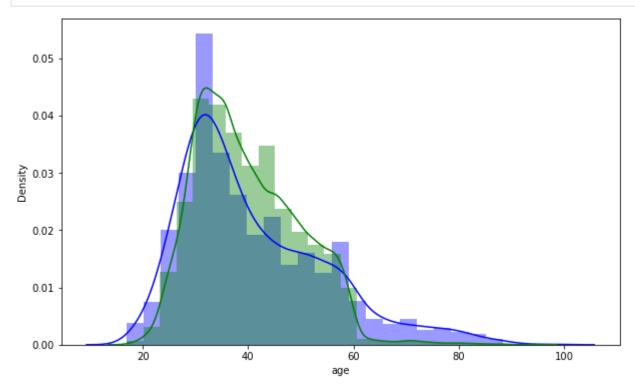
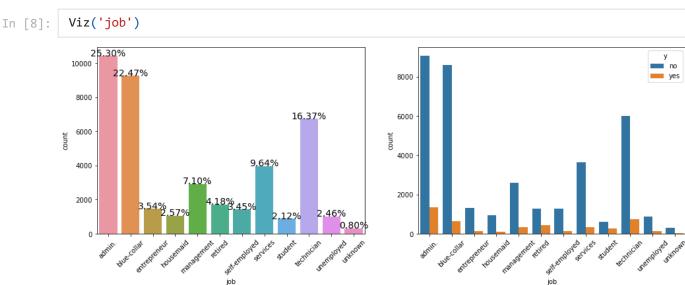
```
In [32]:
          import numpy as np # linear algebra
           import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
           import matplotlib.pyplot as plt
           import seaborn as sns
           from matplotlib import rcParams
          from sklearn.preprocessing import LabelEncoder
           import warnings
          warnings.filterwarnings('ignore')
           color = sns.color_palette()
           rcParams['figure.figsize'] = 10, 6
           lbl = LabelEncoder()
          %matplotlib inline
           pd.set_option('display.max_columns', 50)
In [33]:
          main = pd.read_csv('bank-additional-full.csv',sep= ';')
In [34]:
          main.head(3)
Out[34]:
                       job marital
                                    education
                                               default housing loan
                                                                      contact month day_of_week dura
            age
          0
             56 housemaid
                            married
                                      basic.4y
                                                                     telephone
                                                   no
                                                            no
                                                                 no
                                                                                 may
                                                                                             mon
             57
          1
                    services
                            married high.school unknown
                                                                     telephone
                                                            no
                                                                 no
                                                                                 may
                                                                                             mon
             37
                           married high.school
                    services
                                                           yes
                                                                     telephone
                                                                                             mon
                                                   no
                                                                 no
                                                                                 may
          main[main['y'] == 'yes'].shape[0],main[main['y'] == 'no'].shape[0]
 In [4]:
         (4640, 36548)
 Out[4]:
 In [5]:
          main.columns
 Out[5]: Index(['age', 'job', 'marital', 'education', 'default', 'housing', 'loan',
                 'contact', 'month', 'day_of_week', 'duration', 'campaign', 'pdays',
                 'previous', 'poutcome', 'emp.var.rate', 'cons.price.idx',
                 'cons.conf.idx', 'euribor3m', 'nr.employed', 'y'],
                dtype='object')
          def Viz(s):
 In [6]:
               if main[s].dtype!= '0':
                   age_target_yes = main[(main[s] > 0) & (main['y'] == 'yes')]
                   age_target_no = main[(main[s] > 0) & (main['y'] == 'no')]
                   plt.figure(figsize=(10, 6))
                   sns.distplot(age_target_yes[s], bins=25, color='b')
                   sns.distplot(age_target_no[s], bins=25, color='g')
                   plt.show()
               else:
                   plt.figure(figsize=(16, 8))
                   tmp = pd.crosstab(main[s], main['y'], normalize='index') * 100
                   tmp = tmp.reset_index()
                   plt.subplot(221)
                   g = sns.countplot(x=s, data=main, order=list(tmp[s].values))
                   for p in g.patches:
                       height = p.get_height()
```

```
g.text(p.get_x()+p.get_width()/2.,
height + 3, '{:1.2f}%'.format(height/main.shape[0]*100),
ha="center",fontsize=14)

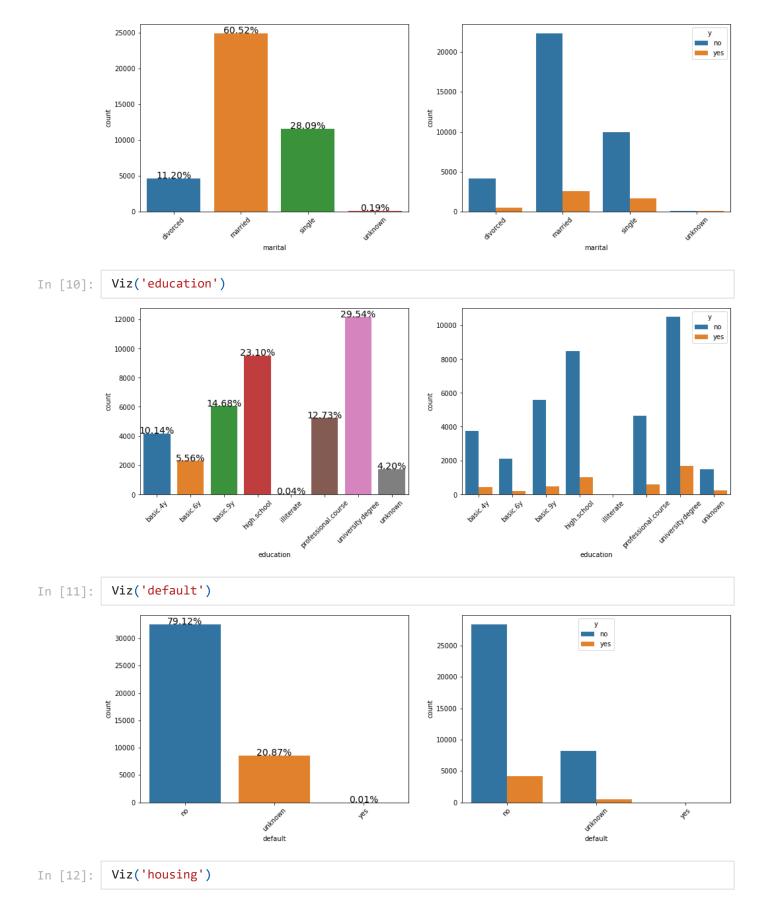
plt.xticks(rotation=45)
plt.subplot(222)
g1 = sns.countplot(x=s, hue='y', data=main,order=list(tmp[s].values))
plt.subplots_adjust(hspace = 0.6, top = 1.4)
plt.xticks(rotation=45)
plt.show()
```

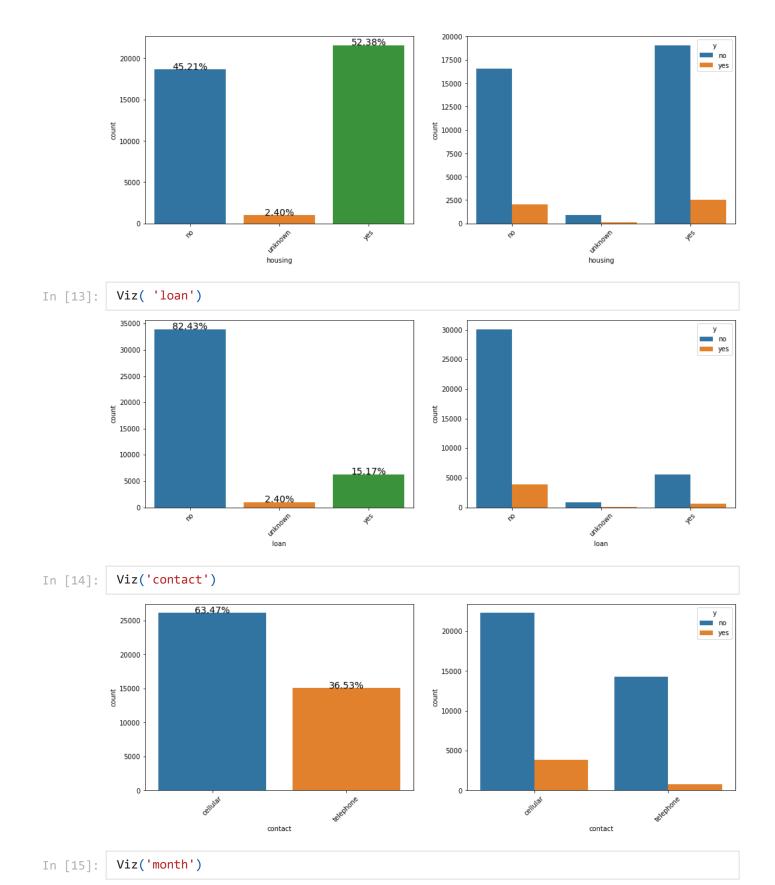
In [7]: Viz('age')

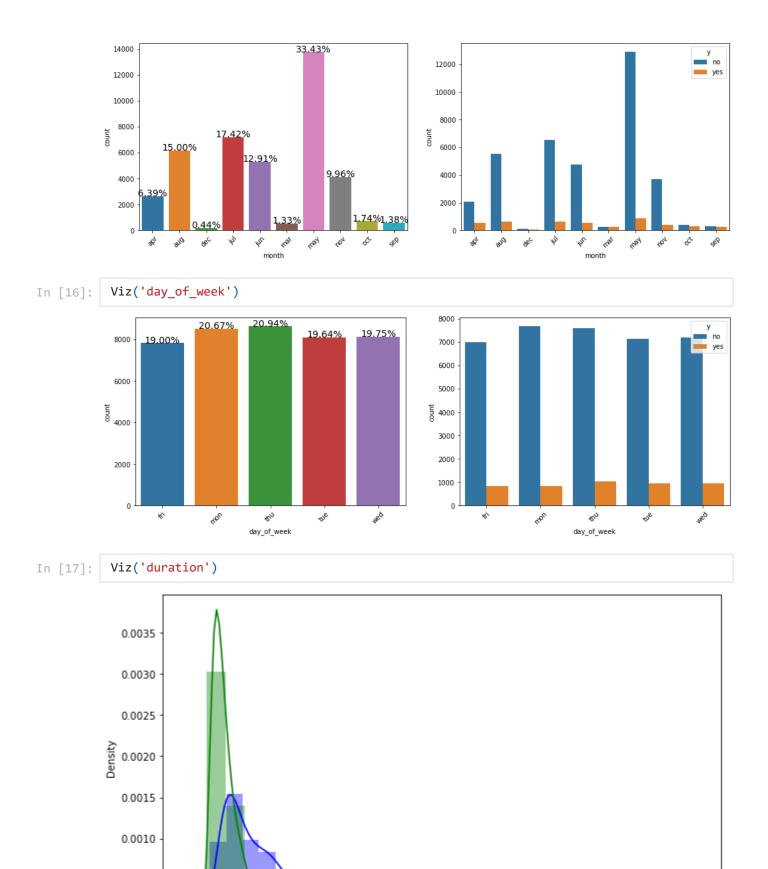




```
In [9]: Viz('marital')
```







In [18]: Viz('campaign')

2000

duration

3000

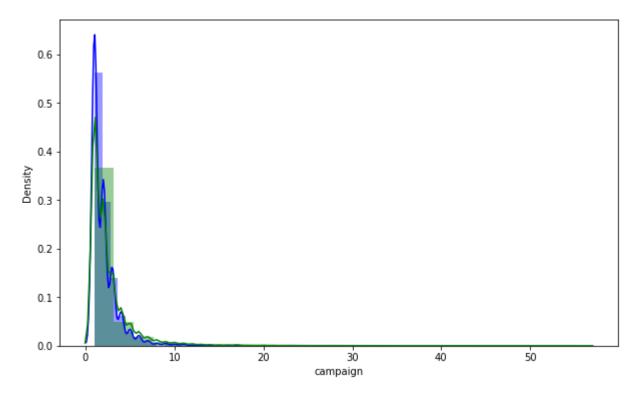
4000

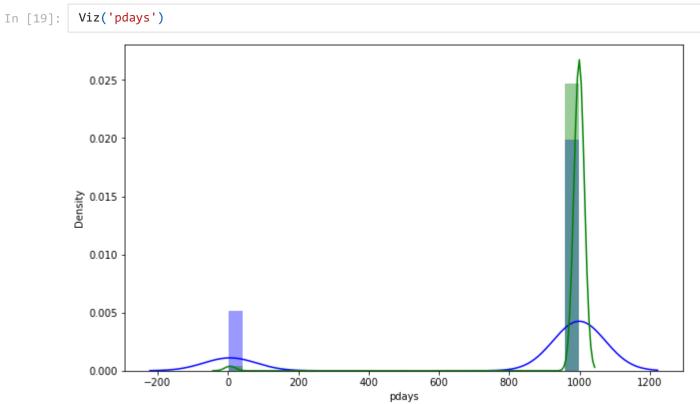
5000

1000

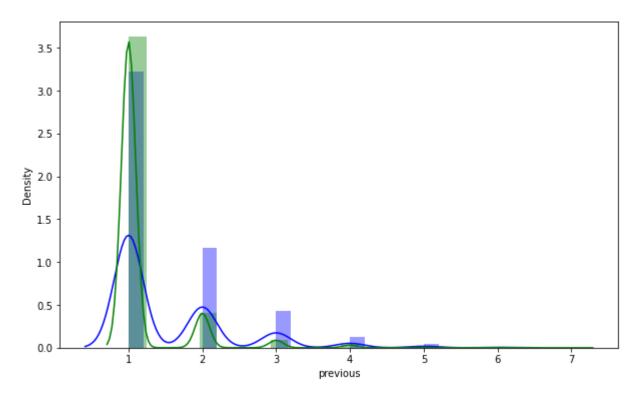
0.0005

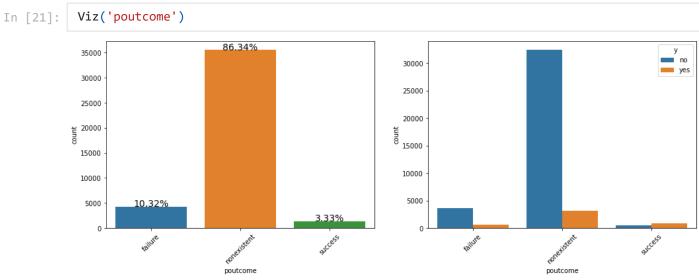
0.0000



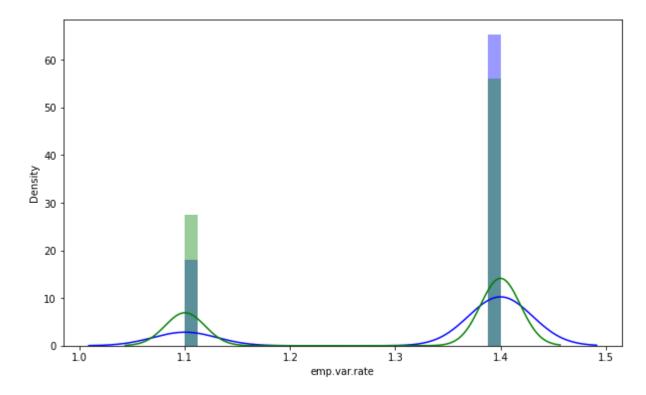


In [20]: Viz('previous')

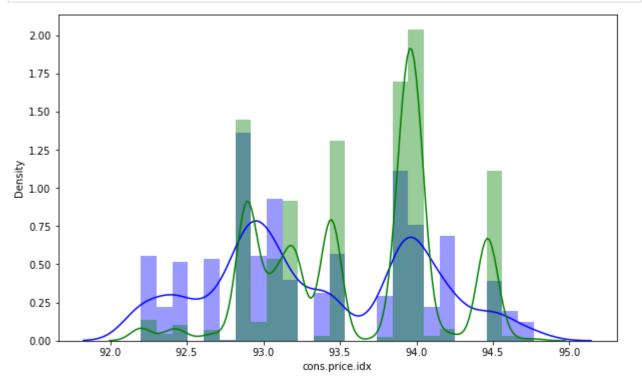




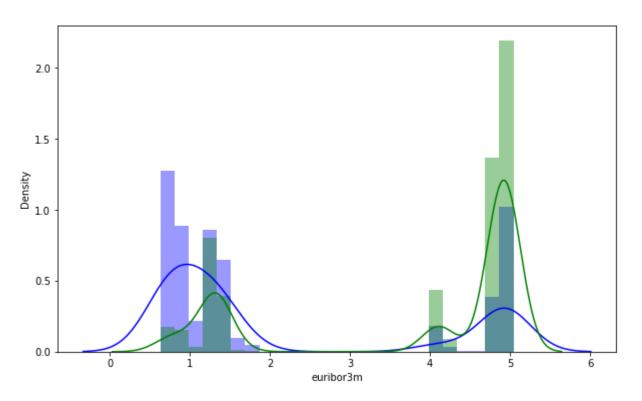
In [22]: Viz('emp.var.rate')



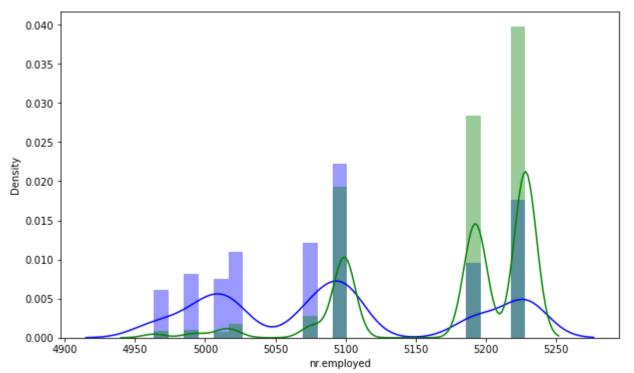
In [23]: Viz('cons.price.idx')



In [24]: Viz('euribor3m')



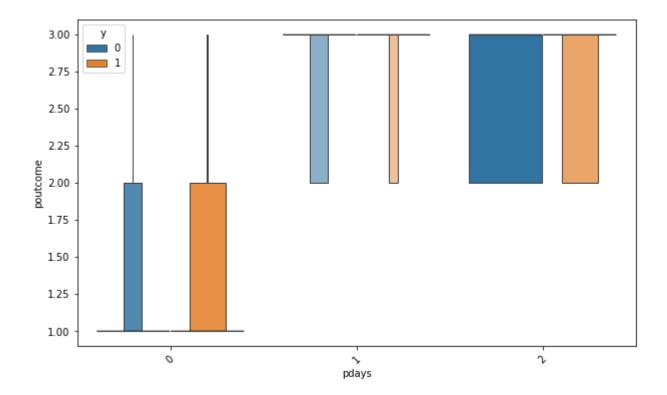




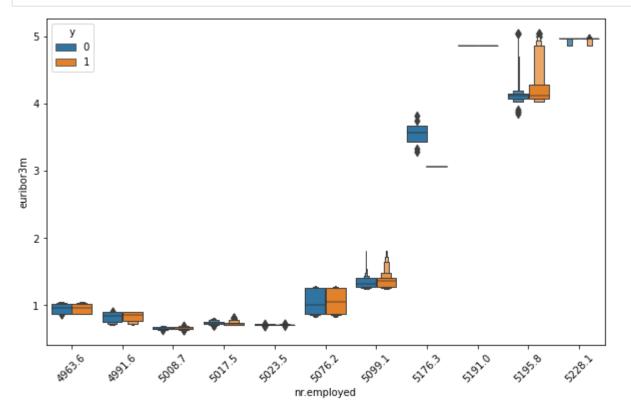
```
df["education"] = df["education"].astype('category')
              df["education"] = df["education"].cat.codes
              df['y'] = df['y'].map({'no':0, 'yes':1})
              d = {range(1, 10): 1, range(10, 20): 2, range(20, 1000): 0}
              df['pdays'] = df['pdays'].apply(lambda x: next((v for k, v in d.items() if x in k), 0))
              d = {range(1, 60): 1, range(60, 200): 2, range(200, 600): 3, range(600, 1000): 4, range(
              df['duration'] = df['duration'].apply(lambda x: next((v for k, v in d.items() if x in k
              cor=df.corr(method="pearson")
In [27]:
              plt.figure(figsize=(25,10))
              sns.heatmap(cor,vmax=1, square=True,annot=True)
              plt.show()
                                                                                                                               - 1.0
                              0.033 -0.39 -0.120.00290.0016.0074.00160.00460.026 0.024 0.0260.000B7000860.13 0.011-0.018 0.03
                                  0.027 0.14 0.00590.0053-0.01 0.012-0.0170.047 0.051 0.049-0.086-0.052 0.02 -0.075-0.086 0.055
                      job
                                                                                                                               - 0.8
                  marital
                                       0.11 -0.0030.0110.00630.00840.00720.037 0.039 0.04 -0.0840.057-0.0340.092-0.086 0.046
                          -0.12 0.14 0.11 1 0.00180.0170.00790.01B.000370.041 0.039 0.04 -0.0440.0820.079-0.0360.041 0.058
                education
                         0.00290.00590.0030.0018 1
                                                -0.0040.00420.00530.00420.0020.00240.00270.00580.00270.0050.00670.00730.003
                  default
                                                                                                                               - 0.6
                          .0016.00530.011 0.017-0.004
                                                  1
                                                     0.046-0.011-0.0110.00840.022 0.024-0.061-0.081-0.034-0.06-0.046 0.012
                  housing
                         0.0074 \cdot 0.010.00630.00790.00420.046 1 0.00360.00540.000800.00190.000410.00140.00490.0130.000180.00420.004
                    loan
                                                                                                                               - 0.4
                                                               -0.14 0.072 0.043 0.052 -0.0470.00650.012 -0.051 -0.07 0.37
                          0.00160.0120.00840.0150.00530.0110.0036 1
                         0.00460.0170.00702000370.00420.0110.0054-0.14
                                                                    -0.049-0.079-0.086 0.15 0.13 -0.014 0.14 0.14 -0.066
                         0.026 0.047 0.037 0.041-0.0020.0084.00088.072-0.049 1 0.54 0.69 -0.25-0.0710.075 -0.28 -0.35 0.3
                                                                                                                               - 0.2
                         0.024 0.051 0.039 0.0390.00240.0220.00190.043-0.079 0.54
                                                                          1 0.87 -0.42 -0.2 -0.051-0.45 -0.5 0.23
                poutcome -0.026 0.049 0.04 0.04 0.00270.0240.0004 D.052-0.086 0.69 0.87 1
                                                                                  -0.46 -0.26 -0.042 -0.48 -0.51 0.27
                                                                                                                               0.0
              emp.var.rate -.00030.086-0.084-0.0440.00580.0610.0014-0.047 0.15 -0.25 -0.42 -0.46
                                                                                                 0.97 0.91 -0.3
             cons.price.idx =.0008@.052-0.057-0.0820.00270.0810.004@.00650.13 -0.071 -0.2 -0.26 0.78 1
                                                                                            0.059 0.69 0.52 -0.14
                                                                                                                                -0.2
              0.28 0.1 0.055
                                                                                              1
               euribor3m -0.011-0.075-0.092-0.0360.0067-0.06-0.00030.051 0.14 -0.28 -0.45 -0.48 0.97 0.69 0.28
                                                                                                      0.95 -0.31
              nr.employed -0.018-0.086-0.086-0.0410.00730.0460.0042-0.07 0.14 -0.35 -0.5 -0.51
                                                                                   0.91 0.52
                                                                                                       1
                                                                                                           -0.35
                                                                                                                               -0.4
                          0.03 0.055 0.046 0.0580.00370.0120.0047 0.37 -0.066 0.3 0.23 0.27 -0.3 -0.14 0.055 -0.31 -0.35
                                                                                    emp.var.rate
                                                                                         ons.price.idx
                                                                                             cons.conf.idx
                                                                                                       employed
In [28]:
              def related(a,b):
                   plt.figure(figsize=(10, 6))
                   sns.boxenplot(x=a, y=b, hue='y', data=df )
                   plt.xticks(rotation=45)
                   plt.show()
```

related('pdays','poutcome')

In [29]:



In [30]: related('nr.employed','euribor3m')



In []: