In [1]:

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
```

In [2]:

```
data = pd.read_csv("/Users/gabriel/Desktop/Tesis/Prediccion bancaria/Datasets/Banco/banco.csv", sep=";")
```

In [3]:

```
data.head()
```

Out[3]:

	age	job	marital	education	default	housing	loar
0	30	blue-collar	married	basic.9y	no	yes	nc
1	39	services	single	high.school	no	no	nc
2	25	services	married	high.school	no	yes	nc
3	38	services	married	basic.9y	no	unknown	unknowr
4	47	admin.	married	university.degree	no	yes	nc

5 rows × 21 columns

In [4]:

```
data.shape
```

Out[4]:

```
(4119, 21)
```

In [5]:

```
data.columns.values
```

Out[5]:

```
array(['age', 'job', 'marital', 'education', 'defa
ult', 'housing', 'loan',
       'contact', 'month', 'day_of_week', 'duratio
n', 'campaign', 'pdays',
       'previous', 'poutcome', 'emp.var.rate', 'co
ns.price.idx',
       'cons.conf.idx', 'euribor3m', 'nr.employed'
, 'y'], dtype=object)
```

In [6]:

```
data["y"] = (data["y"]=="yes").astype(int)
```

In [7]:

```
data.tail()
```

Out[7]:

	age	job	marital	education	default	housing	loan
4114	30	admin.	married	basic.6y	no	yes	yes
4115	39	admin.	married	high.school	no	yes	no
4116	27	student	single	high.school	no	no	no
4117	58	admin.	married	high.school	no	no	no
4118	34	management	single	high.school	no	yes	no

5 rows × 21 columns

In [8]:

```
data["education"].unique()
```

Out[8]:

```
array(['basic.9y', 'high.school', 'university.degr
ee',
       'professional.course', 'basic.6y', 'basic.4
y', 'unknown',
       'illiterate'], dtype=object)
```

In [9]:

```
data["education"] = np.where(data["education"]=="basic.4y", "B
asic", data["education"])
data["education"] = np.where(data["education"]=="basic.6y", "B
asic", data["education"])
data["education"] = np.where(data["education"]=="basic.9y", "B
asic", data["education"])

data["education"] = np.where(data["education"]=="high.school",
"High School", data["education"])
data["education"] = np.where(data["education"]=="professional.
course", "Professional Course", data["education"])
data["education"] = np.where(data["education"]=="university.de
gree", "University Degree", data["education"])

data["education"] = np.where(data["education"]=="illiterate",
"Illiterate", data["education"])
data["education"] = np.where(data["education"]=="unknown", "Un
known", data["education"])
```

In [10]:

```
data["education"].unique()
```

Out[10]:

```
array(['Basic', 'High School', 'University Degree'
, 'Professional Course',
       'Unknown', 'Illiterate'], dtype=object)
```

In [11]:

```
data["y"].value_counts()
```

Out[11]:

```
0    3668
1     451
Name: y, dtype: int64
```

In [12]:

```
data.groupby("y").mean()
```

Out[12]:

y	age	duration	campaign	pdays	previous	emp.var.ra
0	39.895311	219.40976	2.605780	982.763086	0.141767	0.2401
1	41.889135	560.78714	1.980044	778.722838	0.585366	-1.1773

In [13]:

```
data.groupby("education").mean()
```

Out[13]:

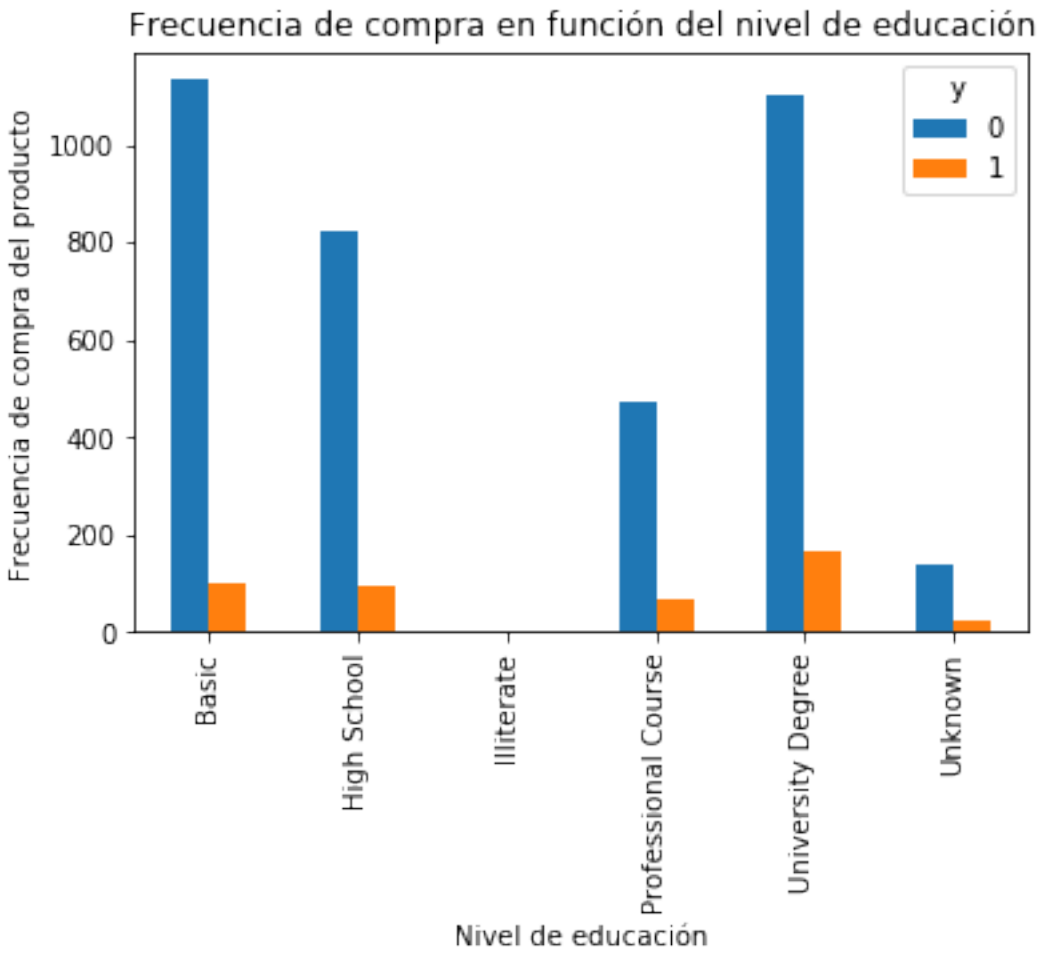
education	age	duration	campaign	pdays	previous
Basic	42.337124	253.898457	2.429732	978.815597	0.149472
High School	38.097720	258.534202	2.630836	958.022801	0.206298
Illiterate	42.000000	146.000000	4.000000	999.000000	0.000000
Professional Course	40.207477	278.816822	2.512150	958.211215	0.194393
University Degree	39.017405	247.707278	2.583070	947.900316	0.207278
Unknown	42.826347	267.281437	2.538922	939.700599	0.263473

In [14]:

```
%matplotlib inline
pd.crosstab(data.education, data.y).plot(kind="bar")
plt.title("Frecuencia de compra en función del nivel de educac
ión")
plt.xlabel("Nivel de educación")
plt.ylabel("Frecuencia de compra del producto")
```

Out[14]:

```
Text(0, 0.5, 'Frecuencia de compra del producto')
```

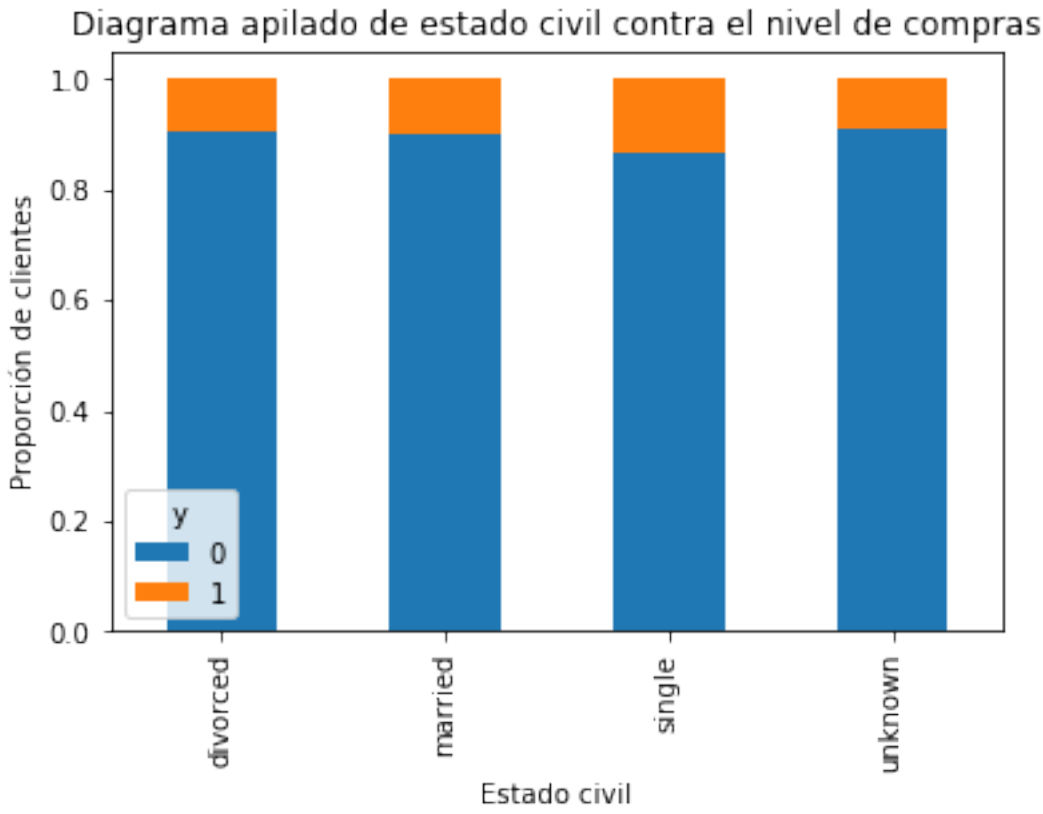


In [15]:

```
table=pd.crosstab(data.marital, data.y)
table.div(table.sum(1).astype(float), axis=0).plot(kind="bar",
stacked=True)
plt.title("Diagrama apilado de estado civil contra el nivel de
compras")
plt.xlabel("Estado civil")
plt.ylabel("Proporción de clientes")
```

Out[15]:

```
Text(0, 0.5, 'Proporción de clientes')
```

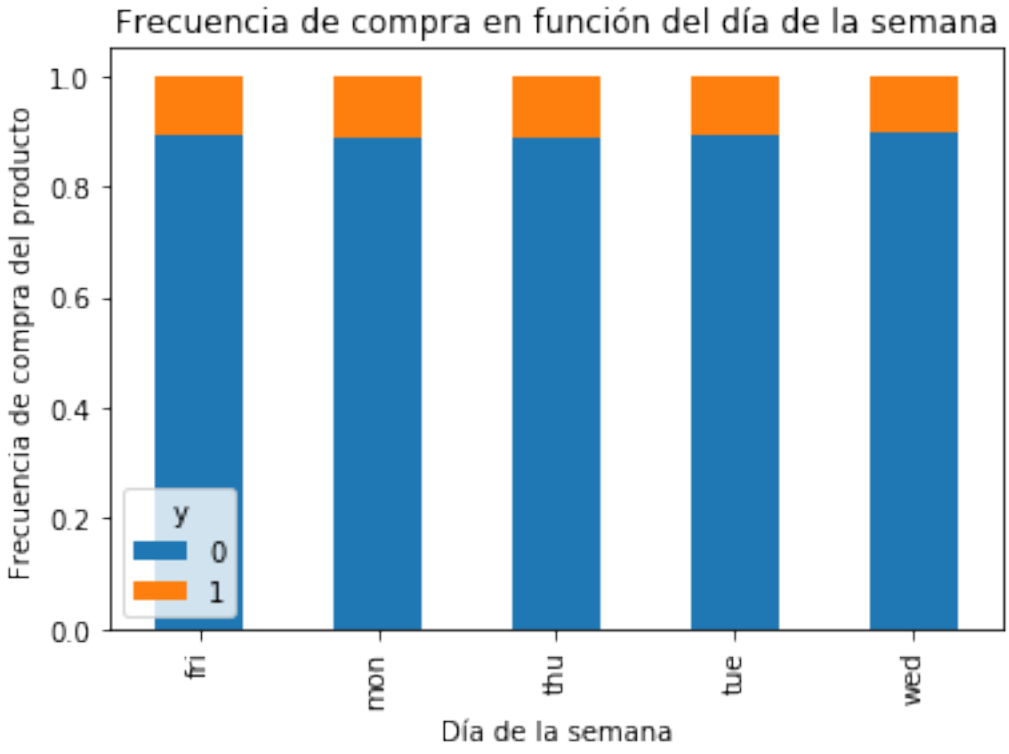


In [16]:

```
%matplotlib inline
table= pd.crosstab(data.day_of_week, data.y)
table.div(table.sum(1).astype(float), axis=0).plot(kind="bar",
stacked=True)
plt.title("Frecuencia de compra en función del día de la seman
a")
plt.xlabel("Día de la semana")
plt.ylabel("Frecuencia de compra del producto")
```

Out[16]:

```
Text(0, 0.5, 'Frecuencia de compra del producto')
```

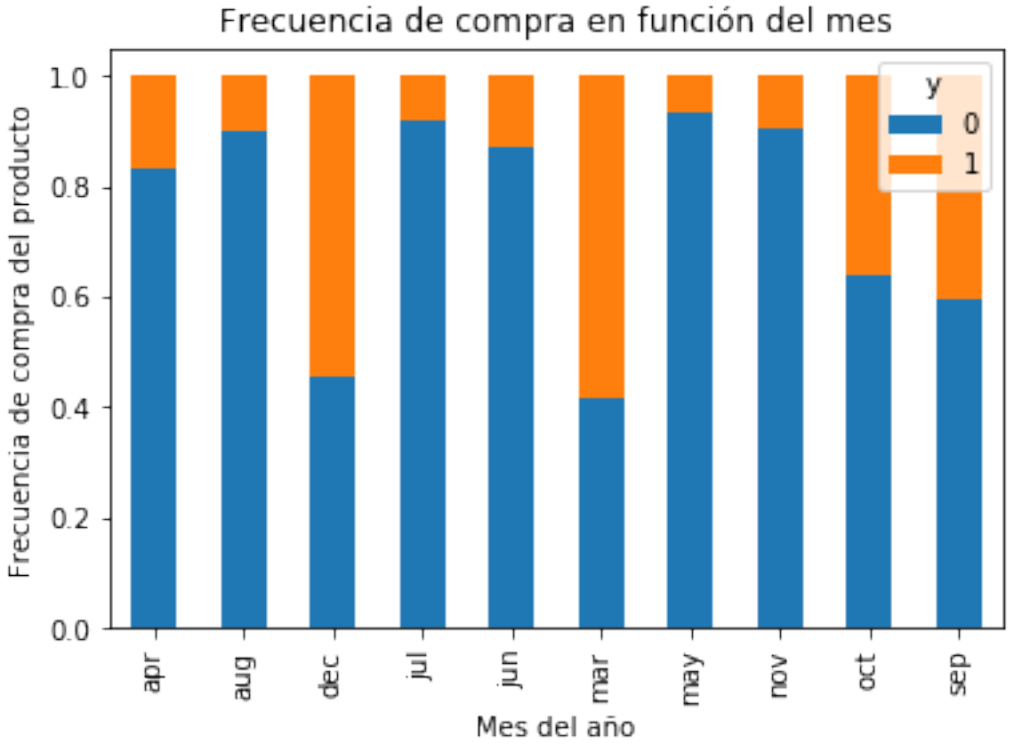


In [17]:

```
%matplotlib inline
table= pd.crosstab(data.month, data.y)
table.div(table.sum(1).astype(float), axis=0).plot(kind="bar",
stacked=True)
plt.title("Frecuencia de compra en función del mes")
plt.xlabel("Mes del año")
plt.ylabel("Frecuencia de compra del producto")
```

Out[17]:

```
Text(0, 0.5, 'Frecuencia de compra del producto')
```

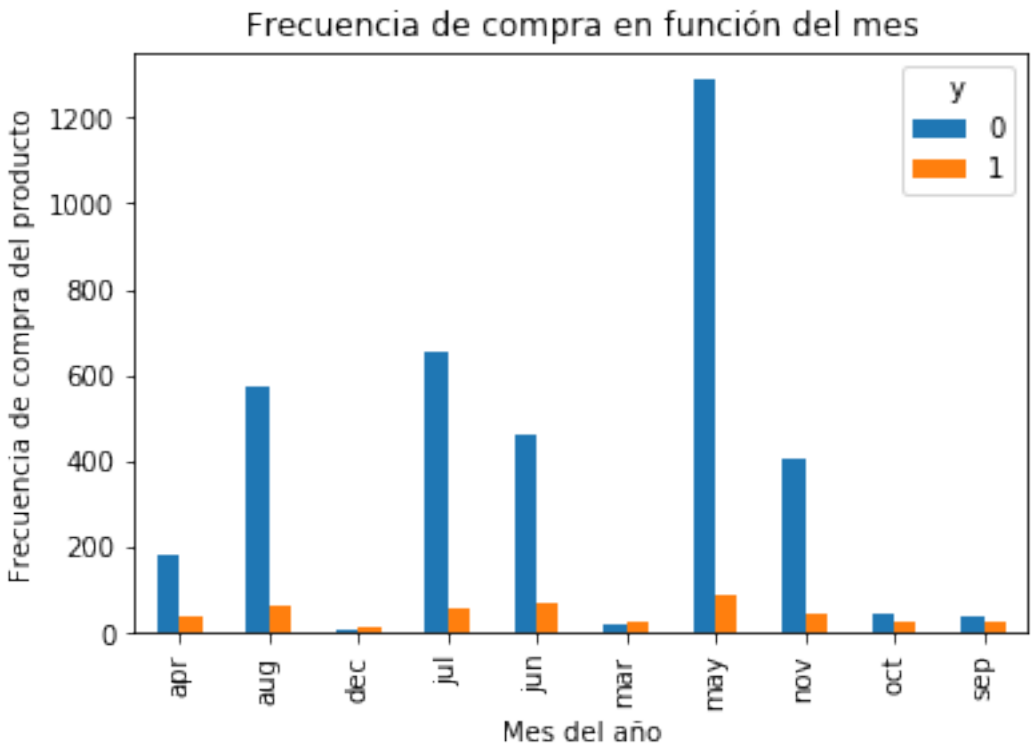


In [18]:

```
%matplotlib inline
table.plot(kind="bar", stacked=False)
plt.title("Frecuencia de compra en función del mes")
plt.xlabel("Mes del año")
plt.ylabel("Frecuencia de compra del producto")
```

Out[18]:

```
Text(0, 0.5, 'Frecuencia de compra del producto')
```

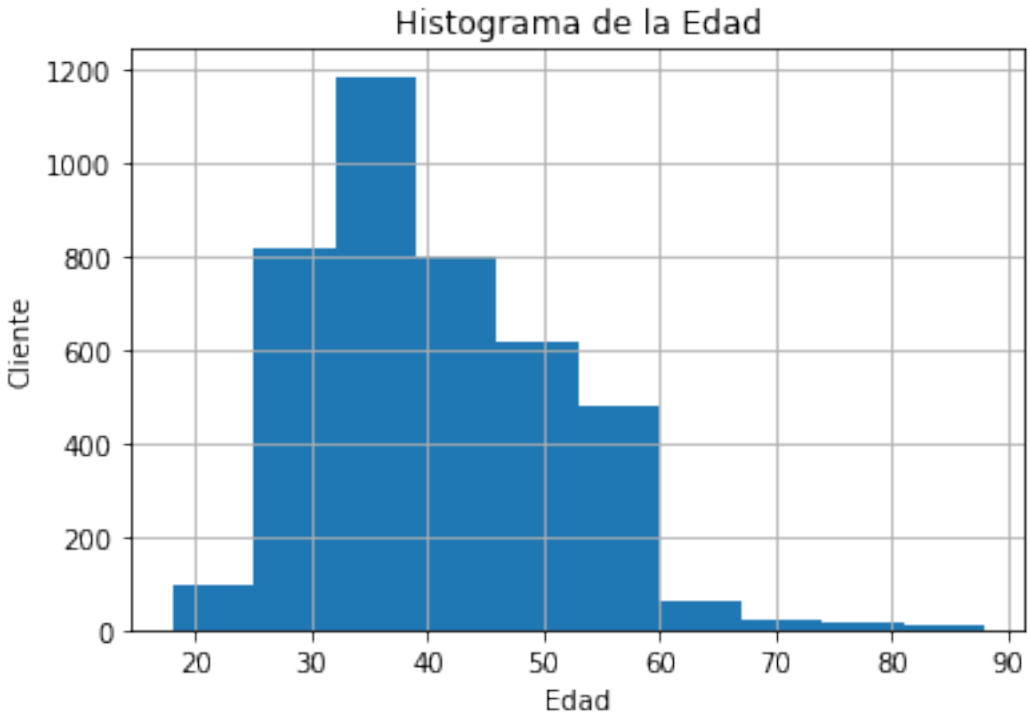


In [19]:

```
%matplotlib inline
data.age.hist()
plt.title("Histograma de la Edad")
plt.xlabel("Edad")
plt.ylabel("Cliente")
```

Out[19]:

```
Text(0, 0.5, 'Cliente')
```

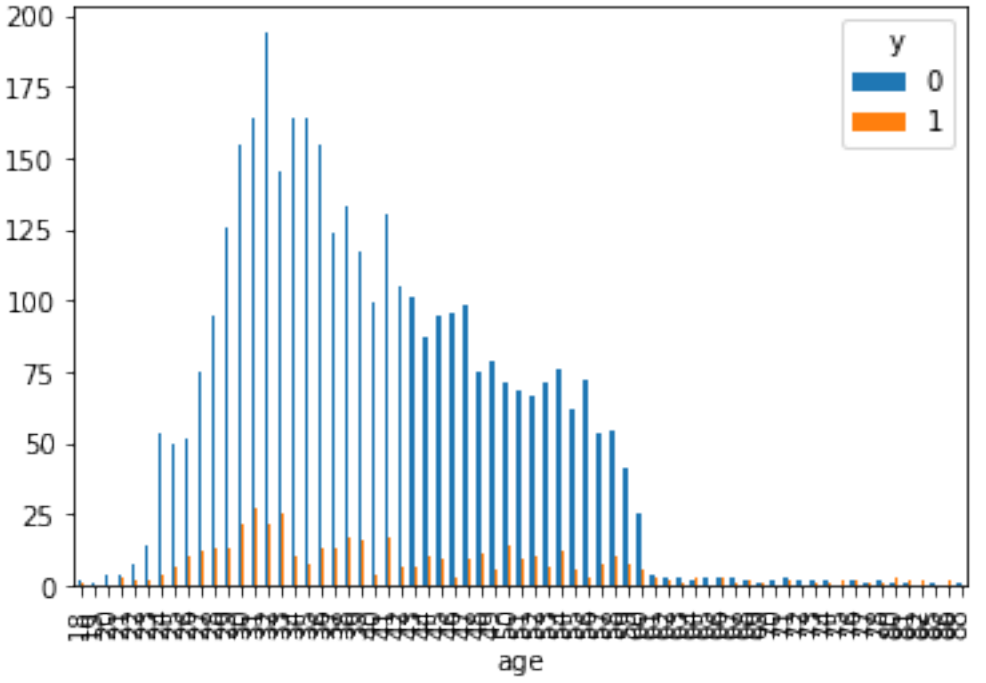


In [20]:

```
pd.crosstab(data.age, data.y).plot(kind="bar")
```

Out[20]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x117aab
050>
```

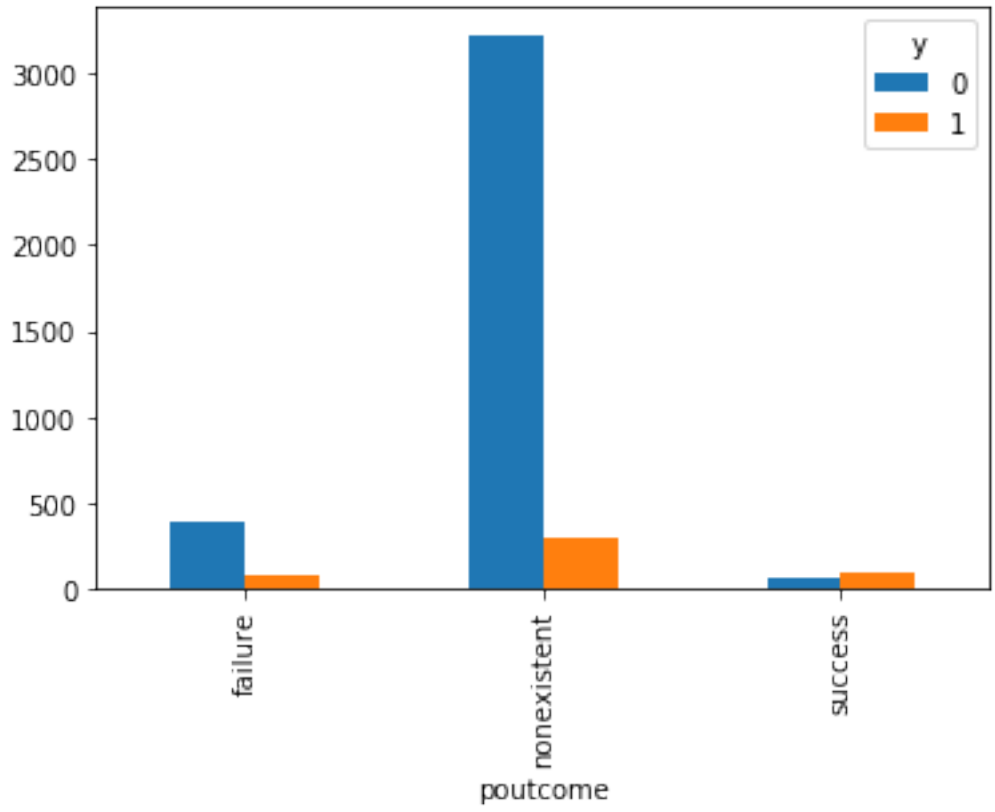


In [21]:

```
pd.crosstab(data.poutcome, data.y).plot(kind="bar")
```

Out[21]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x117de7
750>
```



In [22]:

```
categories = ["job", "marital", "education", "housing", "loan"
, "contact",
              "month", "day_of_week", "poutcome"]
for category in categories:
    cat_list = "cat"+ "_"+category
    cat_dummies = pd.get_dummies(data[category], prefix=catego
ry)
    data_new = data.join(cat_dummies)
    data = data_new
```

In [23]:

```
data_vars = data.columns.values.tolist()
```

In [24]:

```
to_keep = [v for v in data_vars if v not in categories]
to_keep = [v for v in to_keep if v not in ["default"]]
```

In [25]:

```
bank_data = data[to_keep]
bank_data.columns.values
```

Out[25]:

```
array(['age', 'duration', 'campaign', 'pdays', 'pr
evious', 'emp.var.rate',
       'cons.price.idx', 'cons.conf.idx', 'euribor
3m', 'nr.employed', 'y',
       'job_admin.', 'job_blue-collar', 'job_entre
preneur',
       'job_housemaid', 'job_management', 'job_ret
ired',
       'job_self-employed', 'job_services', 'job_s
tudent',
       'job_technician', 'job_unemployed', 'job_un
known',
       'marital_divorced', 'marital_married', 'mar
ital_single',
       'marital_unknown', 'education_Basic', 'educ
ation_High School',
       'education_Illiterate', 'education_Professi
onal Course',
       'education_University Degree', 'education_U
nknown', 'housing_no',
       'housing_unknown', 'housing_yes', 'loan_no'
, 'loan_unknown',
       'loan_yes', 'contact_cellular', 'contact_te
lephone', 'month_apr',
       'month_aug', 'month_dec', 'month_jul', 'mon
th_jun', 'month_mar',
       'month_may', 'month_nov', 'month_oct', 'mon
th_sep',
       'day_of_week_fri', 'day_of_week_mon', 'day_
of_week_thu',
       'day_of_week_tue', 'day_of_week_wed', 'pout
come_failure',
       'poutcome_nonexistent', 'poutcome_success']
, dtype=object)
```

In [26]:

```
bank_data_vars = bank_data.columns.values.tolist()
Y = ['y']
X = [v for v in bank_data_vars if v not in Y]
```

In [27]:

```
n = 12
```

In [28]:

```
from sklearn import datasets
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
```

In [29]:

```
lr = LogisticRegression()
```

In [30]:

```
rfe = RFE(lr, n)
rfe = rfe.fit(bank_data[X], bank_data[Y].values.ravel())
```

```
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
```

```
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
```

In [31]:

```
print(rfe.support_)
```

```
[False False False False  True False False False
True False False False
 False False False  True False False False False
True False False False
 False False False False False False False False F
alse False False False
 False False False False False False  True  True
True  True  True False
  True False False False False False False  True F
alse  True]
```

In [32]:

```
print(rfe.ranking_)
```

```
[33 40 22 43  1 18 30 27  1 36  4 14  9 17 13  1 1
0 39  2  3  1 45 20 42
 32 47 41 16 46 23 38  8 25 11 12 19  7 31 24  5 3
4 26  1  1  1  1  1 21
  1 44 37 28 35 29 15  1  6  1]
```

In [33]:

```
z=zip(bank_data_vars,rfe.support_, rfe.ranking_)
```

In [34]:

```
list(z)
```

Out[34]:

```
[('age', False, 33),
 ('duration', False, 40),
 ('campaign', False, 22),
 ('pdays', False, 43),
 ('previous', True, 1),
 ('emp.var.rate', False, 18),
 ('cons.price.idx', False, 30),
 ('cons.conf.idx', False, 27),
 ('euribor3m', True, 1),
 ('nr.employed', False, 36),
 ('y', False, 4),
 ('job_admin.', False, 14),
 ('job_blue-collar', False, 9),
 ('job_entrepreneur', False, 17),
 ('job_housemaid', False, 13),
 ('job_management', True, 1),
 ('job_retired', False, 10),
 ('job_self-employed', False, 39),
 ('job_services', False, 2),
 ('job_student', False, 3),
 ('job_technician', True, 1),
 ('job_unemployed', False, 45),
 ('job_unknown', False, 20),
 ('marital_divorced', False, 42),
 ('marital_married', False, 32),
 ('marital_single', False, 47),
 ('marital_unknown', False, 41),
 ('education_Basic', False, 16),
 ('education_High School', False, 46),
 ('education_Illiterate', False, 23),
 ('education_Professional Course', False, 38),
 ('education_University Degree', False, 8),
 ('education_Unknown', False, 25),
 ('housing_no', False, 11),
 ('housing_unknown', False, 12),
 ('housing_yes', False, 19),
 ('loan_no', False, 7),
 ('loan_unknown', False, 31),
 ('loan_yes', False, 24),
```

```
('loan_yes', False, 24),
('contact_cellular', False, 5),
('contact_telephone', False, 34),
('month_apr', False, 26),
('month_aug', True, 1),
('month_dec', True, 1),
('month_jul', True, 1),
('month_jun', True, 1),
('month_mar', True, 1),
('month_may', False, 21),
('month_nov', True, 1),
('month_oct', False, 44),
('month_sep', False, 37),
('day_of_week_fri', False, 28),
('day_of_week_mon', False, 35),
('day_of_week_thu', False, 29),
('day_of_week_tue', False, 15),
('day_of_week_wed', True, 1),
('poutcome_failure', False, 6),
('poutcome_nonexistent', True, 1)]
```

In [35]:

```
cols = ["previous", "euribor3m", "job_blue-collar", "job_retir
ed", "month_aug", "month_dec",
        "month_jul", "month_jun", "month_mar", "month_nov", "d
ay_of_week_wed", "poutcome_nonexistent"]
```

In [36]:

```
X = bank_data[cols]
Y = bank_data["y"]
```

Implementación del modelo en Python con statsmodel.api

In [37]:

```
import statsmodels.api as sm
```

In [38]:

```
logit_model = sm.Logit(Y, X) ## Y es la variable que hay que p
redecir y luego las variables predictoras (X)
```

In [39]:

```
result = logit_model.fit()
```

```
Optimization terminated successfully.
         Current function value: 0.291770
         Iterations 7
```

In [40]:

```
result.summary2()
```

Out[40]:

Model:	Logit	Pseudo R-squared:	0.155
Dependent Variable:	y	AIC:	2427.6025
Date:	2019-10-28 18:54	BIC:	2503.4828
No. Observations:	4119	Log-Likelihood:	-1201.8
Df Model:	11	LL-Null:	-1422.9
Df Residuals:	4107	LLR p-value:	6.4492e-88
Converged:	1.0000	Scale:	1.0000
No. Iterations:	7.0000		

	Coef.	Std.Err.	z	P>|z|	[0.025	(
previous	-0.1229	0.0700	-1.7545	0.0793	-0.2601	0
euribor3m	-0.6049	0.0383	-15.7882	0.0000	-0.6800	-0
job_blue-collar	-0.5032	0.1519	-3.3136	0.0009	-0.8009	-0
job_retired	0.2235	0.2191	1.0205	0.3075	-0.2058	0
month_aug	0.6048	0.1759	3.4374	0.0006	0.2600	0
month_dec	1.1358	0.4493	2.5281	0.0115	0.2552	2
month_jul	1.0327	0.1910	5.4071	0.0000	0.6584	1
month_jun	1.0775	0.1752	6.1493	0.0000	0.7341	1
month_mar	1.6448	0.3139	5.2407	0.0000	1.0297	2
month_nov	0.3828	0.1950	1.9634	0.0496	0.0007	0
day_of_week_wed	-0.0649	0.1391	-0.4665	0.6409	-0.3375	0
poutcome_nonexistent	-0.7753	0.1221	-6.3492	0.0000	-1.0147	-0

Implementación del modelo en Python con scikit-learn

(Ajustando el modelo)

In [41]:

```
from sklearn import linear_model
```

In [42]:

```
logit_model = linear_model.LogisticRegression()
logit_model.fit(X,Y)
```

```
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
```

Out[42]:

```
LogisticRegression(C=1.0, class_weight=None, dual=
False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=N
one, max_iter=100,
                   multi_class='warn', n_jobs=None
, penalty='l2',
                   random_state=None, solver='warn
', tol=0.0001, verbose=0,
                   warm_start=False)
```

In [43]:

```
### Porcentaje de acertar la predicción
logit_model.score(X,Y)
```

Out[43]:

```
0.8963340616654528
```

In [44]:

```
pd.DataFrame(list(zip(X.columns, np.transpose(logit_model.coef_))))
```

Out[44]:

	0	1
0	previous	[0.5076571354371748]
1	euribor3m	[-0.5464961328095875]
2	job_blue-collar	[-0.3591553621981977]
3	job_retired	[0.3560383887828734]
4	month_aug	[0.625398308645033]
5	month_dec	[1.1822172985997683]
6	month_jul	[0.9622633627645103]
7	month_jun	[1.0543179248746157]
8	month_mar	[1.6306366297853532]
9	month_nov	[0.4519576818260996]
10	day_of_week_wed	[0.0417143385318625]
11	poutcome_nonexistent	[0.30569877121046607]

Validación del modelo logístico diviendo en conjunto de Test y Entrenamiento

In [45]:

```
from sklearn.model_selection import train_test_split
```

In [46]:

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = 0.3, random_state=0)
```

In [47]:

```
lm = linear_model.LogisticRegression()
lm.fit(X_train, Y_train)
```

```
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
```

Out[47]:

```
LogisticRegression(C=1.0, class_weight=None, dual=
False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=N
one, max_iter=100,
                   multi_class='warn', n_jobs=None
, penalty='l2',
                   random_state=None, solver='warn
', tol=0.0001, verbose=0,
                   warm_start=False)
```

In [48]:

```
from IPython.display import display, Math, Latex
```

In [49]:

```
display(Math(r'Y_p=\begin{cases}0& si\ p\leq0.5\\1&si\ p >0.5\
end{cases}'))
```

$\displaystyle Y_p=\begin{cases}0& si\ p\leq0.5\\1&si\ p >0.5\end{cases}$

In [50]:

```
probs = lm.predict_proba(X_test)
```

In [51]:

```
probs
```

Out[51]:

```
array([[0.95462912, 0.04537088],
       [0.83762689, 0.16237311],
       [0.93244632, 0.06755368],
       ...,
       [0.65044409, 0.34955591],
       [0.97383524, 0.02616476],
       [0.57021896, 0.42978104]])
```

In [52]:

```
prediction = lm.predict(X_test)
```

In [53]:

```
prediction
```

Out[53]:

```
array([0, 0, 0, ..., 0, 0, 0])
```

In [54]:

```
display(Math(r'\varepsilon\in (0,1), Y_p=\begin{cases}0& si\ p
\leq \varepsilon\\1&si\ p >\varepsilon\end{cases}'))
```

$\displaystyle \varepsilon\in (0,1), Y_p=\begin{cases}0& si\ p\leq
\varepsilon\\1&si\ p >\varepsilon\end{cases}$

In [55]:

```
prob = probs[:,1]
prob_df = pd.DataFrame(prob)
threshold = 0.1
prob_df["prediction"] = np.where(prob_df[0]>threshold, 1, 0)
prob_df.head()
```

Out[55]:

	0	prediction
0	0.045371	0
1	0.162373	1
2	0.067554	0
3	0.062144	0
4	0.041582	0

In [56]:

```
pd.crosstab(prob_df.prediction, columns="count")
##390 compradores y 846 posibles compradores
```

Out[56]:

col_0	count
prediction	
0	846
1	390

In [57]:

```
390/len(prob_df)*100
```

Out[57]:

```
31.55339805825243
```

In [58]:

```
threshold = 0.15
prob_df["prediction"] = np.where(prob_df[0]>threshold, 1, 0)
pd.crosstab(prob_df.prediction, columns="count")
```

Out[58]:

prediction	col_0 count
0	905
1	331

In [59]:

```
331/len(prob_df)*100
```

Out[59]:

```
26.779935275080906
```

In [60]:

```
threshold = 0.05
prob_df["prediction"] = np.where(prob_df[0]>threshold, 1, 0)
pd.crosstab(prob_df.prediction, columns="count")
```

Out[60]:

prediction	col_0 count
0	504
1	732

In [61]:

```
732/len(prob_df)*100
```

Out[61]:

```
59.22330097087378
```

In [62]:

```
from sklearn import metrics
```

In [63]:

```
metrics.accuracy_score(Y_test, prediction)
## Da 90%... aumentó un solo un poco por el hecho de utilizar conjunto de entrenamiento y test.
```

Out[63]:

```
0.9004854368932039
```

Validación cruzada

Para que el modelo predictivo no sufra overfitting

In [64]:

```
from sklearn.model_selection import cross_val_score
```

In [65]:

```
scores = cross_val_score(linear_model.LogisticRegression(), X, Y, scoring="accuracy", cv=10)
```

```
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
```

```
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
```

In [66]:

```
scores
```

Out[66]:

```
array([0.9031477 , 0.88834951, 0.90533981, 0.89563
107, 0.90048544,
       0.8907767 , 0.88349515, 0.89320388, 0.89537
713, 0.88807786])
```

In [67]:

```
scores.mean() #Promedio de las predicciones.
```

Out[67]:

```
0.8943884240990478
```

Matrices de Confusión y curva ROC

In [69]:

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y,test_s
ize=0.3, random_state=0)
```

In [70]:

```
lm = linear_model.LogisticRegression()
lm.fit(X_train, Y_train)
```

```
/Users/gabriel/opt/anaconda3/lib/python3.7/site-pa
ckages/sklearn/linear_model/logistic.py:432: Futur
eWarning: Default solver will be changed to 'lbfgs
' in 0.22. Specify a solver to silence this warnin
g.
  FutureWarning)
```

Out[70]:

```
LogisticRegression(C=1.0, class_weight=None, dual=
False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=N
one, max_iter=100,
                   multi_class='warn', n_jobs=None
, penalty='l2',
                   random_state=None, solver='warn
', tol=0.0001, verbose=0,
                   warm_start=False)
```

In [71]:

```
probs = lm.predict_proba(X_test)
```

In [72]:

```
prob=probs[:,1]
prob_df = pd.DataFrame(prob)
threshold = 0.1
prob_df["prediction"] = np.where(prob_df[0]>=threshold, 1, 0)
prob_df["actual"] = list(Y_test)
prob_df.head()
```

Out[72]:

	0	prediction	actual
0	0.045371	0	0
1	0.162373	1	0
2	0.067554	0	0
3	0.062144	0	0
4	0.041582	0	0

In [73]:

```
confusion_matrix = pd.crosstab(prob_df.prediction, prob_df.actual)
```

In [74]:

```
TN=confusion_matrix[0][0]
TP=confusion_matrix[1][1]
FN=confusion_matrix[0][1]
FP=confusion_matrix[1][0]
```

In [75]:

```
sens = TP/(TP+FN)
sens
```

Out[75]:

```
0.21025641025641026
```

In [76]:

```
espc_1 = 1-TN/(TN+FP)
espc_1
```

Out[76]:

```
0.047281323877068515
```

In [77]:

```
thresholds = [0.04, 0.05, 0.07, 0.10, 0.12, 0.15, 0.18, 0.20,
0.25, 0.3, 0.4, 0.5]
sensitivities = [1]
especifities_1 = [1]

for t in thresholds:
    prob_df["prediction"] = np.where(prob_df[0]>=t, 1, 0)
    prob_df["actual"] = list(Y_test)
    prob_df.head()

    confusion_matrix = pd.crosstab(prob_df.prediction, prob_df
.actual)
    TN=confusion_matrix[0][0]
    TP=confusion_matrix[1][1]
    FP=confusion_matrix[0][1]
    FN=confusion_matrix[1][0]

    sens = TP/(TP+FN)
    sensitivities.append(sens)
    espc_1 = 1-TN/(TN+FP)
    especifities_1.append(espc_1)

sensitivities.append(0)
especifities_1.append(0)
```

In [78]:

```
sensitivities
```

Out[78]:

```
[1,
 0.9344262295081968,
 0.8442622950819673,
 0.680327868852459,
 0.6721311475409836,
 0.6639344262295082,
 0.6475409836065574,
 0.5163934426229508,
 0.45901639344262296,
 0.4016393442622951,
 0.36065573770491804,
 0.1721311475409836,
 0.11475409836065574,
 0]
```

In [79]:

```
especifities_1
```

Out[79]:

```
[1,
 0.7782764811490126,
 0.5646319569120287,
 0.2989228007181328,
 0.2764811490125674,
 0.24596050269299818,
 0.22621184919210058,
 0.12387791741472176,
 0.1077199281867145,
 0.07181328545780974,
 0.06463195691202872,
 0.02333931777378817,
 0.013464991023339312,
 0]
```

In [80]:

```
import matplotlib.pyplot as plt
```

In [81]:

```
%matplotlib inline
plt.plot(especifities_1, sensitivities, marker="o", linestyle=
"--", color="r")
x=[i*0.01 for i in range(100)]
y=[i*0.01 for i in range(100)]
plt.plot(x,y)
plt.xlabel("1-Especifidad")
plt.ylabel("Sensibilidad")
plt.title("Curva ROC")
```

Out[81]:

```
Text(0.5, 1.0, 'Curva ROC')
```