Final Project Submission

Please fill out:

- Student name: Tia Plagata
- Student pace: full time
- Scheduled project review date/time:
- Instructor name: Rafael Carrasco
- Blog post URL:

Data Cleaning (Obtain, Scrub)

Cleaning Outline

- Check data types
- Check for null values
- Check for duplicates
- Check for placeholder values/nonsensical values
- Check for imbalance of churn True vs False
- Check for outlier

In [1]:

```
# Import Statements
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
df = pd.read_csv('/Users/jordanrjohnson/DataScienceCourseMaterial/phase_3/ds

df.head()
```

Out[2]:

	state	account length	area code	phone number	international plan	voice mail plan	number vmail messages	total day minutes	total day calls	total day charge	...	tc e ca
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	...	
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	...	1
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	...	1
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	...	
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	...	1

5 rows × 21 columns

In [12]:

```
# Check data types
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
Data columns (total 21 columns):
state                     3333 non-null object
account length            3333 non-null int64
area code                 3333 non-null int64
phone number              3333 non-null object
international plan        3333 non-null object
voice mail plan           3333 non-null object
number vmail messages     3333 non-null int64
total day minutes         3333 non-null float64
total day calls           3333 non-null int64
total day charge          3333 non-null float64
total eve minutes         3333 non-null float64
total eve calls           3333 non-null int64
total eve charge          3333 non-null float64
total night minutes       3333 non-null float64
total night calls         3333 non-null int64
total night charge        3333 non-null float64
total intl minutes        3333 non-null float64
total intl calls          3333 non-null int64
total intl charge         3333 non-null float64
customer service calls    3333 non-null int64
churn                     3333 non-null bool
dtypes: bool(1), float64(8), int64(8), object(4)
memory usage: 524.2+ KB
```

In [26]:

```
# Check for null values
df.isna().sum()
```

Out[26]:

```
state                     0
account length            0
area code                 0
phone number              0
international plan        0
voice mail plan           0
number vmail messages     0
total day minutes         0
total day calls           0
total day charge          0
total eve minutes         0
total eve calls           0
total eve charge          0
total night minutes       0
total night calls         0
total night charge        0
total intl minutes        0
total intl calls          0
total intl charge         0
customer service calls    0
churn                     0
dtype: int64
```

In [10]:

```
# Check for duplicates
df.duplicated().sum()
```

Out[10]:

```
0
```

In [25]:

```
# Check for nonsensical or placeholder values
for col in df.columns:
    print(col)
    print(df[col].unique())
    print('\n-------------------------------------------------\n')
```

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

-------------------------------------------------

account length
[128 107 137  84  75 118 121 147 117 141  65  74 168  95  62 161  85
93
  76  73  77 130 111 132 174  57  54  20  49 142 172  12  72  36  78 1
36
 149  98 135  34 160  64  59 119  97  52  60  10  96  87  81  68 125 1
16
  38  40  43 113 126 150 138 162  90  50  82 144  46  70  55 106  94 1
55
  80 104  99 120 108 122 157 103  63 112  41 193  61  92 131 163  91 1
27
 110 140  83 145  56 151 139   6 115 146 185 148  32  25 179  67  19 1
70
 164  51 208  53 105  66  86  35  88 123  45 100 215  22  33 114  24 1
01
 143  48  71 167  89 199 166 158 196 209  16  39 173 129  44  79  31 1
24
  37 159 194 154  21 133 224  58  11 109 102 165  18  30 176  47 190 1
52
  26  69 186 171  28 153 169  13  27   3  42 189 156 134 243  23   1 2
05
 200   5   9 178 181 182 217 177 210  29 180   2  17   7 212 232 192 1
95
 197 225 184 191 201  15 183 202   8 175   4 188 204 221]

-------------------------------------------------

area code
[415 408 510]

-------------------------------------------------

phone number
['382-4657' '371-7191' '358-1921' ... '328-8230' '364-6381' '400-434
4']

-------------------------------------------------

international plan
['no' 'yes']

-------------------------------------------------

voice mail plan
['yes' 'no']
```

```
------------------------------------------------

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

------------------------------------------------

total day minutes
[265.1 161.6 243.4 ... 321.1 231.1 180.8]

------------------------------------------------

total day calls
[110 123 114  71 113  98  88  79  97  84 137 127  96  70  67 139  66
90
 117  89 112 103  86  76 115  73 109  95 105 121 118  94  80 128  64 1
06
 102  85  82  77 120 133 135 108  57  83 129  91  92  74  93 101 146
72
  99 104 125  61 100  87 131  65 124 119  52  68 107  47 116 151 126 1
22
 111 145  78 136 140 148  81  55  69 158 134 130  63  53  75 141 163
59
 132 138  54  58  62 144 143 147  36  40 150  56  51 165  30  48  60
42
   0  45 160 149 152 142 156  35  49 157  44]

------------------------------------------------

total day charge
[45.07 27.47 41.38 ... 54.59 39.29 30.74]

------------------------------------------------

total eve minutes
[197.4 195.5 121.2 ... 153.4 288.8 265.9]

------------------------------------------------

total eve calls
[ 99 103 110  88 122 101 108  94  80 111  83 148  71  75  76  97  90
65
  93 121 102  72 112 100  84 109  63 107 115 119 116  92  85  98 118
74
 117  58  96  66  67  62  77 164 126 142  64 104  79  95  86 105  81 1
13
 106  59  48  82  87 123 114 140 128  60  78 125  91  46 138 129  89 1
33
 136  57 135 139  51  70 151 137 134  73 152 168  68 120  69 127 132 1
43
  61 124  42  54 131  52 149  56  37 130  49 146 147  55  12  50 157 1
55
  45 144  36 156  53 141  44 153 154 150  43   0 145 159 170]

------------------------------------------------

total eve charge
[16.78 16.62 10.3  ... 13.04 24.55 22.6 ]
```

```
------------------------------------------------

total night minutes
[244.7 254.4 162.6 ... 280.9 120.1 279.1]

------------------------------------------------

total night calls
[ 91 103 104  89 121 118  96  90  97 111  94 128 115  99  75 108  74 1
33
  64  78 105  68 102 148  98 116  71 109 107 135  92  86 127  79  87 1
29
  57  77  95  54 106  53  67 139  60 100  61  73 113  76 119  88  84
62
 137  72 142 114 126 122  81 123 117  82  80 120 130 134  59 112 132 1
10
 101 150  69 131  83  93 124 136 125  66 143  58  55  85  56  70  46
42
 152  44 145  50 153  49 175  63 138 154 140 141 146  65  51 151 158 1
55
 157 147 144 149 166  52  33 156  38  36  48 164]

------------------------------------------------

total night charge
[11.01 11.45  7.32  8.86  8.41  9.18  9.57  9.53  9.71 14.69  9.4   8.
82
  6.35  8.65  9.14  7.23  4.02  5.83  7.46  8.68  9.43  8.18  8.53 10.
67
 11.28  8.22  4.59  8.17  8.04 11.27 11.08 13.2  12.61  9.61  6.88  5.
82
 10.25  4.58  8.47  8.45  5.5  14.02  8.03 11.94  7.34  6.06 10.9   6.
44
  3.18 10.66 11.21 12.73 10.28 12.16  6.34  8.15  5.84  8.52  7.5   7.
48
  6.21 11.95  7.15  9.63  7.1   6.91  6.69 13.29 11.46  7.76  6.86  8.
16
 12.15  7.79  7.99 10.29 10.08 12.53  7.91 10.02  8.61 14.54  8.21  9.
09
  4.93 11.39 11.88  5.75  7.83  8.59  7.52 12.38  7.21  5.81  8.1  11.
04
 11.19  8.55  8.42  9.76  9.87 10.86  5.36 10.03 11.15  9.51  6.22  2.
59
  7.65  6.45  9.    6.4   9.94  5.08 10.23 11.36  6.97 10.16  7.88 11.
91
  6.61 11.55 11.76  9.27  9.29 11.12 10.69  8.8  11.85  7.14  8.71 11.
42
  4.94  9.02 11.22  4.97  9.15  5.45  7.27 12.91  7.75 13.46  6.32 12.
13
 11.97  6.93 11.66  7.42  6.19 11.41 10.33 10.65 11.92  4.77  4.38  7.
41
 12.1   7.69  8.78  9.36  9.05 12.7   6.16  6.05 10.85  8.93  3.48 10.
4
  5.05 10.71  9.37  6.75  8.12 11.77 11.49 11.06 11.25 11.03 10.82  8.
91
  8.57  8.09 10.05 11.7  10.17  8.74  5.51 11.11  3.29 10.13  6.8   8.
49
  9.55 11.02  9.91  7.84 10.62  9.97  3.44  7.35  9.79  8.89  8.14  6.
94
 10.49 10.57 10.2   6.29  8.79 10.04 12.41 15.97  9.1  11.78 12.75 11.
```

```
07
 12.56  8.63  8.02 10.42  8.7   9.98  7.62  8.33  6.59 13.12 10.46  6.
63
  8.32  9.04  9.28 10.76  9.64 11.44  6.48 10.81 12.66 11.34  8.75 13.
05
 11.48 14.04 13.47  5.63  6.6   9.72 11.68  6.41  9.32 12.95 13.37  9.
62
  6.03  8.25  8.26 11.96  9.9   9.23  5.58  7.22  6.64 12.29 12.93 11.
32
  6.85  8.88  7.03  8.48  3.59  5.86  6.23  7.61  7.66 13.63  7.9  11.
82
  7.47  6.08  8.4   5.74 10.94 10.35 10.68  4.34  8.73  5.14  8.24  9.
99
 13.93  8.64 11.43  5.79  9.2  10.14 12.11  7.53 12.46  8.46  8.95  9.
84
 10.8  11.23 10.15  9.21 14.46  6.67 12.83  9.66  9.59 10.48  8.36  4.
84
 10.54  8.39  7.43  9.06  8.94 11.13  8.87  8.5   7.6  10.73  9.56 10.
77
  7.73  3.47 11.86  8.11  9.78  9.42  9.65  7.    7.39  9.88  6.56  5.
92
  6.95 15.71  8.06  4.86  7.8   8.58 10.06  5.21  6.92  6.15 13.49  9.
38
 12.62 12.26  8.19 11.65 11.62 10.83  7.92  7.33 13.01 13.26 12.22 11.
58
  5.97 10.99  8.38  9.17  8.08  5.71  3.41 12.63 11.79 12.96  7.64  6.
58
 10.84 10.22  6.52  5.55  7.63  5.11  5.89 10.78  3.05 11.89  8.97 10.
44
 10.5   9.35  5.66 11.09  9.83  5.44 10.11  6.39 11.93  8.62 12.06  6.
02
  8.85  5.25  8.66  6.73 10.21 11.59 13.87  7.77 10.39  5.54  6.62 13.
33
  6.24 12.59  6.3   6.79  8.28  9.03  8.07  5.52 12.14 10.59  7.54  7.
67
  5.47  8.81  8.51 13.45  8.77  6.43 12.01 12.08  7.07  6.51  6.84  9.
48
 13.78 11.54 11.67  8.13 10.79  7.13  4.72  4.64  8.96 13.03  6.07  3.
51
  6.83  6.12  9.31  9.58  4.68  5.32  9.26 11.52  9.11 10.55 11.47  9.
3
 13.82  8.44  5.77 10.96 11.74  8.9  10.47  7.85 10.92  4.74  9.74 10.
43
  9.96 10.18  9.54  7.89 12.36  8.54 10.07  9.46  7.3  11.16  9.16 10.
19
  5.99 10.88  5.8   7.19  4.55  8.31  8.01 14.43  8.3  14.3   6.53  8.
2
 11.31 13.    6.42  4.24  7.44  7.51 13.1   9.49  6.14  8.76  6.65 10.
56
  6.72  8.29 12.09  5.39  2.96  7.59  7.24  4.28  9.7   8.83 13.3  11.
37
  9.33  5.01  3.26 11.71  8.43  9.68 15.56  9.8   3.61  6.96 11.61 12.
81
 10.87 13.84  5.03  5.17  2.03 10.34  9.34  7.95 10.09  9.95  7.11  9.
22
  6.13 11.05  9.89  9.39 14.06 10.26 13.31 15.43 16.39  6.27 10.64 11.
5
 12.48  8.27 13.53 10.36 12.24  8.69 10.52  9.07 11.51  9.25  8.72  6.
78
  8.6  11.84  5.78  5.85 12.3   5.76 12.07  9.6   8.84 12.39 10.1   9.
73
```

```
  2.85  6.66  2.45  5.28 11.73 10.75  7.74  6.76  6.    7.58 13.69  7.
93
  7.68  9.75  4.96  5.49 11.83  7.18  9.19  7.7   7.25 10.74  4.27 13.
8
  9.12  4.75  7.78 11.63  7.55  2.25  9.45  9.86  7.71  4.95  7.4  11.
17
 11.33  6.82 13.7   1.97 10.89 12.77 10.31  5.23  5.27  9.41  6.09 10.
61
  7.29  4.23  7.57  3.67 12.69 14.5   5.95  7.87  5.96  5.94 12.23  4.
9
 12.33  6.89  9.67 12.68 12.87  3.7   6.04 13.13 15.74 11.87  4.7   4.
67
  7.05  5.42  4.09  5.73  9.47  8.05  6.87  3.71 15.86  7.49 11.69  6.
46
 10.45 12.9   5.41 11.26  1.04  6.49  6.37 12.21  6.77 12.65  7.86  9.
44
  4.3   7.38  5.02 10.63  2.86 17.19  8.67  8.37  6.9  10.93 10.38  7.
36
 10.27 10.95  6.11  4.45 11.9  15.01 12.84  7.45  6.98 11.72  7.56 11.
38
 10.    4.42  9.81  5.56  6.01 10.12 12.4  16.99  5.68 11.64  3.78  7.
82
  9.85 13.74 12.71 10.98 10.01  9.52  7.31  8.35 11.35  9.5  14.03  3.
2
  7.72 13.22 10.7   8.99 10.6  13.02  9.77 12.58 12.35 12.2  11.4  13.
91
  3.57 14.65 12.28  5.13 10.72 12.86 14.    7.12 12.17  4.71  6.28  8.
  7.01  5.91  5.2  12.   12.02 12.88  7.28  5.4  12.04  5.24 10.3  10.
41
 13.41 12.72  9.08  7.08 13.5   5.35 12.45  5.3  10.32  5.15 12.67  5.
22
  5.57  3.94  4.41 13.27 10.24  4.25 12.89  5.72 12.5  11.29  3.25 11.
53
  9.82  7.26  4.1  10.37  4.98  6.74 12.52 14.56  8.34  3.82  3.86 13.
97
 11.57  6.5  13.58 14.32 13.75 11.14 14.18  9.13  4.46  4.83  9.69 14.
13
  7.16  7.98 13.66 14.78 11.2   9.93 11.    5.29  9.92  4.29 11.1  10.
51
 12.49  4.04 12.94  7.09  6.71  7.94  5.31  5.98  7.2  14.82 13.21 12.
32
 10.58  4.92  6.2   4.47 11.98  6.18  7.81  4.54  5.37  7.17  5.33 14.
1
  5.7  12.18  8.98  5.1  14.67 13.95 16.55 11.18  4.44  4.73  2.55  6.
31
  2.43  9.24  7.37 13.42 12.42 11.8  14.45  2.89 13.23 12.6  13.18 12.
19
 14.81  6.55 11.3  12.27 13.98  8.23 15.49  6.47 13.48 13.59 13.25 17.
77
 13.9   3.97 11.56 14.08 13.6   6.26  4.61 12.76 15.76  6.38  3.6  12.
8
  5.9   7.97  5.   10.97  5.88 12.34 12.03 14.97 15.06 12.85  6.54 11.
24
 12.64  7.06  5.38 13.14  3.99  3.32  4.51  4.12  3.93  2.4  11.75  4.
03
 15.85  6.81 14.25 14.09 16.42  6.7  12.74  2.76 12.12  6.99  6.68 11.
81
  7.96  5.06 13.16  2.13 13.17  5.12  5.65 12.37 10.53]

------------------------------------------------
```

```
total intl minutes
[10.  13.7 12.2  6.6 10.1  6.3  7.5  7.1  8.7 11.2 12.7  9.1 12.3 13.1
  5.4 13.8  8.1 13.  10.6  5.7  9.5  7.7 10.3 15.5 14.7 11.1 14.2 12.6
 11.8  8.3 14.5 10.5  9.4 14.6  9.2  3.5  8.5 13.2  7.4  8.8 11.   7.8
  6.8 11.4  9.3  9.7 10.2  8.   5.8 12.1 12.  11.6  8.2  6.2  7.3  6.1
 11.7 15.   9.8 12.4  8.6 10.9 13.9  8.9  7.9  5.3  4.4 12.5 11.3  9.
  9.6 13.3 20.   7.2  6.4 14.1 14.3  6.9 11.5 15.8 12.8 16.2  0.  11.9
  9.9  8.4 10.8 13.4 10.7 17.6  4.7  2.7 13.5 12.9 14.4 10.4  6.7 15.4
  4.5  6.5 15.6  5.9 18.9  7.6  5.   7.  14.  18.  16.  14.8  3.7  2.
  4.8 15.3  6.  13.6 17.2 17.5  5.6 18.2  3.6 16.5  4.6  5.1  4.1 16.3
 14.9 16.4 16.7  1.3 15.2 15.1 15.9  5.5 16.1  4.  16.9  5.2  4.2 15.7
 17.   3.9  3.8  2.2 17.1  4.9 17.9 17.3 18.4 17.8  4.3  2.9  3.1  3.3
  2.6  3.4  1.1 18.3 16.6  2.1  2.4  2.5]

--------------------------------------------------

total intl calls
[ 3  5  7  6  4  2  9 19  1 10 15  8 11  0 12 13 18 14 16 20 17]

--------------------------------------------------

total intl charge
[2.7  3.7  3.29 1.78 2.73 1.7  2.03 1.92 2.35 3.02 3.43 2.46 3.32 3.54
 1.46 3.73 2.19 3.51 2.86 1.54 2.57 2.08 2.78 4.19 3.97 3.   3.83 3.4
 3.19 2.24 3.92 2.84 2.54 3.94 2.48 0.95 2.3  3.56 2.   2.38 2.97 2.11
 1.84 3.08 2.51 2.62 2.75 2.16 1.57 3.27 3.24 3.13 2.21 1.67 1.97 1.65
 3.16 4.05 2.65 3.35 2.32 2.94 3.75 2.4  2.13 1.43 1.19 3.38 3.05 2.43
 2.59 3.59 5.4  1.94 1.73 3.81 3.86 1.86 3.11 4.27 3.46 4.37 0.   3.21
 2.67 2.27 2.92 3.62 2.89 4.75 1.27 0.73 3.65 3.48 3.89 2.81 1.81 4.16
 1.22 1.76 4.21 1.59 5.1  2.05 1.35 1.89 3.78 4.86 4.32 4.   1.   0.54
 1.3  4.13 1.62 3.67 4.64 4.73 1.51 4.91 0.97 4.46 1.24 1.38 1.11 4.4
 4.02 4.43 4.51 0.35 4.1  4.08 4.29 1.49 4.35 1.08 4.56 1.4  1.13 4.24
 4.59 1.05 1.03 0.59 4.62 1.32 4.83 4.67 4.97 4.81 1.16 0.78 0.84 0.89
 0.7  0.92 0.3  4.94 4.48 0.57 0.65 0.68]

--------------------------------------------------

customer service calls
[1 0 2 3 4 5 7 9 6 8]

--------------------------------------------------

churn
[False  True]

--------------------------------------------------
```

In [24]:

```
len(df['state'].unique())
# DC is included as the 51st state
```

Out[24]:

```
51
```

In [13]:

```
# Check balance of target data
df['churn'].value_counts()
```

Out[13]:

```
False    2850
True      483
Name: churn, dtype: int64
```

In [40]:

```
# Check balance with percentages
# There is definitely imbalance, which I will have to deal with during the M
df['churn'].value_counts(normalize=True)
```

Out[40]:

```
False    0.855086
True     0.144914
Name: churn, dtype: float64
```

In [34]:

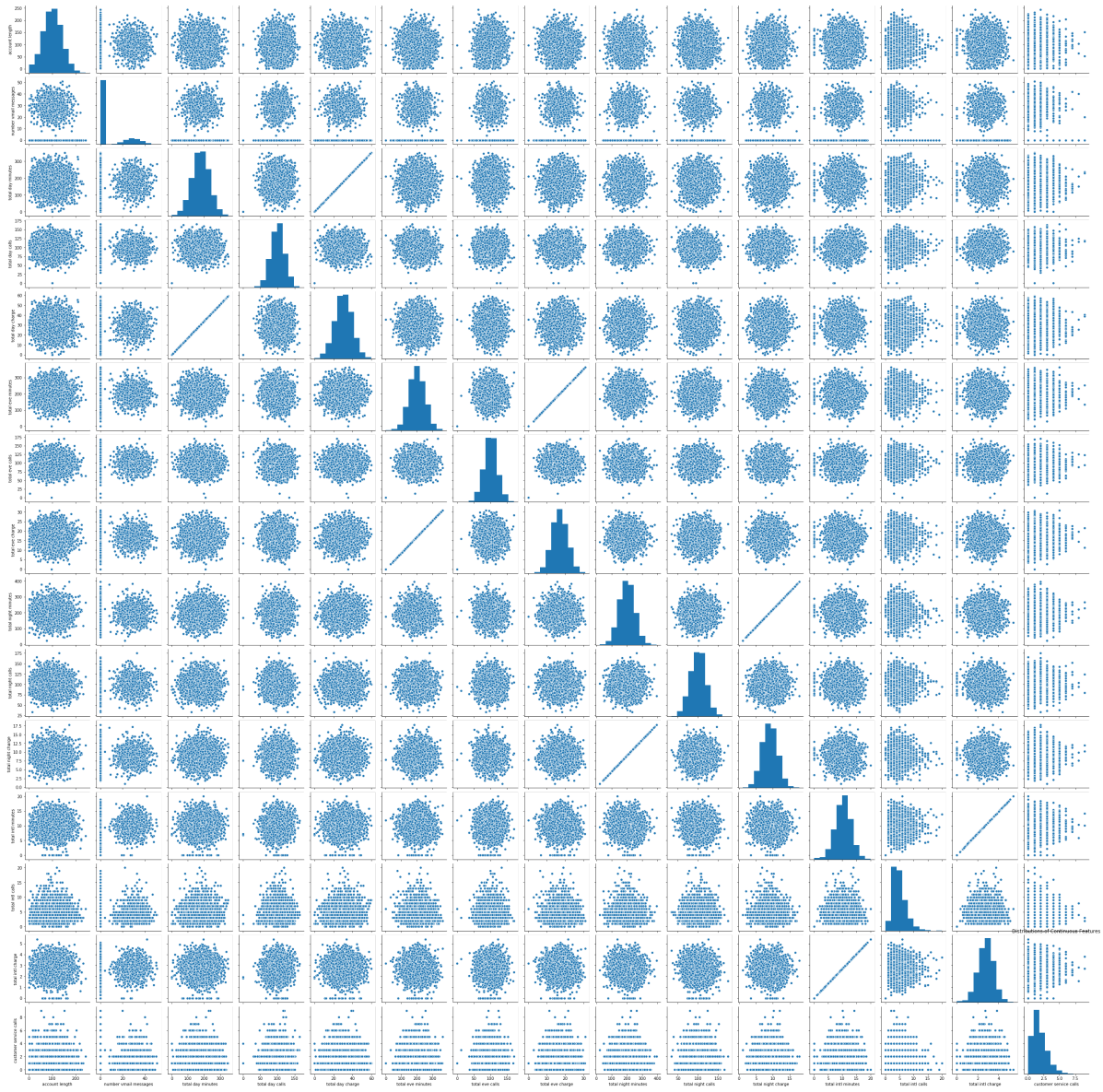
```
# Check out spread of data and outliers
df.describe()
```

Out[34]:

	account length	area code	number vmail messages	total day minutes	total day calls	total day charge	total minu
count	3333.000000	3333.000000	3333.000000	3333.000000	3333.000000	3333.000000	3333.000(
mean	101.064806	437.182418	8.099010	179.775098	100.435644	30.562307	200.980(
std	39.822106	42.371290	13.688365	54.467389	20.069084	9.259435	50.713(
min	1.000000	408.000000	0.000000	0.000000	0.000000	0.000000	0.000(
25%	74.000000	408.000000	0.000000	143.700000	87.000000	24.430000	166.600(
50%	101.000000	415.000000	0.000000	179.400000	101.000000	30.500000	201.400(
75%	127.000000	510.000000	20.000000	216.400000	114.000000	36.790000	235.300(
max	243.000000	510.000000	51.000000	350.800000	165.000000	59.640000	363.700(

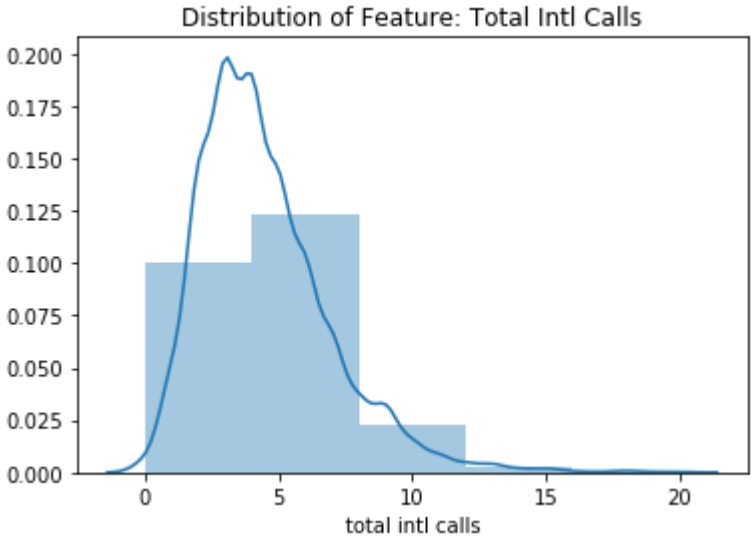
In [35]:

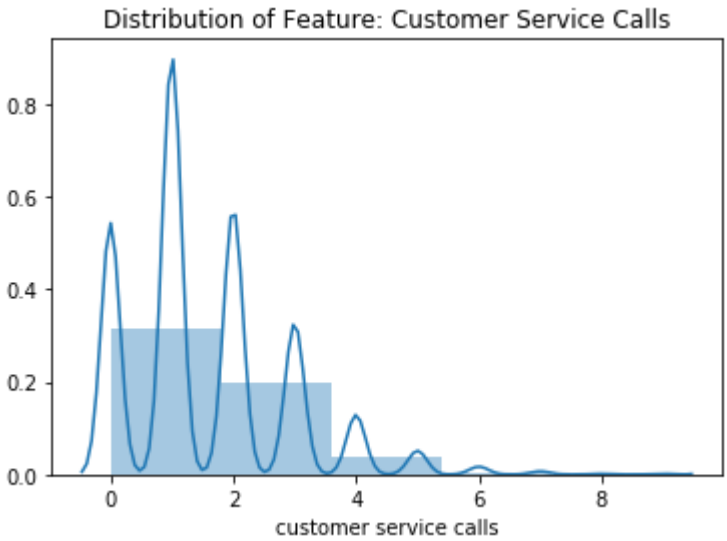
```
# Check spread of data for major outliers
cont_df = df.drop(columns=['state', 'phone number', 'international plan', 'v

sns.pairplot(cont_df)
plt.title('Distributions of Continuous Features')
plt.show()
```

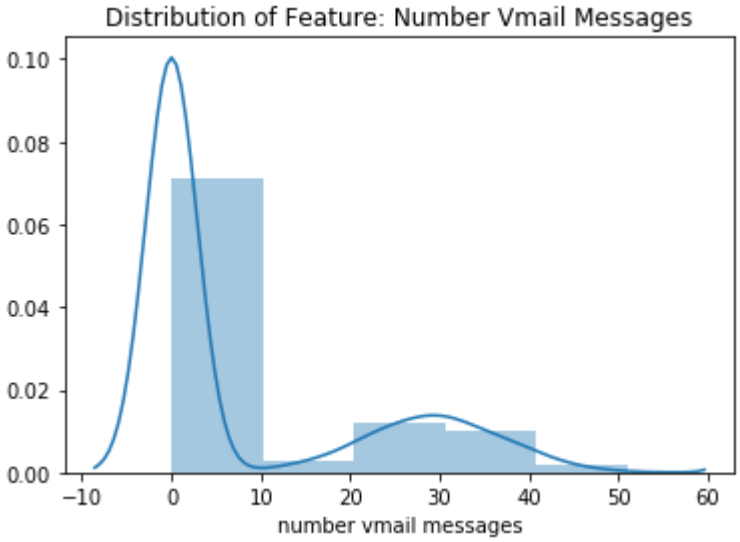


In [39]:

```
# A closer look at these 3 weird distributions
for col in ['total intl calls', 'customer service calls', 'number vmail mess
    sns.distplot(df[col], bins=5)
    plt.title(f'Distribution of Feature: {col.title()}')
    plt.show()
```







Most of this data is pretty normally distributed with no major outliers.

The only 3 continuous features with strange distributions are the 3 above. However, I don't believe that outliers are large enough to deal with right now. I might try to feature engineer something later to deal v them.

Cleaning Conclusion

This dataset was already VERY clean. I didn't need to remove any nulls, duplicates, placeholder values outliers.

That being said, the scrubbing phase of the OSEMN method is not over yet. I may try to deal with som outliers with feature engineering during the EDA/Explore phase. I also need to deal with the imbalance target column during the Modeling phase.

In [ ]: