MATH2319 Machine Learning Project Phase 1
Predicting the contraceptive method choice of a woman based on demographic and socio-economic characteristics

Names: Udeshika Dissanayake
Student ID: s3400652

April 27, 2019

Contents

Chapter 1

Introduction

1.1 Objective

The objective of this study is to to predict the contraceptive methods (no use, long-term methods,or short-term methods) of a woman based on her demographic and socio-economic characteristics.

A data-set of 1473 married women with their demographic and socio-economic characteristics used in this study. The Source for the data-set is the UCI Machine Learning Repository at, http://http://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice [**?**].

This study consists with two phases. Objective of the Phase I is to preprocess and explore the data-set in order to build the model in Phase II. All the activities have been performed in `Python` package in this study and Compiled from Jupyter Notebook This report covers both narratives and the `Python` codesudes for the data preprocessing and exploration which performed under the phase I.

Content of this report is organized as follows. Section 1 describes the data sets and their attributes. Section 2 covers data preprocessing. In Section 3, each attribute and their inter-relationships are explored.

1.2 Data Set

The date-set contains contraceptive methods used & nine other demographic and socio-economic characteristics of 1473 married women in Indonesia, which obtains from National Indonesia Contraceptive Prevalence Survey in 1987. The data-set has 9 descriptive features and one target feature.

1.2.1 Target Feature

The response feature is contraceptive method which is given as:

$$\text{contraceptive method} = \begin{cases} long - term & \text{if the contraceptive method is long term method} \\ short - term & \text{if the contraceptive method is short term method} \\ no - use & \text{if no contraceptive method is used} \end{cases}$$

The target feature has three classes.

1.2.2 Descriptive Features

Following are the variables in the data-set.

- **`Wife's age`**: numerical
- **`Wife's education`**: categorical (low, medium low, medium high, high)
- **`Husband's education`**: categorical (low, medium low, medium high, high)

- `Number of children ever born`: numerical
- `Wife's religion`: binary (Non-Islam, Islam)
- `Wife's now working?`: binary (Yes, No)
- `Husband's occupation`: categorical (Cat1, Cat2, Cat3, Cat4)
- `Standard-of-living index`: categorical (low, medium low, medium high, high)
- `Media exposure`: binary (Good, Bad)

All the descriptive features are self-explanatory.

Chapter 2

Data Pre-processing

2.1 Data Retrieving

The contraceptive methods data-set has been loaded as "df_c" into Python using pandas.

```
In [1]: import pandas as pd
        df_c=pd.read_csv('Data_Set.csv')
```

The data-set has been inspected to check whether the features and descriptions outlined in the documentation are aligning with the data-set.

```
In [2]: #print bold
        from IPython.display import Markdown, display
        def printmd(string):
            display(Markdown(string))

        print("Dimension of the data set is ({},{}).\n".format(df_c.shape[0],df_c.shape[1]) )
        print("Data Types are: \n")
        print(df_c.dtypes)
        #print("\n First 5 rows in the Data-set is:")
        from IPython.display import display, HTML
        print("\n")
        printmd("**\nTable 1: First three rows in the original Data-set**")
        df_c.head(3)
```

```
Dimension of the data set is (1473,10).

Data Types are:

Wifes_age                          int64
Wifes_education                    int64
Husbands_education                 int64
Number_of_children_ever_born       int64
Wifes_religion                     int64
Wifes_now_working%3F               int64
Husbands_occupation                int64
Standard-of-living_index           int64
Media_exposure                     int64
Contraceptive_method_used          int64
dtype: object
```

Table 1: First three rows in the original Data-set

```
Out[2]:    Wifes_age  Wifes_education  Husbands_education  \
        0         24                2                   3
        1         45                1                   3
        2         43                2                   3

           Number_of_children_ever_born  Wifes_religion  Wifes_now_working%3F  \
        0                             3               1                     1
        1                            10               1                     1
        2                             7               1                     1

           Husbands_occupation  Standard-of-living_index  Media_exposure  \
        0                    2                         3               0
        1                    3                         4               0
        2                    3                         4               0

           Contraceptive_method_used
        0                          1
        1                          1
        2                          1
```

2.2 Data Cleaning and Transformation

2.2.1 Relabeling column names

Since the original attribute names are fairly long, a new set of attribute names have been specified for the convenience of this study.

```
In [3]: df_c.columns=[
            'wife_age',
            'wife_edu',
            'husb_edu',
            'children',
            'wife_religion',
            'wife-working',
            'husb-occup',
            's-living_index',
            'media_exp',
            'contrac_mthd']
```

2.2.2 Replacing labels with descriptive labels

Then the labels of the categorical attributes were replaced with descriptive labels instead of original numerical labels. For an example, the original wife's education data 1, 2,3,& 4, representing low, middle low,middle high, and high respectively were replaced by descriptive labels of low, middle low,middle high, and high. Similarly, the numerical labels of the other categorical attributes have been replaced by descriptive labels.

```
In [4]: #Replacing labels for wife's education with descriptive labels
        df_c['wife_edu'].replace(1, "low", inplace=True)
        df_c['wife_edu'].replace(2, "middle low", inplace=True)
```

```
df_c['wife_edu'].replace(3, "middle high", inplace=True)
df_c['wife_edu'].replace(4, "high", inplace=True)

#Replacing labels for husband's education descriptive labels
df_c['husb_edu'].replace(1, "low", inplace=True)
df_c['husb_edu'].replace(2, "middle low", inplace=True)
df_c['husb_edu'].replace(3, "middle high", inplace=True)
df_c['husb_edu'].replace(4, "high", inplace=True)

#Replacing labels for wife's religion with descriptive labels
df_c['wife_religion'].replace(1, "Islam", inplace=True)
df_c['wife_religion'].replace(0, "Other", inplace=True)

#Replacing labels for wifes current working status with descriptive labels
df_c['wife-working'].replace(1, "No", inplace=True)
df_c['wife-working'].replace(0, "Yes", inplace=True)

#Replacing labels for husband's occupation with descriptive labels
df_c['husb-occup'].replace(1, "Cat1", inplace=True)
df_c['husb-occup'].replace(2, "Cat2", inplace=True)
df_c['husb-occup'].replace(3, "Cat3", inplace=True)
df_c['husb-occup'].replace(4, "Cat4", inplace=True)

#Replacing labels for standards of living index with descriptive labels
df_c['s-living_index'].replace(1, "low", inplace=True)
df_c['s-living_index'].replace(2, "middle low", inplace=True)
df_c['s-living_index'].replace(3, "middle high", inplace=True)
df_c['s-living_index'].replace(4, "high", inplace=True)

#Replacing labels for madia exposure with descriptive labels
df_c['media_exp'].replace(1, "bad", inplace=True)
df_c['media_exp'].replace(0, "good", inplace=True)

#Replacing labels for contraceptive methods used with descriptive labels
df_c['contrac_mthd'].replace(1, "No-use", inplace=True)
df_c['contrac_mthd'].replace(2, "Long-term", inplace=True)
df_c['contrac_mthd'].replace(3, "Short-term", inplace=True)
```

2.2.3 Data Type conversion

The 'wife_edu', 'husb_edu', 'wife_religion', 'wife-working', 'husb-occup', 's-living_index', 'media_exp', and 'contrac_mthd' variables should be a factor data type. However in the data set they are defined as numerical variables. In below steps, The data types of these variables, changed from numerical to categorical accordingly.

```
In [5]: for col in ['wife_edu',
                    'husb_edu',
                    'wife_religion',
                    'wife-working',
                    'husb-occup',
                    's-living_index',
                    'media_exp',
                    'contrac_mthd']:
            df_c[col] = df_c[col].astype('category')
```

The updated data-set has been inspected again.

```
In [6]: print("Dimension of the data set is ({},{}).\n".format(df_c.shape[0],df_c.shape[1]) )
        print("Data Types are: \n")
        print(df_c.dtypes)
        printmd("**\nTable 2: Four random rows in the updated Data-set**")
        df_c.sample(4)
```

```
Dimension of the data set is (1473,10).

Data Types are:

wife_age             int64
wife_edu          category
husb_edu          category
children             int64
wife_religion     category
wife-working      category
husb-occup        category
s-living_index    category
media_exp         category
contrac_mthd      category
dtype: object
```

Table 2: Four random rows in the updated Data-set

Out[6]:

	wife_age	wife_edu	husb_edu	children	wife_religion	wife-working	husb-occup	s-living_index	media_exp	contrac_mthd
1443	21	middle low	middle low	0	Other	No	Cat4	high	good	Short-term
59	49	high	high	6	Islam	No	Cat1	middle low	good	No-use
431	18	middle high	high	1	Islam	No	Cat3	high	good	Long-term
68	23	middle high	high	2	Islam	No	Cat3	high	good	No-use

2.2.4 Checking for missing values in the data

Below codes have been executed to identify the missing values in the data-set. It is clearly evident that there are no missing values in the data-set.

```
In [7]: print("Number of missing value for each feature:")
        print(df_c.isnull().sum())
```

```
Number of missing value for each feature:
wife_age          0
wife_edu          0
husb_edu          0
children          0
wife_religion     0
wife-working      0
husb-occup        0
```

```
s-living_index    0
media_exp         0
contrac_mthd      0
dtype: int64
```

2.2.5 Checking for typo in Categorical Features

Typos of all categorical features, including the target feature in the data-set has been checked by investigating the Frequency tables. As can be seen below, there are no typos in the Categorical Features in the data-set.

```
In [8]: for col in df_c.columns:
            if (df_c[col].dtype.name == 'category'):
                print('Unique values for ' + col+':')
                print (df_c[col].value_counts(), '\n''\n')
```

```
Unique values for wife_edu:
high           577
middle high    410
middle low     334
low            152
Name: wife_edu, dtype: int64


Unique values for husb_edu:
high           899
middle high    352
middle low     178
low             44
Name: husb_edu, dtype: int64


Unique values for wife_religion:
Islam    1253
Other     220
Name: wife_religion, dtype: int64


Unique values for wife-working:
No     1104
Yes     369
Name: wife-working, dtype: int64


Unique values for husb-occup:
Cat3    585
Cat1    436
Cat2    425
Cat4     27
Name: husb-occup, dtype: int64


Unique values for s-living_index:
high           684
```

```
middle high    431
middle low     229
low            129
Name: s-living_index, dtype: int64


Unique values for media_exp:
good    1364
bad      109
Name: media_exp, dtype: int64


Unique values for contrac_mthd:
No-use        629
Short-term    511
Long-term     333
Name: contrac_mthd, dtype: int64
```

2.2.6 Checking extra whitespaces & Capital Letter mismatches in Categorical Features

Extra whitespaces & Capital letter mismatches in the categorical data has already been checked while investigating the Frequency tables in previous section (Checking for typo in Categorical Features). However strip() & .lower() functions can be applied to get rid of Extra whitespaces & Capital letter mismatches respectively.

2.2.7 Summary of categorical features

All eight categorical features including the target feature have been summarized below. Each feature consist of 1473 records with definitive unique classes as mentioned in the table.

In [9]:
```python
printmd("**\nTable 3: Summary of categorical features**")
display(df_c.describe(include = 'category'))
```

Table 3: Summary of categorical features

	wife_edu	husb_edu	wife_religion	wife-working	husb-occup	s-living_index
count	1473	1473	1473	1473	1473	1473
unique	4	4	2	2	4	4
top	high	high	Islam	No	Cat3	high
freq	577	899	1253	1104	585	684

	media_exp	contrac_mthd
count	1473	1473
unique	2	3
top	good	No-use
freq	1364	629

2.2.8 Summary of Numerical Features

Wife's age and number of children are the only numerical attributes in this data-set.

```
In [10]: from IPython.display import display, HTML
         display(HTML('<b>Table 4: Summary of continuous features</b>'))
         display(df_c.describe(include = 'int64').round(2))
```

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

	wife_age	children
count	1473.00	1473.00
mean	32.54	3.26
std	8.23	2.36
min	16.00	0.00
25%	26.00	1.00
50%	32.00	3.00
75%	39.00	4.00
max	49.00	16.00

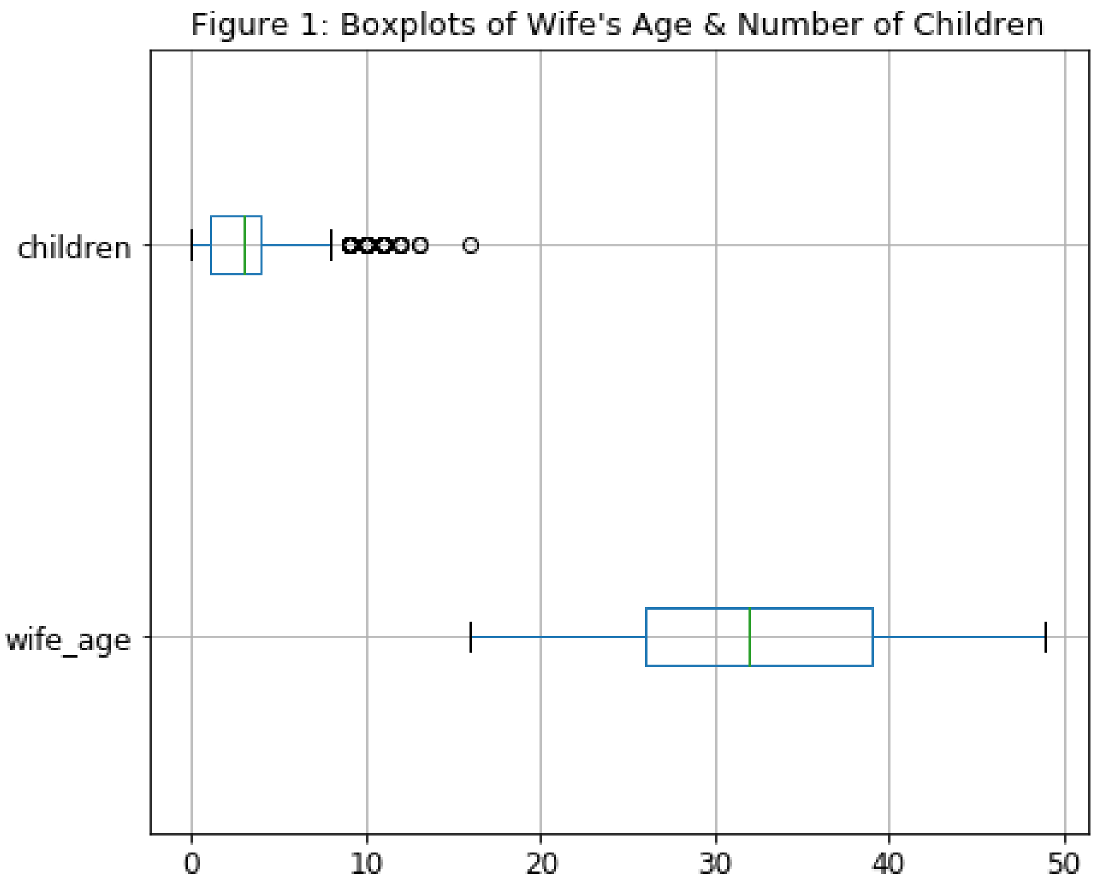
2.2.9 Checking for impossible numerical values in Numerical Features

After examining Table 4, it is evident that this data-set is not consisted with any impossible numerical values.

2.2.10 Checking for outliers in Numerical Features

Boxplot is a best method to visualize outliers of numerical attributes. The box captures the middle 50% of the data, the line shows the median and the whiskers of the plots show the reasonable extent of data. Any dots outside the whiskers are good candidates for outliers. The outlier of each numerical attributes was identified using box diagrams as follows:

```
In [11]: import matplotlib.pyplot as plt

         i=1

In [12]: df_c.boxplot(column=['wife_age','children'],vert = False, figsize=(7,6))
         plt.title("Figure " + str(i) + ": Boxplots of Wife's Age & Number of Children",size=13)
         plt.xticks(size=12)
         plt.yticks(size=12)
         plt.show()
         i=i+1
```



As shown in the boxplot above, there are few outliers for number of children(considerably high number) and IQR Score method has used to remove the outliers from the data-set. Firstly, first and third quartiles have been calculated in order to get the Interquartile Range.

```
In [13]: Q1 = df_c.quantile(0.25) #First Quartile
         Q3 = df_c.quantile(0.75) #Third Quartile
         IQR = Q3 - Q1

         print('\nInterquartile Range is:')
         print(IQR)

         print('\nLower Outlier Boundarye is:')
         print(Q1-(1.5*IQR))

         print('\nUpper Outlier Boundarye is:')
         print(Q3+(1.5*IQR))
```

```
Interquartile Range is:
wife_age    13.0
children     3.0
dtype: float64
```

```
Lower Outlier Boundarye is:
wife_age    6.5
children   -3.5
dtype: float64

Upper Outlier Boundarye is:
wife_age    58.5
children     8.5
dtype: float64
```

Then, removed the outlier rows from the data-set.

```
In [14]: df2=df_c[['wife_age','children']]
         df_oo = df_c[~((df2 < (Q1 - 1.5 * IQR)) |(df2 > (Q3 + 1.5 * IQR))).any(axis=1)]
```

The updated data-set has been inspected again to check the shape of the data-set.

```
In [15]: print("Dimension of the data set before removing outliers is ({},{}).\n".
               format(df_c.shape[0],df_c.shape[1]) )

Dimension of the data set before removing outliers is (1473,10).
```

```
In [16]: print("Dimension of the data set after removing outliers is ({},{}).\n".
               format(df_oo.shape[0],df_oo.shape[1]) )

Dimension of the data set after removing outliers is (1428,10).
```

As per the new dimention of the data-set, 45 rows has been removed.

Chapter 3

Data Exploration

3.1 Univariate Visualization

Univariate visualization are plots of individual attributes without interactions, which can be used to investigate the distribution and the characteristics of each attribute. The histogram and box plot have been used to explore the numerical features, while pie charts, counter plots, and frequency plots have been used to explore categorical features.

3.1.1 Univariate Visualization for numerical attributes

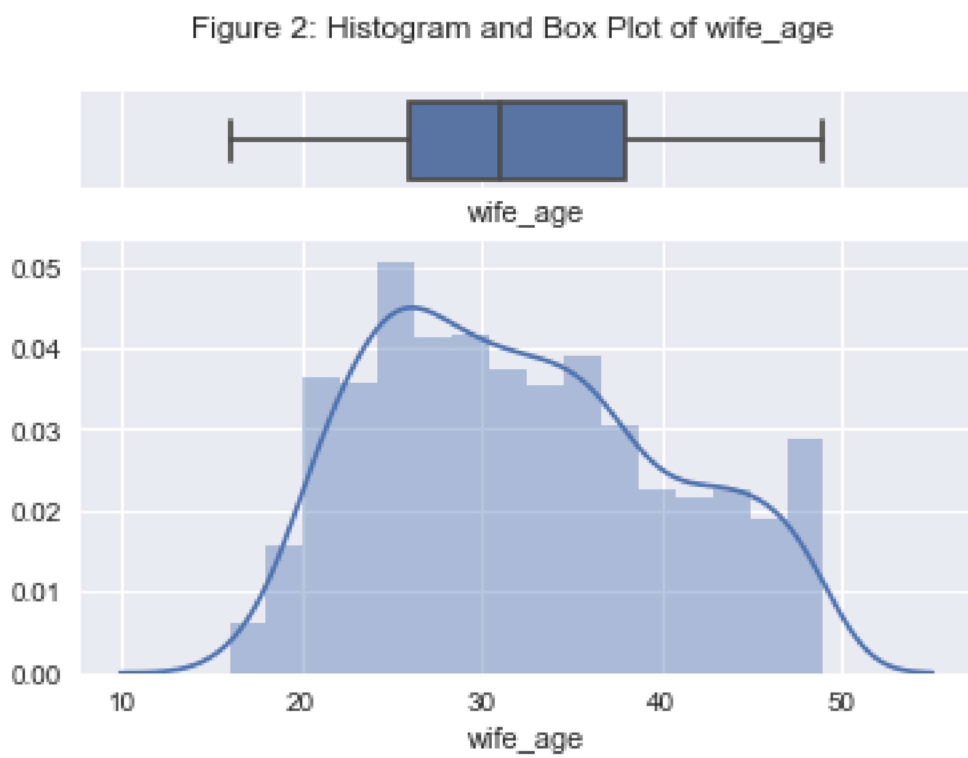
Histogram is the one of the best and accurate method to visualized numerical data. It shows the frequency distribution within a attribute. Boxplot is another insightful method to visualize the distribution of numerical attributes. Therefore, in order to analyze the 'wife_age' and 'children' attributes, both the Histogram & Boxplot have been plotted together.

```
In [17]: import seaborn as sns
```

```
In [18]: # to hide warinings

         def warn(*args, **kwargs):
             pass
         import warnings
         warnings.warn = warn
```

```
In [19]: df=df_oo.copy()
         sns.set(color_codes=True)

         def BarPlot(x):
             total = float(len(df_o))
             ax = df[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": (.2, .9)})
             plt.suptitle("Figure " + str(i) + ": Histogram and Box Plot of " + col,size=12)
             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 ['wife_age','children']:
    BoxHistogramPlot(df[col])
    plt.show()
    i = 1 + i
```



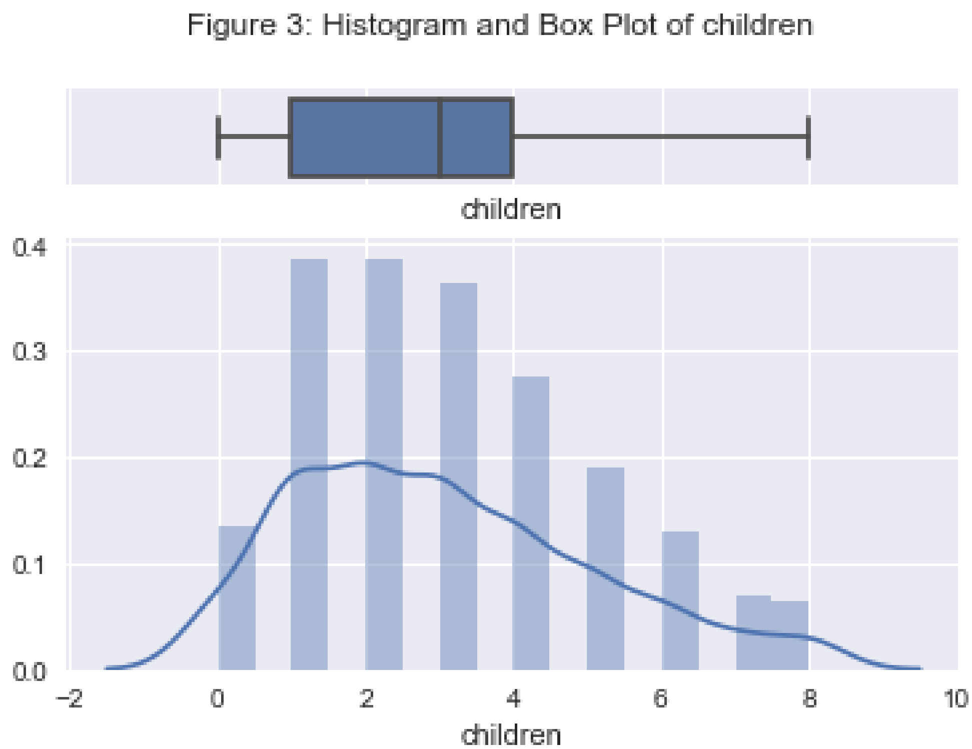


Figure 3 shows the wife_age of the data-set spanning from around 16 years to almost 50 years. The middle 50% of the wife_age resides between 26 age to 39 age as can be seen from the box plot. The histogram clearly shows the right skewness of the data-set. Also, the highest proportion of records has wife_age between ~25 to ~37 years as shown in the histogram.

Figure 4 clearly evident from the box plot that the middle 50% of children count ranges from 1 to 4. Also, the histogram shows the right skewness of the children count in the data-set.

3.1.2 Univariate Visualization for categorical attributes

Pie chart is one of the simplest yet very strong data vitalization tool which enables someone to see the proportion of each data category. Therefore, each of categorical attribute has been visually represented using the the pie charts.

```
In [20]: sns.set_palette("husl", 4)
         for col in df.columns:
             if (df[col].dtype.name == 'category'):
                 df[col].value_counts().plot(kind='pie',fontsize=10,autopct='%.1f%%',
                                             pctdistance=0.6, startangle=90,labels=None,
                                             wedgeprops={'alpha':0.7,'edgecolor':'white'},
                                            figsize=(2, 2))
                 plt.ylabel('')
                 plt.axis('equal')
                 plt.legend(labels=df[col].unique(),bbox_to_anchor=(1.8,0.6), loc="center right",
                            fontsize=10,bbox_transform=plt.gcf().transFigure)
                 plt.title('Figure ' + str(i) + ':  Box Plot of ' + col, size=10)
                 plt.subplots_adjust(left=0.0, bottom=0.1, right=1)
                 plt.show(), '\n'
                 i=i+1
```

Figure 4: Box Plot of wife_edu

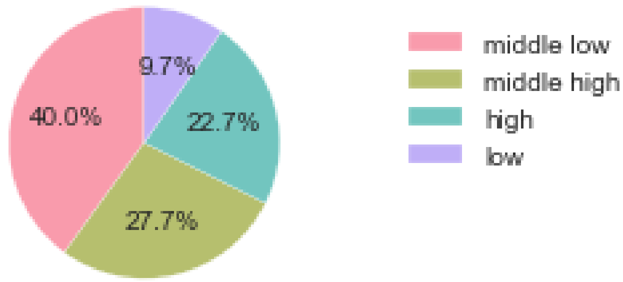


Figure 5: Box Plot of husb_edu

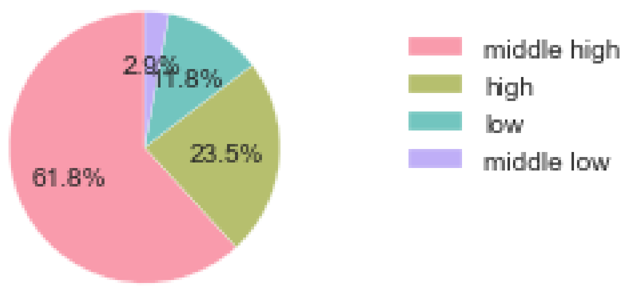


Figure 6: Box Plot of wife_religion

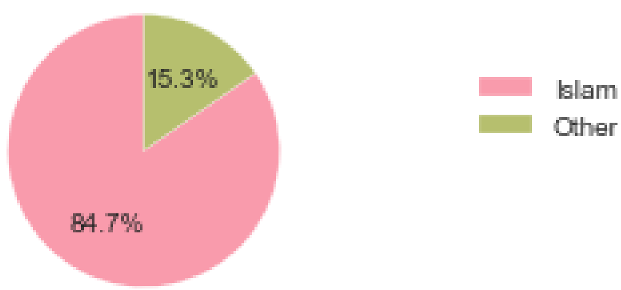


Figure 7: Box Plot of wife-working

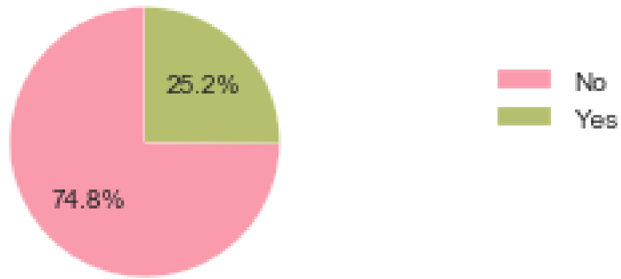


Figure 8: Box Plot of husb-occup

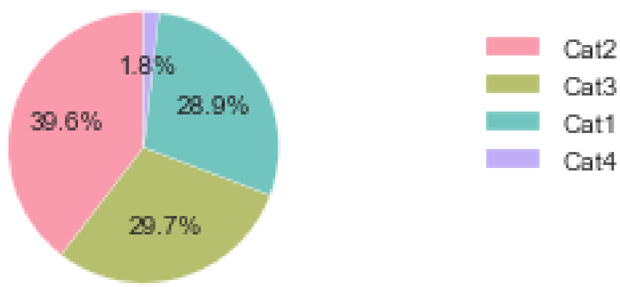


Figure 9: Box Plot of s-living_index

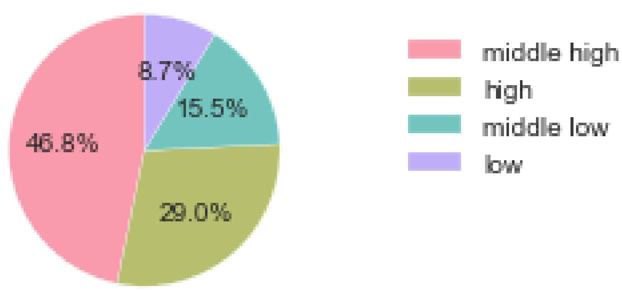


Figure 10: Box Plot of media_exp

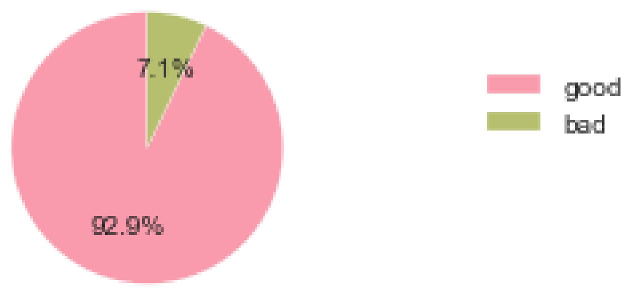


Figure 11: Box Plot of contrac_mthd

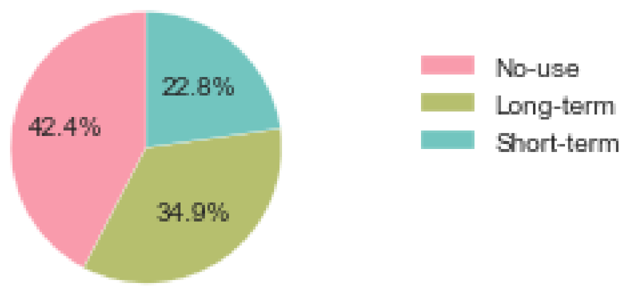


Figure 5 shows approximately equal amount of low (combine low and middle low) and high (combine high and middle high) education level for wifes in the data-set. In contrast, the figure 6 shows the husband eduction level is predominantly high (combine high and middle-high is ~85%) in the data-set. Figure 7 and 8 show that majority of wifes in the data-set are Islam and not working, respectively. Standard of living is fairly high (combine high and middle-high is ~75%) and exposure to media is more than 90%, according to figure 8 and 11, respectively. The distribution of the target feature---contrac_methd---is finally shown in the figure 12.

3.2 Multivariate Visualisation

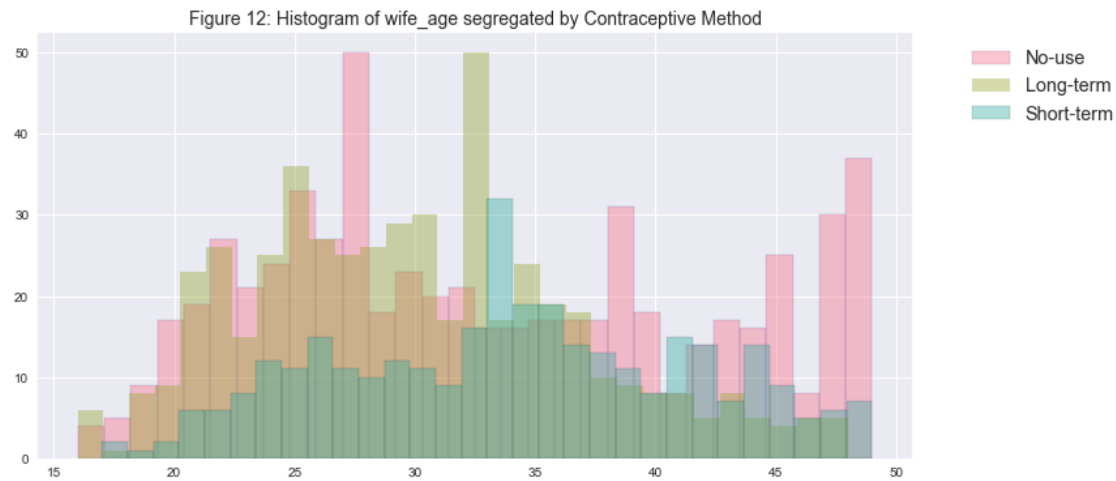
3.2.1 Histogram of Numeric Features Segregated by Contraceptive Method

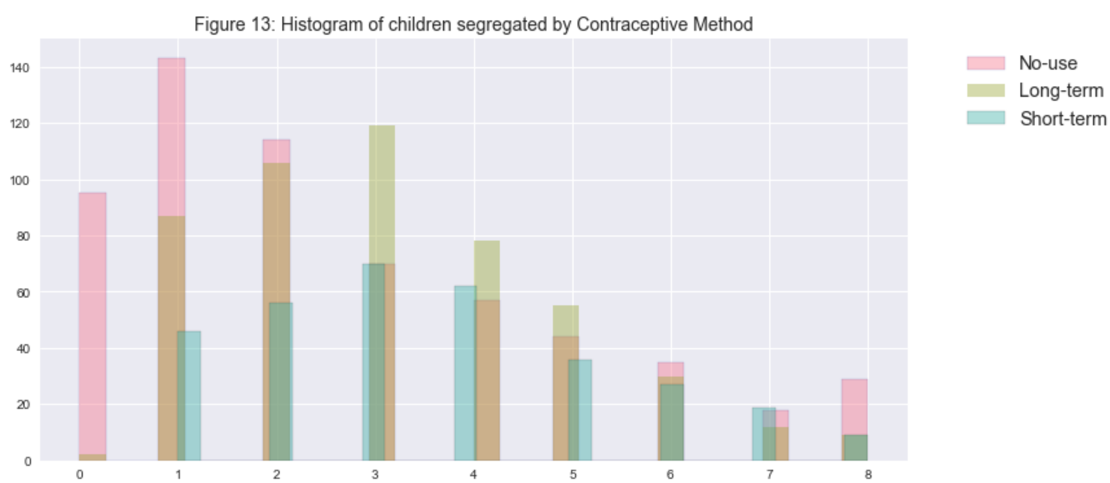
Below are histograms for two numerical features segregated by contraceptive method. As can be seen in Figure 59, highest proportion of low and high aged wifes are using no contraceptive method. In contrast, the highest proportion of middle aged wifes are using long-term contraceptive method, while short-term contraceptive methods seem to normally distribute against the wife's age.

Figure 60 depict the majority of wifes that have less than three children tends to use no contraceptive methods, while majority of wifes with 3 to 5 children are tends to use long-term contraceptive methods. Again, the short-term methods seems to be normally distributed.

```
In [21]: import numpy as np
         for col in ['wife_age', 'children']:
             data1 = df.loc[df['contrac_mthd']=="No-use", col]
             data2 = df.loc[df['contrac_mthd']=="Short-term", col]
```

```
data3 = df.loc[df['contrac_mthd']=="Long-term", col]
plt.figure(figsize=(12,6))
plt.hist(data1, alpha = 0.4, bins = 30,edgecolor="darkblue")
plt.hist(data2, alpha = 0.4, bins = 30)
plt.hist(data3, alpha = 0.4, bins = 30,edgecolor="black")
plt.title("Figure " + str(i) + ": Histogram of " + col +
          " segregated by Contraceptive Method",size=14)
i = i + 1
plt.legend(df['contrac_mthd'].unique(), bbox_to_anchor=(1.05, 1), loc=2,
           borderaxespad=0.5, prop = {'size':'large'})
plt.show()
```





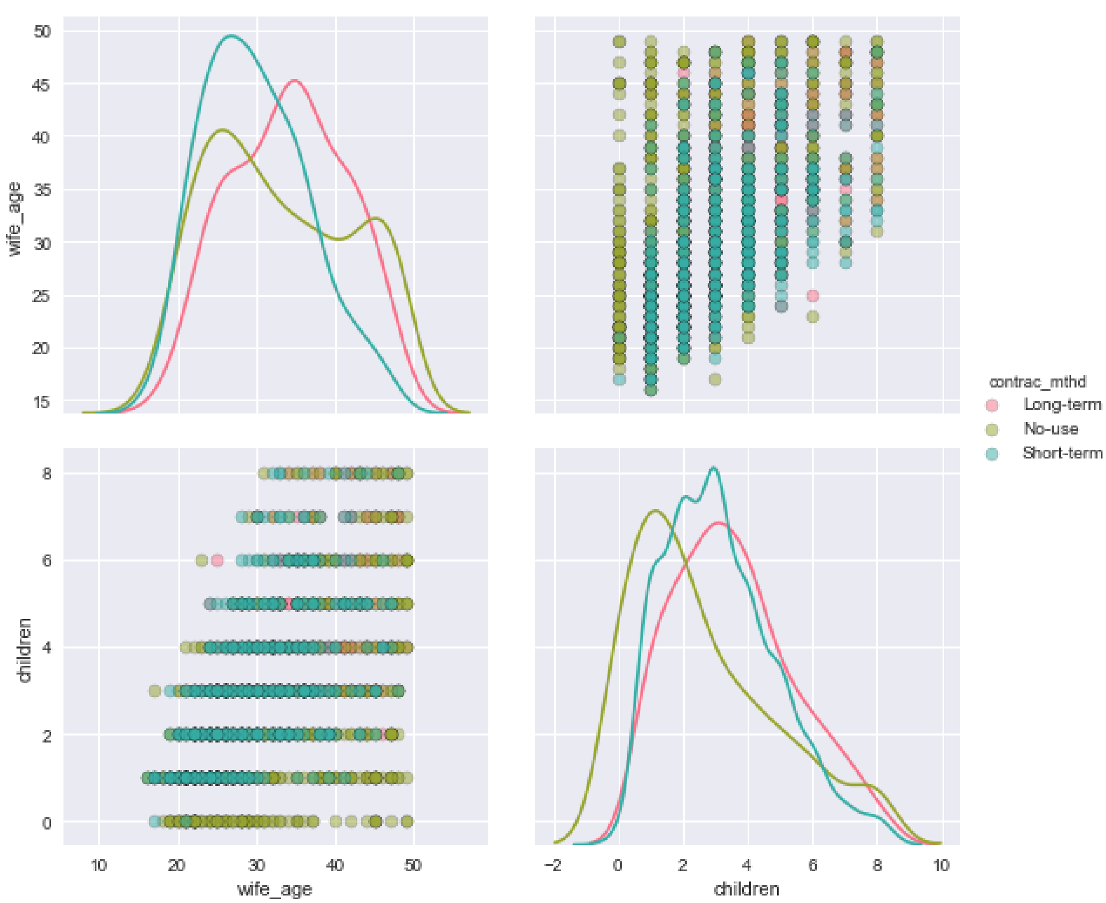
3.2.2 Pair Plots between Wife's Age & Number of Children by Contraceptive Method

Pair plots are useful to analyze the target feature against two numerical features. In this case, the contraceptive method against wife's age and number of children as can be seen in Figure 15. Both density graph and scatter plot for each numerical features are clearly shown in the figure. As can be seen in the first density graph, short-term contraceptive methods are comparatively in high use among young aged wifes (20 to 30 years), while long-term contraceptive methods are comparatively in high use among middle aged wifes (35 to 45 years). Majority of elderly wifes (above 45 years) tends to not use any contraceptive method.

The second density graph depict majority of wifes with less than two children do not use contraceptive method, while wifes with more than 4 children use long-term and short-term methods in fairly equal manner. Short-term contraceptive methods seems to be more popular among the wifes with children between 2 to 4.

```
In [22]: sns.pairplot(df, hue = 'contrac_mthd', diag_kind = 'kde',
                      plot_kws = {'alpha': 0.5, 's': 50, 'edgecolor': 'k'},
                      size = 4)
         plt.suptitle("Figure " + str(i) +
                      ": Pair Plot between Wife's Age and Number of children by Contraceptive Method",
                      size = 14, verticalalignment='top')
         plt.subplots_adjust(top=0.9)
         plt.show()
         i=i+1
```

Figure 14: Pair Plot between Wife's Age and Number of children by Contraceptive Method

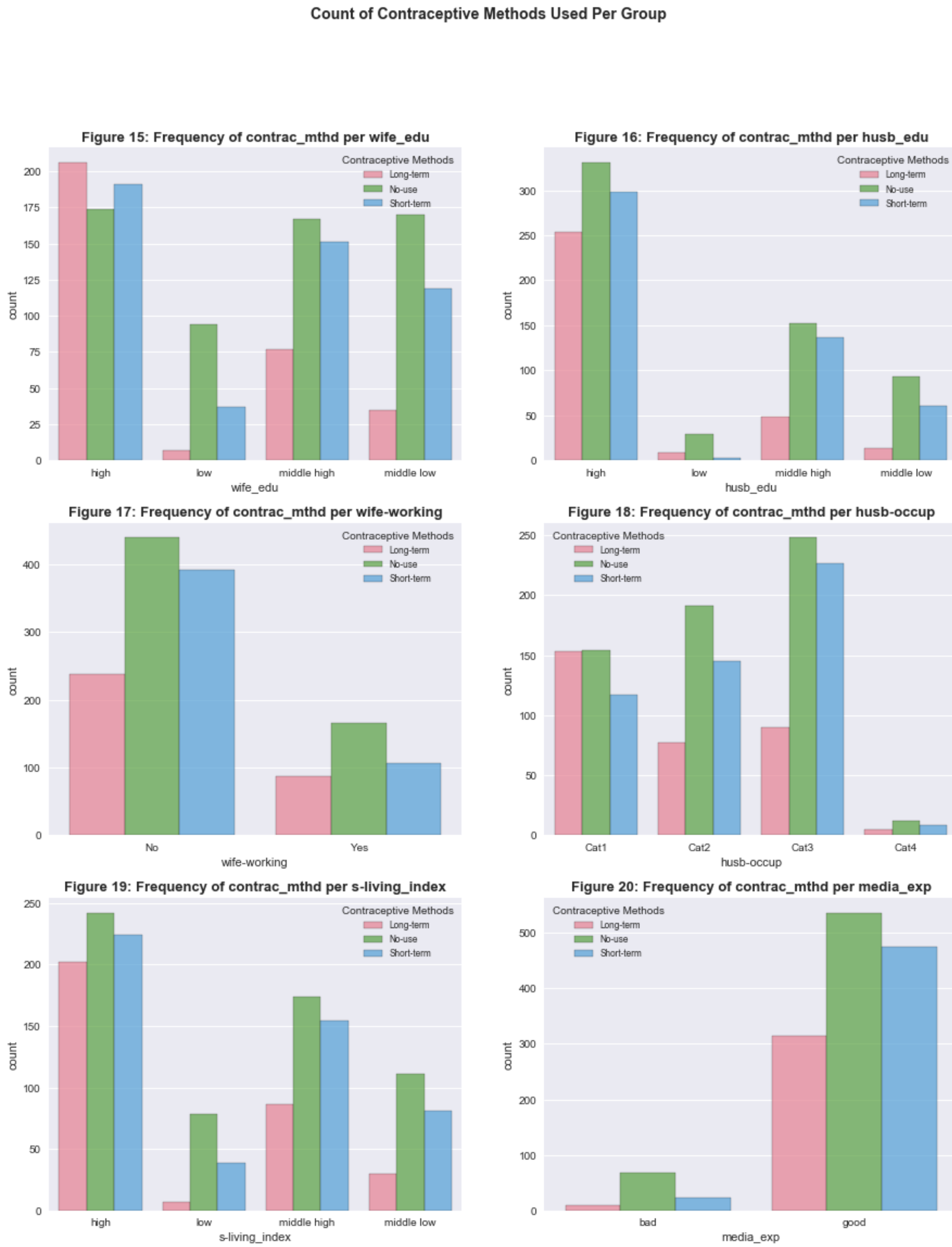


3.2.3 Count barplot of Categorical Features Segregated by Contraceptive Methods

The count barplots are useful to visualize and analyze categorical features against the target feature---contraceptive methods. Seven categorical features have been plotted and the count of each categorical classes and their target feature classification is clearly shown.

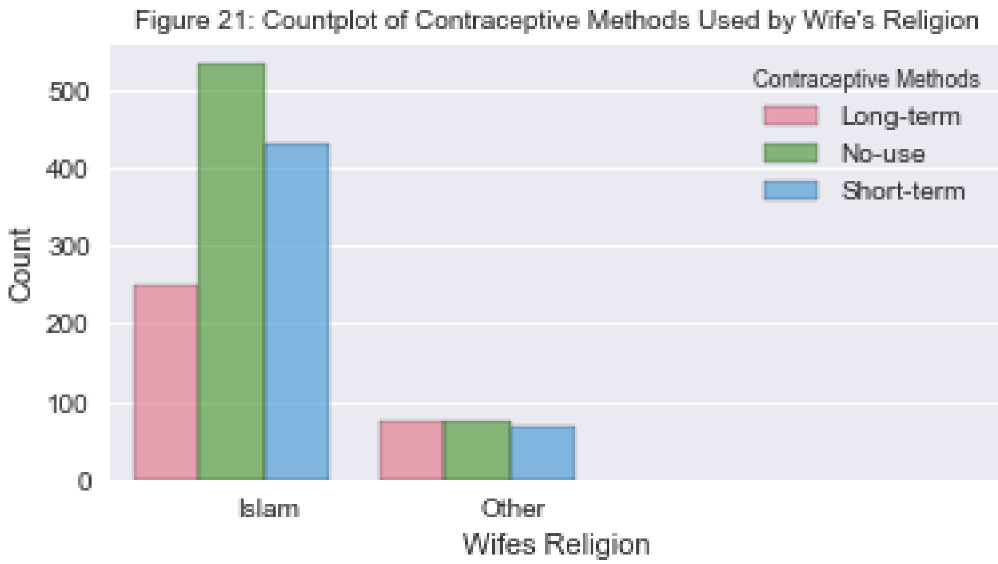
Figure 16 shows the long-term and short-term contraceptive methods are highly used among high education wife's, while the majority of wifes with low eduction, tends to use no contraceptive methods. As can be seen from figure 17 and figure 18, the husband's eduction level and wife's working status, seems to have no correlation with contraceptive method used by wifes. Similarly, other count plots do not shown direct correlation of categorical features to the target feature. However, if these plots are transfered to the proportion plots, then better insight could be obtained.

In [23]:
```
# Initialize Figure and Axes object
fig, ax = plt.subplots(3, 2, figsize=(15,18))
sns.set_palette("husl", 3)


for col, ax in zip(['wife_edu', 'husb_edu', 'wife-working','husb-occup','s-living_index',
                    'media_exp'], ax.flatten()):
    sns.countplot(x=col, hue='contrac_mthd', data=df, ax=ax,alpha=0.7,edgecolor="black")
    fig.suptitle("Count of Contraceptive Methods Used Per Group", weight='bold', size=14)
    ax.set_title("Figure " + str(i) + ": Frequency of contrac_mthd per " + col,
                 weight='bold', size=13)
    ax.legend(prop = {'size':'x-small'}).set_title('Contraceptive Methods',
                                                    prop = {'size':'small'})

    i=i+1
```




```
In [24]: N = 3.5
         ind = np.arange(N)  # the x locations for the groups
         width = 0.30        # the width of the bars
         sns.set(font_scale = 1)
         sns.set_palette("husl", 3)
```

```
fig = plt.gcf()
fig.set_size_inches( 6, 3)


fig=sns.countplot(x='wife_religion',hue='contrac_mthd',data=df,alpha=0.7,edgecolor="black")

fig.set_xlabel('Wifes Religion')
fig.set_ylabel('Count')
fig.set_title("Figure " + str(i) +
              ": Countplot of Contraceptive Methods Used by Wife's Religion", size=10)
fig.set_xticks(ind + width / 2)
fig.legend().set_title('Contraceptive Methods', prop = {'size':'x-small'})

i=i+1
```



3.2.4 Propotional barplot of Categorical Features Segregated by Contraceptive Methods

Above count plots have been transformed to the proportional plots in order to obtain better insight by comparing the normalized values instead of counts. This means each category feature is now plotted to show their proportion of occurrences instead of the actual count as shown in previous section.

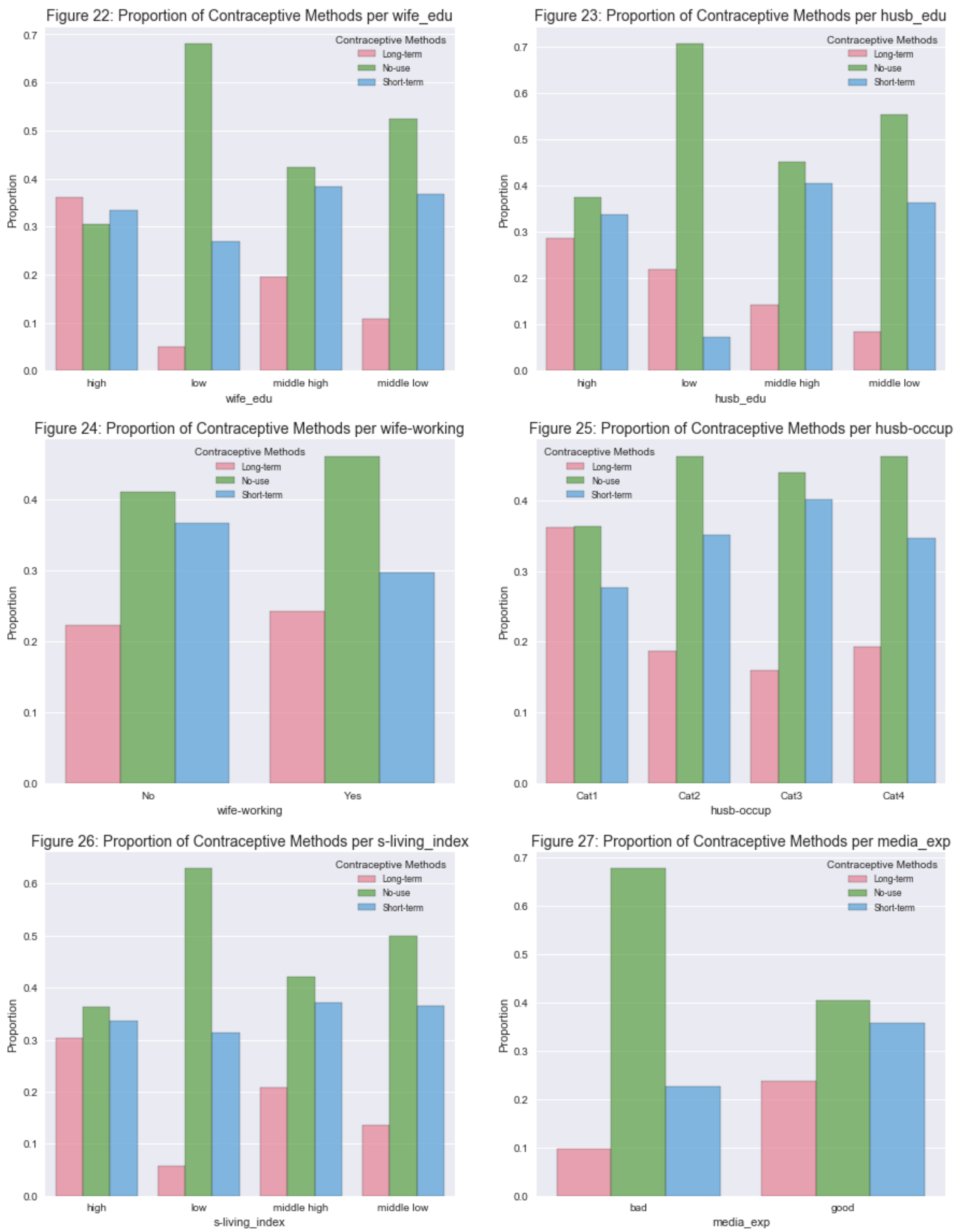
Figure 59 and 60 clearly show that the proportion of wifes not using contraceptive method has direct correlation with wife's and husband's eduction level. Lower the education level of wife's and husband's, the higher the proportion of not using contraceptive methods. Similarly, according to figure 63, standard of living seems to show clear correlation to the target feature: this means higher the standard of living, lower the proportion of not using contraceptive methods. The relationship between exposure to the medial and contraceptive method used is shown in figure 64. It also shows an increase of the proportion of using short and long term contraceptive methods with better exposure to media.

Finally, according to Figure 65, non-Islam wifes in the data-set have higher proportion of using long-term contraceptive methods, while they also have lesser proportion of using no contraceptive methods.

```
In [25]: # Initialize Figure and Axes object
         df_new=df.copy()
```

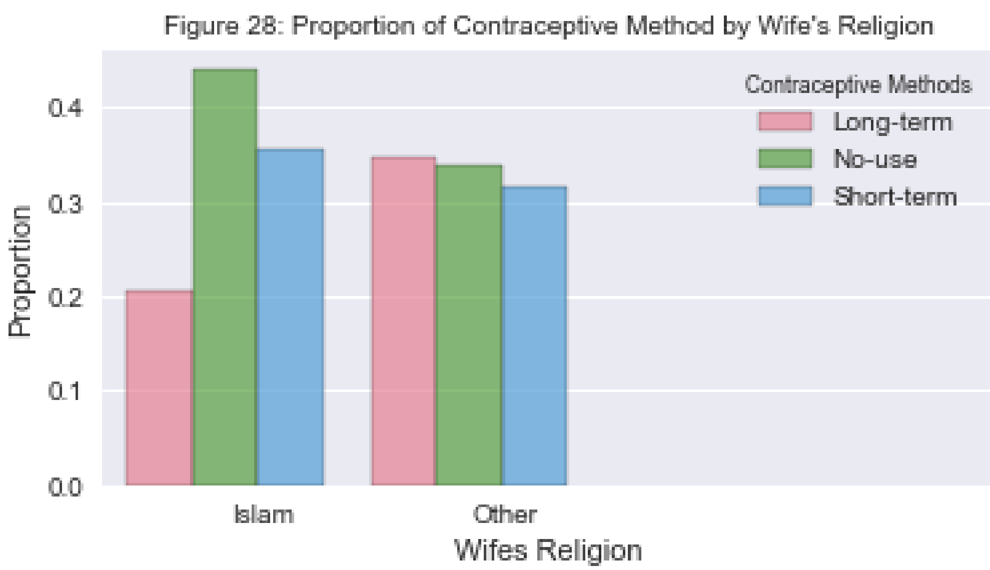
```
df_new['Proportion'] = 0 # a dummy column to refer to
fig, ax = plt.subplots(3, 2, figsize=(15,20))
sns.set_palette("husl", 3)

for col, ax in zip(['wife_edu', 'husb_edu', 'wife-working','husb-occup','s-living_index',
                    'media_exp'], ax.flatten()):
    counts = df_new.groupby([col, 'contrac_mthd']).count()
    group_freq = counts.div(counts.groupby(col).transform('sum')).reset_index()
    sns.barplot(x=col, y='Proportion', hue='contrac_mthd', data=group_freq,
                ax=ax,alpha=0.7,edgecolor="black")
    fig.suptitle("Proportion of Contraceptive Methods Used per Group",
                 size=14)
    ax.set_title("Figure " + str(i) + ": Proportion of Contraceptive Methods per " +
                 col, size=14)
    ax.legend(prop = {'size':'x-small'}).set_title('Contraceptive Methods',
                                                   prop = {'size':'small'})

    i=i+1
```

Proportion of Contraceptive Methods Used per Group




```
In [26]: N = 3.5
```

```
ind = np.arange(N)  # the x locations for the groups
width = 0.30        # the width of the bars
sns.set(font_scale = 1)
sns.set_palette("husl", 3)
fig = plt.gcf()
fig.set_size_inches( 6, 3)

counts = df_new.groupby(['wife_religion', 'contrac_mthd']).count()
group_freq = counts.div(counts.groupby('wife_religion').transform('sum')).reset_index()

fig=sns.barplot(x='wife_religion', y='Proportion', hue='contrac_mthd',
                data=group_freq,alpha=0.7,edgecolor="black")
fig.set_xlabel('Wifes Religion')
fig.set_ylabel('Proportion')
fig.set_title("Figure " + str(i) +
              ": Proportion of Contraceptive Method by Wife's Religion", size=10)
fig.set_xticks(ind + width / 2)
fig.legend().set_title('Contraceptive Methods', prop = {'size':'x-small'})
plt.show()
i=i+1
```



3.2.5 Interaction between Categorical and Numeric Features

The relationships between categorical and numerical features have been visually represented in below grouped boxplots. Each boxplot have been plotted segregated by contraceptive methods. It is easily to observe that wife's with lower eduction level tends to use short-term methods in early stage of their lives and use no methods as getting older. Also, from Figure 35 onwards show that the short and long-term contraceptive methods are much used when number of children in the family are between 3 to 6 across most of the categorical features.These boxplots will be really useful in Phase 2, when developing the classification model.

```
In [27]: plt.rcParams["font.family"] = "DejaVu Sans"
         sns.set(font_scale = 1.1)
```

```
for col in ['wife_age','children']:
    for k in ['wife_edu', 'husb_edu', 'wife-working',
              'husb-occup','s-living_index','media_exp']:
        ax = df.groupby('contrac_mthd').boxplot(column = col, by = k,
                                                vert = False,

                                                figsize=(15,10))
        plt.suptitle("Figure " + str(i) + ": Box Plot of " + col + " grouped by " + k +
                     " and segregated by Contraceptive Method"
                    )
        plt.yticks()
        plt.show()
        i = 1 + i
```

Figure 29: Box Plot of wife_age grouped by wife_edu and segregated by Contraceptive Method

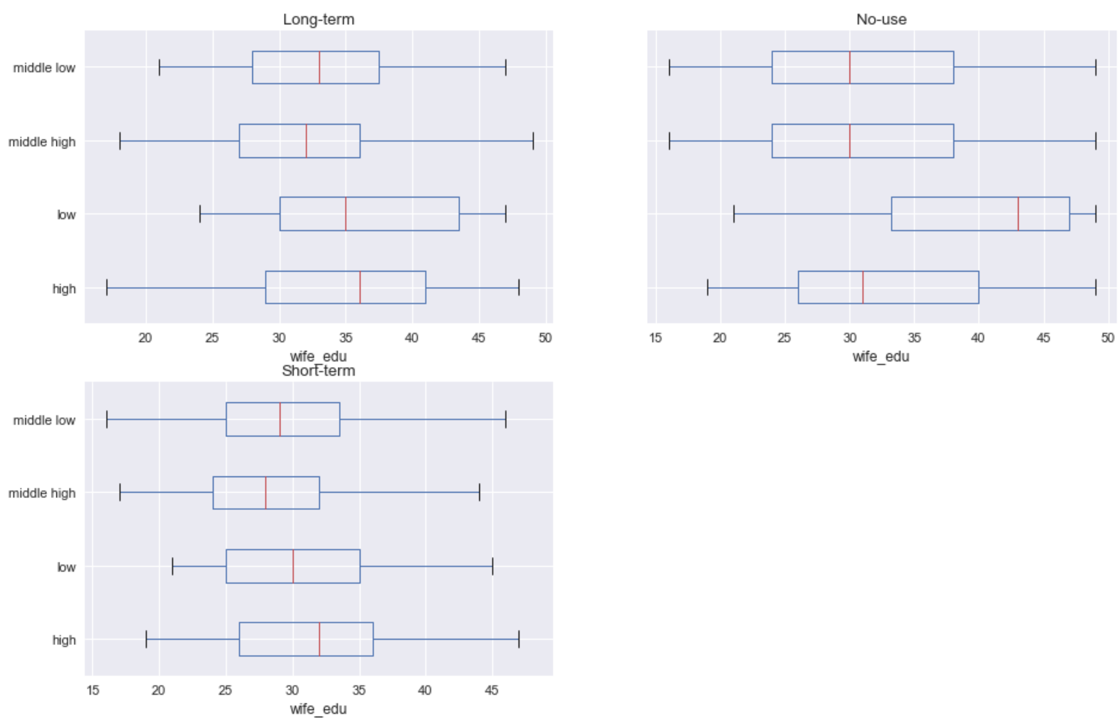


Figure 30: Box Plot of wife_age grouped by husb_edu and segregated by Contraceptive Method

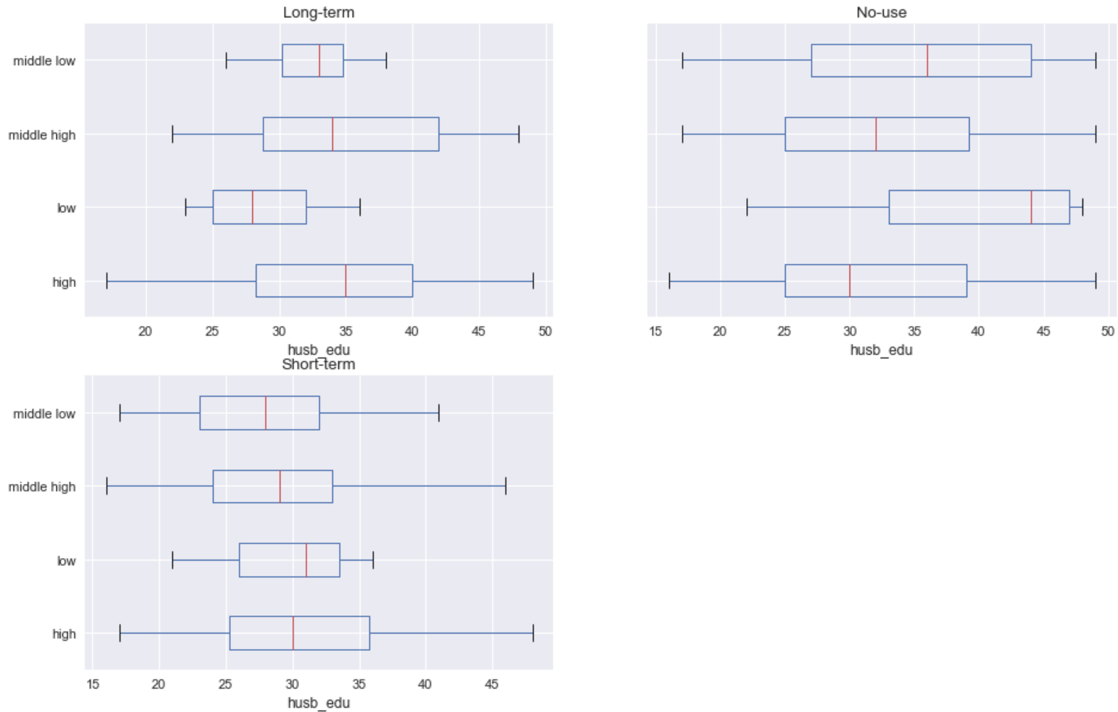


Figure 31: Box Plot of wife_age grouped by wife-working and segregated by Contraceptive Method

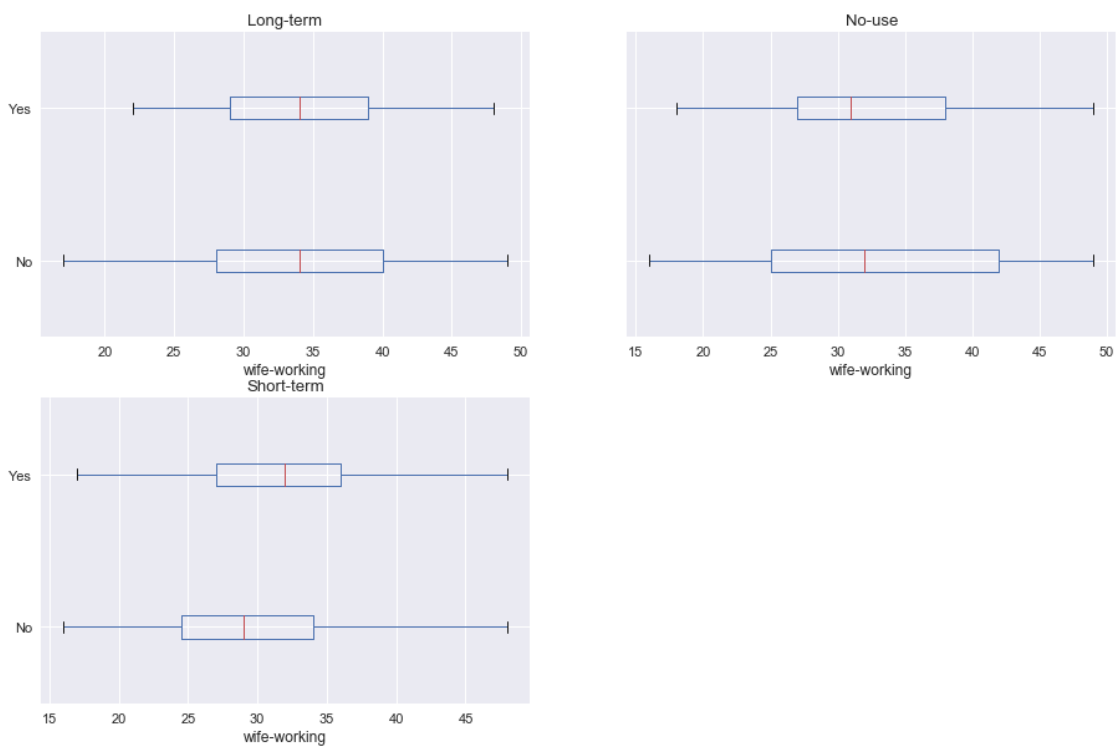


Figure 32: Box Plot of wife_age grouped by husb-occup and segregated by Contraceptive Method

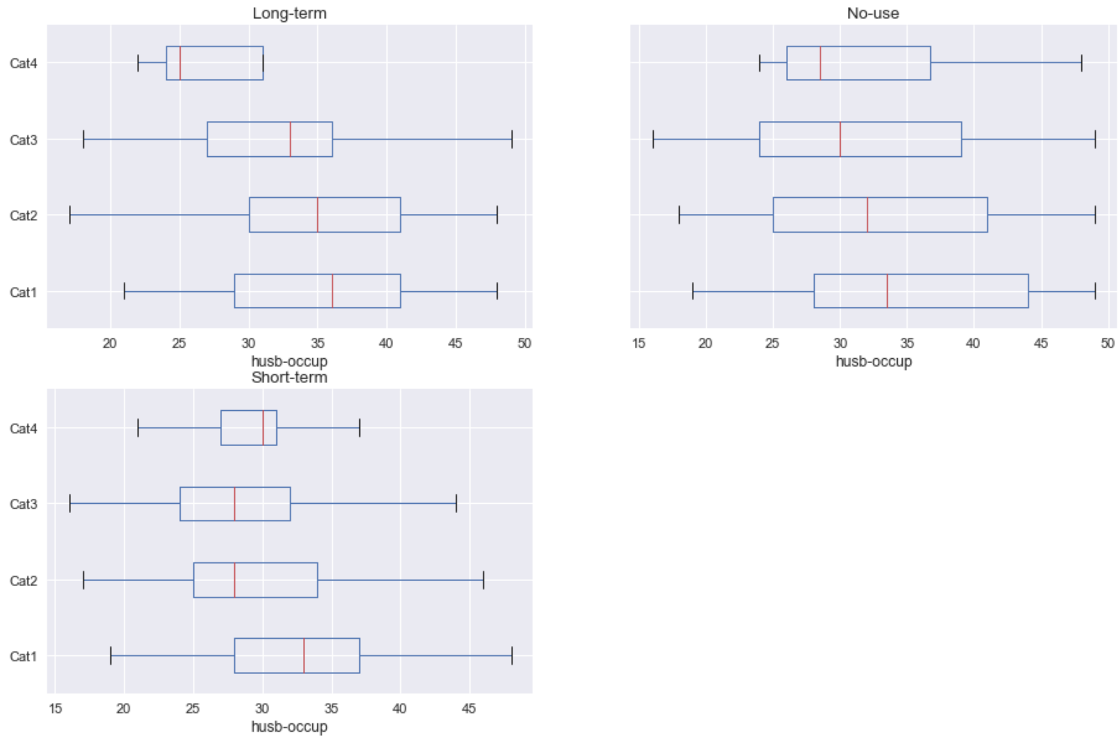


Figure 33: Box Plot of wife_age grouped by s-living_index and segregated by Contraceptive Method

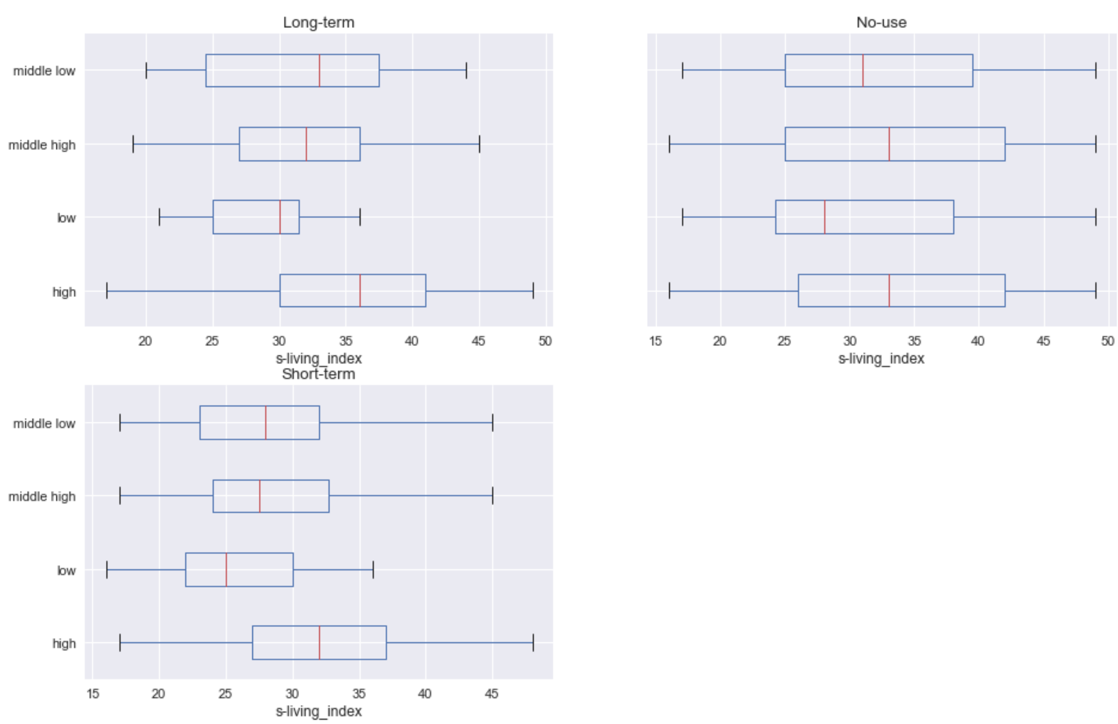


Figure 34: Box Plot of wife_age grouped by media_exp and segregated by Contraceptive Method

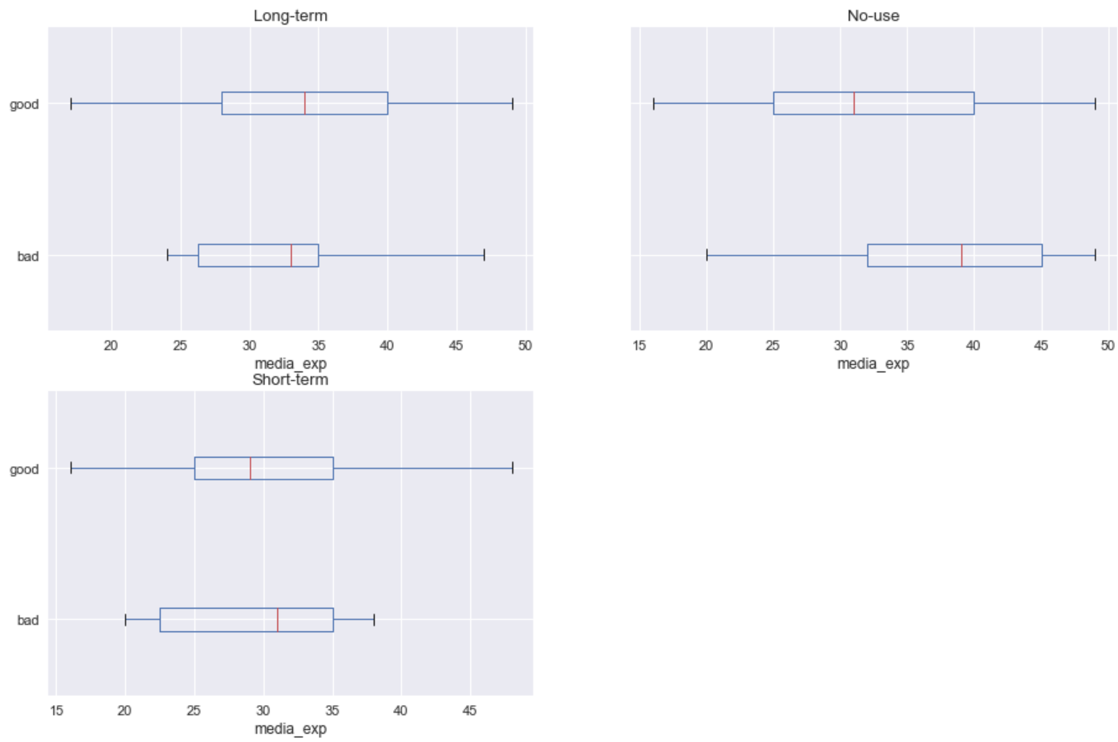


Figure 35: Box Plot of children grouped by wife_edu and segregated by Contraceptive Method

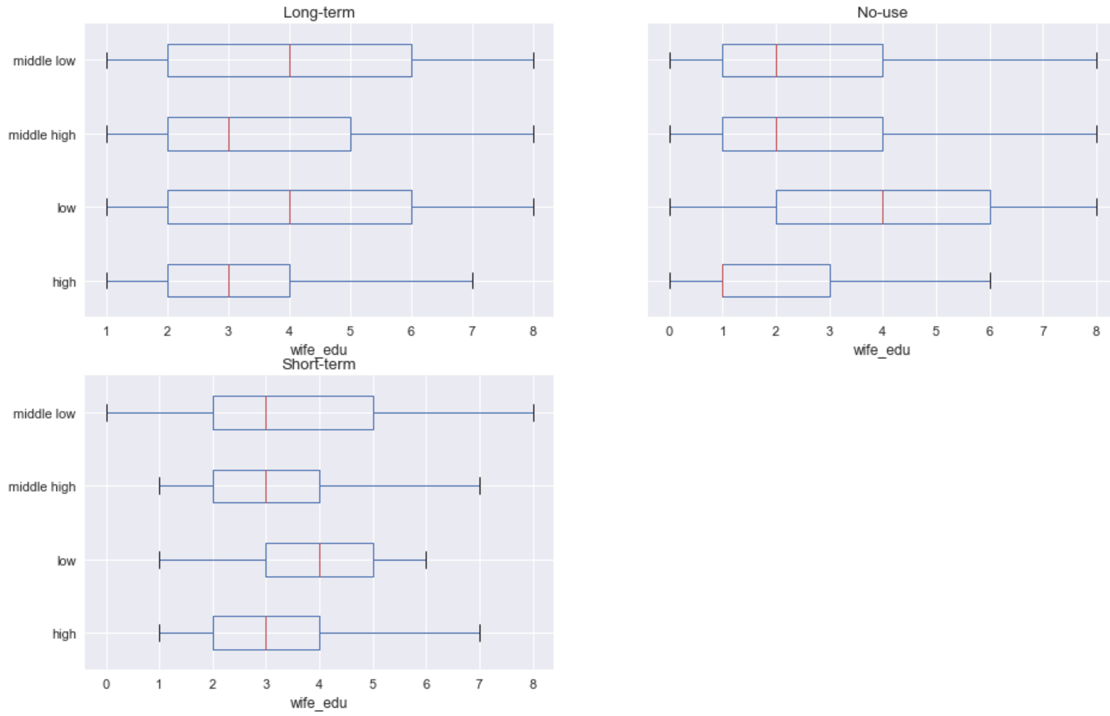


Figure 36: Box Plot of children grouped by husb_edu and segregated by Contraceptive Method

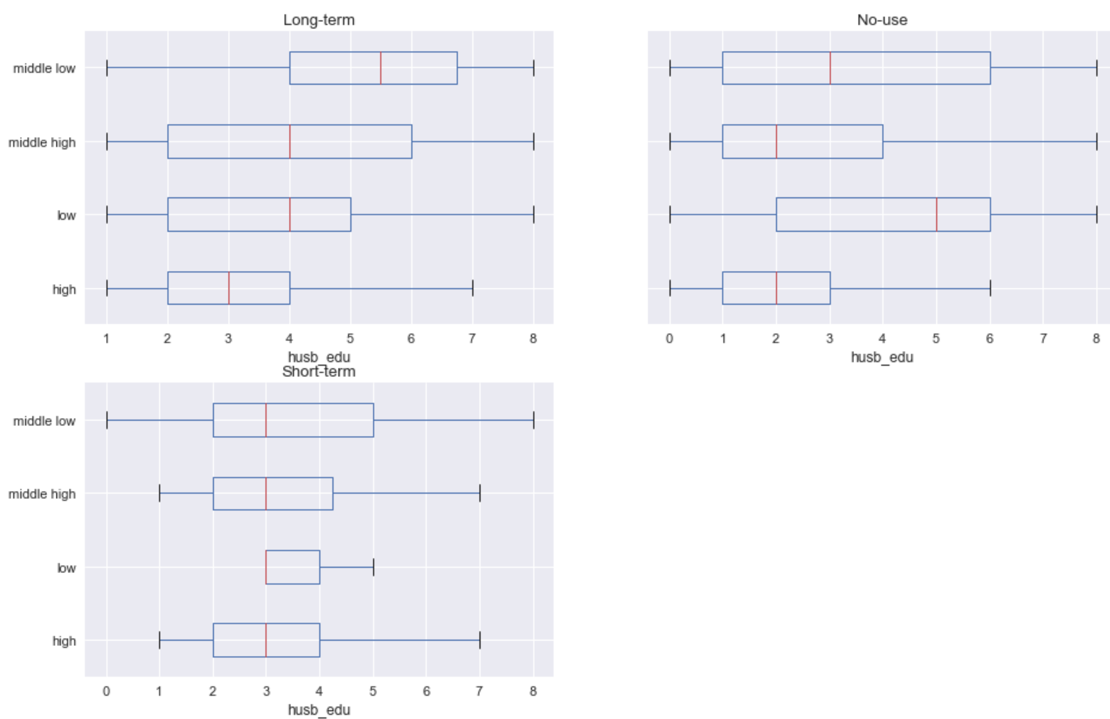


Figure 37: Box Plot of children grouped by wife-working and segregated by Contraceptive Method

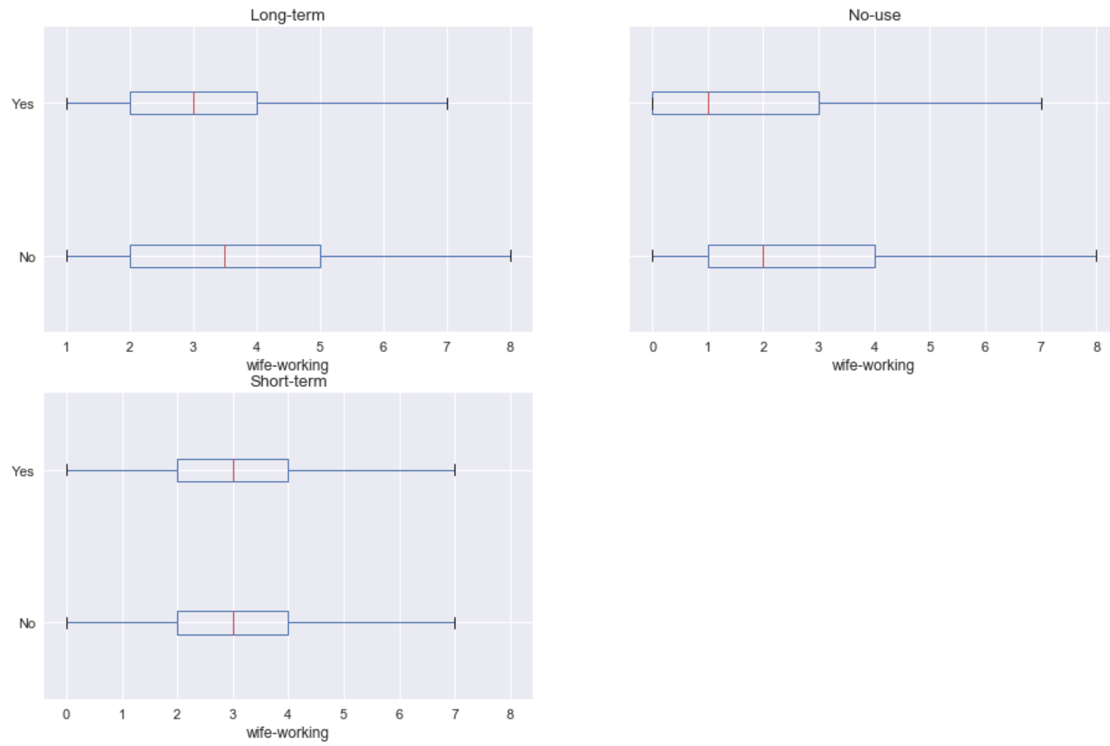


Figure 38: Box Plot of children grouped by husb-occup and segregated by Contraceptive Method

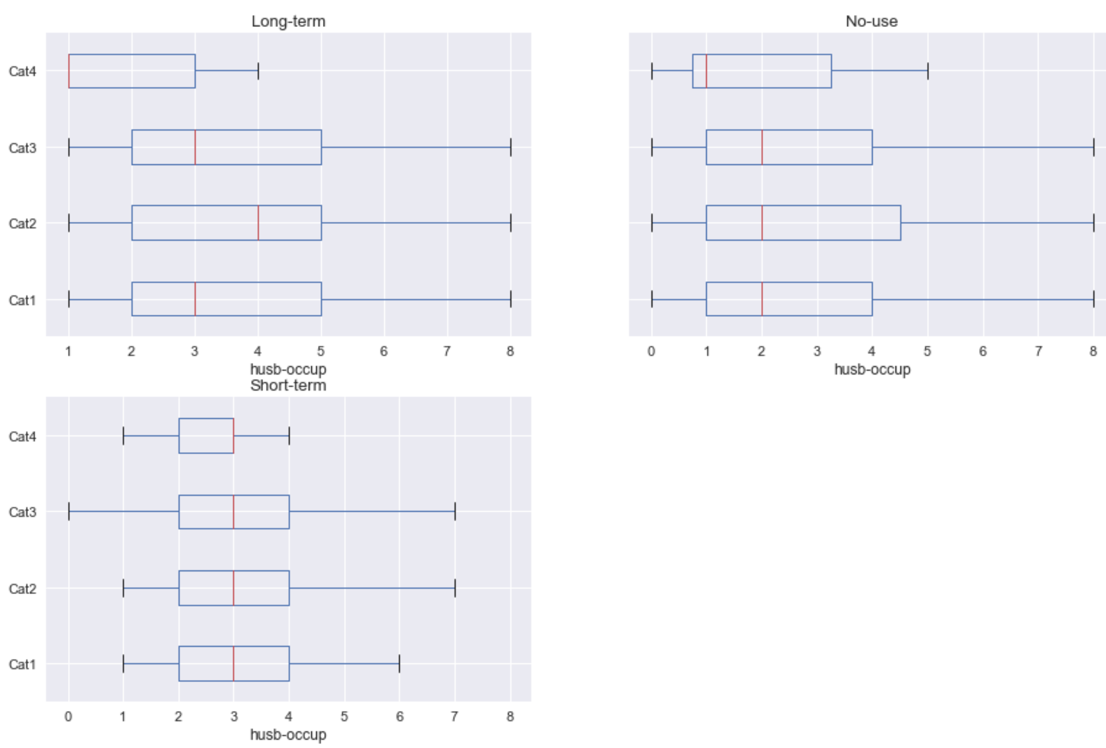


Figure 39: Box Plot of children grouped by s-living_index and segregated by Contraceptive Method

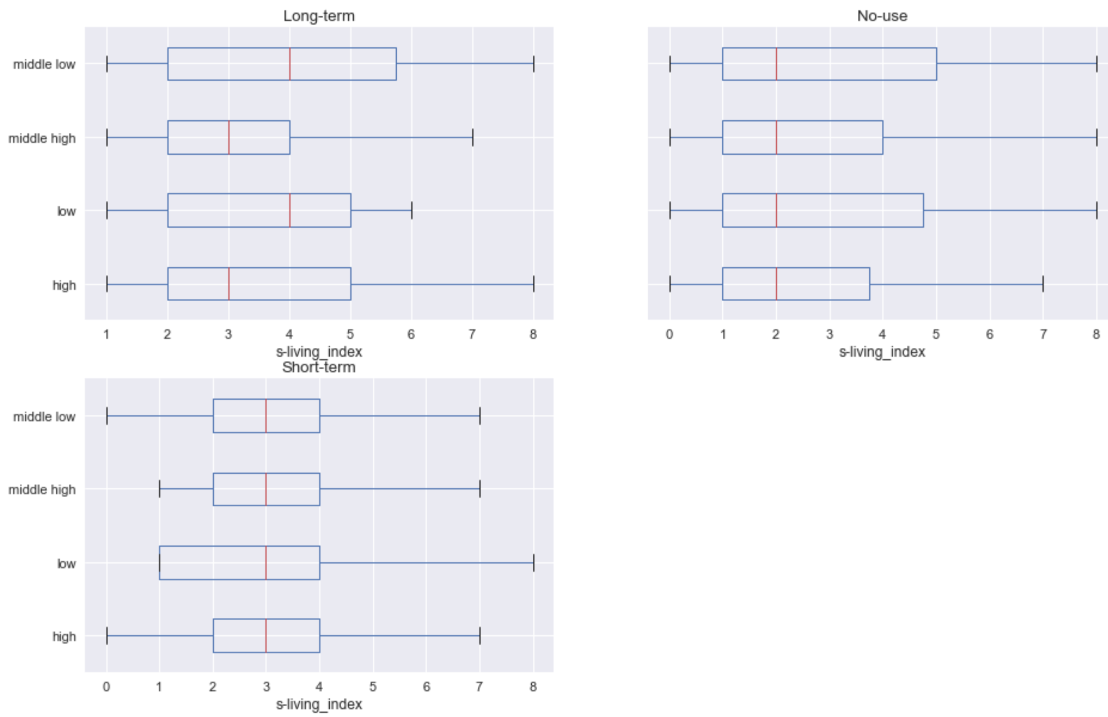
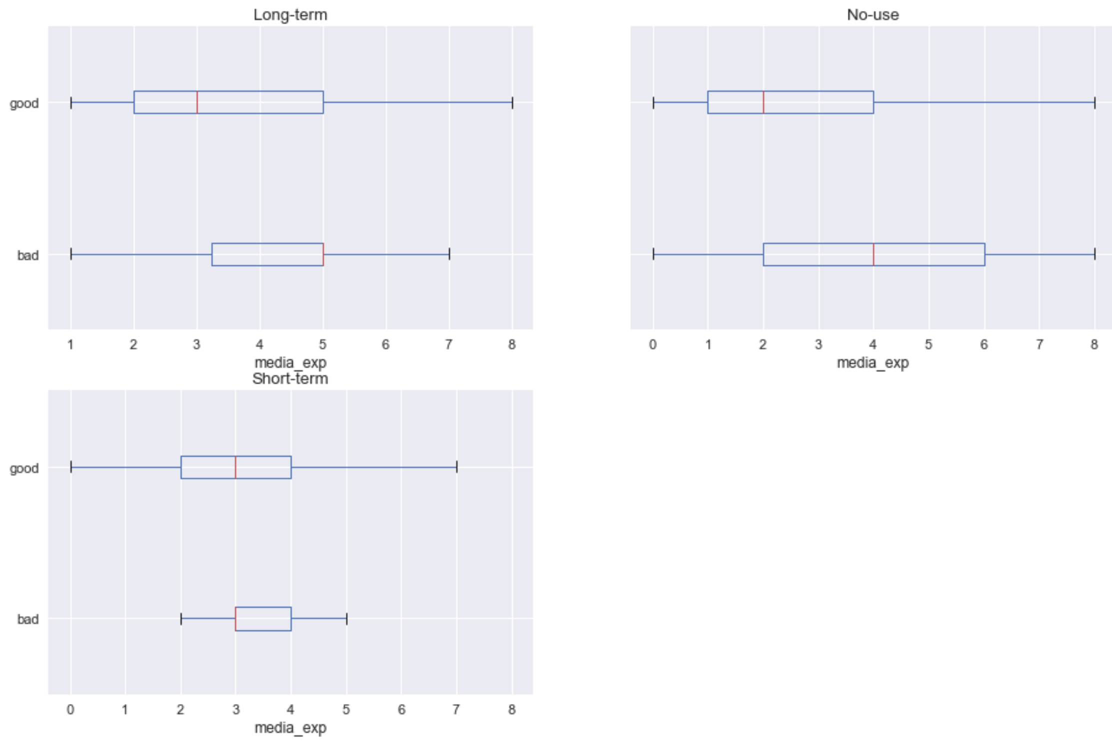


Figure 40: Box Plot of children grouped by media_exp and segregated by Contraceptive Method



Chapter 4

Summary

In Phase 1, the raw data-set was loaded to phython using pandas and descriptive labes have been introduced and data types have been changed to match description of the original data-set. Next, the data pre-processing was done to handle missing values, typos, and outliers. Finally, the categorical and numerical features have been visualized using different plotting methods to visually undestand any correlation and characteristics between descriptive and target features.

Bibliography