MATH2319 Machine Learning Project Phase 1
Predicting Wheather the client will subscribe a term deposit

Name: Junaid Ahmed Syed
Student ID: s3731300

May 22, 2019

Contents

Chapter 1

Introduction

1.1 Objective

The objective of this project is to predict whether the client subscribed a term deposit. The data sets were sourced from the UCI Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/Bank+MarketingUCI [1]. This project has two phases. Phase I focused on data preprocessing and exploration, as covered in this report. We shall present a model building in Phase II. The rest of this report is organized as follows. Section 2 describes the data sets and their attributes. Section 3 covers data pre-processing. In Section 4, we explore each attribute and their inter-relationships. The last section is to present a brief summary.

1.2 Data Set

The UCI Machine Learning Repository has 3 datasets, but only bank-additional-full.csv is useful for this project. This data set has 45211 observations and 17 variables.

1.2.1 Target Feature

The classification goal is to predict if the client will subscribe (yes/no) a term deposit (variable y).

$$TargetFeature = \begin{cases} Yes & \text{if } clientwillsubscribeatermdeposit = True \\ No & \text{if } clientwillnotsubscribeatermdeposit = Flase \end{cases}$$

1.2.2 Descriptive Features

1 - age (numeric)

2 - job: type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')

3 - marital: marital status (categorical: 'divorced','married','single','unknown')

4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','

5 - default: has credit in default? (categorical: 'no','yes','unknown')

6 - housing: has housing loan? (categorical: 'no','yes','unknown')

7 - loan: has personal loan? (categorical: 'no','yes','unknown')

8 - contact: contact communication type (categorical: 'cellular','telephone')

9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')

10 - day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')

11 - duration: last contact duration, in seconds (numeric)

12 - campaign: number of contacts performed during this campaign and for this client (numeric)

13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)

14 - previous: number of contacts performed before this campaign and for this client (numeric)

15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')

16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)

17 - cons.price.idx: consumer price index - monthly indicator (numeric)

18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)

19 - euribor3m: euribor 3 month rate - daily indicator (numeric)

20 - nr.employed: number of employees - quarterly indicator (numeric)

Among all the descriptive features, only the first 11 rows represent clients data and the last five rows are related to social and economic context attributes.

Chapter 2

Data Pro-processsing

2.1 Preliminaries

In the first place, we need to confirm that the feature type matched the description as outlined in the document.

```
In [1]: import pandas as pd
        Bank=pd.read_csv('bank-additional-full.csv', sep=';')
```

2.1.1 Data Cleaning and Transformation

```
In [2]: print("Dimension of the data set is{Bank.shape}\n")
        print("Data Types are:")
        print(Bank.dtypes)
```

```
Dimension of the data set is{Bank.shape}

Data Types are:
age                 int64
job                object
marital            object
education          object
default            object
housing            object
loan               object
contact            object
month              object
day_of_week        object
duration            int64
campaign            int64
pdays               int64
previous            int64
poutcome           object
emp.var.rate      float64
cons.price.idx    float64
cons.conf.idx     float64
euribor3m         float64
nr.employed       float64
y                  object
dtype: object
```

On surface, no attributes containNaNvalues (though the missing values might be coded with different-labels) as shown in the code chunk.

```
In [3]: print("\nNumber of missing value for each feature:")
        print(Bank.isnull().sum())
```

```
Number of missing value for each feature:
age               0
job               0
marital           0
education         0
default           0
housing           0
loan              0
contact           0
month             0
day_of_week       0
duration          0
campaign          0
pdays             0
previous          0
poutcome          0
emp.var.rate      0
cons.price.idx    0
cons.conf.idx     0
euribor3m         0
nr.employed       0
y                 0
dtype: int64
```

Table 1 shows Summary of continuous features of both int and float values whereas Table 2 shows Summary of categorical features. From Table 2, The cardinality of target feature is binary.

```
In [4]: from IPython.display import display, HTML
        display(HTML('<b>Table 1: Summary of continuous features</b>'))
        display(Bank.describe(include='float64'))
        display(Bank.describe(include='int64'))


        display(HTML('<b>Table 2: Summary of categorical (object) features</b>'))
        display(Bank.describe(include='object'))
```

```
<IPython.core.display.HTML object>
```

	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed
count	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000
mean	0.081886	93.575664	-40.502600	3.621291	5167.035911
std	1.570960	0.578840	4.628198	1.734447	72.251528
min	-3.400000	92.201000	-50.800000	0.634000	4963.600000
25%	-1.800000	93.075000	-42.700000	1.344000	5099.100000
50%	1.100000	93.749000	-41.800000	4.857000	5191.000000
75%	1.400000	93.994000	-36.400000	4.961000	5228.100000
max	1.400000	94.767000	-26.900000	5.045000	5228.100000

```
                age      duration      campaign         pdays      previous
count  41188.00000  41188.000000  41188.000000  41188.000000  41188.000000
mean      40.02406    258.285010      2.567593    962.475454      0.172963
std       10.42125    259.279249      2.770014    186.910907      0.494901
min       17.00000      0.000000      1.000000      0.000000      0.000000
25%       32.00000    102.000000      1.000000    999.000000      0.000000
50%       38.00000    180.000000      2.000000    999.000000      0.000000
75%       47.00000    319.000000      3.000000    999.000000      0.000000
max       98.00000   4918.000000     56.000000    999.000000      7.000000
```

```
<IPython.core.display.HTML object>
```

```
           job  marital          education default housing   loan   contact  \
count    41188    41188              41188   41188   41188  41188     41188
unique      12        4                  8       3       3      3         2
top     admin.  married  university.degree      no     yes     no  cellular
freq     10422    24928              12168   32588   21576  33950     26144

        month day_of_week     poutcome      y
count   41188       41188        41188  41188
unique     10           5            3      2
top       may         thu  nonexistent     no
freq    13769        8623        35563  36548
```

2.2 Continuous Features

The variable duration should be removed because this attribute highly affects the output target. We can notice a good amount of outliers in all the continuous features. Since Outliers can be a great source of information, I decided to include them in my predictive analytics model. However, I need to choose techniques and statistical methods that excel at handling outliers without influencing the analysis

```
In [5]: Bank=Bank.drop(['duration'],1)
```

2.3 Categorical Features

None of the categorical features has whitespaces in it.

Chapter 3

Data Exploration

3.1 Univariate Visualisation

Two definitions are defined named BarPlot(x) and BoxHistogramPlot(x)for categorical and numerical features respectively for the sake of simplicity. For given an input categorical column, BarPlot(x)returns a bar chart with the percentage on top of each bar. A bar chart is useful to present the proportions by categories. Forgiven an input numerical column, BoxHistogramPlot(x)plots a histogram and a box plot. A histogram is are useful to visualize the shape of the underlying distribution whereas A box plot tells the range of the attribute and helps detect any outliers. The following chunk codes show how these functions were defined using the numpy library and the matplotlib library.

```
In [6]: import matplotlib.pyplot as plt
        import seaborn as sns
        sns.set(color_codes=True)
        def BarPlot(x):
            total=float(len(Bank))
            ax=Bank[x].value_counts(normalize=True).plot(kind="bar", alpha=0.5)
            def BoxHistogramPlot(x):
                f, (ax_box, ax_hist)=plt.subplots(2, sharex=True,gridspec_kw={"height_ratios": (.15,.85)
                sns.boxplot(x, ax=ax_box)
                sns.distplot(x, ax=ax_hist)
                ax_box.set(yticks=[])
                sns.despine(ax=ax_hist)
                sns.despine(ax=ax_box, left=True)
                plt.show()
```

In figure 1, admin and the blue collar is the most popular jobs whereas the least number of jobs were mostly either not specified or a student. In figure 2 uncovers that more than 50% of the clients are married. Figure 3 shows that most of the clients have at least a university degree with a negligible number of illiterates. From figures 4,5,6,7 we can see that nearly 70 % no credit in default and more than half of the clients have a housing loan but only 20 % have a personal loan and also it can be seen that most of them are preferring a cellular phone over a telephone. Figures 8,9,10 tells us that the last contract period was mostly May and Thursday with inconsistency in the outcome of the previous marketing campaign. Finally, from the target feature, the most common outcome is that the client is not subscribed a term deposit.

```
In [7]: i=1# initialize figure labelling4)
        for col in['job','marital','education','default','housing','loan','contact','month','day_of_week
            plt.figure(figsize=(6,2))
            plt.title("Figure"+str(i)+": Bar Chart of "+col, fontsize=19)
            BarPlot(col)
```

```
plt.show()
i=i+1
```

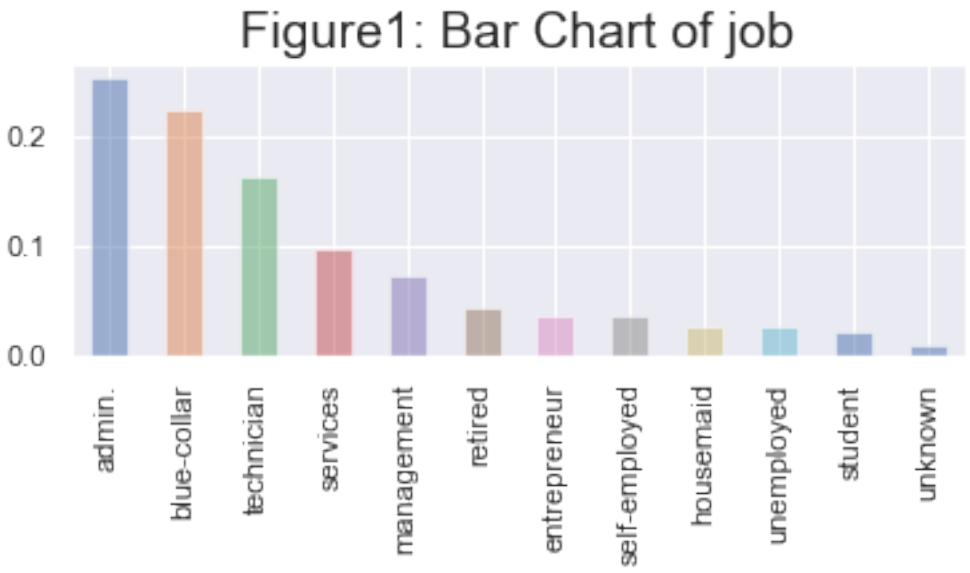


Figure1: Bar Chart of job
0.2
0.1
0.0
admin.
blue-collar
technician
services
management
retired
entrepreneur
self-employed
housemaid
unemployed
student
unknown

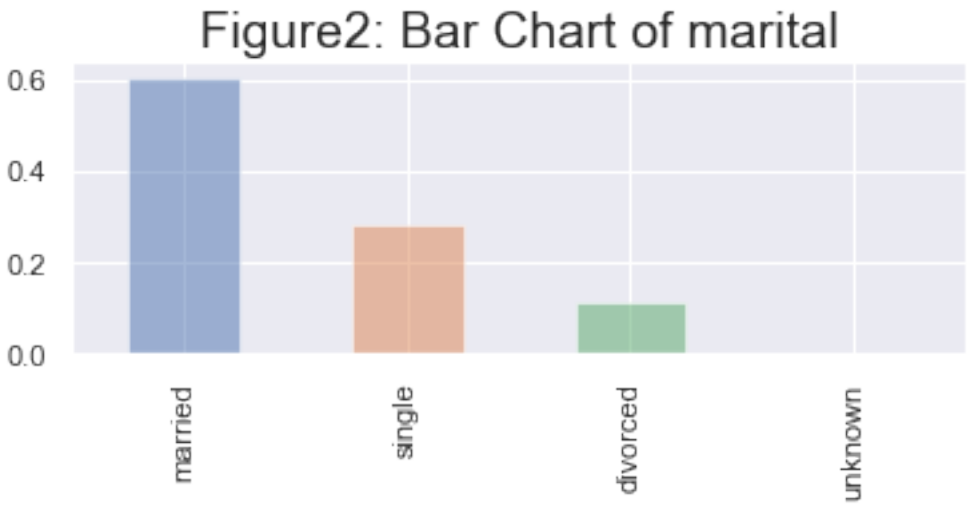


Figure2: Bar Chart of marital
0.6
0.4
0.2
0.0
married
single
divorced
unknown

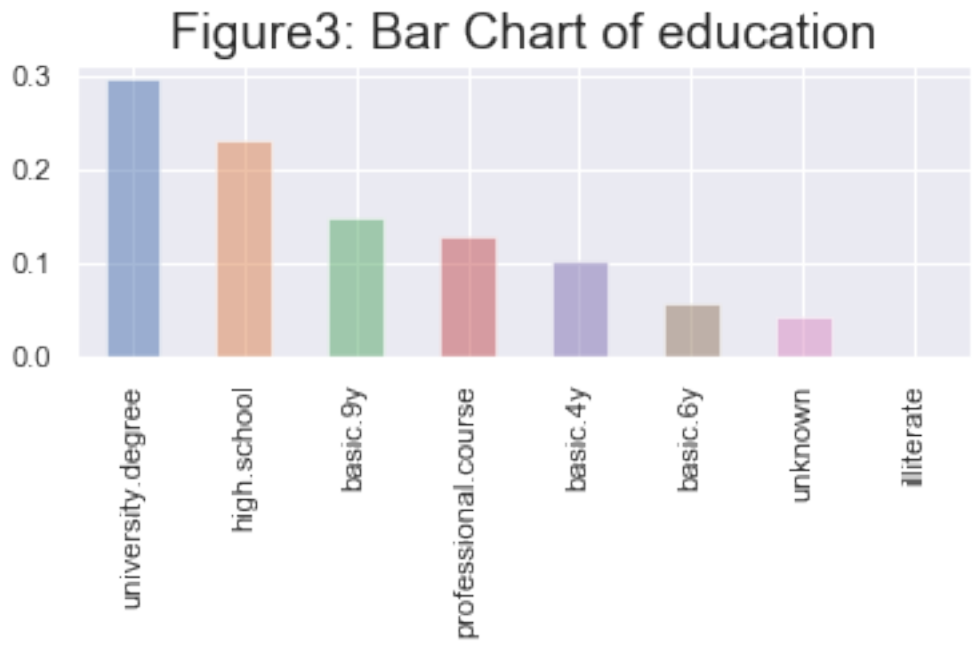


Figure3: Bar Chart of education
0.3
0.2
0.1
0.0
university.degree
high.school
basic.9y
professional.course
basic.4y
basic.6y
unknown
illiterate

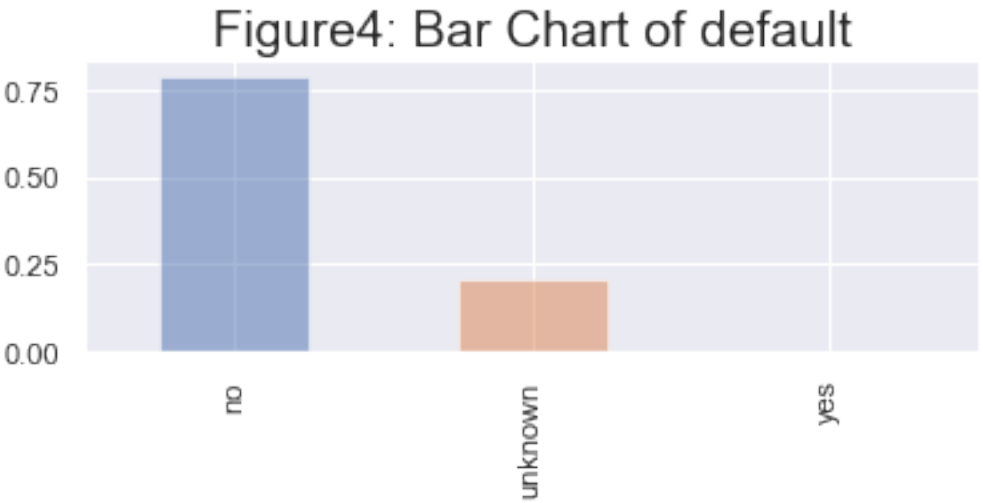


Figure4: Bar Chart of default
0.75
0.50
0.25
0.00
no
unknown
yes

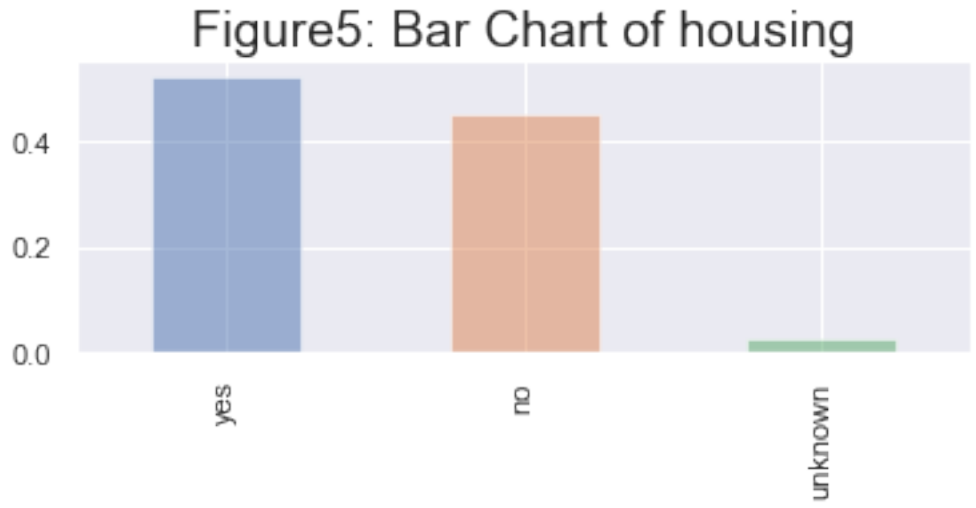


Figure5: Bar Chart of housing
0.4
0.2
0.0
yes
no
unknown

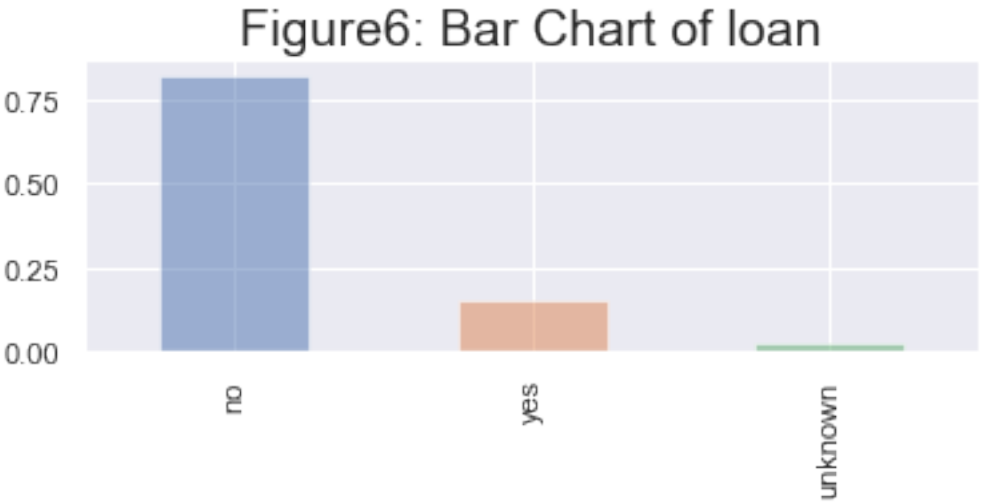


Figure6: Bar Chart of loan
0.75
0.50
0.25
0.00
no
yes
unknown

Figure7: Bar Chart of contact

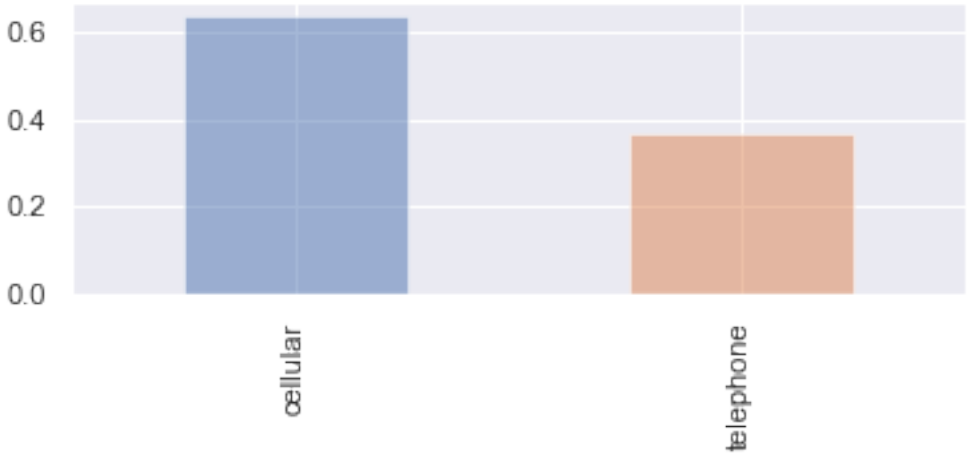


Figure8: Bar Chart of month

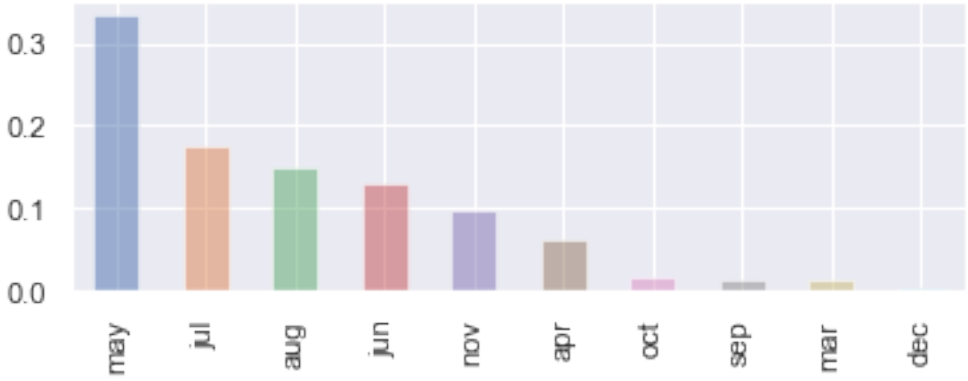
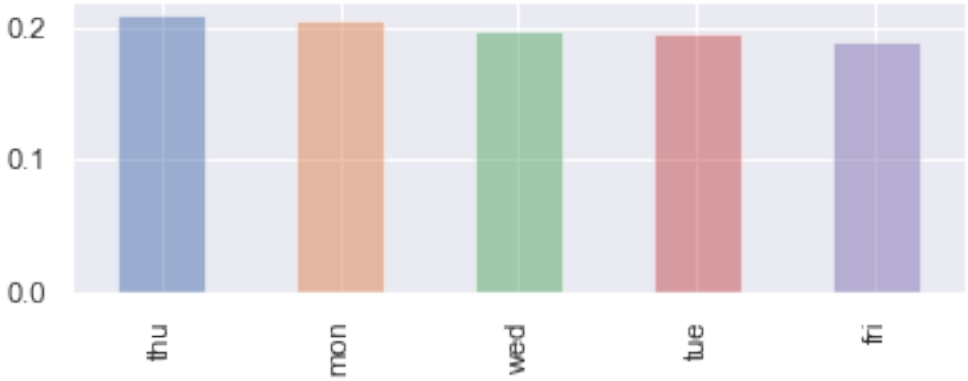
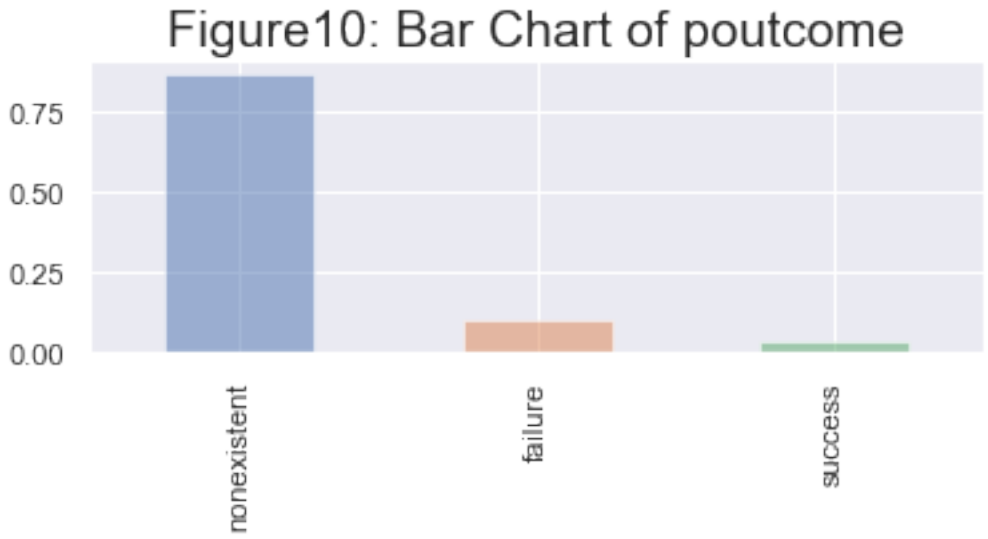
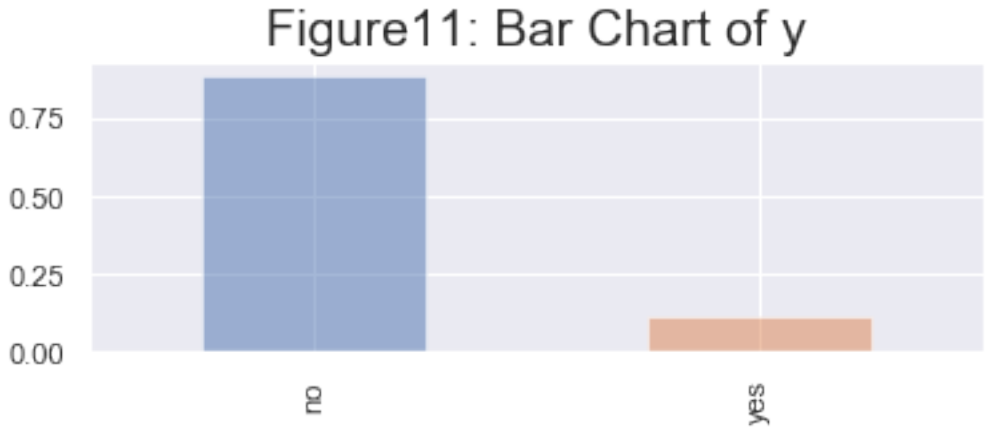


Figure9: Bar Chart of day_of_week







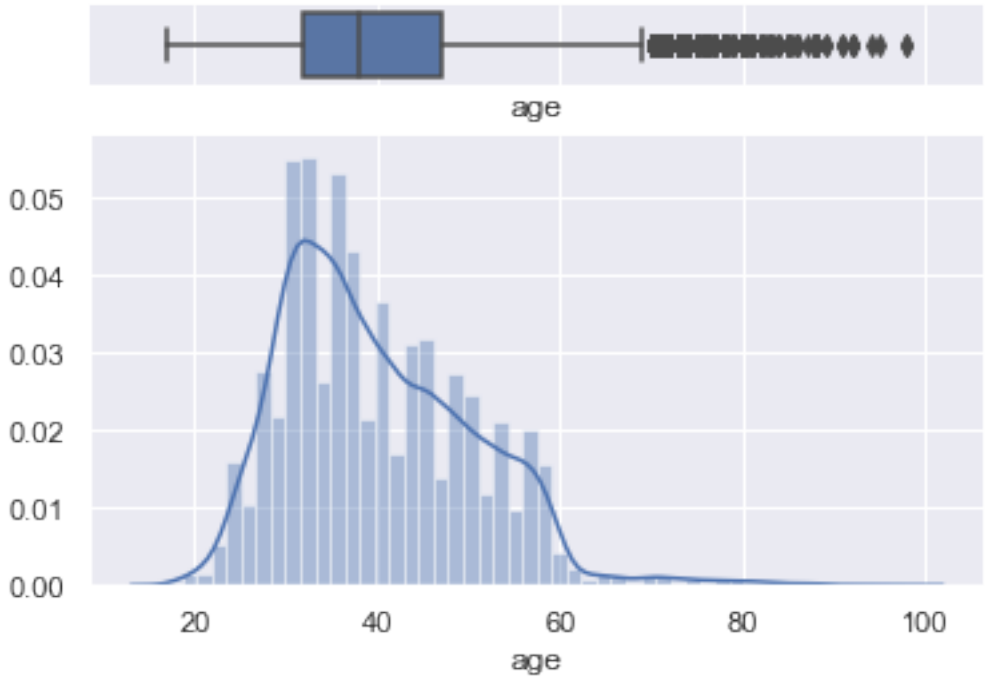
The BoxHistogramPlot function was applied to all numeric variables, to investigate the numeric columns like age, duration, campaign, pdays, previous. None of these features is perfectly normal. The most common age of the clients is around between 30 and 50 and has left skewed distribution as we can see from the histogram. similarly, the campaign attribute which is the number of contacts performed during this campaign has a left-skewed distribution and has the highest number of outliers among all. No clear relation can be seen in pdays with the most common value of 999 which means the client was not previously contacted. Likewise, there is no quite a relation with previous attribute because the common value is 0. Due to limited number of pages in the report, we have picked out a few variabes to state our point

```
In [8]: def BoxHistogramPlot(x):
            f, (ax_box, ax_hist)=plt.subplots(2, sharex=True,gridspec_kw={"height_ratios": (.15,.85)
            sns.boxplot(x, ax=ax_box)
            sns.distplot(x, ax=ax_hist)
            ax_box.set(yticks=[])
```

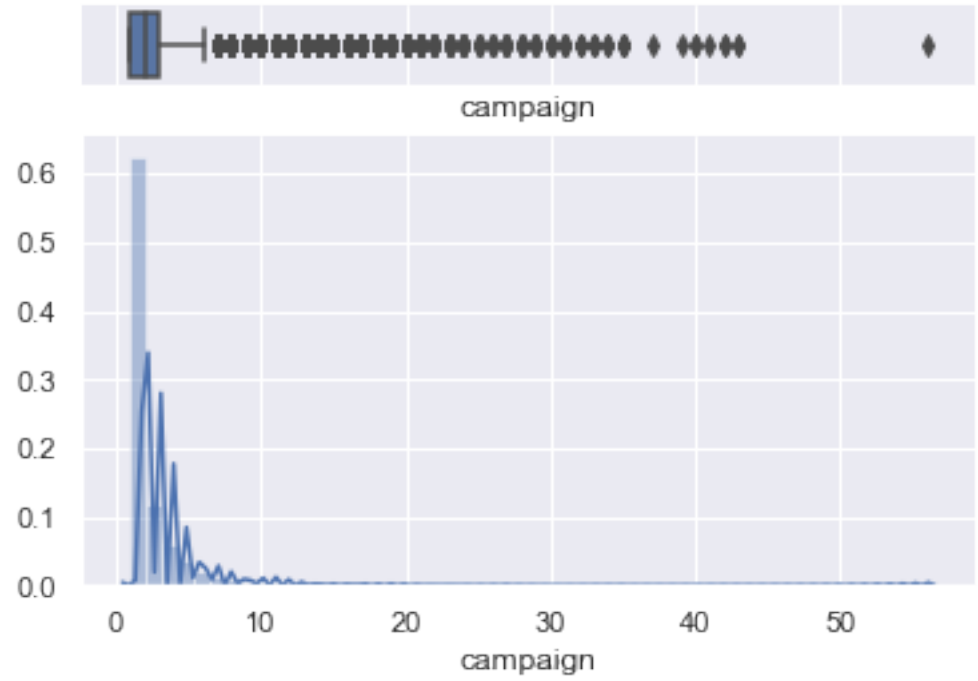
```
            sns.despine(ax=ax_hist)
            sns.despine(ax=ax_box, left=True)
            plt.show()
    for col in['age','campaign','pdays','previous']:
        plt.suptitle("Figure"+str(i)+": Histogram and Box Plot of" +col)
        BoxHistogramPlot(Bank[col])
        plt.show()
        i=1+i
```

```
/Users/junaid/anaconda2/lib/python2.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a n
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

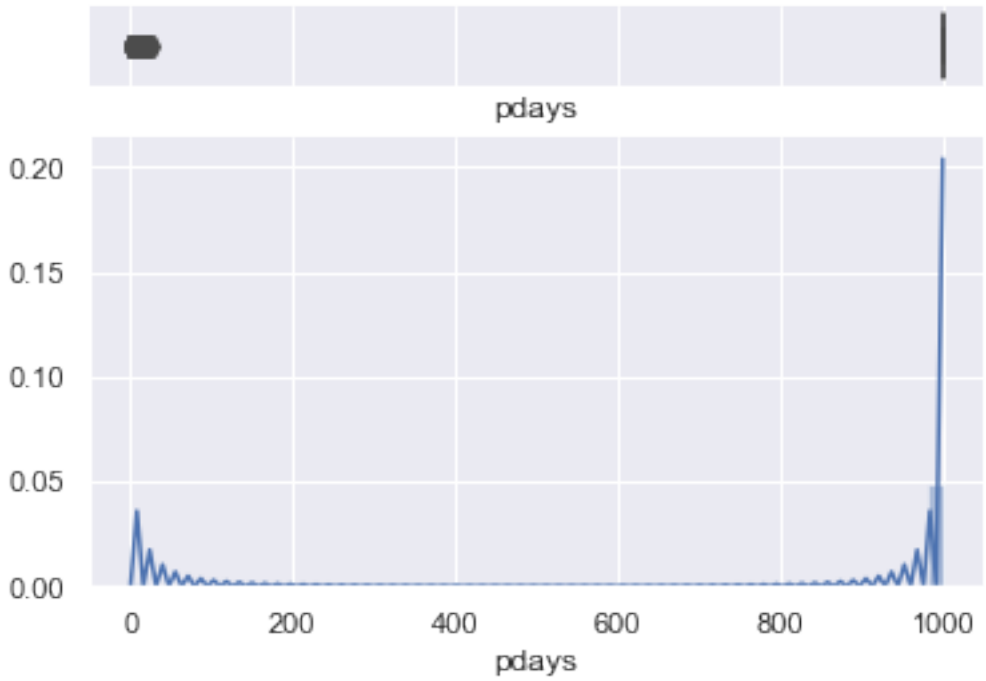
```
<Figure size 432x288 with 0 Axes>
```



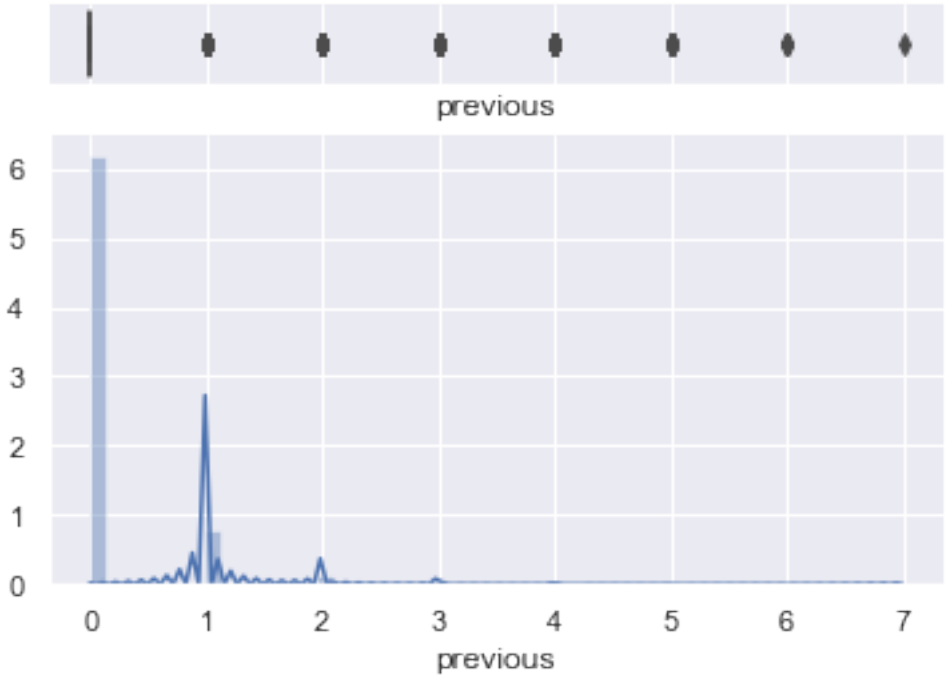

```
<Figure size 432x288 with 0 Axes>
```




```
<Figure size 432x288 with 0 Axes>
```



```
<Figure size 432x288 with 0 Axes>
```



3.2 Multivariate Visualisation

3.2.1 Histogram of Numeric Features Segregated by term deposit

The following are histograms for each numeric attributes segregated by term deposit level. figure 10 shows that the negatively skewed for both clients who subscribed and not subscribed to a term deposit. Similarly, the campaign has the same type of skewness as the age attribute. As expected the client was not previously contacted has more chance of subscribing to the term deposit.

```
In [9]: for col in['age','campaign','pdays','previous']:
            data1=Bank.loc[Bank['y']=="yes", col]
            data2=Bank.loc[Bank['y']=="no", col]
            plt.hist(data1, alpha=0.3, bins=30)
            plt.hist(data2, alpha=0.3, bins=30)
            plt.title("Figure"+str(i)+": Histogram of " +col+ " segregated by term deposit level")
            i=i+1
            plt.legend(Bank['y'].unique(), bbox_to_anchor=(1.05,1), loc=2, borderaxespad=0.)
            plt.show()
```

Figure16: Histogram of age segregated by term deposit level

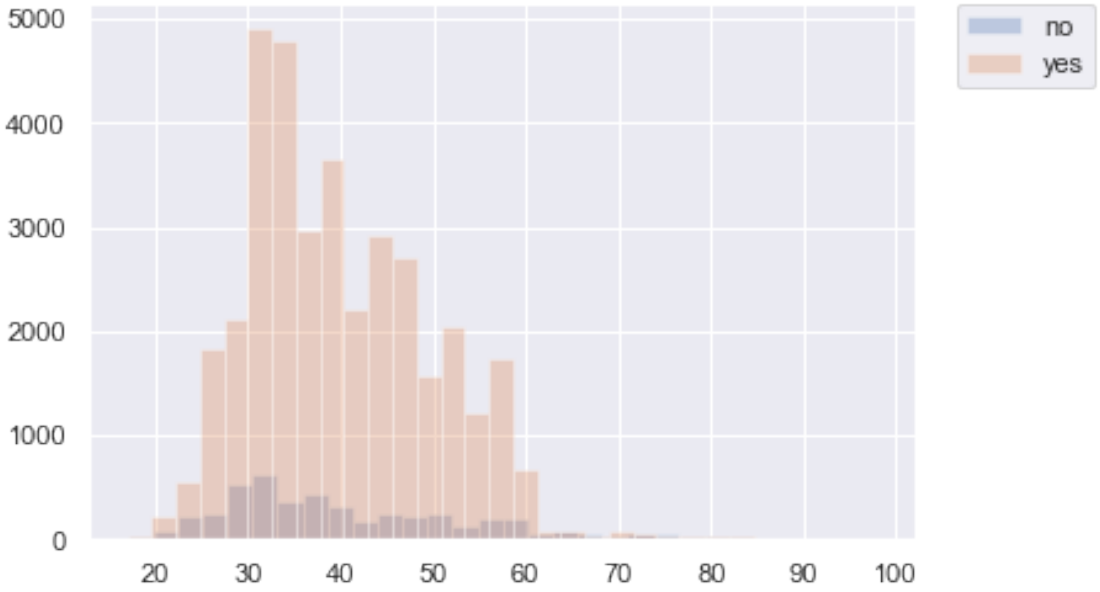


Figure17: Histogram of campaign segregated by term deposit level

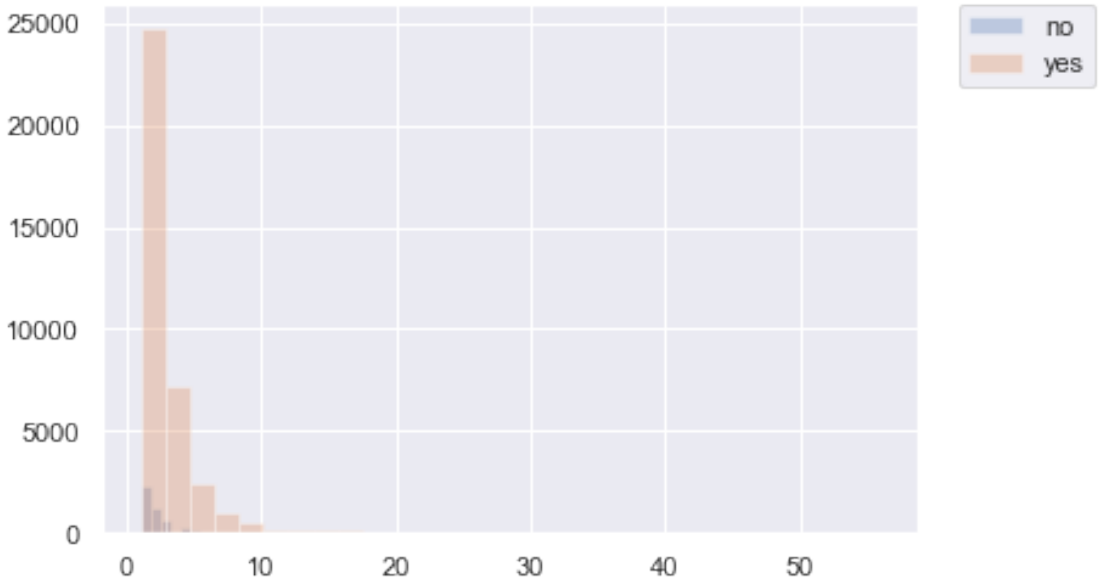


Figure18: Histogram of pdays segregated by term deposit level

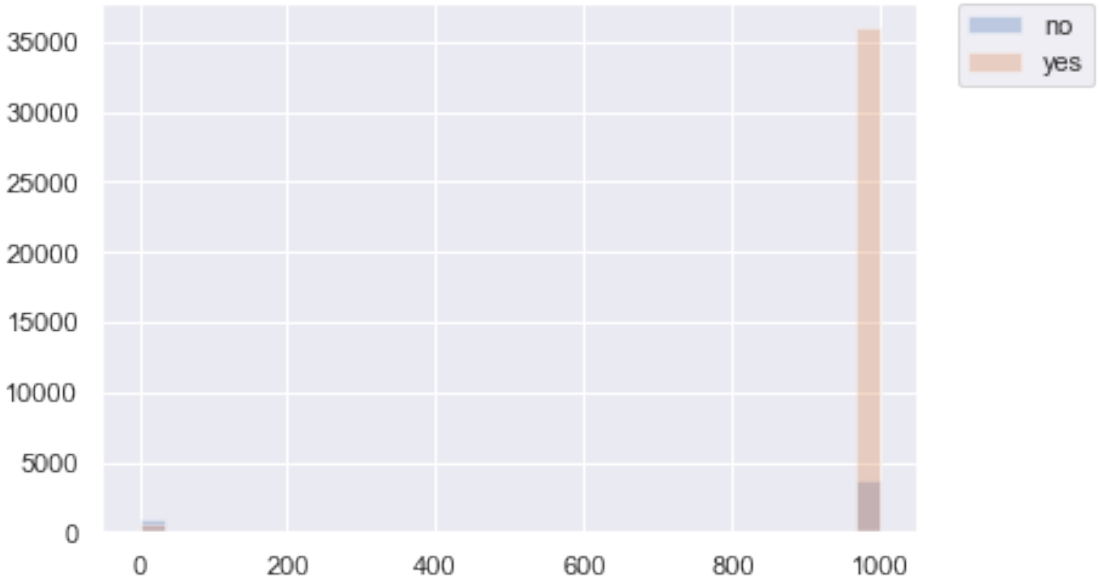
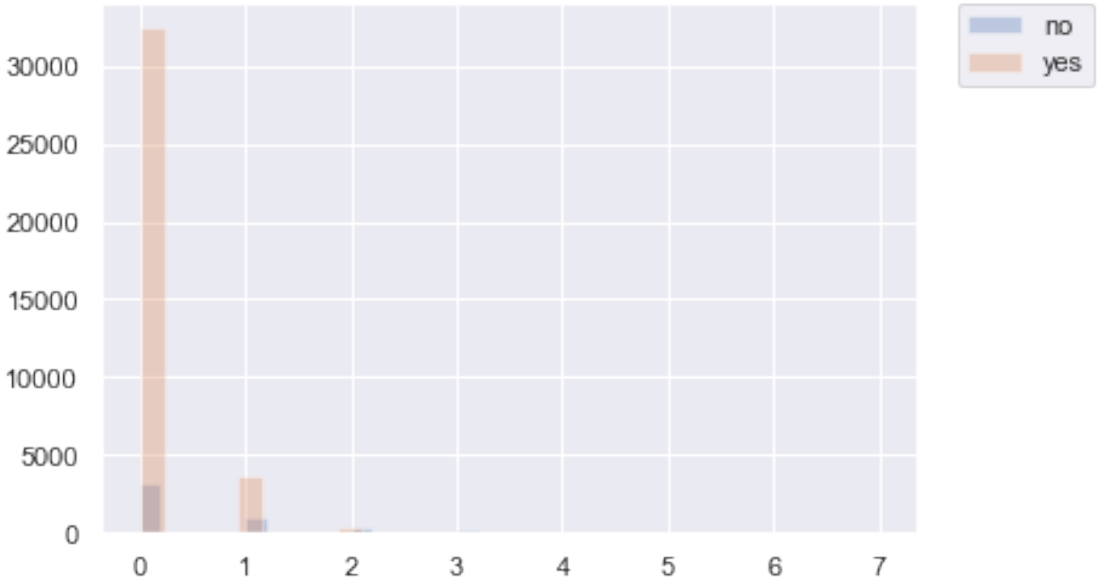


Figure19: Histogram of previous segregated by term deposit level



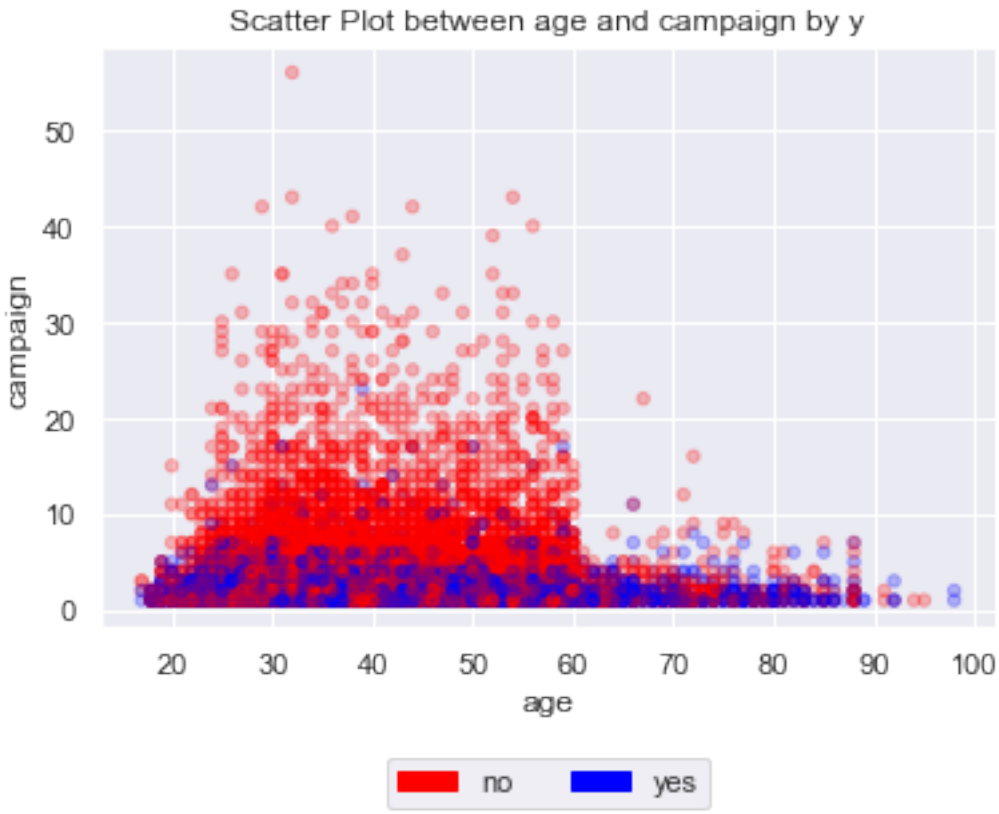
3.2.2 Pairwise Scatter Plots between Two Numeric Features by Term deposit Level

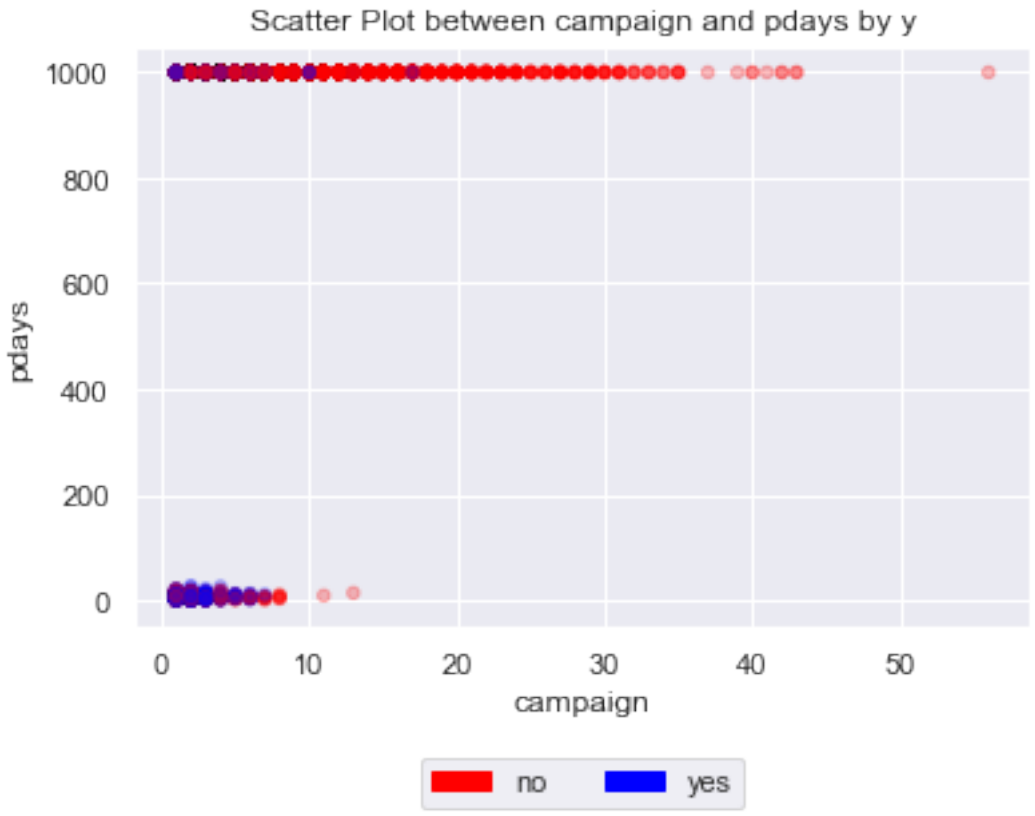
A function named scatterplotByCategory(c, x, y, D) is designed to draw a scatterplot between two numeric attributes y and x labelled by a categorical attribute c given an input data D. In the case, D is Bank and c is the term deposit level. The codes below show the details of this function. Among all these graphs we can notice the graph with age and campaign has good correlation and remaining scatterplots show no clear

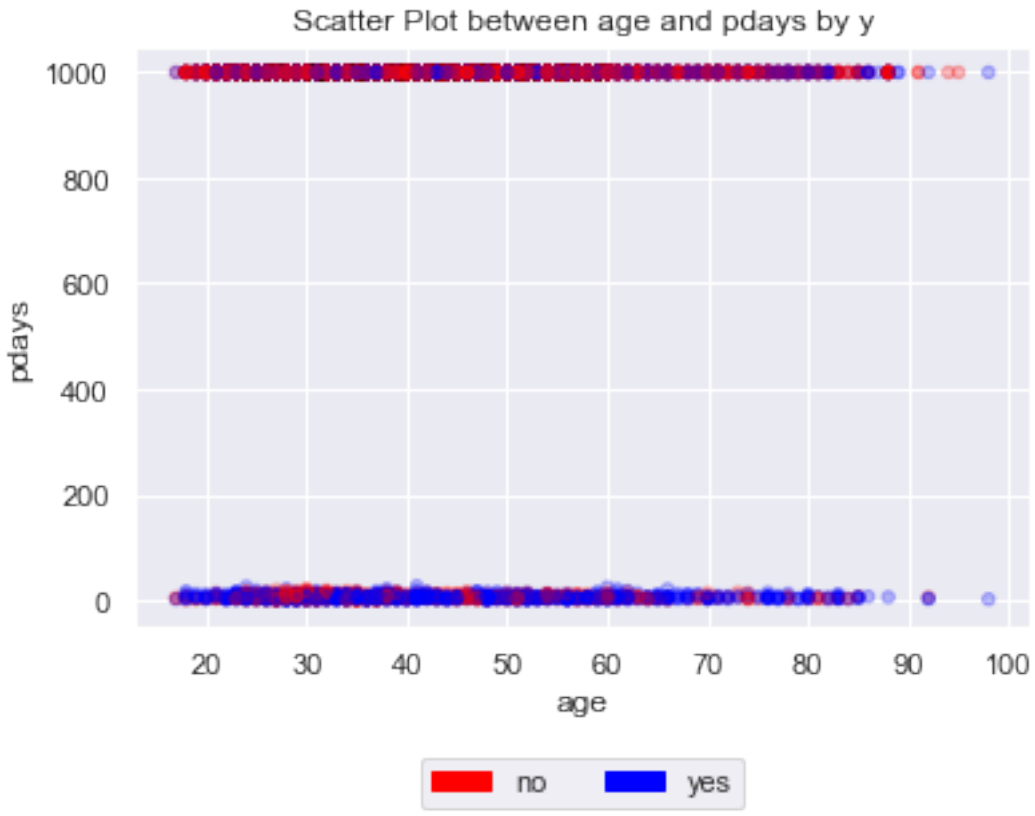
correlation between any two numeric variables. Therefore, numeric features are likely to be independent which each other.

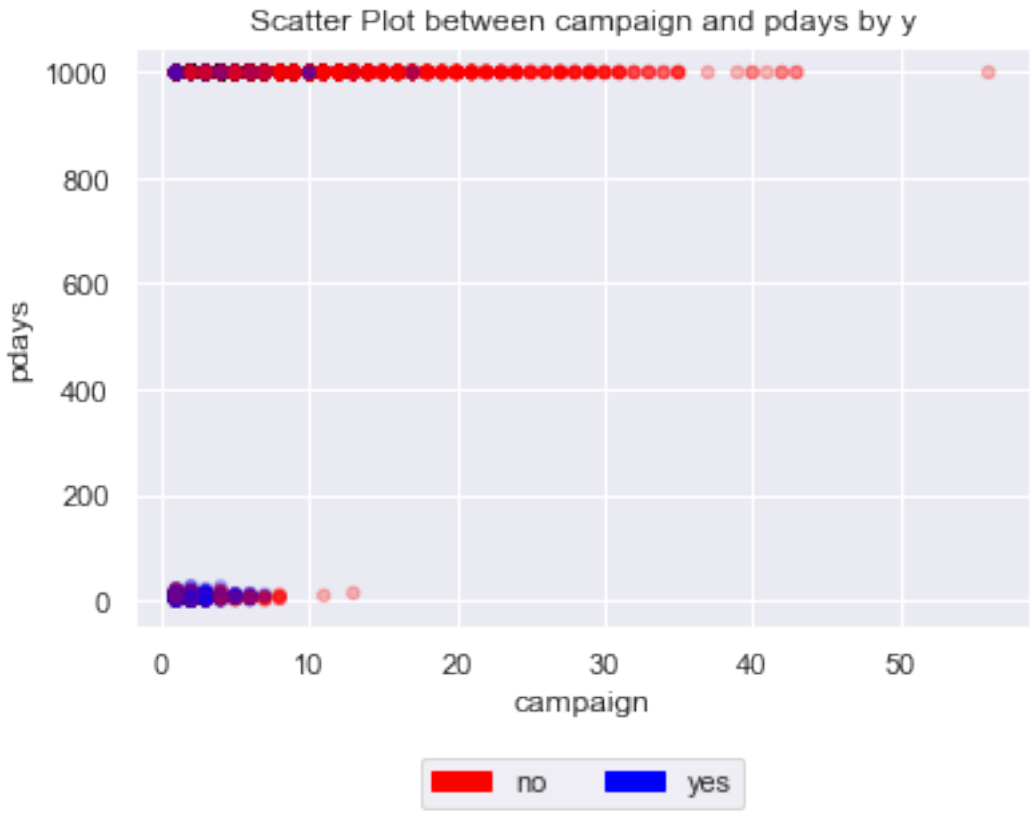
```
In [10]: import matplotlib.patches as mpatches
         def scatterplotByCategory(c, x, y, D):
             data=D
             v=data[c].unique()
             m_1=data[c]==v[0]
             m_2=data[c]==v[1]
             data.loc[m_1, c]=0
             data.loc[m_2, c]=1
             data[c].value_counts()
             colors_palette = {0:'red',1:'blue'}
             colors = [colors_palette[i] for i in data[c]]
             red_patch=mpatches.Patch(color='red', label=v[0])
             blue_patch=mpatches.Patch(color='blue', label=v[1])
             data[[x, y]].plot(kind='scatter', x=0, y=1, c=colors, alpha=0.25)
             plt.title('Scatter Plot between ' +x+ ' and ' +y+ ' by ' +c)
             plt.legend(loc=9, handles=[red_patch, blue_patch],bbox_to_anchor=(0.5,-0.2), ncol=2)
             plt.show()# return the original string  values
             n_1=data[c]==0
             n_2=data[c]==1
             data.loc[n_1, c]=v[0]
             data.loc[n_2, c]=v[1]

In [11]: for col in['campaign','pdays']:
             scatterplotByCategory('y','age', col , Bank)
             for col in['pdays']:
                 scatterplotByCategory('y','campaign', col , Bank)
```


Scatter Plot between age and campaign by y
campaign
0
10
20
30
40
50
20
30
40
50
60
70
80
90
100
age
no
yes


Scatter Plot between campaign and pdays by y
1000
800
600
400
200
0
pdays
0
10
20
30
40
50
campaign
no
yes

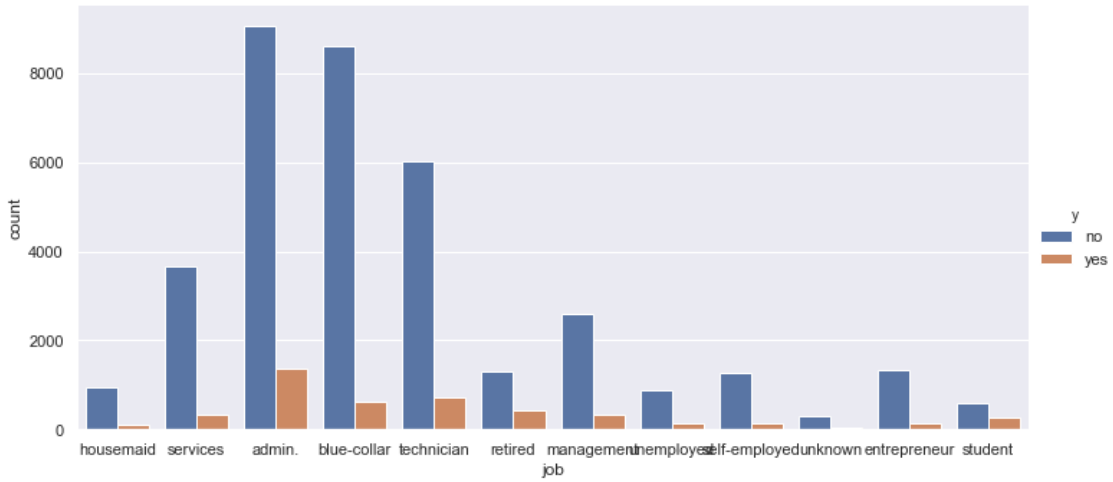

Scatter Plot between age and pdays by y
1000
800
600
400
200
0
pdays
20
30
40
50
60
70
80
90
100
age
no
yes



3.3 Categorical Attributes Segregated by Term deposit Level

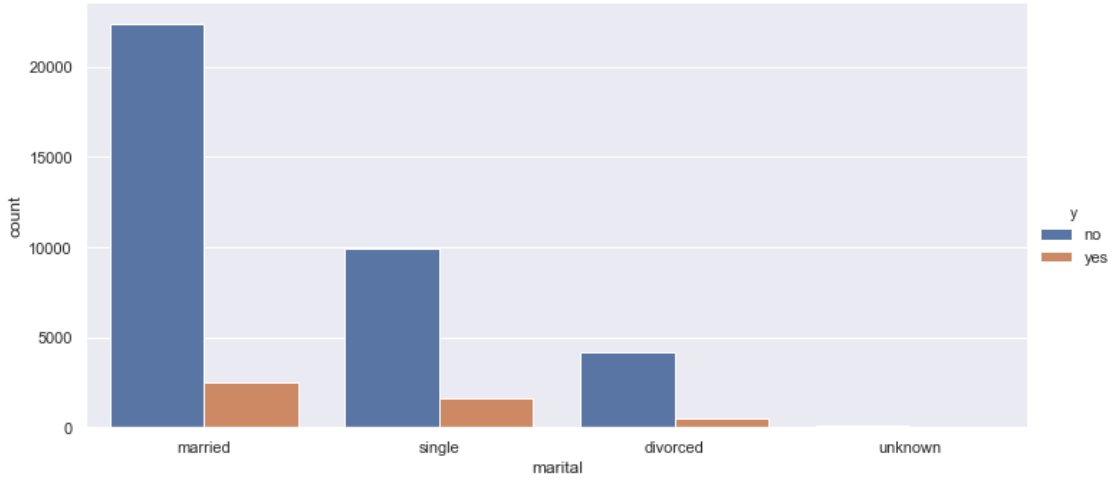
```
In [12]: import seaborn as sns
         sns.set(style="darkgrid")

         g = sns.catplot(x="job", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



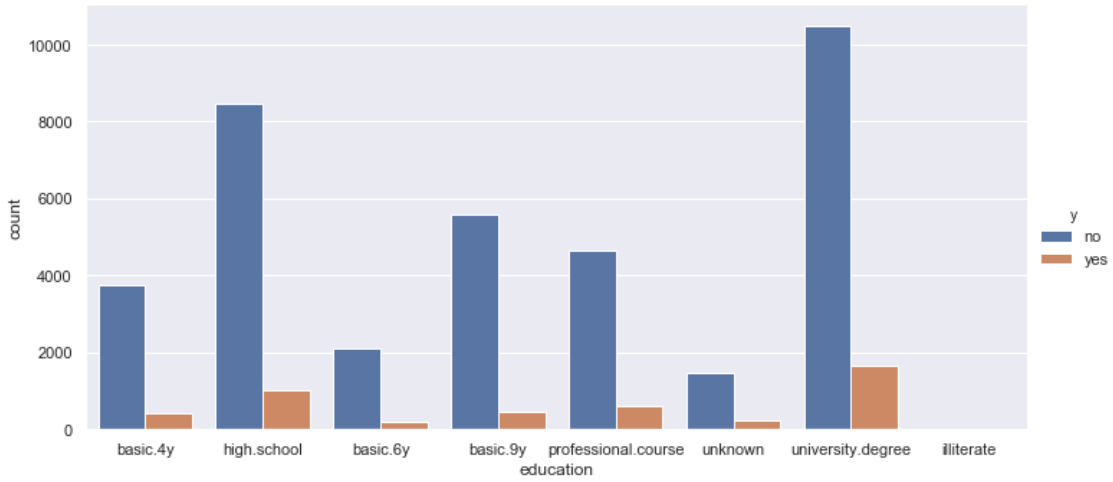
admin jobs are most popular jobs and has a highest change to subscribing and also not subscribing.

```
In [13]: g = sns.catplot(x="marital", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



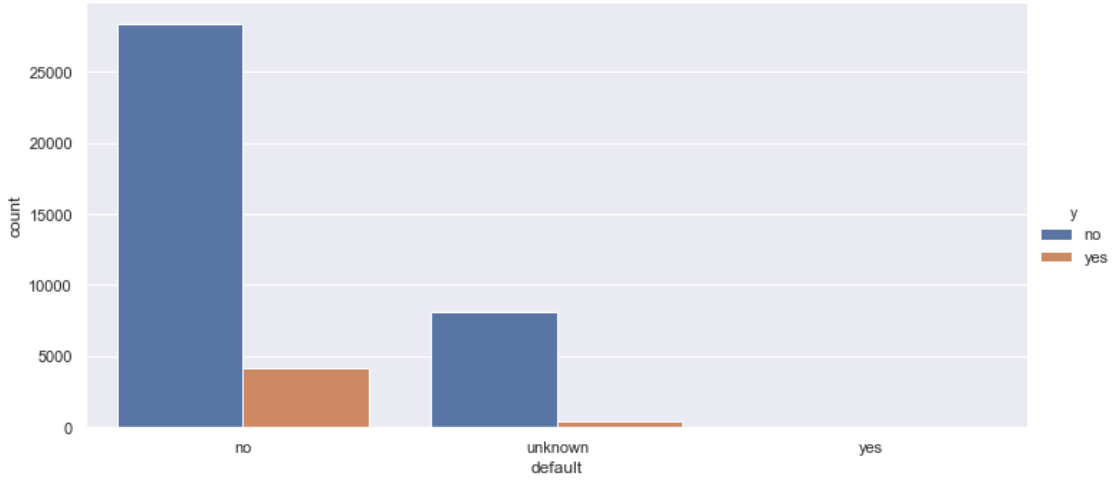
Distribution is almost same for the client irrespective of his marital status.

```
In [14]: g = sns.catplot(x="education", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



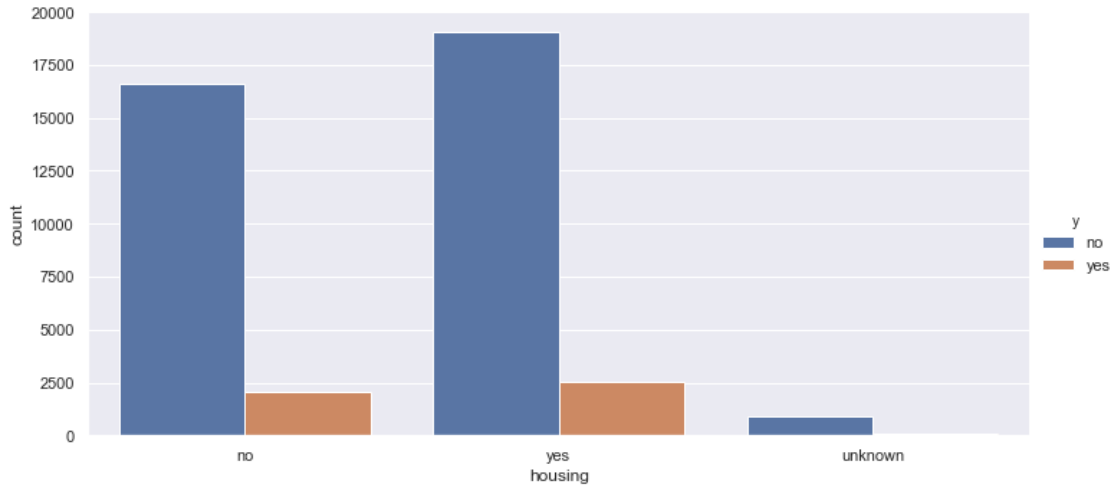
Looks like the client who has a higher degree education is less likely to have a term deposit subscription.

```
In [15]: g = sns.catplot(x="default", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



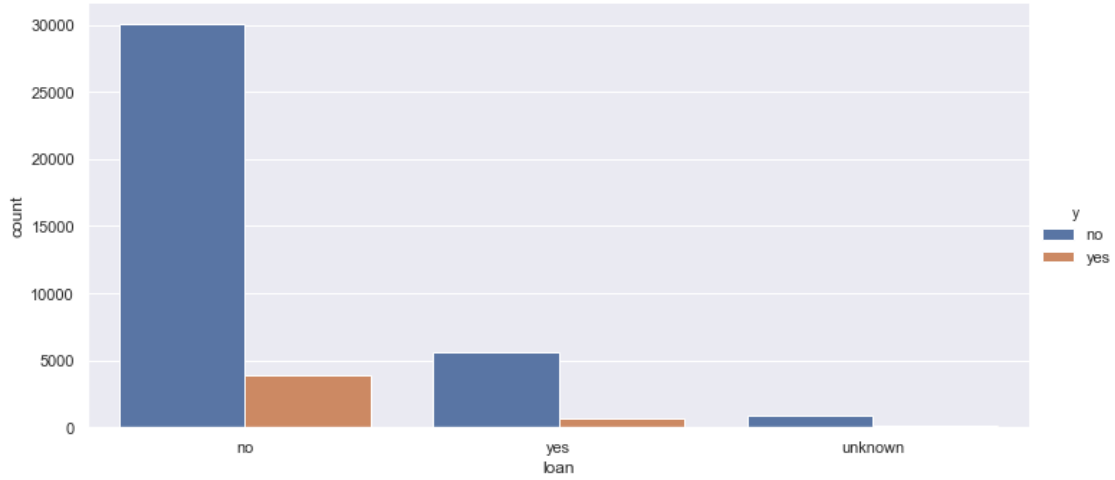
The person who doesn't have credit default has 70% for not subscribing to a term deposit. But the client with unknown credit default has a 95% chance of not subscribing.

```
In [16]: g = sns.catplot(x="housing", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



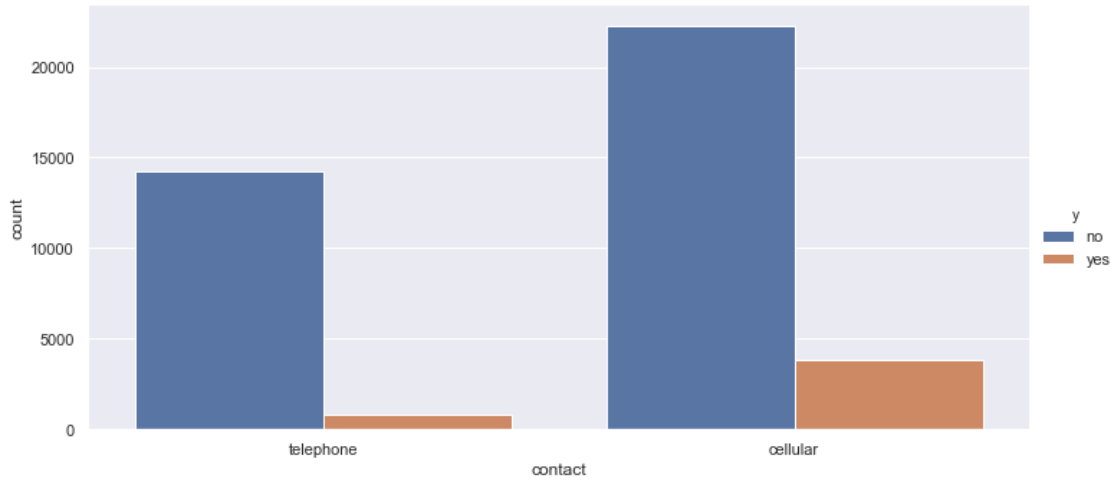
The clients who are having a housing loan has less chance of having subscribed to a term deposit.

```
In [17]: g = sns.catplot(x="loan", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



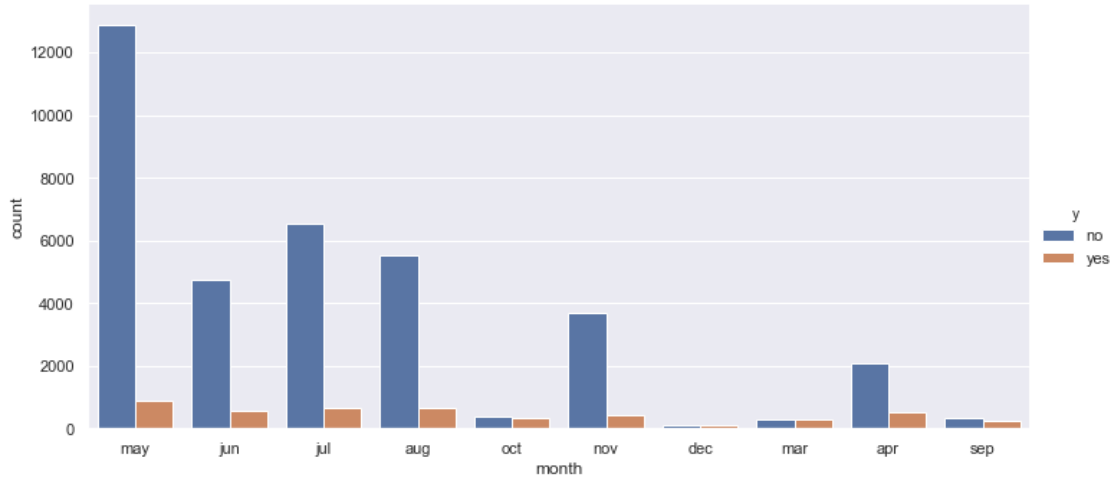
The clients who don't have a personal loan also has a less chance of having subscribed to a term deposit.

```
In [18]: g = sns.catplot(x="contact", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



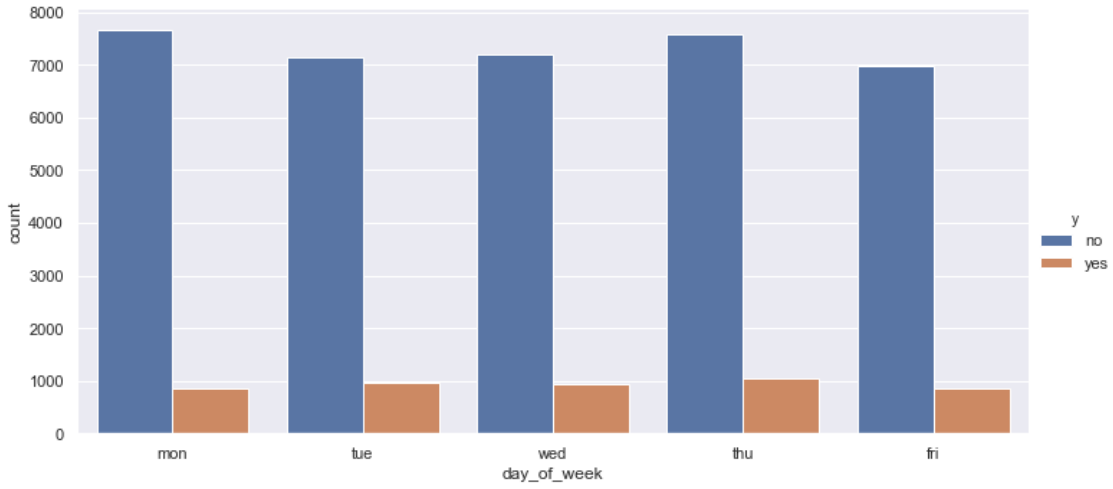
Apparently, the clients who have a cellular network is more likely to not have a term deposit subscription.

```
In [19]: g = sns.catplot(x="month", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



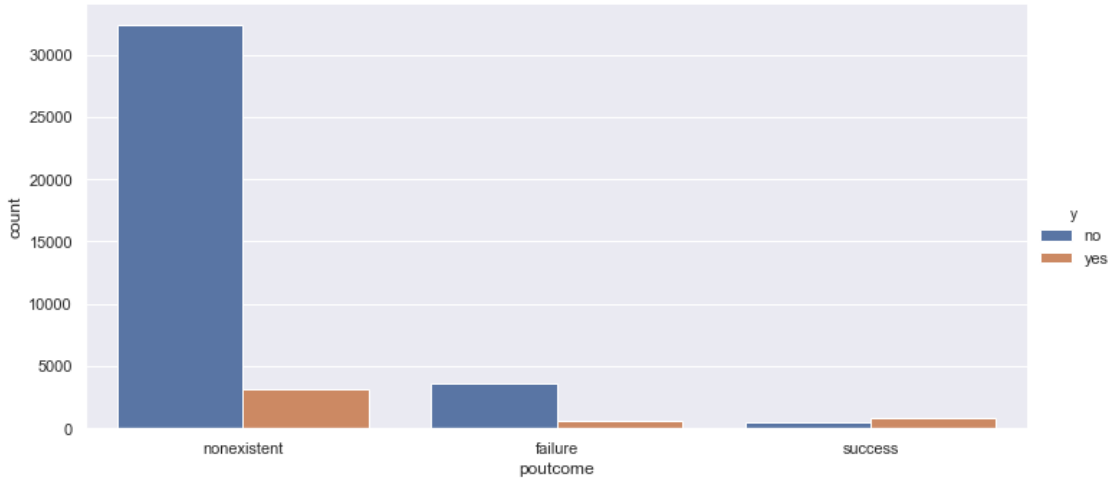
It could be a coincidence that the clients who have may as last month for the contract, is not subscribed to a term deposit.

```
In [20]: g = sns.catplot(x="day_of_week", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```



We can see that no matter which day in a week was last contact day. The client has more chance of not having a term deposit.

```
In [21]: g = sns.catplot(x="poutcome", hue="y",
                         data=Bank, kind="count",
                          height=5, aspect=2);
```

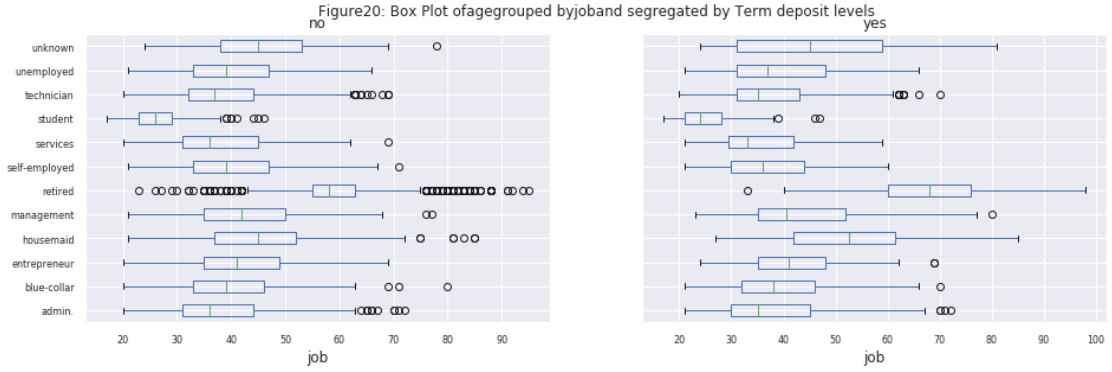


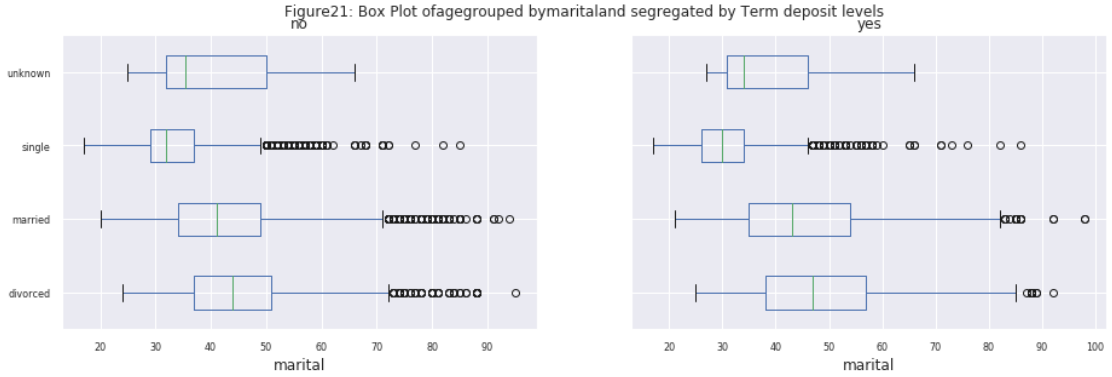
If the outcome of the previous marketing campaign is non-existent that client has more chance for not subscribing the term deposit

3.4 Interaction between Categorical and Numeric Features

In the grouped boxplots segregated by term deposit levels below, We can see that most the clients who have the age between 35 to 35 has term deposit subscription and those people are more educated. The boxplot "disappears" between pdays and jobs and also with previous and martial indicating our initial suspicion that these variables carry have similar information. So, we decided to remove pdays and previous attributes.

```
In [22]: plt.rcParams["font.family"]="DejaVu Sans"
         for col in['age','campaign','pdays','previous','emp.var.rate','cons.price.idx','cons.conf.idx'
             for k in['job','marital','education','default','housing','loan']:
                 ax=Bank.groupby('y').boxplot(column=col, by=k,vert=False,fontsize=8,figsize=(15,4.5))
                 plt.suptitle("Figure"+str(i)+": Box Plot of"+col+"grouped by"+k+"and segregated by Ter
                 plt.yticks()
                 plt.show()
                 i=1+i
```





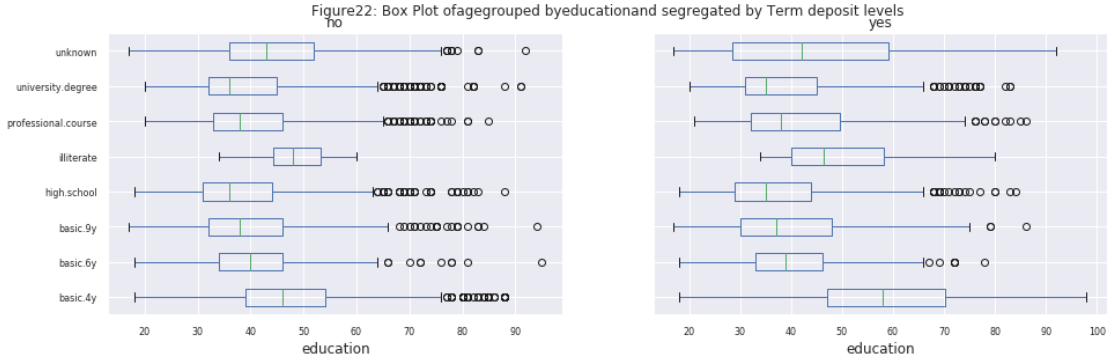


Figure23: Box Plot ofagegrouped bydefaultand segregated by Term deposit levels

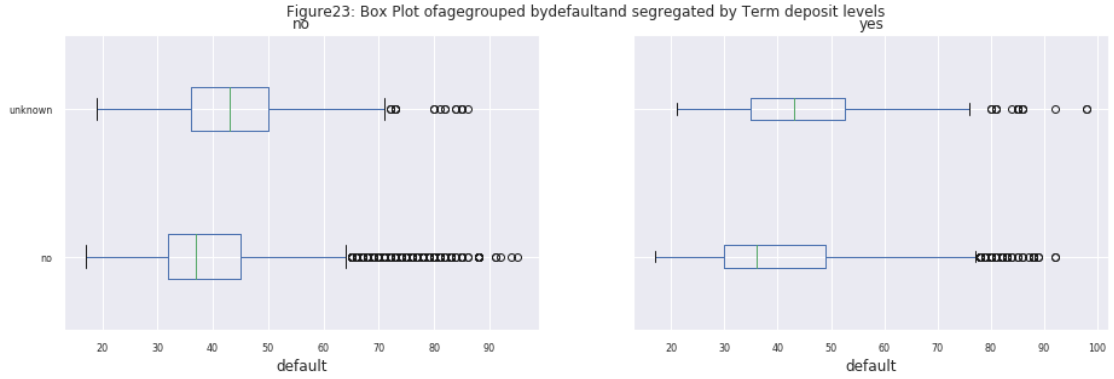


Figure24: Box Plot ofagegrouped byhousingand segregated by Term deposit levels

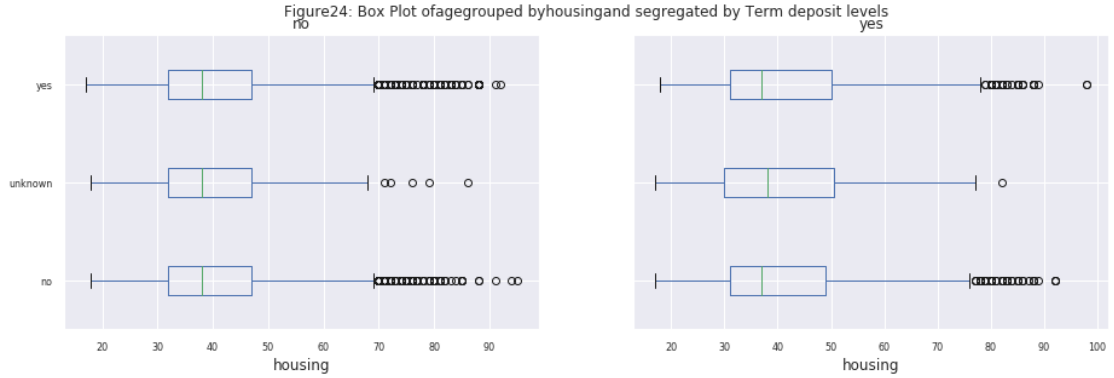


Figure25: Box Plot ofagegrouped byloanand segregated by Term deposit levels

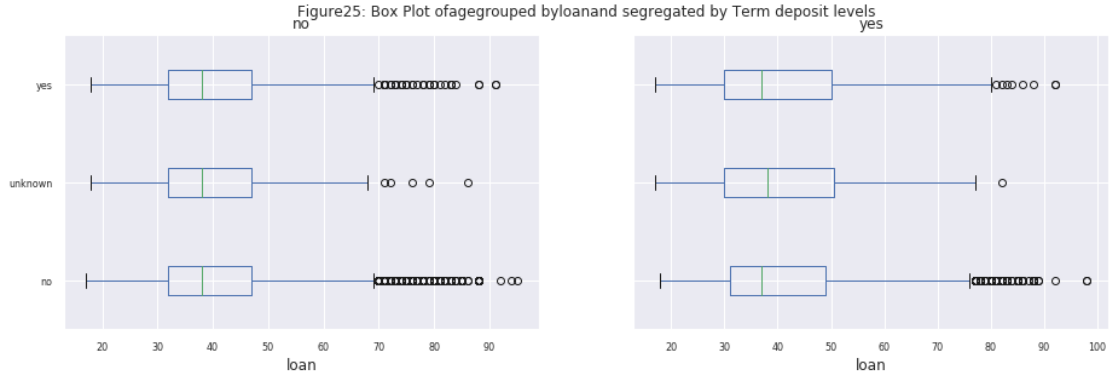


Figure26: Box Plot ofcampaigngrouped byjoband segregated by Term deposit levels

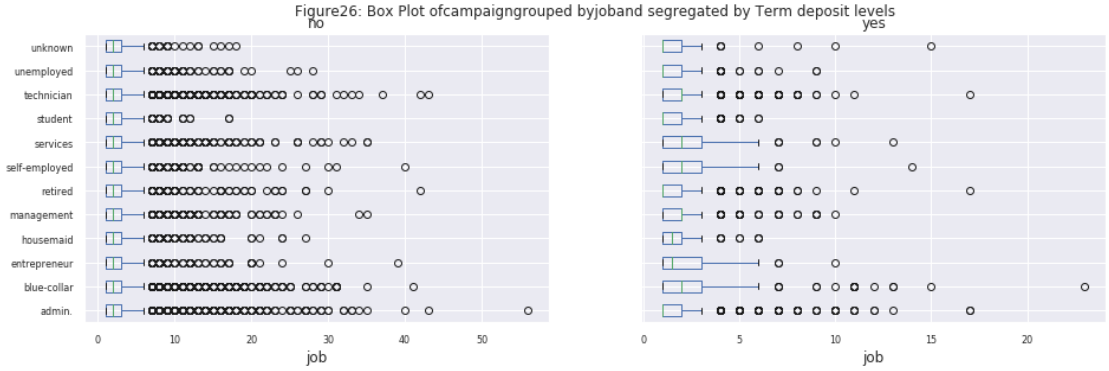


Figure27: Box Plot ofcampaigngrouped bymaritaland segregated by Term deposit levels

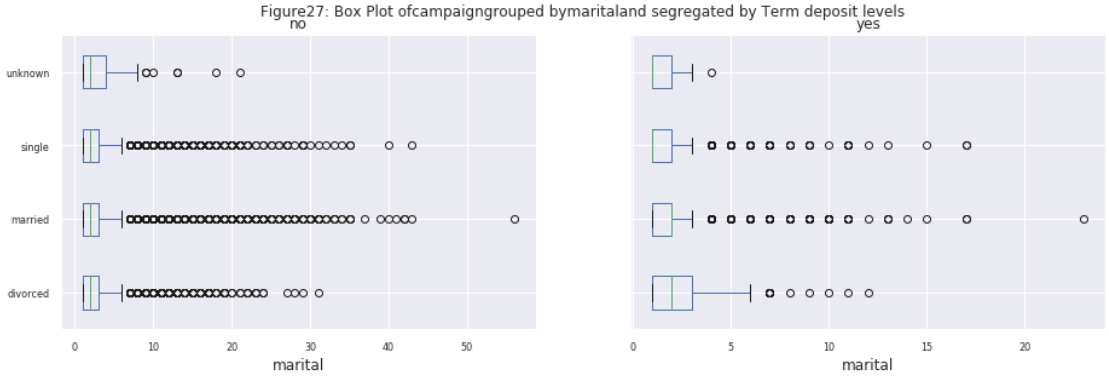


Figure28: Box Plot ofcampaigngrouped byeducationand segregated by Term deposit levels

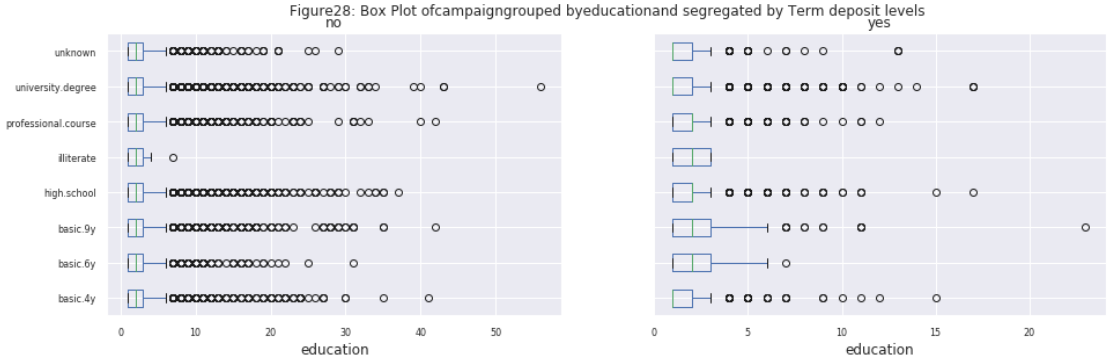


Figure29: Box Plot ofcampaigngrouped bydefaultand segregated by Term deposit levels

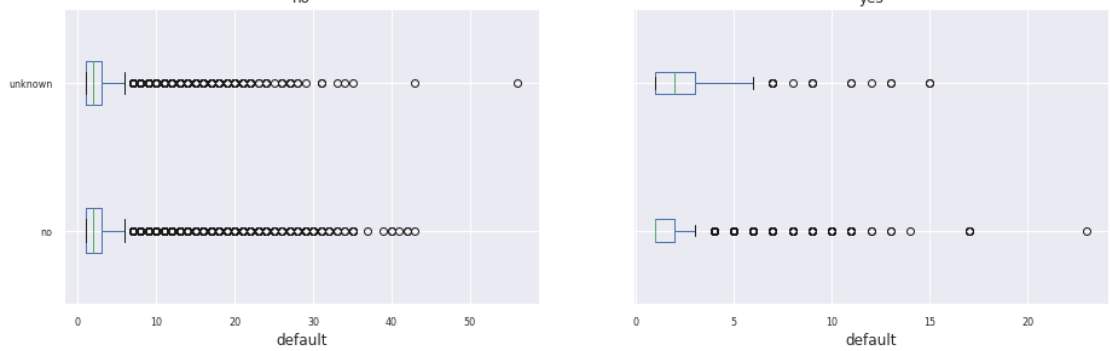


Figure30: Box Plot ofcampaigngrouped byhousingand segregated by Term deposit levels

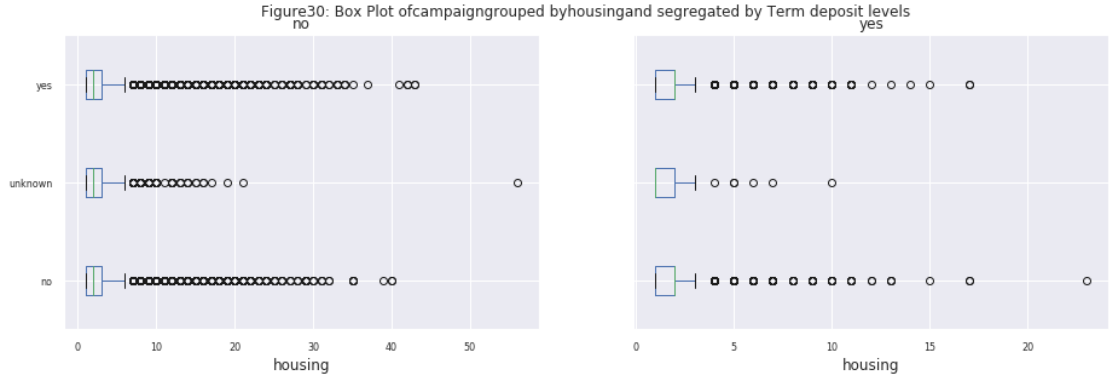


Figure31: Box Plot ofcampaigngrouped byloanand segregated by Term deposit levels

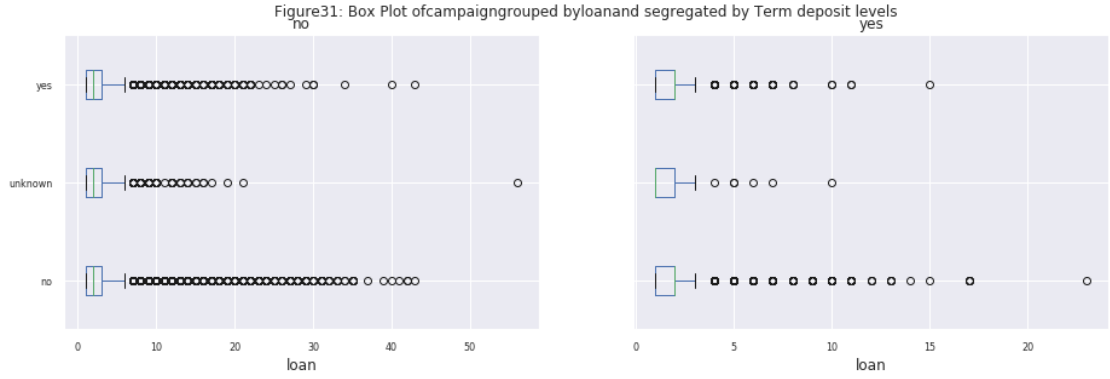


Figure32: Box Plot ofpdaysgrouped byjoband segregated by Term deposit levels

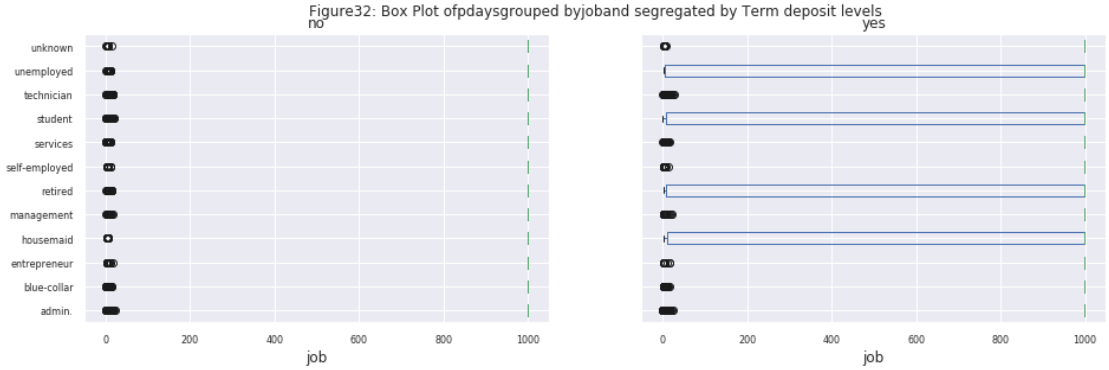


Figure33: Box Plot ofpdaysgrouped bymaritaland segregated by Term deposit levels

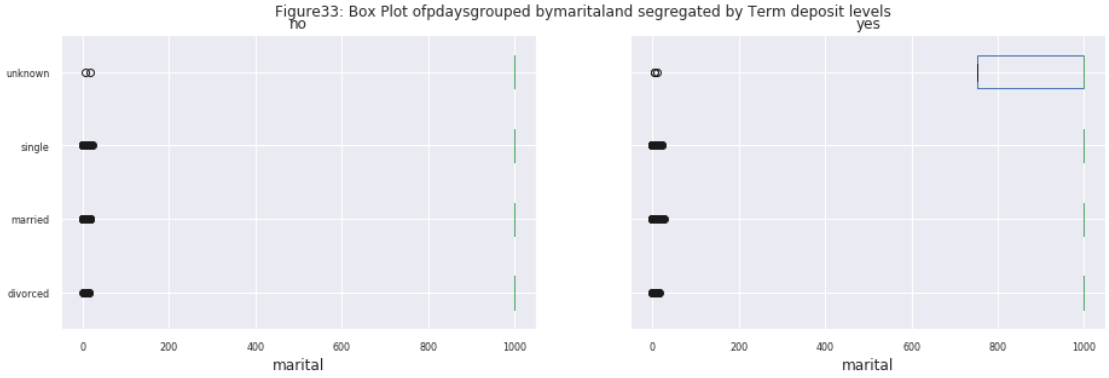


Figure34: Box Plot ofpdaysgrouped byeducationand segregated by Term deposit levels

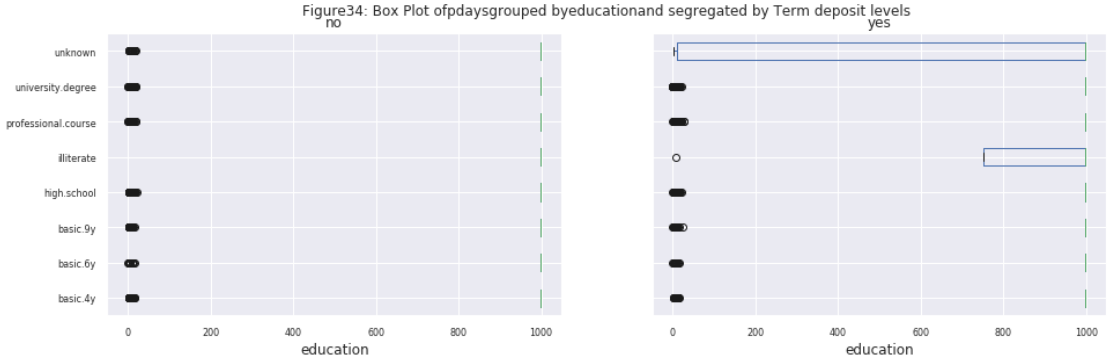


Figure35: Box Plot ofpdaysgrouped bydefaultand segregated by Term deposit levels

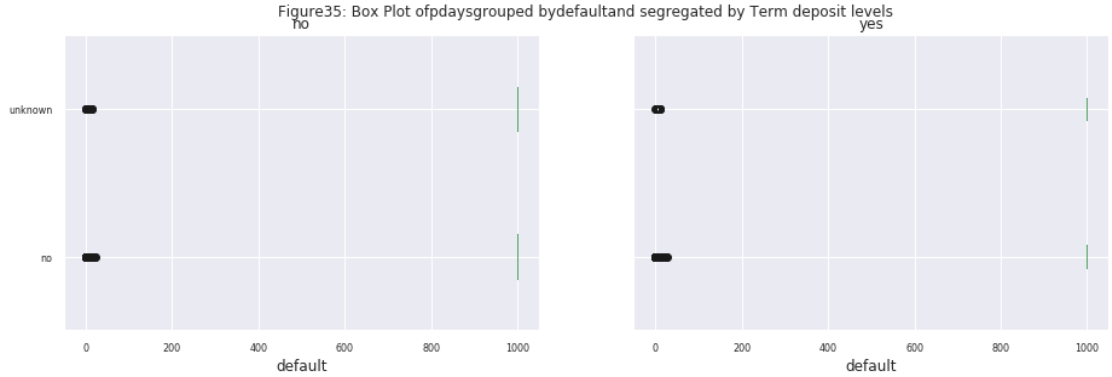


Figure36: Box Plot ofpdaysgrouped byhousingand segregated by Term deposit levels

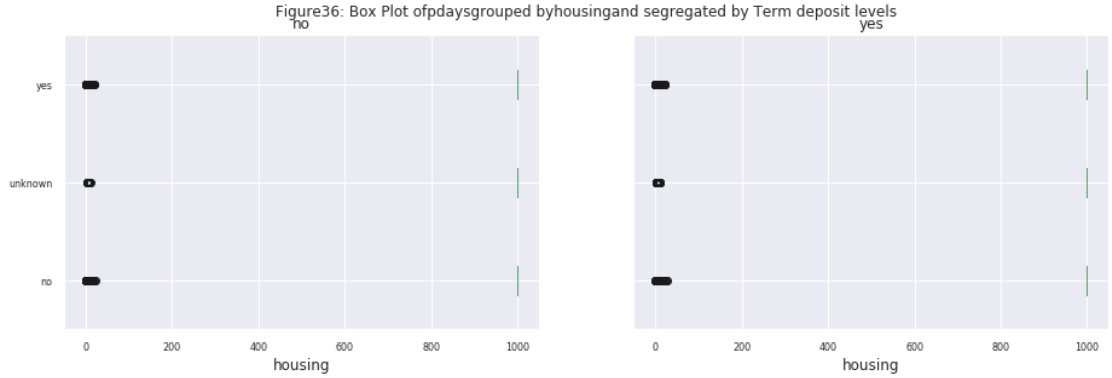


Figure37: Box Plot ofpdaysgrouped byloanand segregated by Term deposit levels

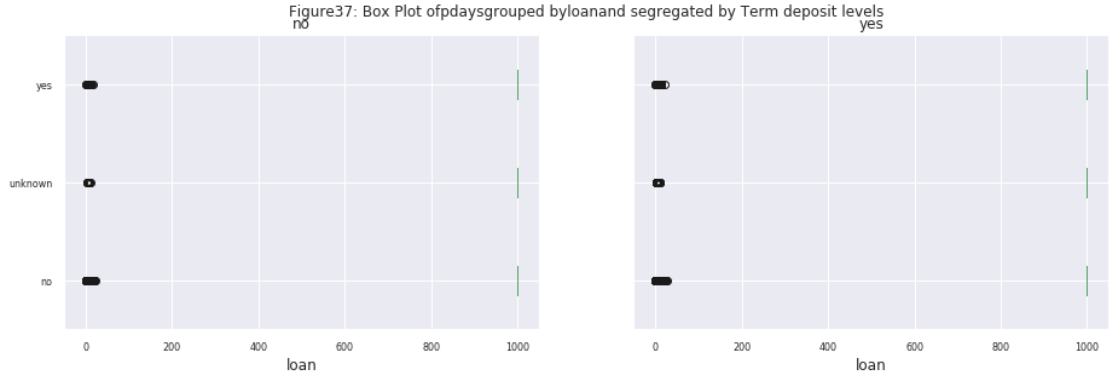


Figure38: Box Plot ofpreviousgrouped byjoband segregated by Term deposit levels

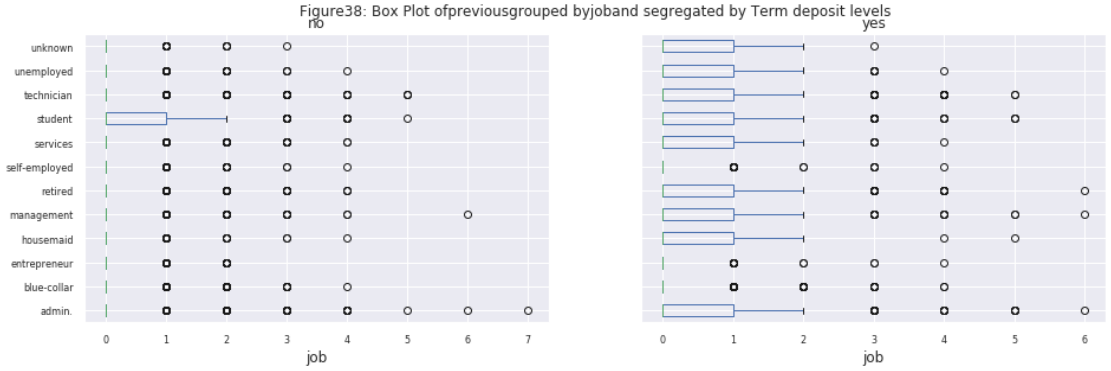


Figure39: Box Plot ofpreviousgrouped bymaritaland segregated by Term deposit levels

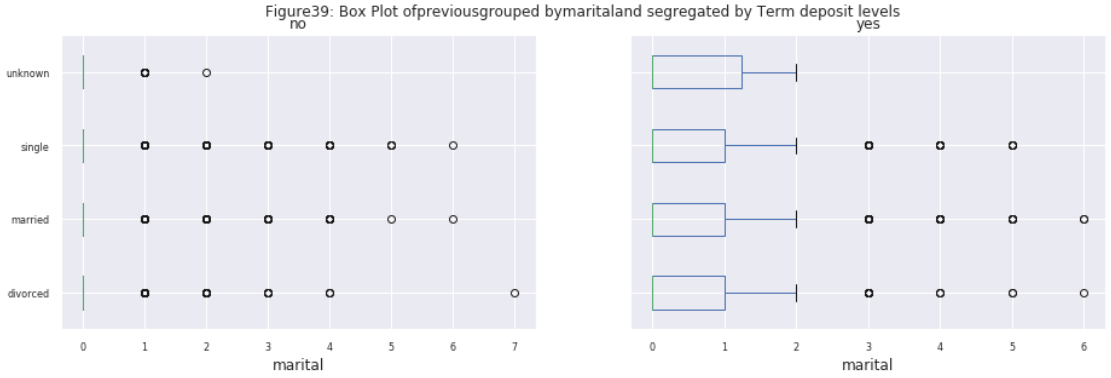


Figure40: Box Plot ofpreviousgrouped byeducationand segregated by Term deposit levels

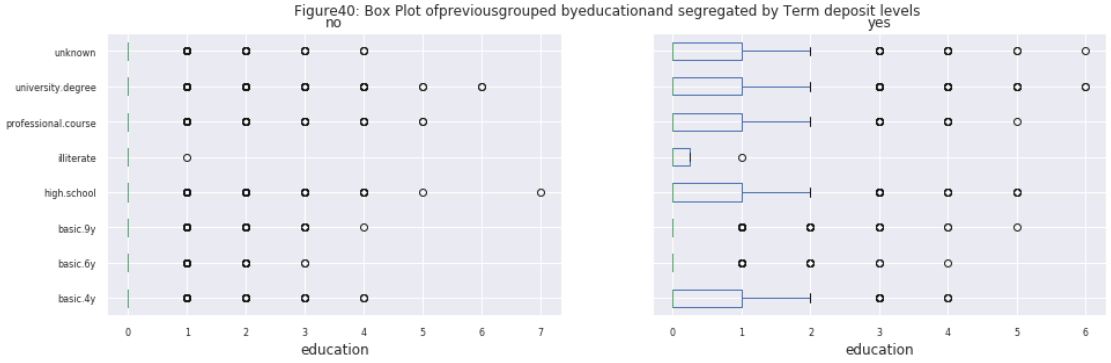


Figure41: Box Plot ofpreviousgrouped bydefaultand segregated by Term deposit levels

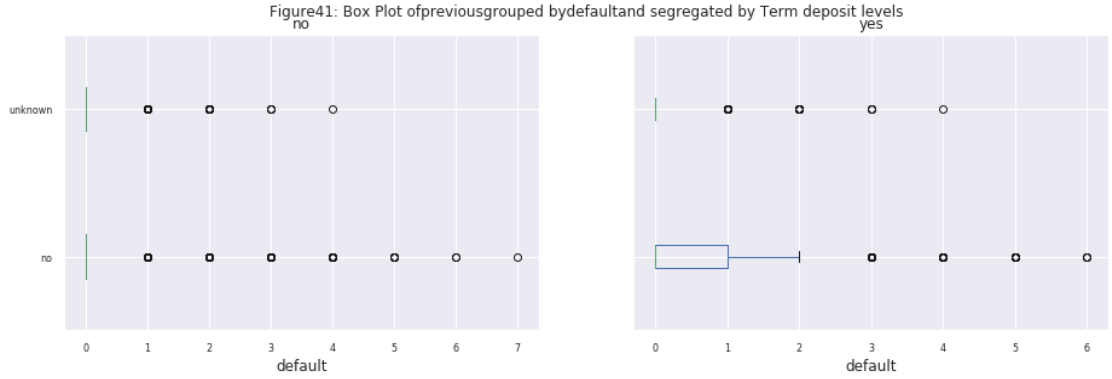


Figure42: Box Plot ofpreviousgrouped byhousingand segregated by Term deposit levels

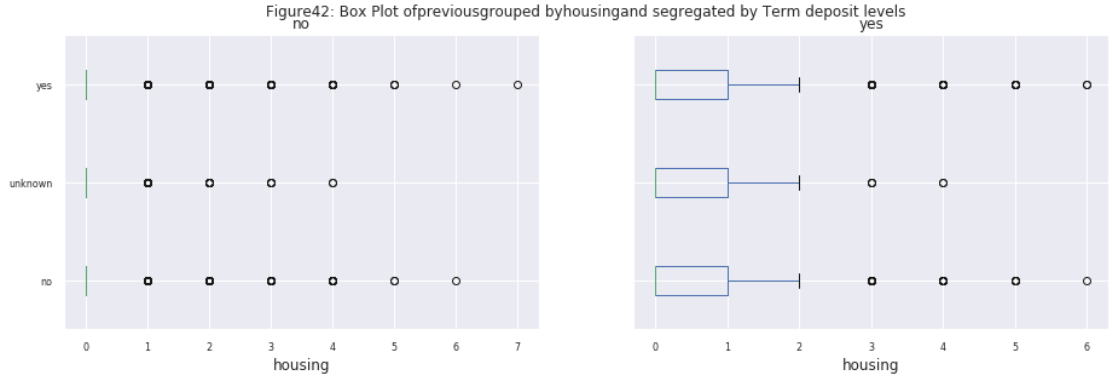


Figure43: Box Plot ofpreviousgrouped byloanand segregated by Term deposit levels

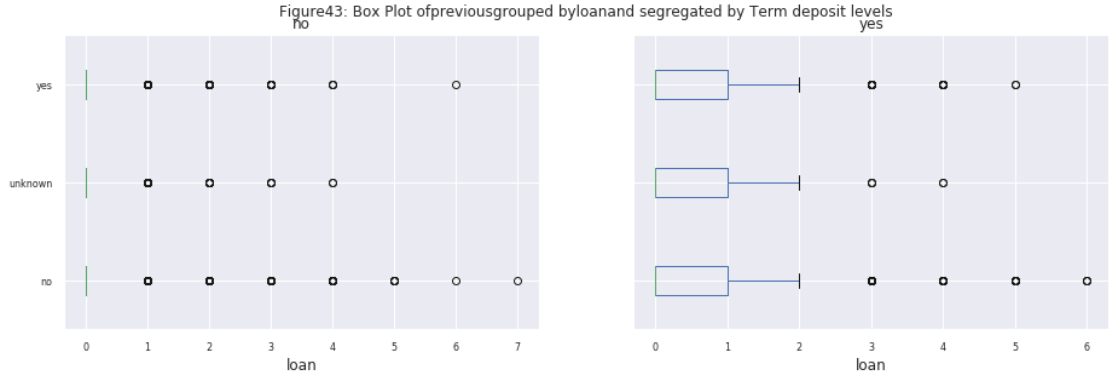


Figure44: Box Plot ofemp.var.rategrouped byjoband segregated by Term deposit levels

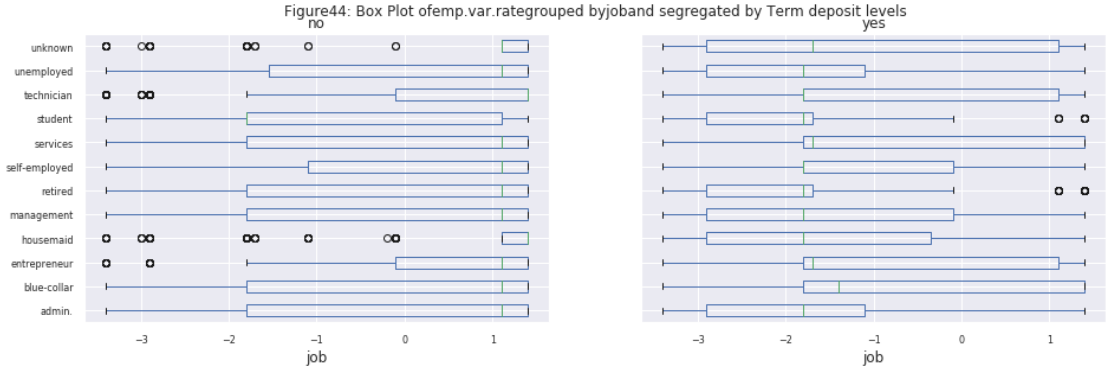


Figure45: Box Plot ofemp.var.rategrouped bymaritaland segregated by Term deposit levels

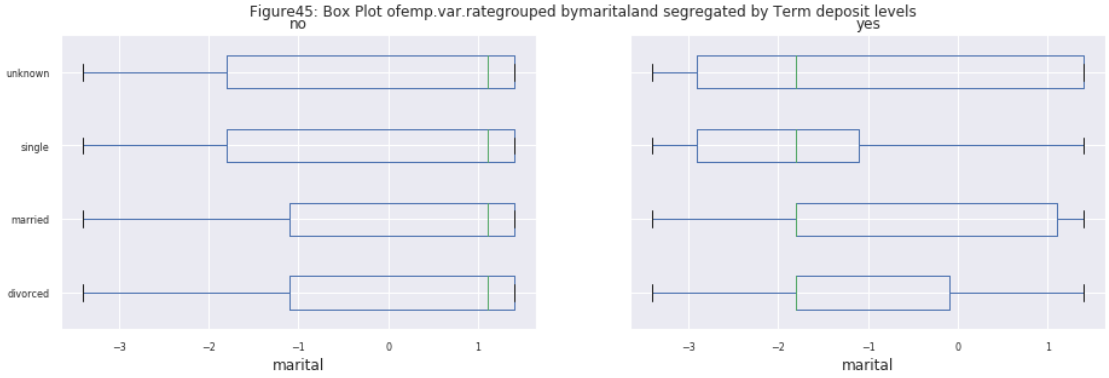


Figure46: Box Plot ofemp.var.rategrouped byeducationand segregated by Term deposit levels

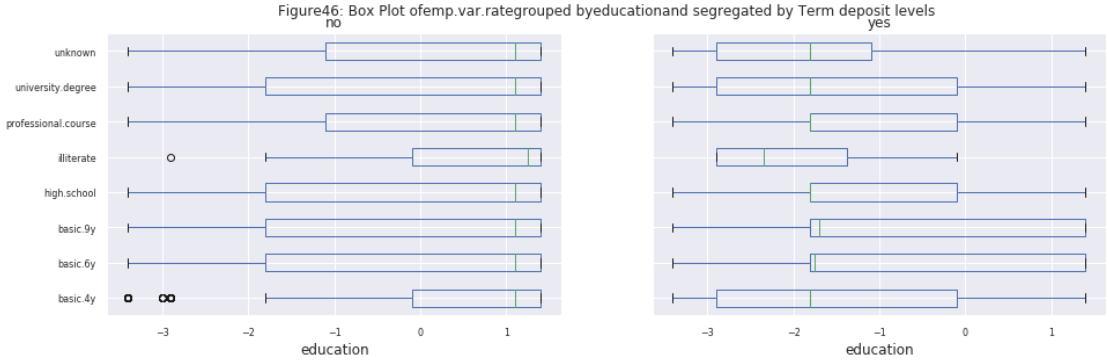


Figure47: Box Plot ofemp.var.rategrouped bydefaultand segregated by Term deposit levels

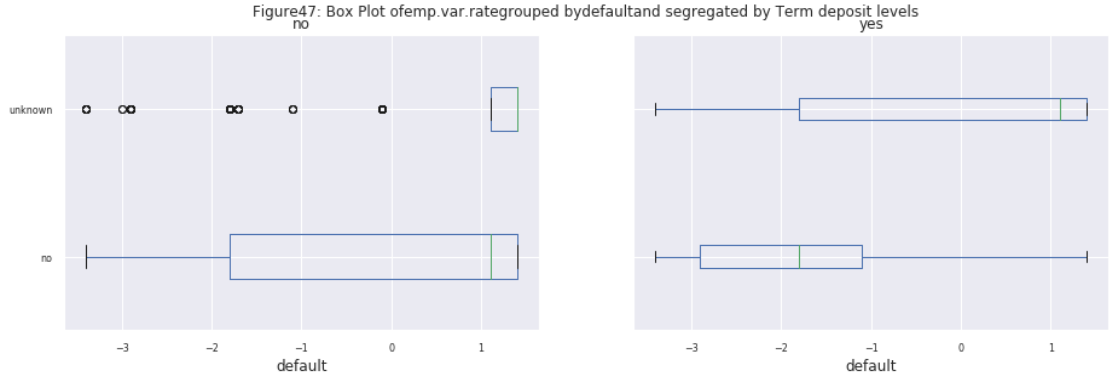


Figure48: Box Plot ofemp.var.rategrouped byhousingand segregated by Term deposit levels

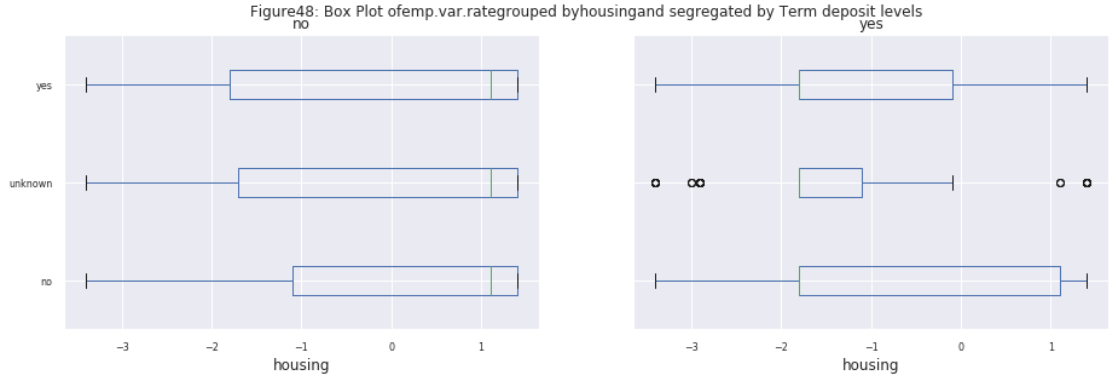


Figure49: Box Plot ofemp.var.rategrouped byloanand segregated by Term deposit levels

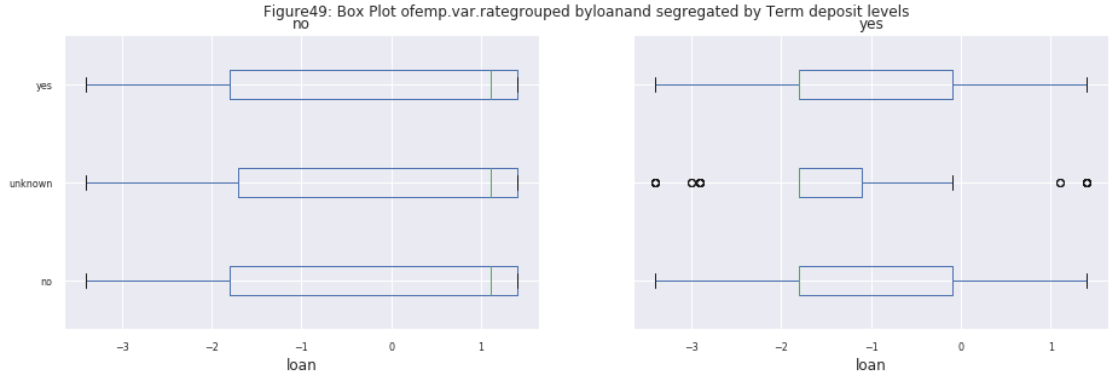


Figure50: Box Plot ofcons.price.idxgrouped byjoband segregated by Term deposit levels

Figure51: Box Plot ofcons.price.idxgrouped bymaritaland segregated by Term deposit levels

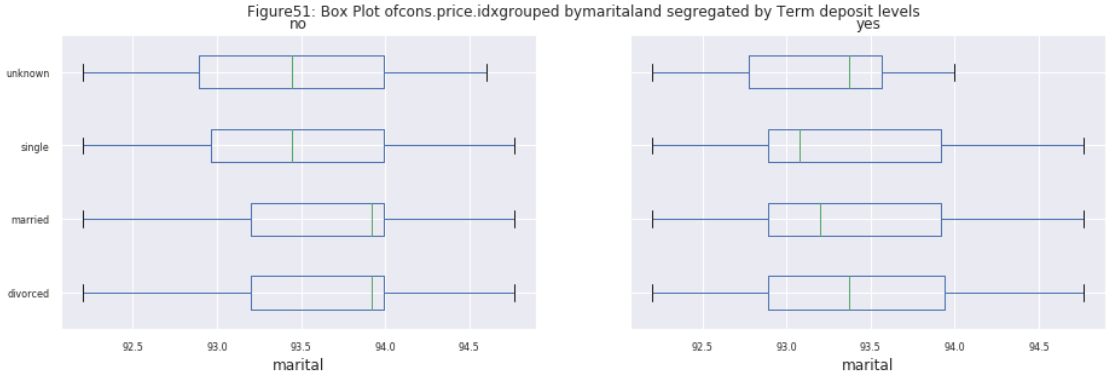


Figure52: Box Plot ofcons.price.idxgrouped byeducationand segregated by Term deposit levels

Figure53: Box Plot ofcons.price.idxgrouped bydefaultand segregated by Term deposit levels

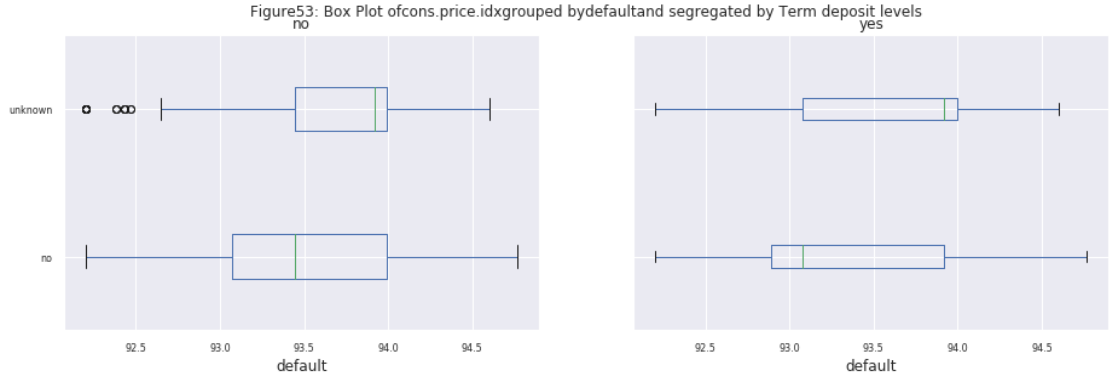


Figure54: Box Plot ofcons.price.idxgrouped byhousingand segregated by Term deposit levels

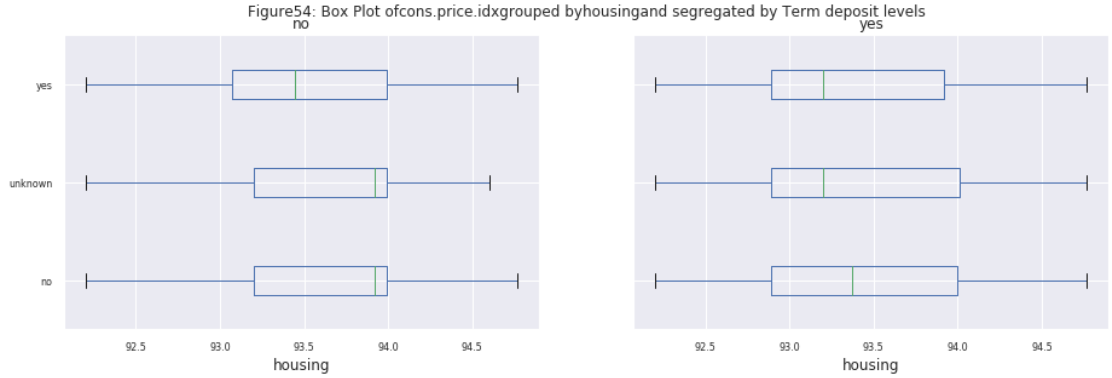


Figure55: Box Plot ofcons.price.idxgrouped byloanand segregated by Term deposit levels

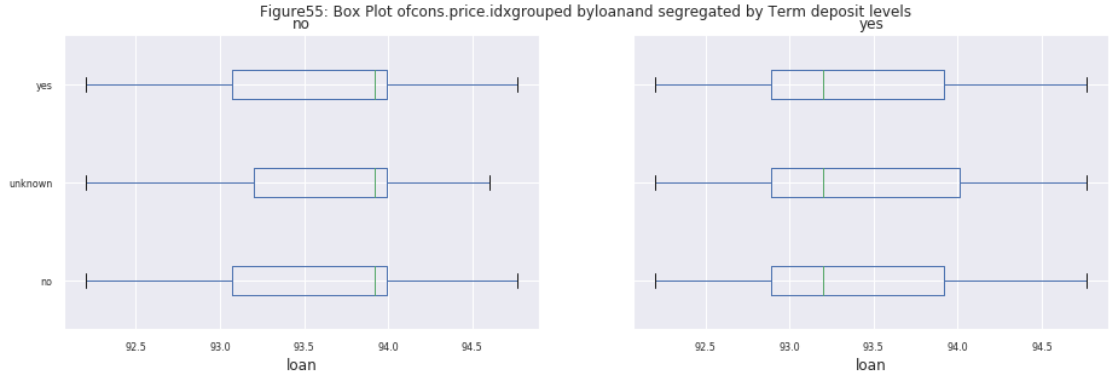


Figure56: Box Plot ofcons.conf.idxgrouped byjoband segregated by Term deposit levels

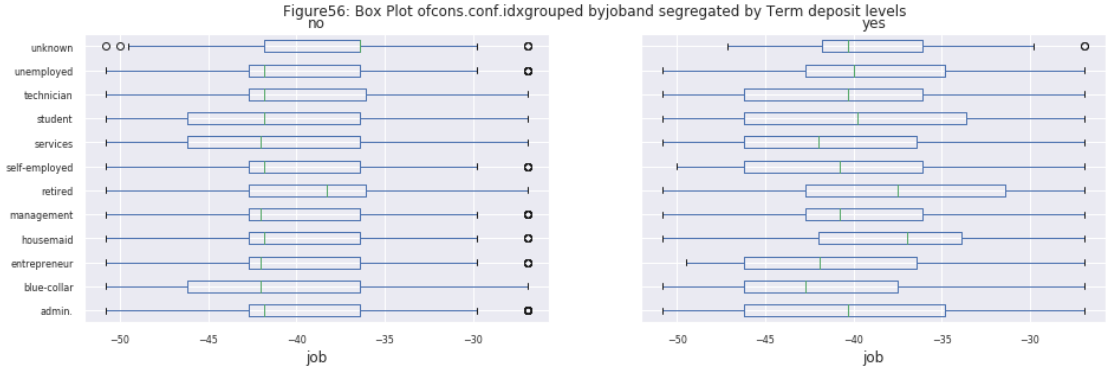


Figure57: Box Plot ofcons.conf.idxgrouped bymaritaland segregated by Term deposit levels

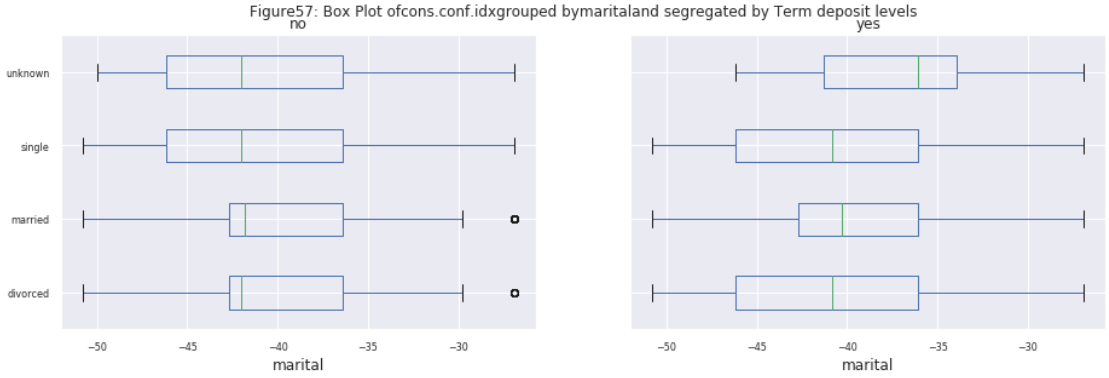


Figure58: Box Plot ofcons.conf.idxgrouped byeducationand segregated by Term deposit levels

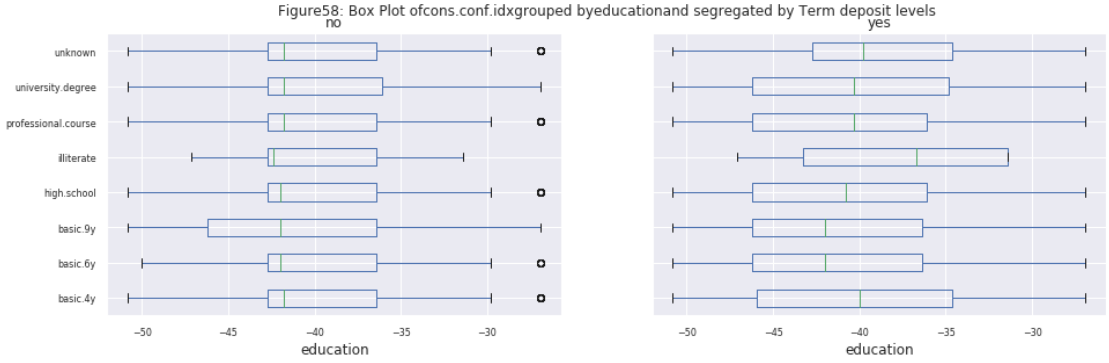


Figure59: Box Plot ofcons.conf.idxgrouped bydefaultand segregated by Term deposit levels

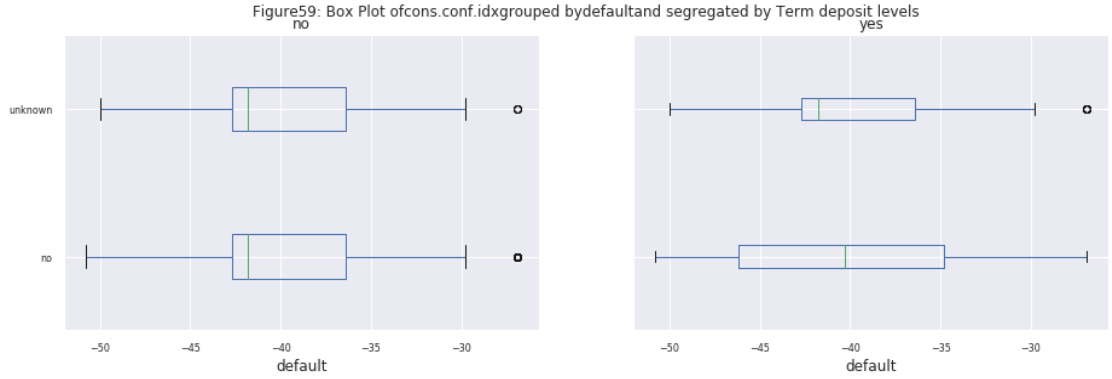


Figure60: Box Plot ofcons.conf.idxgrouped byhousingand segregated by Term deposit levels

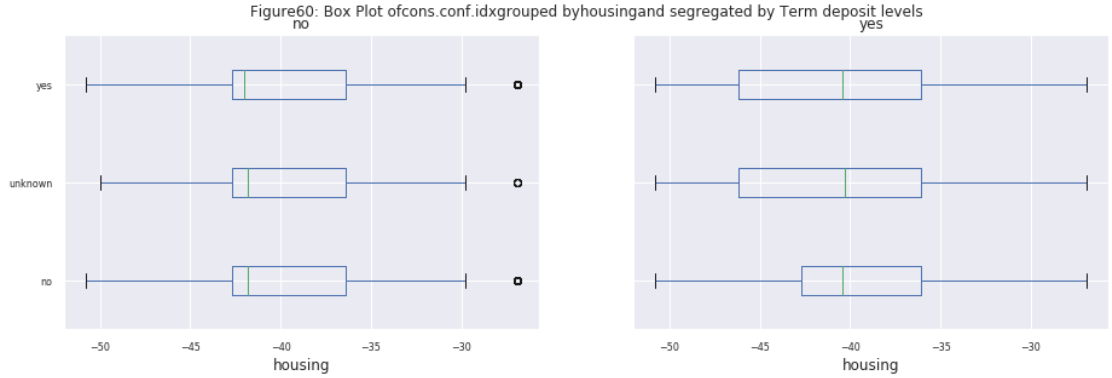


Figure61: Box Plot ofcons.conf.idxgrouped byloanand segregated by Term deposit levels

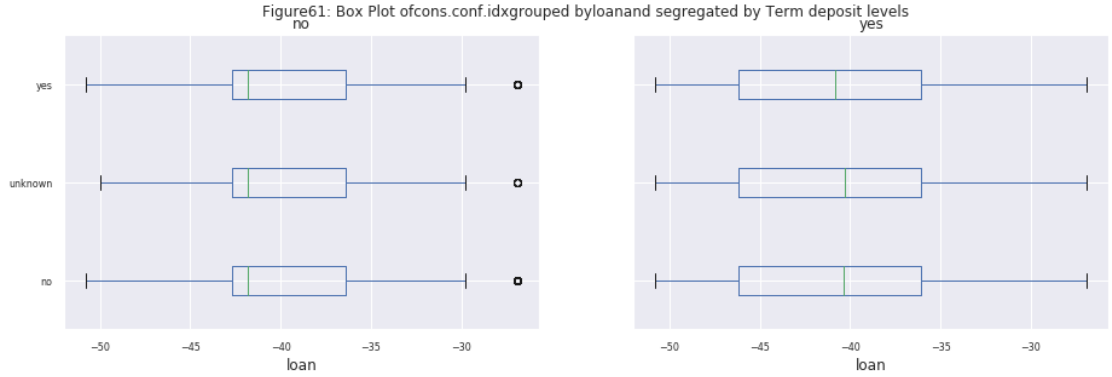


Figure62: Box Plot ofeuribor3mgrouped byjoband segregated by Term deposit levels

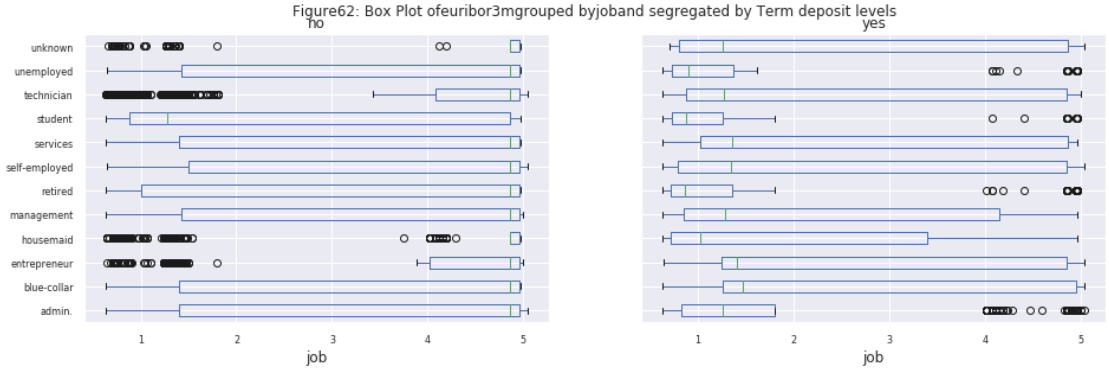


Figure63: Box Plot ofeuribor3mgrouped bymaritaland segregated by Term deposit levels

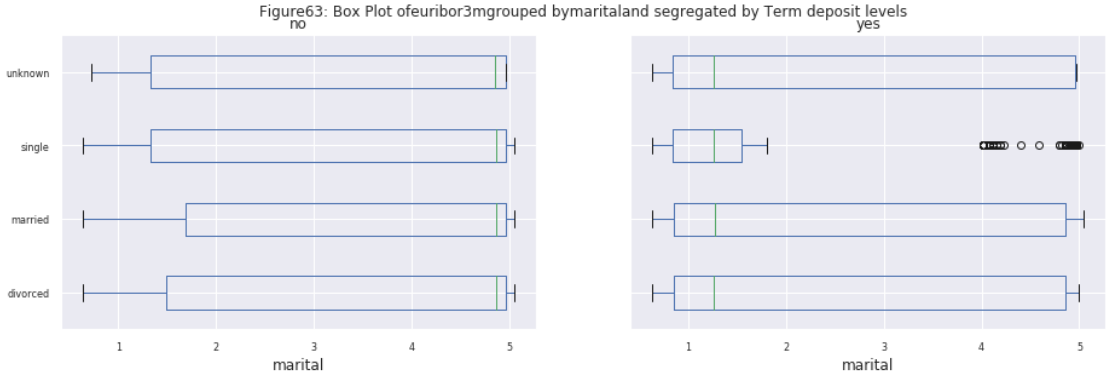


Figure64: Box Plot ofeuribor3mgrouped byeducationand segregated by Term deposit levels

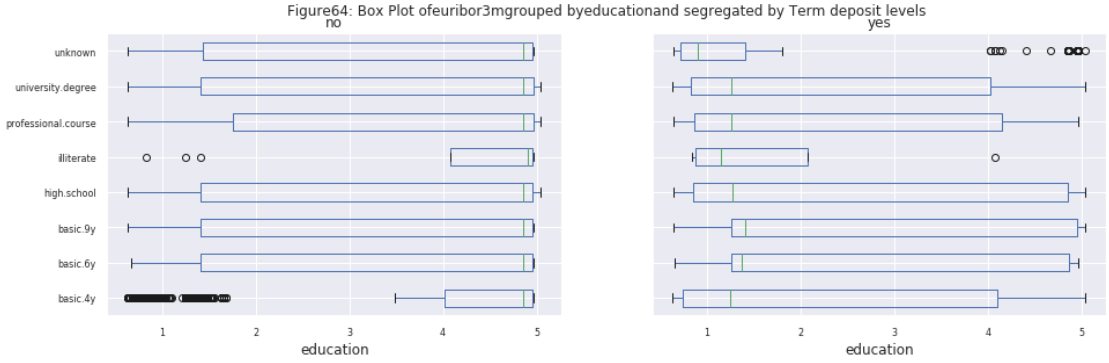


Figure65: Box Plot ofeuribor3mgrouped bydefaultand segregated by Term deposit levels

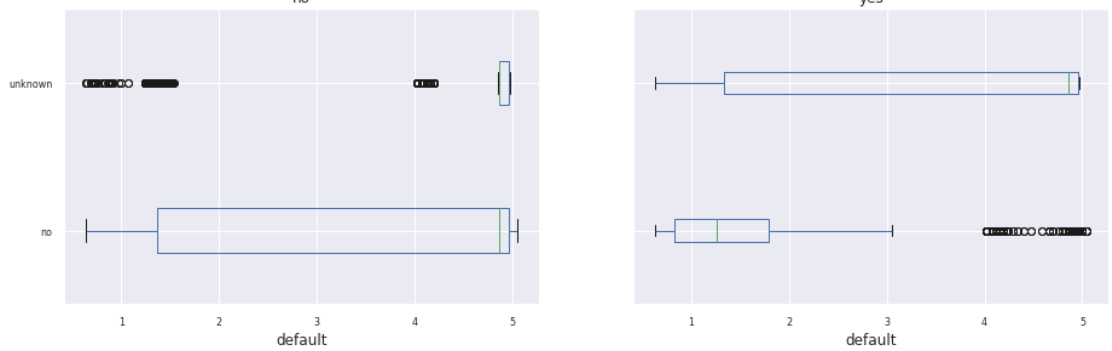


Figure66: Box Plot ofeuribor3mgrouped byhousingand segregated by Term deposit levels

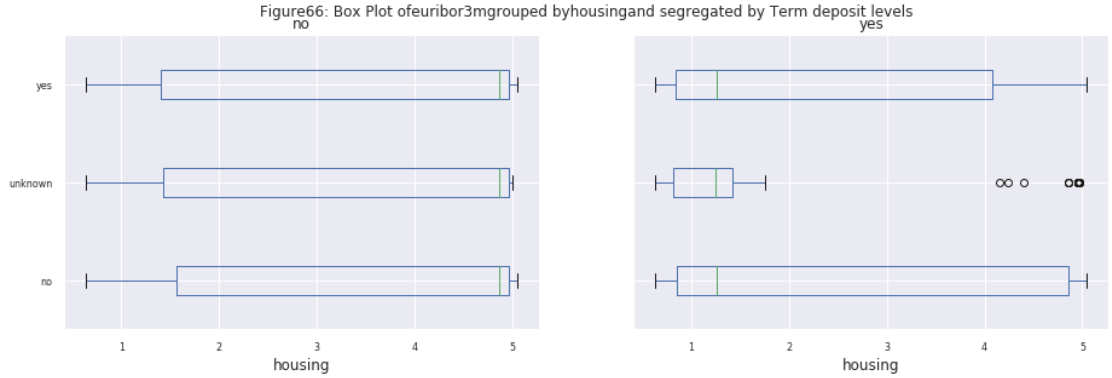


Figure67: Box Plot ofeuribor3mgrouped byloanand segregated by Term deposit levels

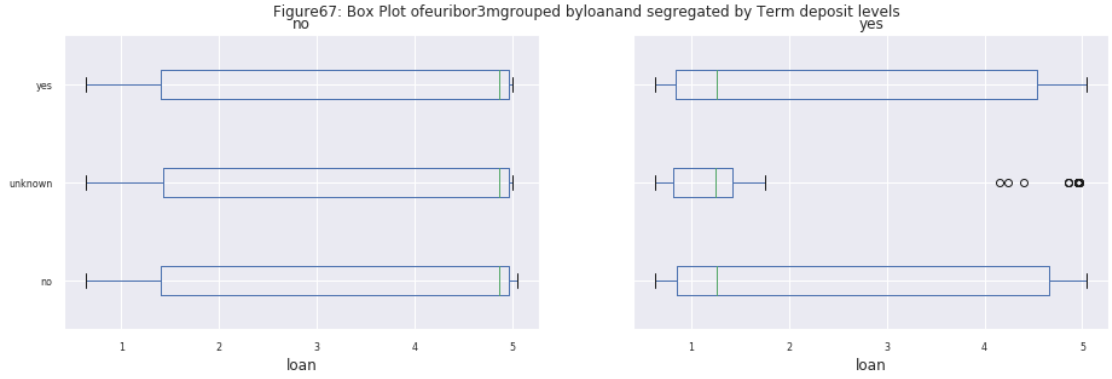


Figure68: Box Plot ofnr.employedgrouped byjoband segregated by Term deposit levels

Figure69: Box Plot ofnr.employedgrouped bymaritaland segregated by Term deposit levels

Figure70: Box Plot ofnr.employedgrouped byeducationand segregated by Term deposit levels

Figure71: Box Plot ofnr.employedgrouped bydefaultand segregated by Term deposit levels

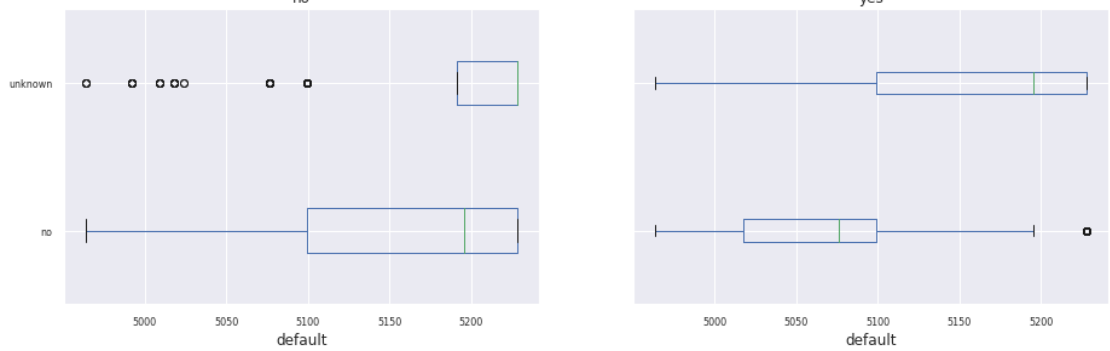
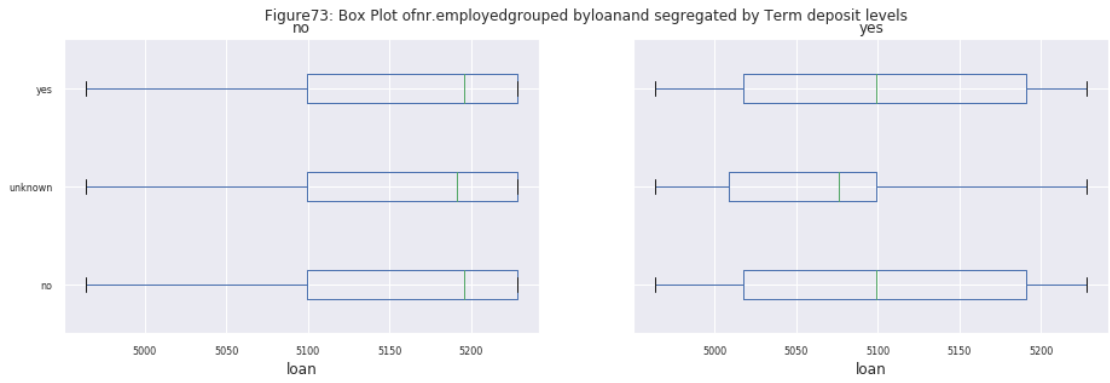


Figure72: Box Plot ofnr.employedgrouped byhousingand segregated by Term deposit levels

Figure73: Box Plot ofnr.employedgrouped byloanand segregated by Term deposit levels




```
In [23]: Bank=Bank.drop(['pdays'],1)
         Bank=Bank.drop(['previous'],1)
```

Chapter 4

Summary

In Phase 1, We removed the duration column because it was doesn't have any predictive power. We investigated the relationship between all attributes with each other and also their relationship with the target feature and removed pdays and previous columns since it carried the same information and noticed that age, job, marital, education, default, housing, loan are important features for predictive analysis.

Bibliography

[1] P. Cortez S. Moro and P. Rita. UCI Machine Learning Repository: Bank Data Set.