Ex 1

April 15, 2020

Churn dataset preprocessing

1 Ex 1

1.1 Churn dataset preprocessing

1.1.1 A documented code to show what should we do in churn dataset preprocessing

First of all, we check the columns and in general the shape of the dataset to see what are we dealing with

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt
```

[2]:
```
churn = pd.read_csv('/home/hakim/Documents/semester 8/DM/HW_2/churn.csv')
```

1.2 Data columns

As we can see below, the churn dataset having too many columns what should we do first of all is to **drop some of the columns** because we do not need all of them and they may overfit or underfit our model.

[3]:
```
print('dataset shape: ',churn.shape)
print('\n\ndataset columns: ')
for c in range(len(churn.columns)):
    print([c], churn.columns[c])
```

```
dataset shape:  (3333, 21)


dataset columns:
[0] State
[1] Account Length
[2] Area Code
```

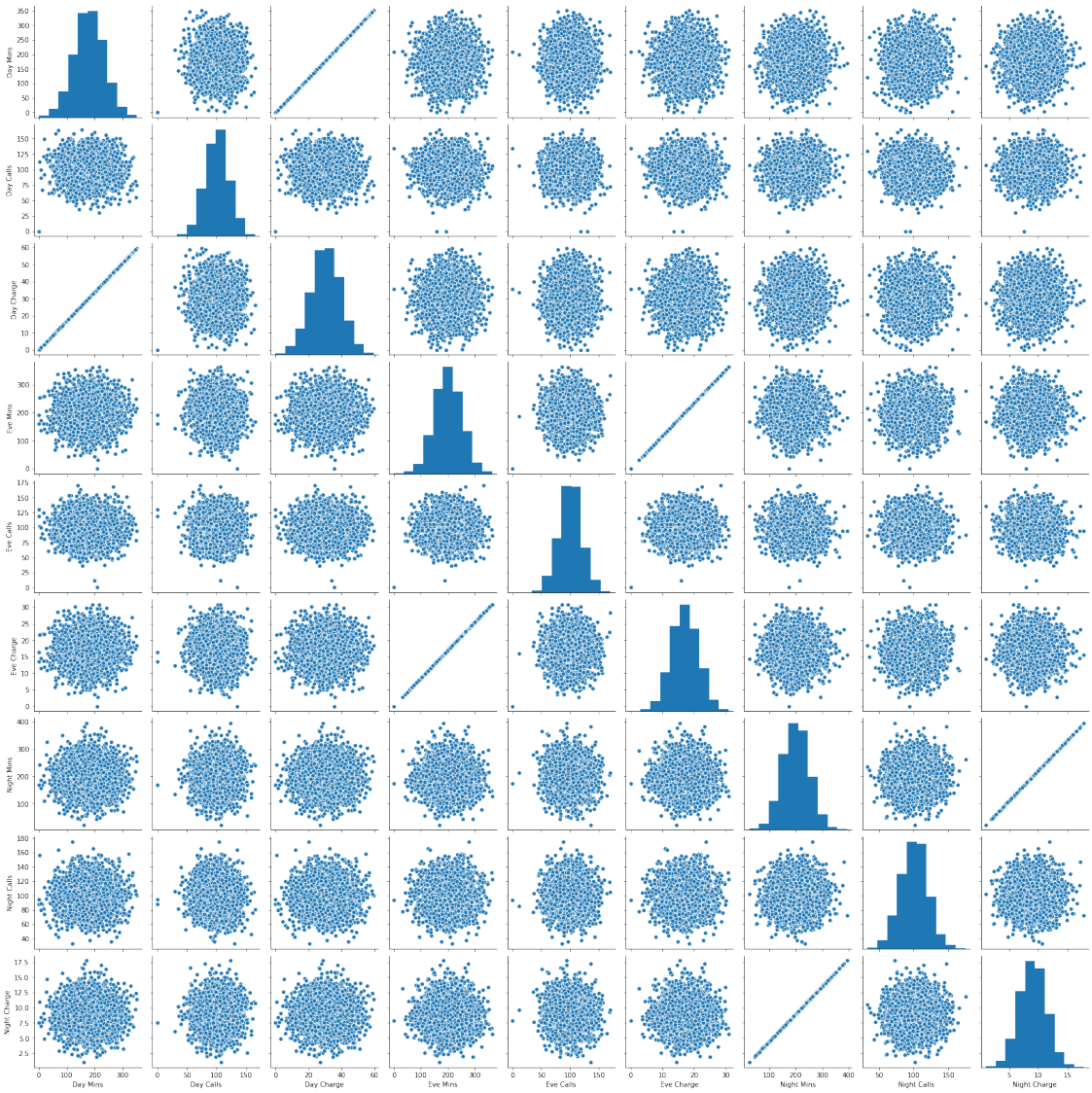
```
[3] Phone
[4] Int'l Plan
[5] VMail Plan
[6] VMail Message
[7] Day Mins
[8] Day Calls
[9] Day Charge
[10] Eve Mins
[11] Eve Calls
[12] Eve Charge
[13] Night Mins
[14] Night Calls
[15] Night Charge
[16] Intl Mins
[17] Intl Calls
[18] Intl Charge
[19] CustServ Calls
[20] Churn?
```

1.3 Drop columns

in the below code we drop some of the columns to see the more important and relevant values

[4]:
```
to_drop = ['State', 'Account Length', 'Area Code', 'Phone', "Int'l Plan",
       'VMail Plan', 'VMail Message', 'Intl Mins', 'Intl Calls', 'Intl Charge',
       'CustServ Calls', 'Churn?']
churn = churn.drop(to_drop, axis=1)
```

[5]:
```
sb.pairplot(churn)
plt.show()
```



1.4 all possible values in the columns

in here we read dataset again to see all of the possible values in each column

[6]:
```
churn = pd.read_csv('/home/hakim/Documents/semester 8/DM/HW_2/churn.csv')
```

[7]:
```
np.set_printoptions(threshold=160)
colls = churn.columns
for c in range(len(colls)):
    print('\n\n',colls[c], ': ')
    print(np.sort( churn[colls[c]].unique()))
```

```
 State :
['AK' 'AL' 'AR' 'AZ' 'CA' 'CO' 'CT' 'DC' 'DE' 'FL' 'GA' 'HI' 'IA' 'ID'
 'IL' 'IN' 'KS' 'KY' 'LA' 'MA' 'MD' 'ME' 'MI' 'MN' 'MO' 'MS' 'MT' 'NC'
 'ND' 'NE' 'NH' 'NJ' 'NM' 'NV' 'NY' 'OH' 'OK' 'OR' 'PA' 'RI' 'SC' 'SD'
 'TN' 'TX' 'UT' 'VA' 'VT' 'WA' 'WI' 'WV' 'WY']


 Account Length :
[  1   2   3 … 225 232 243]


 Area Code :
[408 415 510]


 Phone :
['327-1058' '327-1319' '327-3053' … '422-8333' '422-8344' '422-9964']


 Int'l Plan :
['no' 'yes']


 VMail Plan :
['no' 'yes']


 VMail Message :
[ 0  4  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51]


 Day Mins :
[  0.    2.6   7.8 … 345.3 346.8 350.8]


 Day Calls :
[  0  30  35  36  40  42  44  45  47  48  49  51  52  53  54  55  56  57
  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75
  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93
  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110 111
 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129
 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147
 148 149 150 151 152 156 157 158 160 163 165]
```

```
 Day Charge :
[ 0.    0.44  1.33 … 58.7  58.96 59.64]


 Eve Mins :
[  0.   31.2  42.2 … 354.2 361.8 363.7]


 Eve Calls :
[  0  12  36  37  42  43  44  45  46  48  49  50  51  52  53  54  55  56
  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74
  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92
  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110
 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128
 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146
 147 148 149 150 151 152 153 154 155 156 157 159 164 168 170]


 Eve Charge :
[ 0.    2.65  3.59 … 30.11 30.75 30.91]


 Night Mins :
[ 23.2  43.7  45.  … 377.5 381.9 395. ]


 Night Calls :
[ 33  36  38  42  44  46  48  49  50  51  52  53  54  55  56  57  58  59
  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77
  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95
  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110 111 112 113
 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131
 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149
 150 151 152 153 154 155 156 157 158 164 166 175]


 Night Charge :
[ 1.04  1.97  2.03 … 16.99 17.19 17.77]


 Intl Mins :
[ 0.   1.1  1.3 … 18.4 18.9 20. ]


 Intl Calls :
[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
```

```
 Intl Charge :
[0.   0.3  0.35 … 4.97 5.1  5.4 ]


 CustServ Calls :
[0 1 2 3 4 5 6 7 8 9]


 Churn? :
['False.' 'True.']
```

1.4.1 Data Cleaning

due to outliers and also a large number of columns and values, based on what we are going to do, we should drop a part of our dataset to be ready for further analysis.

1.4.2 Aggregation

again base on our type of analysis and our target value, we could aggregate some of the values and make our analysis faster and better.

1.4.3 Discretization

Discretization can be a very big help for analyzing this dataset. due to the large amount of data that we have. we can discretize most of the date and perform the classification and other types of data analyzing on it so much better and faster.

1.4.4 Binarization

Binarization can help too by dividing the variables into two types of groupe so that we can classify the data much better.

1.4.5 Dimensionality Reduction

I do not have any comment on that because according to the plot above, I don't think that could be helpfull. but again, I am not an expert (yet!) so that is just my oppinion.

1.5 conclusion

As told above, we can perform some of the preprocessing processes on this dataset for making our work easier and much better.