def mono\_bin(Y, X, n):

df1 = pd.DataFrame({"X": X, "Y": Y})

justmiss = df1[['X','Y']][df1.X.isnull()]

notmiss = df1[['X','Y']][df1.X.notnull()]

r = 0

while np.abs(r) < 1:

try:

d1 = pd.DataFrame({"X": notmiss.X, "Y": notmiss.Y, "Bucket": pd.qcut(notmiss.X, n)})

d2 = d1.groupby('Bucket', as\_index=True)

r, p = stats.spearmanr(d2.mean().X, d2.mean().Y)

n = n - 1

except Exception as e:

n = n - 1

if len(d2) == 1:

n = force\_bin

bins = algos.quantile(notmiss.X, np.linspace(0, 1, n))

if len(np.unique(bins)) == 2:

bins = np.insert(bins, 0, 1)

bins[1] = bins[1]-(bins[1]/2)

d1 = pd.DataFrame({"X": notmiss.X, "Y": notmiss.Y, "Bucket": pd.cut(notmiss.X, np.unique(bins),include\_lowest=True)})

d2 = d1.groupby('Bucket', as\_index=True)

d3 = pd.DataFrame({},index=[])

d3["MIN\_VALUE"] = d2.min().X

d3["MAX\_VALUE"] = d2.max().X

d3["COUNT"] = d2.count().Y

d3["EVENT"] = d2.sum().Y

d3["NONEVENT"] = d2.count().Y -

d2.sum().Y

d3=d3.reset\_index(drop=True)

if len(justmiss.index) > 0:

d4 = pd.DataFrame({'MIN\_VALUE':np.nan},index=[0])

d4["MAX\_VALUE"] = np.nan

d4["COUNT"] = justmiss.count().Y

d4["EVENT"] = justmiss.sum().Y

d4["NONEVENT"] = justmiss.count().Y - justmiss.sum().Y

d3 = d3.append(d4,ignore\_index=True)

d3["EVENT\_RATE"] = d3.EVENT/d3.COUNT

d3["NON\_EVENT\_RATE"] = d3.NONEVENT/d3.COUNT

d3["DIST\_EVENT"] = d3.EVENT/

d3.sum().EVENT

d3["DIST\_NON\_EVENT"] = d3.NONEVENT/d3.sum().NONEVENT

d3["WOE"] = np.log(d3.DIST\_NON\_EVENT/d3.DIST\_EVENT)

d3["IV"] = (d3.DIST\_EVENT-d3.DIST\_NON\_EVENT)\*np.log(d3.DIST\_EVENT/d3.DIST\_NON\_EVENT)

d3["VAR\_NAME"] = "VAR"

d3 = d3[['VAR\_NAME','MIN\_VALUE', 'MAX\_VALUE', 'COUNT', 'EVENT', 'EVENT\_RATE', 'NONEVENT', 'NON\_EVENT\_RATE', 'DIST\_EVENT','DIST\_NON\_EVENT','WOE', 'IV']]

d3 = d3.replace([np.inf, -np.inf], 0)

d3.IV = d3.IV.sum()

return(d3)