.0	7880.0	Electronico
2682	4.0	7760.0	Presencial
2683	8.0	7320.0	Presencial
2684	5.0	7850.0	Electronico
2685	4.0	7820.0	Electronico
2686	5.0	7170.0	Electronico
2687	6.0	7530.0	Electronico
2688	7.0	7805.0	Electronico
2689	4.0	7820.0	Electronico

	PQR_Id	PQR_Filtro	Campaña
1940	NaN	No	NaN
1941	NaN	No	NaN

1942	NaN	No	NaN
1943	NaN	No	NaN
1944	NaN	No	NaN
1945	NaN	No	NaN
1946	NaN	No	NaN
1947	NaN	No	NaN
1948	NaN	No	NaN
1949	NaN	No	NaN
1950	NaN	No	NaN
1951	NaN	No	NaN
1952	NaN	No	NaN
1953	NaN	No	NaN
1954	NaN	No	NaN
1955	NaN	No	NaN
1956	NaN	No	NaN
1957	NaN	No	NaN
1958	NaN	No	NaN
1959	NaN	No	NaN
1960	NaN	No	NaN
1961	NaN	No	NaN
1962	NaN	No	NaN
1963	NaN	No	NaN
1964	NaN	No	NaN
1965	NaN	No	NaN
1966	NaN	No	NaN
1967	NaN	No	NaN
1968	NaN	No	NaN
1969	NaN	No	NaN
...	...	...	...
2660	A0A28865923	Si	NaN
2661	A0A28865923	Si	NaN
2662	A0A28865923	Si	NaN
2663	A0A28865923	Si	NaN
2664	A0A28865923	Si	NaN
2665	A0A28865923	Si	NaN
2666	A0A28865923	Si	NaN
2667	A0A28865923	Si	NaN
2668	A0A28865923	Si	NaN
2669	A0A28865923	Si	NaN
2670	A0A28865923	Si	NaN
2671	A0A28865923	Si	NaN
2672	A0A28865923	Si	NaN
2673	A0A28865923	Si	NaN
2674	A0A28865923	Si	NaN
2675	A0A28865923	Si	NaN
2676	A0A28865923	Si	NaN
2677	A0A28865923	Si	NaN
2678	A0A28865923	Si	NaN

```
2679  A0A28865923         Si     NaN
2680  A0A28865923         Si     NaN
2681  A0A28865923         Si     NaN
2682  A0A28865923         Si     NaN
2683  A0A28865923         Si     NaN
2684  A0A28865923         Si     NaN
2685  A0A28865923         Si     NaN
2686  A0A28865923         Si     NaN
2687  A0A28865923         Si     NaN
2688  A0A28865923         Si     NaN
2689          NaN         No     NaN

[750 rows x 18 columns]
```

In [66]:
```
# Graficando variables brutas
dfErase_750.index.tolist()
```

Out[66]:
```
[1940,
 1941,
 1942,
 1943,
 1944,
 1945,
 1946,
 1947,
 1948,
 1949,
 1950,
 1951,
 1952,
 1953,
 1954,
 1955,
 1956,
 1957,
 1958,
 1959,
 1960,
 1961,
 1962,
 1963,
 1964,
 1965,
 1966,
 1967,
 1968,
 1969,
 1970,
```

1971,
1972,
1973,
1974,
1975,
1976,
1977,
1978,
1979,
1980,
1981,
1982,
1983,
1984,
1985,
1986,
1987,
1988,
1989,
1990,
1991,
1992,
1993,
1994,
1995,
1996,
1997,
1998,
1999,
2000,
2001,
2002,
2003,
2004,
2005,
2006,
2007,
2008,
2009,
2010,
2011,
2012,
2013,
2014,
2015,
2016,
2017,
2018,

2019,
2020,
2021,
2022,
2023,
2024,
2025,
2026,
2027,
2028,
2029,
2030,
2031,
2032,
2033,
2034,
2035,
2036,
2037,
2038,
2039,
2040,
2041,
2042,
2043,
2044,
2045,
2046,
2047,
2048,
2049,
2050,
2051,
2052,
2053,
2054,
2055,
2056,
2057,
2058,
2059,
2060,
2061,
2062,
2063,
2064,
2065,
2066,

2067,
2068,
2069,
2070,
2071,
2072,
2073,
2074,
2075,
2076,
2077,
2078,
2079,
2080,
2081,
2082,
2083,
2084,
2085,
2086,
2087,
2088,
2089,
2090,
2091,
2092,
2093,
2094,
2095,
2096,
2097,
2098,
2099,
2100,
2101,
2102,
2103,
2104,
2105,
2106,
2107,
2108,
2109,
2110,
2111,
2112,
2113,
2114,

2115,
2116,
2117,
2118,
2119,
2120,
2121,
2122,
2123,
2124,
2125,
2126,
2127,
2128,
2129,
2130,
2131,
2132,
2133,
2134,
2135,
2136,
2137,
2138,
2139,
2140,
2141,
2142,
2143,
2144,
2145,
2146,
2147,
2148,
2149,
2150,
2151,
2152,
2153,
2154,
2155,
2156,
2157,
2158,
2159,
2160,
2161,
2162,

2163,
2164,
2165,
2166,
2167,
2168,
2169,
2170,
2171,
2172,
2173,
2174,
2175,
2176,
2177,
2178,
2179,
2180,
2181,
2182,
2183,
2184,
2185,
2186,
2187,
2188,
2189,
2190,
2191,
2192,
2193,
2194,
2195,
2196,
2197,
2198,
2199,
2200,
2201,
2202,
2203,
2204,
2205,
2206,
2207,
2208,
2209,
2210,

2211,
2212,
2213,
2214,
2215,
2216,
2217,
2218,
2219,
2220,
2221,
2222,
2223,
2224,
2225,
2226,
2227,
2228,
2229,
2230,
2231,
2232,
2233,
2234,
2235,
2236,
2237,
2238,
2239,
2240,
2241,
2242,
2243,
2244,
2245,
2246,
2247,
2248,
2249,
2250,
2251,
2252,
2253,
2254,
2255,
2256,
2257,
2258,

2259,
2260,
2261,
2262,
2263,
2264,
2265,
2266,
2267,
2268,
2269,
2270,
2271,
2272,
2273,
2274,
2275,
2276,
2277,
2278,
2279,
2280,
2281,
2282,
2283,
2284,
2285,
2286,
2287,
2288,
2289,
2290,
2291,
2292,
2293,
2294,
2295,
2296,
2297,
2298,
2299,
2300,
2301,
2302,
2303,
2304,
2305,
2306,

2307,
2308,
2309,
2310,
2311,
2312,
2313,
2314,
2315,
2316,
2317,
2318,
2319,
2320,
2321,
2322,
2323,
2324,
2325,
2326,
2327,
2328,
2329,
2330,
2331,
2332,
2333,
2334,
2335,
2336,
2337,
2338,
2339,
2340,
2341,
2342,
2343,
2344,
2345,
2346,
2347,
2348,
2349,
2350,
2351,
2352,
2353,
2354,

2355,
2356,
2357,
2358,
2359,
2360,
2361,
2362,
2363,
2364,
2365,
2366,
2367,
2368,
2369,
2370,
2371,
2372,
2373,
2374,
2375,
2376,
2377,
2378,
2379,
2380,
2381,
2382,
2383,
2384,
2385,
2386,
2387,
2388,
2389,
2390,
2391,
2392,
2393,
2394,
2395,
2396,
2397,
2398,
2399,
2400,
2401,
2402,

2403,
2404,
2405,
2406,
2407,
2408,
2409,
2410,
2411,
2412,
2413,
2414,
2415,
2416,
2417,
2418,
2419,
2420,
2421,
2422,
2423,
2424,
2425,
2426,
2427,
2428,
2429,
2430,
2431,
2432,
2433,
2434,
2435,
2436,
2437,
2438,
2439,
2440,
2441,
2442,
2443,
2444,
2445,
2446,
2447,
2448,
2449,
2450,

2451,
2452,
2453,
2454,
2455,
2456,
2457,
2458,
2459,
2460,
2461,
2462,
2463,
2464,
2465,
2466,
2467,
2468,
2469,
2470,
2471,
2472,
2473,
2474,
2475,
2476,
2477,
2478,
2479,
2480,
2481,
2482,
2483,
2484,
2485,
2486,
2487,
2488,
2489,
2490,
2491,
2492,
2493,
2494,
2495,
2496,
2497,
2498,

```
2499,
2500,
2501,
2502,
2503,
2504,
2505,
2506,
2507,
2508,
2509,
2510,
2511,
2512,
2513,
2514,
2515,
2516,
2517,
2518,
2519,
2520,
2521,
2522,
2523,
2524,
2525,
2526,
2527,
2528,
2529,
2530,
2531,
2532,
2533,
2534,
2535,
2536,
2537,
2538,
2539,
2540,
2541,
2542,
2543,
2544,
2545,
2546,
```

2547,
2548,
2549,
2550,
2551,
2552,
2553,
2554,
2555,
2556,
2557,
2558,
2559,
2560,
2561,
2562,
2563,
2564,
2565,
2566,
2567,
2568,
2569,
2570,
2571,
2572,
2573,
2574,
2575,
2576,
2577,
2578,
2579,
2580,
2581,
2582,
2583,
2584,
2585,
2586,
2587,
2588,
2589,
2590,
2591,
2592,
2593,
2594,

2595,
2596,
2597,
2598,
2599,
2600,
2601,
2602,
2603,
2604,
2605,
2606,
2607,
2608,
2609,
2610,
2611,
2612,
2613,
2614,
2615,
2616,
2617,
2618,
2619,
2620,
2621,
2622,
2623,
2624,
2625,
2626,
2627,
2628,
2629,
2630,
2631,
2632,
2633,
2634,
2635,
2636,
2637,
2638,
2639,
2640,
2641,
2642,

2643,
2644,
2645,
2646,
2647,
2648,
2649,
2650,
2651,
2652,
2653,
2654,
2655,
2656,
2657,
2658,
2659,
2660,
2661,
2662,
2663,
2664,
2665,
2666,
2667,
2668,
2669,
2670,
2671,
2672,
2673,
2674,
2675,
2676,
2677,
2678,
2679,
2680,
2681,
2682,
2683,
2684,
2685,
2686,
2687,
2688,
2689]

0.5 Parte 2: Limpiando de Nan, Droping nan en mas de 5 columnas

In [69]:
```
# Uso otro dataframe por si necesito el original
dfPrinV2 = dfPrin.drop(dfPrin.index[dfErase_750.index.tolist()])
```

In [70]:
```
dfPrinV2.head()
```

Out[70]:

	Ciudad	Cliente	Localidad_Zona	Sexo	Edad	Rango_Edad	Hijos	Categoria_Cliente	Activo	Paquete	Precio_Semanal_Paquete	Cantidad_Horas_Demandadas	Costo	MedioPago	Puntos_Conexion_Cliente	PQR_Id	PQR_Filtro	Campaña
0	Bogota	C_5871	Engativa	Hombre	59.0	a_Mayor_50	3.0	Fiel	1	Basic	6900	5.2	3000.4	Presencial	2	NaN	No	NO_Impacto
1	Bogota 011	C_8093	Engativa	Hombre	52.0	a_Mayor_50	2.0	Nuevo	1	Basic	6900	3.7	2501.1	Presencial	1	NaN	No	NO_Impacto
2	Bogota	C_8387	Teusaquillo	Hombre	75.0	a_Mayor_50	4.0	Fiel	1	Basic	6900	14.3	4000.9	NaN	4	NaN	No	NO_Impacto
3	Bogota	C_9658	Engativa	Hombre	52.0	a_Mayor_50	1.0	Fiel	1	Basic	6900	2.9	3500.3	Presencial	3	NaN	No	NO_Impacto
4	Bogota	C_9721	Teusaquillo	Hombre	62.0	a_Mayor_50	4.0	Fiel	1	Basic	6900	10.1	3002.7	Electronico	2	NaN	No	NO_Impacto

In [71]:
```
# Verificacion del missing
missing_values_table(dfPrinV2)
```

Out[71]:

	Missing Values	% of Total Values
Ciudad	0	0.000000
Cliente	0	0.000000
Localidad_Zona	0	0.000000
Sexo	0	0.000000
Edad	0	0.000000
Rango_Edad	0	0.000000
Hijos	0	0.000000
Categoria_Cliente	0	0.000000
Activo	0	0.000000
Paquete	0	0.000000

```
Precio_Semanal_Paquete                  0           0.000000
Cantidad_Horas_Demandadas               0           0.000000
Costo                                   0           0.000000
MedioPago                               3           0.154639
Puntos_Conexion_Cliente                 0           0.000000
PQR_Id                               1008          51.958763
PQR_Filtro                              0           0.000000
Campaña                                 0           0.000000
```

```
In [72]: # Variables con problemas son PQR_Id, que no se va a usar para hipotesis
         # Hay 3 nan en medio de pago
```

```
In [73]: dfPrinV2.iloc[[2,65,99]]
         # se tomo la decision que si hay la edad es mayor a 50, el medio de pago e
```

Out[73]:

	Ciudad	Cliente	Localidad_Zona	Sexo	Edad	Rango_Edad	Hijos	\
2	Bogota	C_8387	Teusaquillo	Hombre	75.0	a_Mayor_50	4.0	
65	Bogota	C_72055	Engativa	Mujer	51.0	a_Mayor_50	2.0	
99	Bogota	C_106302	Engativa	Mujer	68.0	a_Mayor_50	3.0	

	Categoria_Cliente	Activo	Paquete	Precio_Semanal_Paquete	\
2	Fiel	1	Basic	6900	
65	Flotante	1	Basic	6900	
99	Fiel	1	Basic	6900	

	Cantidad_Horas_Demandadas	Costo	MedioPago	Puntos_Conexion_Cliente
2	14.3	4000.9	NaN	4
65	2.0	2500.1	NaN	1
99	11.6	4501.1	NaN	5

	PQR_Id	PQR_Filtro	Campaña
2	NaN	No	NO_Impacto
65	PRQA23A	Si	NO_Impacto
99	NaN	No	NO_Impacto

```
In [74]: # Cambio de las filas de esta variable
         dfPrinV2.loc[2,'MedioPago'] = 'Presencial'
         dfPrinV2.loc[65,'MedioPago'] = 'Presencial'
         dfPrinV2.loc[99,'MedioPago'] = 'Presencial'
```

```
In [75]: #### Cambio de la variable edad
```

```
In [77]: # Se pidio que fuera mayor a 80 porque arriba se vio que habian personas c
         dfPrinV2[dfPrinV2.Edad > 80]
```

Out[77]:

	Ciudad	Cliente	Localidad_Zona	Sexo	Edad	Rango_Edad	Hijos
56	Bogota	C_61832	Barrios Unidos	Hombre	220.0	a_Mayor_50	1.0
613	Bogota	C_619199	Engativa	Hombre	210.0	a_Mayor_50	0.0
1608	Bogota	C_1614188	Teusaquillo	Hombre	220.0	a_Mayor_50	0.0

```
1718  Bogota  C_1726982            Suba  Hombre  220.0  a_Mayor_50    1.0

     Categoria_Cliente  Activo  Paquete  Precio_Semanal_Paquete  \
56               Nuevo       1    Basic                    6900
613              Nuevo       1  Modular                    4418
1608              Fiel       1  Premium                   12500
1718              Fiel       1  Premium                   12500

      Cantidad_Horas_Demandadas   Costo    MedioPago  Puntos_Conexion_Clie
56                          3.9  2500.3   Presencial
613                        18.1  2591.9  Electronico
1608                        3.6  6295.9  Electronico
1718                        2.9  6045.7   Presencial

           PQR_Id PQR_Filtro     Campaña
56         PRQA269         Si  NO_Impacto
613   A0087657AA5         Si  NO_Impacto
1608          NaN         No  NO_Impacto
1718          NaN         No  NO_Impacto
```

```
In [78]: dfPrinV2.loc[56,'Edad'] = 22.0
         dfPrinV2.loc[613,'Edad'] = 21.0
         dfPrinV2.loc[1608,'Edad'] = 22.0
         dfPrinV2.loc[1718,'Edad'] = 22.0
```

0.6 Recomendacion 1: Hagan una tabla donde ponen el tipo de variable de la base de datos, asi es mas facil saber cual usar para las pruebas estadisticas y no escriben tanto

0.7 Mirando el tipo de variables

Variable	Tipo de Variable
Ciudad	Categorica nominal
Cliente	Categorica con unico id
Localidad_Zona	Categorica nominal
Sexo	Categorica nominal (M o F) , tiene un 27.88% de missing
Edad	Numerica
Rango_Edad	Categorica ordinal, tiene missing, se debe organizar

Variable	Tipo de Variable
Hijos	Numerica discreta, es el numero de hijos del cliente
Categoria_Cliente	Categorica ordinal, aunque se puede ver que cliente paga mas
Activo	Categorico numerico, solo tiene valores de 1. Todos estan activos
Paquete	Categorica nominal
Precio_Semanal_Paquete	Numerica discreta
Cantidad_Horas_Demands	Numerica continua
Costo	Numerica continua
MedioPago	Categorica nominal
Puntos_Conexion_Cliente	Numerica discreta
PQR_Id	Categorica nominal
PQR_Filtro	Categorica nominal
Campaña	Categorica nominal

0.8 Recomendacion 2: Escriban la pregunta del parcial y al final del punto escriben una celda que diga respuesta para que le sea facil al profesor calificar y entender lo que hicieron

0.9 Parte 3: Analisis con pruebas estadisticos y graficos

0.9.1 P1. RESULTADOS EXPLORACION, INDIQUEN QUE DATOS CON PROBLEMAS y QUE DATOS RAROS

Los datos tienen 18 variables y 2690 observaciones
Problemas:

1. Hay 1759 filas que tienen al menos un valor nulo. Si eliminaran todos nos quedariamos con 2690-1759 = 931, lo cual no es bueno

2. Hay 750 filas que tienen igual o mas de 4 valores NAN en diferentes columnas. Las variables que tienen este problema son Sexo, Edad, Rango_edad (Es obvio porque se relaciona con Edad), Hijos y campana

Datos raros:

1. Hay 8 variables con problemas de missing values
2. Hijos tiene valores de 0.1 y 0.8, es posible que sea 1 y 1
3. Hay valores de 210 y 220 en la edad, es posible que sea 21 y 22

0.10 OJO: No se demoren mucho en la limpieza, de nada sirve que se gasten 30 minutos limpiando la BD para luego tener poco tiempo de hacer los otros puntos

0.10.1 P2. PIENSE EN EL NEGOCIO DADO LOS DATOS

0.10.2 P3. PIENSE QUE HIPOTESIS PUEDEN SERVIR PARA EL NEGOCIO Y PIENSE EN EL ANALISIS QUE PUEDE REALIZAR

** REPASO: TIPOS DE ANALISIS Y SUS HIPOTESIS NULAS **:
ANALIS UNIVARIADO
-Graficas exploratorias que puedan indicarme una tendencia. Por ejemplo: Boxplots (Var. Categorica en X), Histogramas (Var. numerica en X), Barplots (Var. categoria en X)
ANALIS BIVARIADO
-Grafico:

```
-Diagramas de torta
-Histogramas por variable categorica
-Barplot de variable categorica
```

-Estadistico:

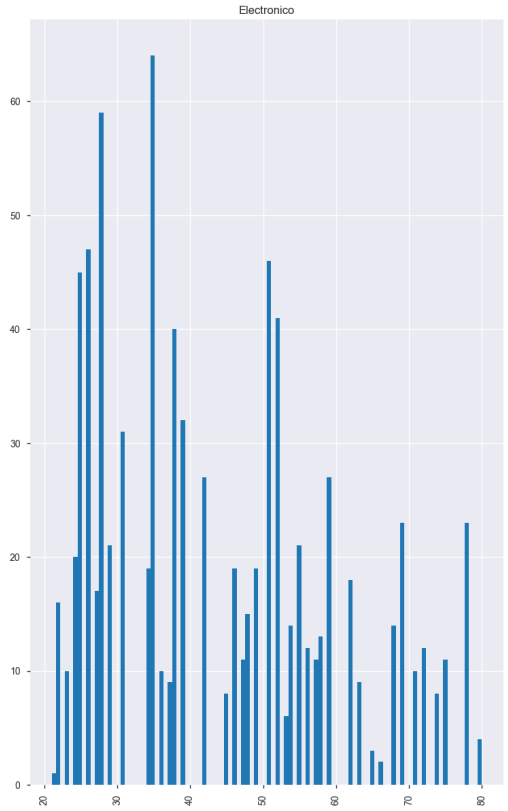
```
-Chi-squared (Tabla de contingencia) [Entre variables categoricas, que asume esta p
-Correlacion [Entre variables numericas, que asume esta prueba ?]
-Prueba de Normalidad [Por que es importante y cuando la hago ?]
```

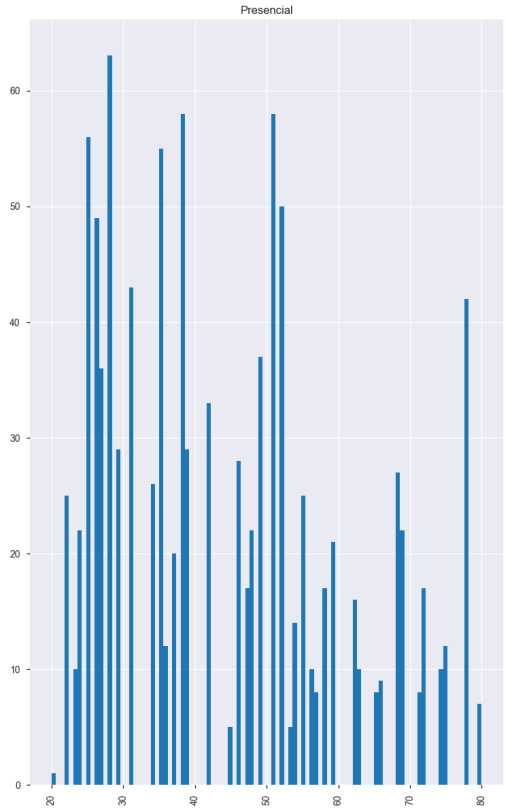
0.11 Recomendacion 3 y Repaso: Pueden usar prints y ifs para evaluar las p-values, esto permite que el analisis sea mas rapido. Sin embargo, es importante que entiendan que significa.

0.12 Por ejemplo: Si una hipotesis es:

0.12.1 H1: Los jovenes (edades entre 20-30) pagan mas con medio de pago electronico

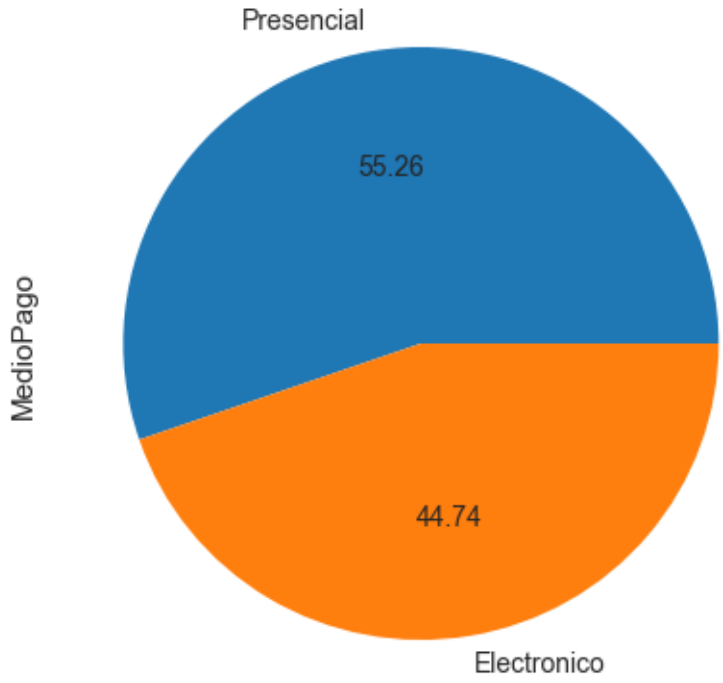
```
In [86]: # Estoy usando el dataframe que limpie rapidamente
         dfPrinV2.hist('Edad',by='MedioPago',figsize=(20,15),bins=100)

Out[86]: array([<matplotlib.axes._subplots.AxesSubplot object at 0x111627358>,
                <matplotlib.axes._subplots.AxesSubplot object at 0x111d2cb70>], dty
```






```
In [87]: # Haciendo pie chart dado MEDIO DE PAGO
         dfPrinV2.MedioPago.value_counts().plot(kind='pie',figsize=(5, 5),autopct='
```

```
Out[87]: <matplotlib.axes._subplots.AxesSubplot at 0x1116c63c8>
```




```
In [88]: # Que tipo de variables son Medio de Pago y Rango de Edad ?
         # Haciendo tabla de contingencia
         TablaH1 = pd.crosstab(index=dfPrinV2.MedioPago,columns = dfPrinV2['Rango_E

In [89]: TablaH1

Out[89]: Rango_Edad   a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30   All
         MedioPago
         Electronico         330                  304         234   868
         Presencial          398                  385         289  1072
         All                 728                  689         523  1940

In [90]: TablaH1.index = ["Electronico","Presencial","TOTAL_RANGO_EDAD"]
         TablaH1.columns=["a_Mayor_50","b_Menor_50_mayor_30","c_Menor_30","TOTAL_ME
```

0.13 OJO: Piense cuando una tabla de contingencia no cumple los requisitos para hacer Chi-cuadrado ? . Esto lo vieron en clase :)

```
In [92]: TablaH1

Out[92]:                  a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30  \
         Electronico             330                  304         234
         Presencial              398                  385         289
```

```
TOTAL_RANGO_EDAD         728                  689         523

                    TOTAL_MEDIO_DE_PAGO
Electronico                         868
Presencial                         1072
TOTAL_RANGO_EDAD                   1940
```

In [93]:
```
# Total sobre filas y columnas
TablaH1/TablaH1.ix['TOTAL_RANGO_EDAD','TOTAL_MEDIO_DE_PAGO']
```

Out[93]:
```
                  a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30  \
Electronico         0.170103             0.156701    0.120619
Presencial          0.205155             0.198454    0.148969
TOTAL_RANGO_EDAD    0.375258             0.355155    0.269588

                    TOTAL_MEDIO_DE_PAGO
Electronico                    0.447423
Presencial                     0.552577
TOTAL_RANGO_EDAD               1.000000
```

In [94]:
```
# Total sobre filas
TablaH1/TablaH1.ix["TOTAL_RANGO_EDAD"]
```

Out[94]:
```
                  a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30  \
Electronico         0.453297             0.441219    0.447419
Presencial          0.546703             0.558781    0.552581
TOTAL_RANGO_EDAD    1.000000             1.000000    1.000000

                    TOTAL_MEDIO_DE_PAGO
Electronico                    0.447423
Presencial                     0.552577
TOTAL_RANGO_EDAD               1.000000
```

In [95]:
```
# Total sobre columnas
TablaH1.div(TablaH1.TOTAL_MEDIO_DE_PAGO, axis=0)
```

Out[95]:
```
                  a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30  \
Electronico         0.380184             0.350230    0.269585
Presencial          0.371269             0.359142    0.269590
TOTAL_RANGO_EDAD    0.375258             0.355155    0.269588

                    TOTAL_MEDIO_DE_PAGO
Electronico                         1.0
Presencial                          1.0
TOTAL_RANGO_EDAD                    1.0
```

In [96]:
```
# Generando TablaH1 para hacer chi-cuadrado
TablaH1=pd.crosstab(index=dfPrinV2.MedioPago,columns=dfPrinV2["Rango_Edad'
```

```
In [98]: TablaH1.index= ["Electronico","Presencial"]
         TablaH1
         # Se genero la tabla de contingencia
```

```
Out[98]: Rango_Edad   a_Mayor_50  b_Menor_50_mayor_30  c_Menor_30
         Electronico         330                  304         234
         Presencial          398                  385         289
```

```
In [99]: # Haciendo prueba de chi-cuadrado
         chi2_contingency(TablaH1)
```

```
Out[99]: (0.2088464653828308,
          0.90084395011802076,
          2,
          array([[ 325.72371134,  308.2742268 ,  234.00206186],
                 [ 402.27628866,  380.7257732 ,  288.99793814]]))
```

```
In [100]: # Guardo el p-value en una variable para luego evaluarla
          # dado las condiciones de abajo
          pvalue = chi2_contingency(TablaH1)[1]
```

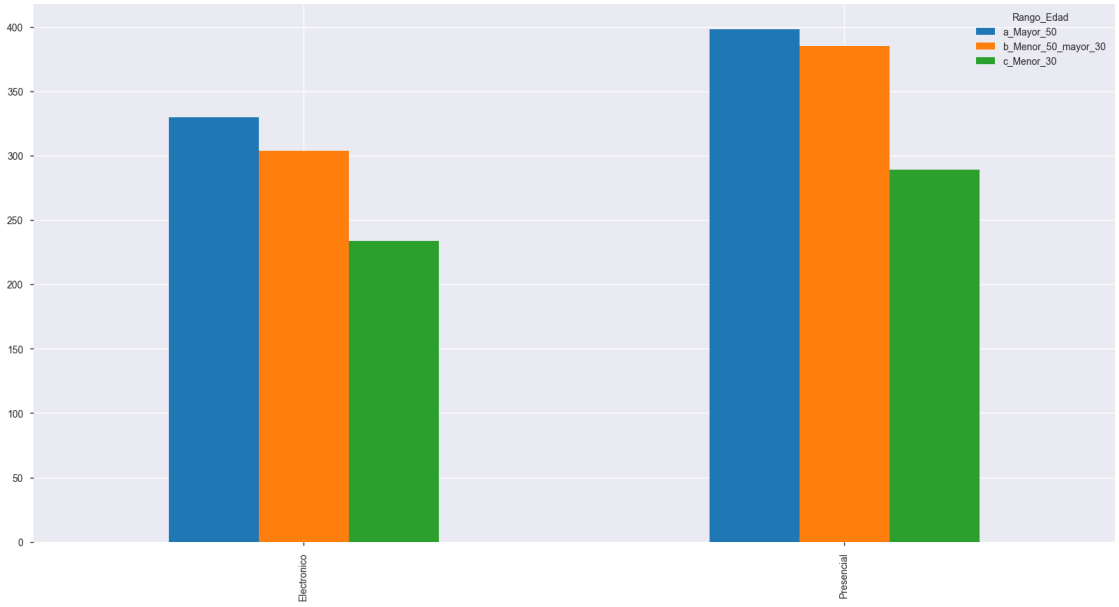
```
In [102]: # Usando prints y if para saber si acepto o no la hipotesis nula y
          # asi concentrarme en el analisis
          print("Hipotesis NULA:", "Las variables son independendientes")
          print("P-Value:", pvalue)

          pval = pvalue

          if pval > 0.05:
              print("Accept NULL hypothesis - Las variables  son independientes es
          if pval < 0.05:
              print("Reject NULL hypothesis - Las variables son dependientes.")
```

```
Hipotesis NULA: Las variables son independendientes
P-Value: 0.900843950118
Accept NULL hypothesis - Las variables  son independientes es decir no son dependie
```

0.14 Recomendacion 4: Siempre es recomendable hacer una grafica para soportar los resultados estadisticos y ver otras tendencias

```
In [103]: TablaH1.plot(kind='bar',figsize=(20,10),legend=True)
```

```
Out[103]: <matplotlib.axes._subplots.AxesSubplot at 0x115f426a0>
```



0.15 Respuesta

La edad no esta relacionada con el medio de pago en los datos, es decir no necesariamente los jovenes pagan con medio electronico.

El valor-p fue mayor a 0.05 por lo cual no se puede rechazar la hipotesis nula

0.16 PRACTICA: En serio les recomiendo que repitan el ejercicio anterior y hagan los dos puntos de abajo

0.16.1 1. Hagan una hipotesis para que sea evaluada con pruebas de correlaciones y evaluan la hipotesis nula de la prueba

0.16.2 2. Hagan una hipotesis para que sea evaluada con pruebas de chi-cuadrado, cambiando una variable numerica a una categorica de acuerdo a las diapositivas de 6-11 de la Sesion 6. Estadistica Bivariada