Project Overview

Predict which flying etiquette behaviors women deem rude (compared with men).

Hypothesis

Women will be more tolerant of what may be considered rude behavior when flying on an airplane.

Data Overview/Cleansing

- Drop null values
- Change column names for easier readability
- Change Yes/No answers to binary
- Create dummy variables for questions that had more than two answers (due to multiple choice)

Analysis and Results

- Split data into train and test (80%, 20%)
- Target variable > ‘Gender’
- Ran Scikit Learn Logistic Regression CV
- Training data score = .90
- Test data score = .72 (could be result of imbalanced classes)
- Test data Accuracy Score = .70
- Test data AUC = .79

Summary

Women and men feel equally the same about rude behavior on airplanes with the exception of two areas: 1) Women believe the armrest between 2 seats should be shared, 2) Women feel it's very rude to ask people to change seats in order to sit near family.

Women scored high for being 5'6" or less and for traveling less than once/month or a few times/month.

Next steps would be to run a new model with a different target variable such as frequency of flying or height to see if either of those two variables had a more direct impact on the rudeness rating scale.

In [1]:

```
import pandas as pd
import numpy as np
import csv
```

In [2]:

```
df = pd.read_csv('https://raw.githubusercontent.com/fivethirtyeight/data/master/fly
```

In [3]:

```
df1 = pd.DataFrame(df)
```

In [4]:

```
df1.head(3)
```

Out[4]:

	RespondentID	How often do you travel by plane?	Do you ever recline your seat when you fly?	How tall are you?	Do you have any children under 18?	In a row of three seats, who should get to use the two arm rests?	In a row of two seats, who should get to use the middle arm rest?	Who should have control over the window shade?	Is itrude to move to an unsold seat on a plane?	Gen spe is it to s mor than few wor toth stra sitti nex you plan
0	3436139758	Once a year or less	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	3434278696	Once a year or less	About half the time	6'3"	Yes	The arm rests should be shared	The arm rests should be shared	Everyone in the row should have some say	No, not rude at all	No, all ru
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	The arm rests should be shared	The person in the window seat should have excl...	No, not rude at all	No, all ru

3 rows × 27 columns

In [5]:

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1040 entries, 0 to 1039
Data columns (total 27 columns):
RespondentID
1040 non-null int64
How often do you travel by plane?
1040 non-null object
Do you ever recline your seat when you fly?
858 non-null object
How tall are you?
858 non-null object
Do you have any children under 18?
851 non-null object
In a row of three seats, who should get to use the two arm rests?
856 non-null object
In a row of two seats, who should get to use the middle arm rest?
856 non-null object
Who should have control over the window shade?
856 non-null object
Is itrude to move to an unsold seat on a plane?
855 non-null object
Generally speaking, is it rude to say more than a few words tothe stra
nger sitting next to you on a plane?
855 non-null object
On a 6 hour flight from NYC to LA, how many times is it acceptable to
get up if you're not in an aisle seat?
855 non-null object
Under normal circumstances, does a person who reclines their seat duri
ng a flight have any obligation to the person sitting behind them?
854 non-null object
Is itrude to recline your seat on a plane?
854 non-null object
Given the opportunity, would you eliminate the possibility of reclinin
g seats on planes entirely?
854 non-null object
Is it rude to ask someone to switch seats with you in order to be clos
er to friends?
850 non-null object
Is itrude to ask someone to switch seats with you in order to be close
r to family?
850 non-null object
Is it rude to wake a passenger up if you are trying to go to the bathr
oom?
850 non-null object
Is itrude to wake a passenger up if you are trying to walk around?
850 non-null object
In general, is itrude to bring a baby on a plane?
849 non-null object
In general, is it rude to knowingly bring unruly children on a plane?
849 non-null object
```

```
Have you ever used personal electronics during take off or landing in
violation of a flight attendant's direction?
849 non-null object
Have you ever smoked a cigarette in an airplane bathroom when it was a
gainst the rules?
849 non-null object
Gender
1007 non-null object
Age
1007 non-null object
Household Income
826 non-null object
Education
1001 non-null object
Location (Census Region)
998 non-null object
dtypes: int64(1), object(26)
memory usage: 219.4+ KB
```

In [6]:

```
df1.isnull().sum()
```

Out[6]:

```
RespondentID
0
How often do you travel by plane?
0
Do you ever recline your seat when you fly?
182
How tall are you?
182
Do you have any children under 18?
189
In a row of three seats, who should get to use the two arm rests?
184
In a row of two seats, who should get to use the middle arm rest?
184
Who should have control over the window shade?
184
Is itrude to move to an unsold seat on a plane?
185
Generally speaking, is it rude to say more than a few words tothe stra
nger sitting next to you on a plane?
185
On a 6 hour flight from NYC to LA, how many times is it acceptable to
get up if you're not in an aisle seat?
185
Under normal circumstances, does a person who reclines their seat duri
ng a flight have any obligation to the person sitting behind them?
186
Is itrude to recline your seat on a plane?
186
```

```
Given the opportunity, would you eliminate the possibility of reclinin
g seats on planes entirely?
186
Is it rude to ask someone to switch seats with you in order to be clos
er to friends?
190
Is itrude to ask someone to switch seats with you in order to be close
r to family?
190
Is it rude to wake a passenger up if you are trying to go to the bathr
oom?
190
Is itrude to wake a passenger up if you are trying to walk around?
190
In general, is itrude to bring a baby on a plane?
191
In general, is it rude to knowingly bring unruly children on a plane?
191
Have you ever used personal electronics during take off or landing in
violation of a flight attendant's direction?
191
Have you ever smoked a cigarette in an airplane bathroom when it was a
gainst the rules?
191
Gender
33
Age
33
Household Income
214
Education
39
Location (Census Region)
42
dtype: int64
```

In [7]:

```
df1 = df1.dropna()
```

In [8]:

```
df1.head()
```

Out[8]:

		How often do	Do you ever recline	How	Do you have	In a row of three seats, who	In a row of two seats, who	Who should have	Is itrude to move

	RespondentID	you travel by plane?	your seat when you fly?	tall are you?	any children under 18?	should get to use the two arm rests?	should get to use the middle arm rest?	control over the window shade?	to an unsold seat on plane?
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	The arm rests should be shared	The person in the window seat should have excl...	No, not rude at a
3	3434268208	Once a year or less	Always	5'11"	No	The arm rests should be shared	The arm rests should be shared	Everyone in the row should have some say	No, not rude at a
4	3434250245	Once a month or less	About half the time	5'7"	No	The person in the middle seat gets both arm rests	The person in aisle	Everyone in the row should have some say	No, not rude at a
5	3434245875	Once a year or less	Usually	5'9"	Yes	Whoever puts their arm on the arm rest first	Whoever puts their arm on the arm rest first	The person in the window seat should have excl...	Yes, somewh rude
		Once				The person in the	Whoever	Everyone in the	

6	3434235351	a month or less	Once in a while	6'2"	Yes	middle seat gets both arm rests	puts their arm on the arm rest first	row should have some say	No, not rude at a

5 rows × 27 columns

In [9]:

```
df1.isnull().sum()
```

Out[9]:

```
RespondentID
0
How often do you travel by plane?
0
Do you ever recline your seat when you fly?
0
How tall are you?
0
Do you have any children under 18?
0
In a row of three seats, who should get to use the two arm rests?
0
In a row of two seats, who should get to use the middle arm rest?
0
Who should have control over the window shade?
0
Is itrude to move to an unsold seat on a plane?
0
Generally speaking, is it rude to say more than a few words tothe stra
nger sitting next to you on a plane?
0
On a 6 hour flight from NYC to LA, how many times is it acceptable to
get up if you're not in an aisle seat?
0
Under normal circumstances, does a person who reclines their seat duri
ng a flight have any obligation to the person sitting behind them?
0
Is itrude to recline your seat on a plane?
0
Given the opportunity, would you eliminate the possibility of reclinin
g seats on planes entirely?
0
Is it rude to ask someone to switch seats with you in order to be clos
er to friends?
0
Is itrude to ask someone to switch seats with you in order to be close
```

```
r to family?
0
Is it rude to wake a passenger up if you are trying to go to the bathr
oom?
0
Is itrude to wake a passenger up if you are trying to walk around?
0
In general, is itrude to bring a baby on a plane?
0
In general, is it rude to knowingly bring unruly children on a plane?
0
Have you ever used personal electronics during take off or landing in
violation of a flight attendant's direction?
0
Have you ever smoked a cigarette in an airplane bathroom when it was a
gainst the rules?
0
Gender
0
Age
0
Household Income
0
Education
0
Location (Census Region)
0
dtype: int64
```

In [10]:

```
df1.columns
```

Out[10]:

```
Index([u'RespondentID', u'How often do you travel by plane?',
       u'Do you ever recline your seat when you fly?', u'How tall are
you?',
       u'Do you have any children under 18?',
       u'In a row of three seats, who should get to use the two arm re
sts?',
       u'In a row of two seats, who should get to use the middle arm r
est?',
       u'Who should have control over the window shade?',
       u'Is itrude to move to an unsold seat on a plane?',
       u'Generally speaking, is it rude to say more than a few words t
othe stranger sitting next to you on a plane?',
       u'On a 6 hour flight from NYC to LA, how many times is it accep
table to get up if you're not in an aisle seat?',
       u'Under normal circumstances, does a person who reclines their
seat during a flight have any obligation to the person sitting behind
them?',
       u'Is itrude to recline your seat on a plane?',
       u'Given the opportunity, would you eliminate the possibility of
reclining seats on planes entirely?',
       u'Is it rude to ask someone to switch seats with you in order t
o be closer to friends?',
       u'Is itrude to ask someone to switch seats with you in order to
be closer to family?',
       u'Is it rude to wake a passenger up if you are trying to go to
the bathroom?',
       u'Is itrude to wake a passenger up if you are trying to walk ar
ound?',
       u'In general, is itrude to bring a baby on a plane?',
       u'In general, is it rude to knowingly bring unruly children on
a plane?',
       u'Have you ever used personal electronics during take off or la
nding in violation of a flight attendant's direction?',
       u'Have you ever smoked a cigarette in an airplane bathroom when
it was against the rules?',
       u'Gender', u'Age', u'Household Income', u'Education',
       u'Location (Census Region)'],
      dtype='object')
```

In [11]:

```
df1.columns = ['ID,',
              'Travel_Freq',
              'Recline_seat',
              'Height',
              'Child_<_18',
              'Threeseats_armrest_use',
              'Twoseats_armrest_use',
              'Control_window_shade',
              'Rude_movetounsoldseat',
              'Rude_morethanfewwords',
              'Sixhrflt_getup_noaisle',
              'Recline_obligation',
              'Rude_reclineseat',
              'Eliminate_reclineseats',
              'Switch_seat_friends',
              'Switch_seat_family',
              'Rude_to_wake_forbathroom',
              'Rude_to_wake_forwalk',
              'Rude_bring_baby',
              'Rude_bring_unrulychild',
              'Use_electronic_takeoff_land',
              'Smoke_in_bathroom',
              'Gender',
              'Age',
              'Income',
              'Education',
              'Location']
```

In [12]:

```
df1.columns
```

Out[12]:

```
Index([u'ID,', u'Travel_Freq', u'Recline_seat', u'Height', u'Child_<_1
8',
       u'Threeseats_armrest_use', u'Twoseats_armrest_use',
       u'Control_window_shade', u'Rude_movetounsoldseat',
       u'Rude_morethanfewwords', u'Sixhrflt_getup_noaisle',
       u'Recline_obligation', u'Rude_reclineseat', u'Eliminate_recline
seats',
       u'Switch_seat_friends', u'Switch_seat_family',
       u'Rude_to_wake_forbathroom', u'Rude_to_wake_forwalk',
       u'Rude_bring_baby', u'Rude_bring_unrulychild',
       u'Use_electronic_takeoff_land', u'Smoke_in_bathroom', u'Gender'
, u'Age',
       u'Income', u'Education', u'Location'],
      dtype='object')
```

In [13]:

```
df1.shape
```

Out[13]:

```
(687, 27)
```

In [14]:

```
df1['Travel_Freq'].value_counts()
# make dummy var
```

Out[14]:

```
Once a year or less      500
Once a month or less     158
A few times per month     26
A few times per week       2
Every day                  1
Name: Travel_Freq, dtype: int64
```

In [ ]:

```

```

In [15]:

```
df1['Recline_seat'].value_counts()
#dummy var
```

Out[15]:

```
Once in a while        199
Usually                154
Never                  129
Always                 112
About half the time     93
Name: Recline_seat, dtype: int64
```

In [16]:

```
df1 = df1[df1.Recline_seat != 'About half the time']
df1
```

Out[16]:

	ID,	Travel_Freq	Recline_seat	Height	Child_<_18	Threeseats_armrest_use	Twoseats_
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	The arm re: shared
3	3434268208	Once a year or less	Always	5'11"	No	The arm rests should be shared	The arm re: shared
5	3434245875	Once a year or less	Usually	5'9"	Yes	Whoever puts their arm on the arm rest first	Whoever p on the arm
6	3434235351	Once a month or	Once in a while	6'2"	Yes	The person in the middle seat gets both arm rests	Whoever p on the arm

In [17]:

```
df1['Height'].value_counts()
#could write function to change to inches but prefer to create dummy var for quicke:
```

Out[17]:

```
5'10"          55
5'4"           54
5'8"           53
5'5"           53
5'6"           51
5'7"           51
5'9"           49
6'0"           42
5'11"          37
5'2"           31
5'3"           24
6'1"           20
6'2"           19
5'1"           12
6'3"           12
6'4"           11
5'0"            8
Under 5 ft.     8
6'5"            4
Name: Height, dtype: int64
```

In [18]:

```
df1['Child_<_18'].value_counts()
#make binary
```

Out[18]:

```
No     463
Yes    131
Name: Child_<_18, dtype: int64
```

In [19]:

```
df1['Threeseats_armrest_use'].value_counts()
#make dummy var
```

Out[19]:

```
The arm rests should be shared                                  412
The person in the middle seat gets both arm rests                79
Whoever puts their arm on the arm rest first                     64
Other (please specify)                                           33
The people in the aisle and window seats get both arm rests       6
Name: Threeseats_armrest_use, dtype: int64
```

In [20]:

```
df1 = df1[df1.Threeseats_armrest_use != 'Other (please specify)']
df1
```

Out[20]:

	ID,	Travel_Freq	Recline_seat	Height	Child_<_18	Threeseats_armrest_use	Twoseats_
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	The arm re shared
3	3434268208	Once a year or less	Always	5'11"	No	The arm rests should be shared	The arm re shared
5	3434245875	Once a year or less	Usually	5'9"	Yes	Whoever puts their arm on the arm rest first	Whoever p on the arm
6	3434235351	Once a month or	Once in a while	6'2"	Yes	The person in the middle seat gets both arm rests	Whoever p on the arm

In [21]:

```
df1 = df1[df1.Threeseats_armrest_use != 'The people in the aisle and window seats ge
df1
```

Out[21]:

	ID,	Travel_Freq	Recline_seat	Height	Child_<_18	Threeseats_armrest_use	Twoseats_
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	The arm re shared
3	3434268208	Once a year or less	Always	5'11"	No	The arm rests should be shared	The arm re shared
5	3434245875	Once a year or less	Usually	5'9"	Yes	Whoever puts their arm on the arm rest first	Whoever p on the arm
6	3434235351	Once a month or	Once in a while	6'2"	Yes	The person in the middle seat gets both arm rests	Whoever p on the arm

In [22]:

```
df1['Twoseats_armrest_use'].value_counts()
#make dummy var
```

Out[22]:

```
The arm rests should be shared                 403
Whoever puts their arm on the arm rest first    84
The person in aisle                             40
The person by the window                        20
Other (please specify)                           8
Name: Twoseats_armrest_use, dtype: int64
```

In [23]:

```
df1 = df1[df1.Twoseats_armrest_use != 'Other (please specify)']
```

In [24]:

```
df1['Control_window_shade'].value_counts()
#drop column
```

Out[24]:

```
Everyone in the row should have some say                        320
The person in the window seat should have exclusive control    227
Name: Control_window_shade, dtype: int64
```

In [25]:

```
df1['Rude_movetounsoldseat'].value_counts()
#make dummy var
```

Out[25]:

```
No, not rude at all    446
Yes, somewhat rude      79
Yes, very rude          22
Name: Rude_movetounsoldseat, dtype: int64
```

In [26]:

```
df1['Rude_morethanfewwords'].value_counts()
#make dummy var
```

Out[26]:

```
No, not at all rude    429
Yes, somewhat rude     106
Yes, very rude          12
Name: Rude_morethanfewwords, dtype: int64
```

In [27]:

```
df1['Sixhrflt_getup_noaisle'].value_counts()
#make dummy var
```

Out[27]:

```
Three times                              194
Twice                                    187
Four times                                69
More than five times times                50
Once                                      41
It is not okay to get up during flight     6
Name: Sixhrflt_getup_noaisle, dtype: int64
```

In [28]:

```
df1 = df1[df1.Sixhrflt_getup_noaisle != 'It is not okay to get up during flight']
```

In [29]:

```
df1['Recline_obligation'].value_counts()
#make binary value
```

Out[29]:

```
Yes, they should not recline their chair if the person behind them ask
s them not to    342
No, the person on the flight has no obligation to the person behind th
em               199
Name: Recline_obligation, dtype: int64
```

In [30]:

```
df1['Rude_reclineseat'].value_counts()
#make dummy var
```

Out[30]:

```
No, not rude at all    323
Yes, somewhat rude     171
Yes, very rude          47
Name: Rude_reclineseat, dtype: int64
```

In [31]:

```
df1['Eliminate_reclineseats'].value_counts()
#make binary value
```

Out[31]:

```
No     374
Yes    167
Name: Eliminate_reclineseats, dtype: int64
```

In [32]:

```
df1['Switch_seat_friends'].value_counts()
#make dummy var
```

Out[32]:

```
No, not at all rude    409
Yes, somewhat rude     115
Yes, very rude          17
Name: Switch_seat_friends, dtype: int64
```

In [33]:

```
df1['Switch_seat_family'].value_counts()
#make dummy var
```

Out[33]:

```
No, not at all rude    459
Yes, somewhat rude      73
Yes, very rude           9
Name: Switch_seat_family, dtype: int64
```

In [34]:

```
df1['Rude_to_wake_forbathroom'].value_counts()
#make dummy var
```

Out[34]:

```
No, not at all rude    357
Yes, somewhat rude     173
Yes, very rude          11
Name: Rude_to_wake_forbathroom, dtype: int64
```

In [35]:

```
df1['Rude_to_wake_forwalk'].value_counts()
#make dummy var
```

Out[35]:

```
Yes, somewhat rude     291
No, not at all rude    144
Yes, very rude         106
Name: Rude_to_wake_forwalk, dtype: int64
```

In [36]:

```
df1['Rude_bring_baby'].value_counts()
#make dummy var
```

Out[36]:

```
No, not at all rude    386
Yes, somewhat rude     110
Yes, very rude          45
Name: Rude_bring_baby, dtype: int64
```

In [37]:

```
df1['Rude_bring_unrulychild'].value_counts()
#make dummy var
```

Out[37]:

```
Yes, somewhat rude     235
Yes, very rude         215
No, not at all rude     91
Name: Rude_bring_unrulychild, dtype: int64
```

In [38]:

```
df1['Use_electronic_takeoff_land'].value_counts()
#make binary value
```

Out[38]:

```
No     458
Yes     83
Name: Use_electronic_takeoff_land, dtype: int64
```

In [39]:

```
df1['Smoke_in_bathroom'].value_counts()
#drop column given low number for Yes
```

Out[39]:

```
No     538
Yes      3
Name: Smoke_in_bathroom, dtype: int64
```

In [40]:

```
df1['Gender'].value_counts()
#make binary value
```

Out[40]:

```
Female    272
Male      269
Name: Gender, dtype: int64
```

In [41]:

```
df1['Age'].value_counts()
#make dummy var
```

Out[41]:

```
45-60    159
30-44    138
> 60     136
18-29    108
Name: Age, dtype: int64
```

In [42]:

```
df1['Income'].value_counts()
# make dummy var
```

Out[42]:

```
$50,000 - $99,999      206
$100,000 - $149,999    111
$25,000 - $49,999      100
150000                  82
$0 - $24,999            42
Name: Income, dtype: int64
```

In [43]:

```
df1['Education'].value_counts()
#drop 'less than high school', make dummy var
```

Out[43]:

```
Bachelor degree                     180
Graduate degree                     164
Some college or Associate degree    147
High school degree                   48
Less than high school degree          2
Name: Education, dtype: int64
```

In [44]:

```
df1 = df1[df1.Education != 'Less than high school degree']
```

In [45]:

```
df1['Location'].value_counts()
#make dummy var
```

Out[45]:

```
Pacific               127
South Atlantic         87
East North Central     76
Middle Atlantic        74
West South Central     53
West North Central     44
Mountain               32
New England            27
East South Central     19
Name: Location, dtype: int64
```

In [46]:

```
#drop ID, Control win shade, Smoke in bath columns
df1.head(2)
```

Out[46]:

	ID,	Travel_Freq	Recline_seat	Height	Child_<_18	Threeseats_armrest_use	T
2	3434275578	Once a year or less	Usually	5'8"	No	Whoever puts their arm on the arm rest first	T s
3	3434268208	Once a year or less	Always	5'11"	No	The arm rests should be shared	T s

2 rows × 27 columns

In [47]:

```
df1.drop(df1.columns[[0,7,21]], axis=1, inplace=True)
```

In [48]:

```
df1.columns
```

Out[48]:

```
Index([u'Travel_Freq', u'Recline_seat', u'Height', u'Child_<_18',
       u'Threeseats_armrest_use', u'Twoseats_armrest_use',
       u'Rude_movetounsoldseat', u'Rude_morethanfewwords',
       u'Sixhrflt_getup_noaisle', u'Recline_obligation', u'Rude_reclin
eseat',
       u'Eliminate_reclineseats', u'Switch_seat_friends',
       u'Switch_seat_family', u'Rude_to_wake_forbathroom',
       u'Rude_to_wake_forwalk', u'Rude_bring_baby', u'Rude_bring_unrul
ychild',
       u'Use_electronic_takeoff_land', u'Gender', u'Age', u'Income',
       u'Education', u'Location'],
      dtype='object')
```

In [49]:

```
df1['Child_<_18'] = df1['Child_<_18'].map({'No':0, 'Yes':1})
```

In [50]:

```
df1['Recline_obligation'] = df1['Recline_obligation'].map(
{'No, the person on the flight has no obligation to the person behind them':0,
'Yes, they should not recline their chair if the person behind them asks them not t
```

In [51]:

```
df1['Eliminate_reclineseats'] = df1['Eliminate_reclineseats'].map({'No':0, "Yes":1}
```

In [52]:

```
df1['Use_electronic_takeoff_land'] = df1['Use_electronic_takeoff_land'].map({'No':0,
```

In [53]:

```
df1['Gender'] = df1['Gender'].map({'Male':0, 'Female':1})
```

In [54]:

```
dummy_TF = pd.get_dummies(df1[['Travel_Freq']], prefix='Trav_Freq')
```

In [55]:

```
dummy_RS = pd.get_dummies(df1[['Recline_seat']], prefix='Recline_seat')
```

In [56]:

```
dummy_H = pd.get_dummies(df1[['Height']], prefix='Height')
```

In [57]:

```
dummy_3S = pd.get_dummies(df1[['Threeseats_armrest_use']], prefix='Threeseat_arm')
```

In [58]:

```
dummy_2S = pd.get_dummies(df1[['Twoseats_armrest_use']], prefix='Twoseat_arm')
```

In [59]:

```
dummy_USS = pd.get_dummies(df1[['Rude_movetounsoldseat']], prefix='Unsold_seat')
```

In [60]:

```
dummy_words = pd.get_dummies(df1[['Rude_morethanfewwords']], prefix='Few_words')
```

In [61]:

```
dummy_6HR = pd.get_dummies(df1[['Sixhrflt_getup_noaisle']], prefix='Sixhrflt')
```

In [62]:

```
dummy_reclineob = pd.get_dummies(df1[['Recline_obligation']], prefix='Rec_obligate'
```

In [63]:

```
dummy_ruderecline = pd.get_dummies(df1[['Rude_reclineseat']], prefix='Rude_recline'
```

In [64]:

```
dummy_elimrecline = pd.get_dummies(df1[['Eliminate_reclineseats']], prefix='Elim_rec
```

In [65]:

```
dummy_switchfriends = pd.get_dummies(df1[['Switch_seat_friends']], prefix='Switch_fr
```

In [66]:

```
dummy_switchfam = pd.get_dummies(df1[['Switch_seat_family']], prefix='Switch_fam')
```

In [67]:

```
dummy_wakebath = pd.get_dummies(df1[['Rude_to_wake_forbathroom']], prefix='Wake_bath
```

In [68]:

```
dummy_wakewalk = pd.get_dummies(df1[['Rude_to_wake_forwalk']], prefix='Wake_walk')
```

In [69]:

```
dummy_baby = pd.get_dummies(df1[['Rude_bring_baby']], prefix='Bring_baby')
```

In [70]:

```
dummy_child = pd.get_dummies(df1[['Rude_bring_unrulychild']], prefix='Unruly_child'
```

In [71]:

```
dummy_age = pd.get_dummies(df1[['Age']], prefix='Age')
```

In [72]:

```
dummy_income = pd.get_dummies(df1[['Income']], prefix='Income')
```

In [73]:

```
dummy_edu = pd.get_dummies(df1[['Education']], prefix='Edu')
```

In [74]:

```
dummy_locate = pd.get_dummies(df1[['Location']], prefix='Loca')
```

In [75]:

```
data = pd.concat([dummy_TF,
                  dummy_RS,
                  dummy_H,
                  dummy_3S,
                  dummy_2S,
                 dummy_USS,
                 dummy_words,
                 dummy_6HR,
                  dummy_reclineob,
                  dummy_rudecline,
                  dummy_elimrecline,
                  dummy_switchfriends,
                  dummy_switchfam,
                  dummy_wakebath,
                  dummy_wakewalk,
                  dummy_baby,
                  dummy_child,
                  dummy_age,
                  dummy_income,
                  dummy_edu,
                  dummy_locate,
                 df1[['Child_<_18']],
                 df1[['Recline_obligation']],
                 df1[['Eliminate_reclineseats']],
                 df1[['Use_electronic_takeoff_land']],
                 df1[['Gender']]],
                 axis=1, join='inner')
```

In [76]:

```
data.head()
```

Out[76]:

	Trav_Freq_A few times per month	Trav_Freq_A few times per week	Trav_Freq_Every day	Trav_Freq_Once a month or less	Trav_Freq_Once a year or less	Recline
2	0	0	0	0	1	0
3	0	0	0	0	1	1
5	0	0	0	0	1	0
6	0	0	0	1	0	0
7	0	0	0	0	1	0

5 rows × 96 columns

In [77]:

```
data.isnull().sum()
```

```
Out[77]:

Trav_Freq_A few times per month                                    0
Trav_Freq_A few times per week                                     0
Trav_Freq_Every day                                                0
Trav_Freq_Once a month or less                                     0
Trav_Freq_Once a year or less                                      0
Recline_seat_Always                                                0
Recline_seat_Never                                                 0
Recline_seat_Once in a while                                       0
Recline_seat_Usually                                               0
Height_5'0"                                                        0
Height_5'1"                                                        0
Height_5'10"                                                       0
Height_5'11"                                                       0
Height_5'2"                                                        0
Height_5'3"                                                        0
Height_5'4"                                                        0
Height_5'5"                                                        0
Height_5'6"                                                        0
Height_5'7"                                                        0
Height_5'8"                                                        0
Height_5'9"                                                        0
Height_6'0"                                                        0
Height_6'1"                                                        0
Height_6'2"                                                        0
Height_6'3"                                                        0
Height_6'4"                                                        0
Height_6'5"                                                        0
Height_Under 5 ft.                                                 0
Threeseat_arm_The arm rests should be shared                       0
Threeseat_arm_The person in the middle seat gets both arm rests    0
                                                                  ..
Unruly_child_No, not at all rude                                   0
Unruly_child_Yes, somewhat rude                                    0
Unruly_child_Yes, very rude                                        0
Age_18-29                                                          0
Age_30-44                                                          0
Age_45-60                                                          0
Age_> 60                                                           0
Income_$0 - $24,999                                                0
Income_$100,000 - $149,999                                         0
Income_$25,000 - $49,999                                           0
Income_$50,000 - $99,999                                           0
Income_150000                                                      0
Edu_Bachelor degree                                                0
Edu_Graduate degree                                                0
Edu_High school degree                                             0
Edu_Some college or Associate degree                               0
Loca_East North Central                                            0
Loca_East South Central                                            0
```

```
Loca_Middle Atlantic                                              0
Loca_Mountain                                                     0
Loca_New England                                                  0
Loca_Pacific                                                      0
Loca_South Atlantic                                               0
Loca_West North Central                                           0
Loca_West South Central                                           0
Child_<_18                                                        0
Recline_obligation                                                0
Eliminate_reclineseats                                            0
Use_electronic_takeoff_land                                       0
Gender                                                            0
dtype: int64
```

In [78]:

```
data = data.copy()
```

In [79]:

```
data.corr()
```

Out[79]:

	Trav_Freq_A few times per month	Trav_Freq_A few times per week	Trav_Freq_Every day	Trav_Freq_Once a month or less	Trav_Freq_ a year or le
Trav_Freq_A few times per month	1.000000	-0.008014	-0.008014	-0.097871	-0.318398
Trav_Freq_A few times per week	-0.008014	1.000000	-0.001859	-0.022701	-0.073852
Trav_Freq_Every day	-0.008014	-0.001859	1.000000	-0.022701	-0.073852
Trav_Freq_Once a month or less	-0.097871	-0.022701	-0.022701	1.000000	-0.901965
Trav_Freq_Once a year or	-0.318398	-0.073852	-0.073852	-0.901965	1.000000

In [80]:

```
from sklearn.cross_validation import train_test_split
train, test = train_test_split(data, train_size=.80, test_size=.20)
```

```
/Users/krys/anaconda2/lib/python2.7/site-packages/sklearn/cross_valida
tion.py:44: DeprecationWarning: This module was deprecated in version
0.18 in favor of the model_selection module into which all the refacto
red classes and functions are moved. Also note that the interface of t
he new CV iterators are different from that of this module. This modul
e will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
```

In [81]:

```
train.shape
```

Out[81]:

```
(431, 96)
```

In [82]:

```
train.head(2)
```

Out[82]:

	Trav_Freq_A few times per month	Trav_Freq_A few times per week	Trav_Freq_Every day	Trav_Freq_Once a month or less	Trav_Freq_Once a year or less	Recli
375	0	0	0	0	1	0
524	0	0	0	0	1	0

2 rows × 96 columns

In [83]:

```
test.shape
```

Out[83]:

```
(108, 96)
```

In [84]:

```
from sklearn import linear_model
from sklearn.linear_model import LogisticRegressionCV
log_model = LogisticRegressionCV()
```

In [85]:

```
X = train.iloc[:, :-1]
y = train['Gender']
```

In [86]:

```
log_model.fit(X,y)
```

Out[86]:

```
LogisticRegressionCV(Cs=10, class_weight=None, cv=None, dual=False,
           fit_intercept=True, intercept_scaling=1.0, max_iter=100,
           multi_class='ovr', n_jobs=1, penalty='l2', random_state=Non
e,
           refit=True, scoring=None, solver='lbfgs', tol=0.0001, verbo
se=0)
```

In [87]:

```
log_model.score(X,y)
```

Out[87]:

```
0.90487238979118334
```

In [88]:

```
print log_model.intercept_
print log_model.coef_
```

```
[-1.63633273]
[[  4.20532451e+00  -1.16251558e+01  -8.01994726e-01   4.72385038e+00
    2.95776630e+00   5.28035478e-01  -1.01238781e+00  -3.45518582e-01
    2.89661572e-01   1.16736127e+01   1.11055081e+01  -1.15019400e+00
   -3.15486055e+00   4.88743320e+00   5.63451710e+00   7.22424535e+00
    4.69868957e+00   3.26469570e+00   6.07712449e-01  -4.65348043e-03
   -8.39394341e-01  -5.08327066e+00  -9.01712498e+00  -1.17674319e+01
   -1.03712655e+01  -9.66987776e+00  -7.10200324e+00   8.52345300e+00
    5.85980691e-02   6.49485288e-01  -1.24829270e+00   1.32973240e+00
   -1.92335605e+00  -1.19471845e+00   1.24813274e+00  -1.34979624e-01
   -1.20788041e+00   8.02650686e-01  -8.24949512e-01   2.64442716e-01
    2.02974504e-02   1.62139998e-01  -1.33423731e+00   9.99652926e-01
   -7.47790934e-01   3.80025974e-01   1.18563049e-01  -9.65442496e-01
    1.44893373e-01   2.80339778e-01   5.67859275e-02  -1.55598545e-01
   -4.64679637e-01   8.00688359e-02  -2.29498170e+00  -2.12008999e+00
    3.87486234e+00  -3.82126055e-01   9.83690550e-01  -1.14177384e+00
    6.06223988e-01   3.54923451e-01  -1.50135678e+00   1.35857303e+00
   -6.60855610e-01  -1.23792676e+00  -3.83170617e-01  -7.55721440e-01
    5.98682711e-01   4.10920049e-01   4.75542183e-01  -5.90700634e-01
   -8.35970943e-01  -1.47475826e+00   7.38138934e-01   3.66570548e-01
    1.02906071e+00  -1.19922127e+00  -6.33408079e-01  -1.22449744e-01
    4.98173277e-01  -2.82524799e-01   4.31940082e-01  -4.36328867e-01
   -1.49115532e+00  -8.79633706e-01   2.32887120e+00  -9.43348029e-01
    7.08215769e-01  -7.29429116e-01   4.70658646e-01  -2.28200166e-01
    1.18563049e-01   5.67859275e-02   8.71737835e-01]]
```

In [91]:

```
X_test = test.iloc[:, :-1]
y_test = test.iloc[:, -1]
```

In [92]:

```
y_predicted = log_model.predict(X_test)
```

In [93]:

```
print y_predicted.mean()
```

```
0.5
```

In [94]:

```
log_model.score(X_test, y_test)
```

Out[94]:

```
0.72222222222222221
```

In [101]:

```
coef_list = pd.DataFrame(zip(X_test.columns, np.transpose(log_model.coef_)))
coef_list.values.tolist()
```

Out[101]:

```
[['Trav_Freq_A few times per month', array([ 4.20532451])],
 ['Trav_Freq_A few times per week', array([-11.62515581])],
 ['Trav_Freq_Every day', array([-0.80199473])],
 ['Trav_Freq_Once a month or less', array([ 4.72385038])],
 ['Trav_Freq_Once a year or less', array([ 2.9577663])],
 ['Recline_seat_Always', array([ 0.52803548])],
 ['Recline_seat_Never', array([-1.01238781])],
 ['Recline_seat_Once in a while', array([-0.34551858])],
 ['Recline_seat_Usually', array([ 0.28966157])],
 ['Height_5\'0"', array([ 11.67361266])],
 ['Height_5\'1"', array([ 11.10550808])],
 ['Height_5\'10"', array([-1.150194])],
 ['Height_5\'11"', array([-3.15486055])],
 ['Height_5\'2"', array([ 4.8874332])],
 ['Height_5\'3"', array([ 5.6345171])],
 ['Height_5\'4"', array([ 7.22424535])],
 ['Height_5\'5"', array([ 4.69868957])],
 ['Height_5\'6"', array([ 3.2646957])],
 ['Height_5\'7"', array([ 0.60771245])],
 ['Height_5\'8"', array([-0.00465348])],
 ['Height_5\'9"', array([-0.83939434])],
 ['Height_6\'0"', array([-5.08327066])],
 ['Height_6\'1"', array([-9.01712498])],
 ['Height_6\'2"', array([-11.76743194])],
 ['Height_6\'3"', array([-10.37126549])],
 ['Height_6\'4"', array([-9.66987776])],
 ['Height_6\'5"', array([-7.10200324])],
 ['Height_Under 5 ft.', array([ 8.523453])],
 ['Threeseat_arm_The arm rests should be shared', array([ 0.05859807])
],
 ['Threeseat_arm_The person in the middle seat gets both arm rests',
  array([ 0.64948529])],
 ['Threeseat_arm_Whoever puts their arm on the arm rest first',
  array([-1.2482927])],
 ['Twoseat_arm_The arm rests should be shared', array([ 1.3297324])],
 ['Twoseat_arm_The person by the window', array([-1.92335605])],
 ['Twoseat_arm_The person in aisle', array([-1.19471845])],
 ['Twoseat_arm_Whoever puts their arm on the arm rest first',
  array([ 1.24813274])],
 ['Unsold_seat_No, not rude at all', array([-0.13497962])],
 ['Unsold_seat_Yes, somewhat rude', array([-1.20788041])],
 ['Unsold_seat_Yes, very rude', array([ 0.80265069])],
 ['Few_words_No, not at all rude', array([-0.82494951])],
 ['Few_words_Yes, somewhat rude', array([ 0.26444272])],
 ['Few_words_Yes, very rude', array([ 0.02029745])],
 ['Sixhrflt_Four times', array([ 0.16214])],
 ['Sixhrflt_More than five times times', array([-1.33423731])],
 ['Sixhrflt_Once', array([ 0.99965293])],
 ['Sixhrflt_Three times', array([-0.74779093])],
 ['Sixhrflt_Twice', array([ 0.38002597])],
 ['Recline_obligation', array([ 0.11856305])],
 ['Rude_recline_No, not rude at all', array([-0.9654425])],
 ['Rude_recline_Yes, somewhat rude', array([ 0.14489337])],
 ['Rude_recline_Yes, very rude', array([ 0.28033978])],
 ['Eliminate_reclineseats', array([ 0.05678593])],
 ['Switch_friend_No, not at all rude', array([-0.15559854])],
 ['Switch_friend_Yes, somewhat rude', array([-0.46467964])],
 ['Switch_friend_Yes, very rude', array([ 0.08006884])],
 ['Switch_fam_No, not at all rude', array([-2.2949817])],
 ['Switch_fam_Yes, somewhat rude', array([-2.12008999])],
 ['Switch_fam_Yes, very rude', array([ 3.87486234])],
 ['Wake_bath_No, not at all rude', array([-0.38212606])],
 ['Wake_bath_Yes, somewhat rude', array([ 0.98369055])],
 ['Wake_bath_Yes, very rude', array([-1.14177384])],
 ['Wake_walk_No, not at all rude', array([ 0.60622399])],
 ['Wake_walk_Yes, somewhat rude', array([ 0.35492345])],
 ['Wake_walk_Yes, very rude', array([-1.50135678])],
 ['Bring_baby_No, not at all rude', array([ 1.35857303])],
 ['Bring_baby_Yes, somewhat rude', array([-0.66085561])],
 ['Bring_baby_Yes, very rude', array([-1.23792676])],
 ['Unruly_child_No, not at all rude', array([-0.38317062])],
 ['Unruly_child_Yes, somewhat rude', array([-0.75572144])],
 ['Unruly_child_Yes, very rude', array([ 0.59868271])],
 ['Age_18-29', array([ 0.41092005])],
 ['Age_30-44', array([ 0.47554218])],
 ['Age_45-60', array([-0.59070063])],
 ['Age_> 60', array([-0.83597094])],
 ['Income_$0 - $24,999', array([-1.47475826])],
 ['Income_$100,000 - $149,999', array([ 0.73813893])],
 ['Income_$25,000 - $49,999', array([ 0.36657055])],
 ['Income_$50,000 - $99,999', array([ 1.02906071])],
 ['Income_150000', array([-1.19922127])],
 ['Edu_Bachelor degree', array([-0.63340808])],
 ['Edu_Graduate degree', array([-0.12244974])],
 ['Edu_High school degree', array([ 0.49817328])],
 ['Edu_Some college or Associate degree', array([-0.2825248])],
 ['Loca_East North Central', array([ 0.43194008])],
 ['Loca_East South Central', array([-0.43632887])],
 ['Loca_Middle Atlantic', array([-1.49115532])],
 ['Loca_Mountain', array([-0.87963371])],
 ['Loca_New England', array([ 2.3288712])],
 ['Loca_Pacific', array([-0.94334803])],
 ['Loca_South Atlantic', array([ 0.70821577])],
 ['Loca_West North Central', array([-0.72942912])],
 ['Loca_West South Central', array([ 0.47065865])],
 ['Child_<_18', array([-0.22820017])],
 ['Recline_obligation', array([ 0.11856305])],
 ['Eliminate_reclineseats', array([ 0.05678593])],
 ['Use_electronic_takeoff_land', array([ 0.87173783])]]
```

In [103]:

```python
from sklearn.linear_model import LogisticRegression
from sklearn import metrics, cross_validation

predicted = cross_validation.cross_val_predict\
(LogisticRegression(), test.iloc[:, :-1], test['Gender'], cv=10)

print metrics.classification_report(test['Gender'], predicted)
```

```
             precision    recall  f1-score   support

          0       0.73      0.74      0.73        54
          1       0.74      0.72      0.73        54

avg / total       0.73      0.73      0.73       108
```

In [105]:

```
#TEST data - Accuracy
from sklearn import model_selection
from sklearn.linear_model import LogisticRegression
seed = 7
kfold = model_selection.KFold(n_splits=10, random_state=seed)
model = LogisticRegression()
scoring = 'accuracy'
results = model_selection.cross_val_score(model, X_test, y_test,
                                          cv=kfold, scoring=scoring)
print('Accuracy: %.3f (%.3f)') % (results.mean(), results.std())
```

```
Accuracy: 0.696 (0.112)
```

In [106]:

```
#TEST data - AUC

seed = 7
kfold = model_selection.KFold(n_splits=10, random_state=seed)
model = LogisticRegression()
scoring = 'roc_auc'
results = model_selection.cross_val_score(model, X_test, y_test,
                                          cv=kfold, scoring=scoring)
print('AUC: %.3f (%.3f)') % (results.mean(), results.std())
```

```
AUC: 0.792 (0.105)
```

In [107]:

```
#TEST data - LogLoss
seed = 7
kfold = model_selection.KFold(n_splits=10, random_state=seed)
model = LogisticRegression()
scoring = 'neg_log_loss'
results = model_selection.cross_val_score(model, X_test, y_test,
                                          cv=kfold, scoring=scoring)
print('LogLoss: %.3f (%.3f)') % (results.mean(), results.std())
```

```
LogLoss: -0.617 (0.216)
```

In [96]:

```
D = test.columns.values.tolist()
D
```

Out[96]:

```
['Trav_Freq_A few times per month',
 'Trav_Freq_A few times per week',
 'Trav_Freq_Every day',
 'Trav_Freq_Once a month or less',
 'Trav_Freq_Once a year or less',
 'Recline_seat_Always',
 'Recline_seat_Never',
 'Recline_seat_Once in a while',
 'Recline_seat_Usually',
 'Height_5\'0"',
 'Height_5\'1"',
 'Height_5\'10"',
 'Height_5\'11"',
 'Height_5\'2"',
 'Height_5\'3"',
 'Height_5\'4"',
 'Height_5\'5"',
 'Height_5\'6"',
 'Height_5\'7"',
 'Height_5\'8"',
 'Height_5\'9"',
 'Height_6\'0"',
 'Height_6\'1"',
 'Height_6\'2"',
 'Height_6\'3"',
 'Height_6\'4"',
 'Height_6\'5"',
 'Height_Under 5 ft.',
 'Threeseat_arm_The arm rests should be shared',
 'Threeseat_arm_The person in the middle seat gets both arm rests',
 'Threeseat_arm_Whoever puts their arm on the arm rest first',
 'Twoseat_arm_The arm rests should be shared',
 'Twoseat_arm_The person by the window',
 'Twoseat_arm_The person in aisle',
 'Twoseat_arm_Whoever puts their arm on the arm rest first',
 'Unsold_seat_No, not rude at all',
 'Unsold_seat_Yes, somewhat rude',
 'Unsold_seat_Yes, very rude',
 'Few_words_No, not at all rude',
 'Few_words_Yes, somewhat rude',
 'Few_words_Yes, very rude',
 'Sixhrflt_Four times',
 'Sixhrflt_More than five times times',
 'Sixhrflt_Once',
 'Sixhrflt_Three times',
 'Sixhrflt_Twice',
 'Recline_obligation',
 'Rude_recline_No, not rude at all',
 'Rude_recline_Yes, somewhat rude',
 'Rude_recline_Yes, very rude',
 'Eliminate_reclineseats',
 'Switch_friend_No, not at all rude',
 'Switch_friend_Yes, somewhat rude',
 'Switch_friend_Yes, very rude',
 'Switch_fam_No, not at all rude',
 'Switch_fam_Yes, somewhat rude',
 'Switch_fam_Yes, very rude',
 'Wake_bath_No, not at all rude',
 'Wake_bath_Yes, somewhat rude',
 'Wake_bath_Yes, very rude',
 'Wake_walk_No, not at all rude',
 'Wake_walk_Yes, somewhat rude',
 'Wake_walk_Yes, very rude',
 'Bring_baby_No, not at all rude',
 'Bring_baby_Yes, somewhat rude',
 'Bring_baby_Yes, very rude',
 'Unruly_child_No, not at all rude',
 'Unruly_child_Yes, somewhat rude',
 'Unruly_child_Yes, very rude',
 'Age_18-29',
 'Age_30-44',
 'Age_45-60',
 'Age_> 60',
 'Income_$0 - $24,999',
 'Income_$100,000 - $149,999',
 'Income_$25,000 - $49,999',
 'Income_$50,000 - $99,999',
 'Income_150000',
 'Edu_Bachelor degree',
 'Edu_Graduate degree',
 'Edu_High school degree',
 'Edu_Some college or Associate degree',
 'Loca_East North Central',
 'Loca_East South Central',
 'Loca_Middle Atlantic',
 'Loca_Mountain',
 'Loca_New England',
 'Loca_Pacific',
 'Loca_South Atlantic',
 'Loca_West North Central',
 'Loca_West South Central',
 'Child_<_18',
 'Recline_obligation',
 'Eliminate_reclineseats',
 'Use_electronic_takeoff_land',
 'Gender']
```

In [ ]:

```

```

In [ ]:

```

```