User Average Active Time as Metric

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats
```

In [2]:

```
# reading the data sets
t1_user_active_min     = pd.read_csv("t1_user_active_min.csv")
t2_user_variant        = pd.read_csv("t2_user_variant.csv")
t3_user_active_min_pre = pd.read_csv("t3_user_active_min_pre.csv")
t4_user_attributes     = pd.read_csv("t4_user_attributes.csv")
```

Use table 1 and 2

In [3]:

```
t1_user_active_min.head()
```

Out[3]:

	uid	dt	active_mins
0	0	2019-02-22	5.0
1	0	2019-03-11	5.0
2	0	2019-03-18	3.0
3	0	2019-03-22	4.0
4	0	2019-04-03	9.0

In [4]:

```
t2_user_variant.head()
```

Out[4]:

	uid	variant_number	dt	signup_date
0	0	0	2019-02-06	2018-09-24
1	1	0	2019-02-06	2016-11-07
2	2	0	2019-02-06	2018-09-17
3	3	0	2019-02-06	2018-03-04
4	4	0	2019-02-06	2017-03-09

In [5]:

```
# Drop data with active_mins more than (24 hrs * 60 mins) on a single da
te.
t1_user_active_min = t1_user_active_min[t1_user_active_min.active_mins <
= 24*60]
```

In [6]:

```
# Total minutes per user
t1_user_active_min_mean = t1_user_active_min.groupby(by=['uid'], as_index=False).mean()
```

In [21]:

```
# merge t1 and t2 by uid
t12 = pd.merge(t2_user_variant, t1_user_active_min_mean, on='uid')
```

In [22]:

```
t12.head()
```

Out[22]:

	uid	variant_number	dt	signup_date	active_mins
0	0	0	2019-02-06	2018-09-24	3.307692
1	1	0	2019-02-06	2016-11-07	160.052632
2	2	0	2019-02-06	2018-09-17	2.428571
3	3	0	2019-02-06	2018-03-04	3.208333
4	4	0	2019-02-06	2017-03-09	1.950000

In [23]:

```
t12.variant_number.value_counts()
```

Out[23]:

```
0    37425
1     9208
Name: variant_number, dtype: int64
```

Compute confidence interval on difference of means.

In [24]:

```
stats12 = t12.groupby(['variant_number'])['active_mins'].agg(['mean', 'count', 'std','var'])
stats12
```

Out[24]:

	mean	count	std	var
variant_number				
0	8.761316	37425	18.780503	352.707296
1	11.638743	9208	21.102090	445.298205

In [25]:

```
sigma_diff = np.sqrt(stats12.loc[0]['var'] / stats12.loc[0]['count'] +
                     stats12.loc[1]['var'] / stats12.loc[1]['count'])
upper = (stats12.loc[1]['mean'] - stats12.loc[0]['mean']) + (1.96 * sigma_diff)
lower = (stats12.loc[1]['mean'] - stats12.loc[0]['mean']) - (1.96 * sigma_diff)
print([lower, upper])
```

```
[2.406274815429466, 3.3485787569999346]
```

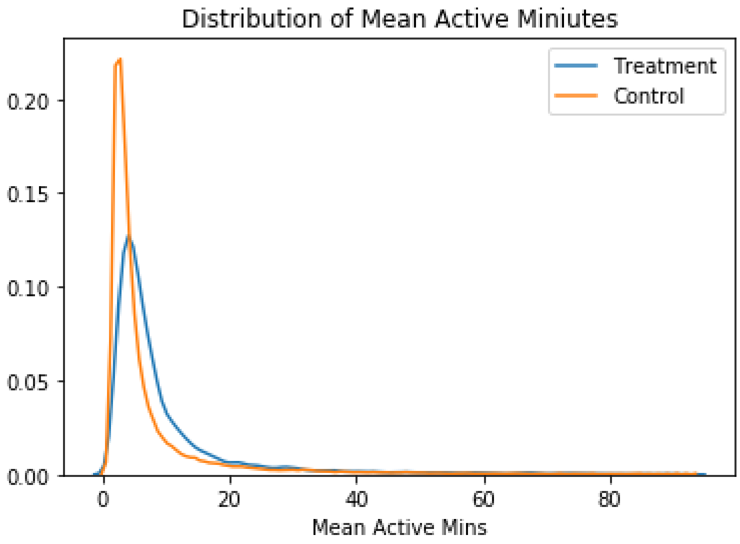
Perform a t-test:

```
In [26]: stats.ttest_ind(t12.active_mins[t12.variant_number==1], t12.active_mins[
         t12.variant_number==0], equal_var =False)
```

```
Out[26]: Ttest_indResult(statistic=11.970143076301513, pvalue=7.574338206129901e
         -33)
```

The low p-value suggests strong evidence for the new UI design to make a positive impact on total active time per user.

```
In [116]: build_dist(t12[(t12.variant_number==1) &
                          (t12.active_mins < t12.active_mins.quantile(0.99)) &
                          (t12.active_mins > t12.active_mins.quantile(0.01))],
                     t12[(t12.variant_number==0) &
                          (t12.active_mins < t12.active_mins.quantile(0.99)) &
                          (t12.active_mins > t12.active_mins.quantile(0.01))],
                     "active_mins","active_mins","Mean Active Mins", "Treatment",
          "Control",
                     title='Distribution of Mean Active Miniutes')
```



Add Table 3

In [27]:

```
t3_user_active_min_pre.head()
```

Out[27]:

	uid	dt	active_mins
0	0	2018-09-24	3.0
1	0	2018-11-08	4.0
2	0	2018-11-24	3.0
3	0	2018-11-28	6.0
4	0	2018-12-02	6.0

In [28]:

```
# Drop data with active_mins more than (24 hrs * 60 mins) on a single date.
t3_user_active_min_pre = t3_user_active_min_pre[t3_user_active_min_pre.active_mins <= 24*60]
```

In [29]:

```
# Total minutes per user
t3_user_active_min_pre_mean = t3_user_active_min_pre.groupby(by=['uid'], as_index=False).mean()
```

In [31]:

```
t123 = pd.merge(t12, t3_user_active_min_pre_mean, on='uid', suffixes=('_post', '_pre'))
```

In [32]:

```
t123.head()
```

Out[32]:

	uid	variant_number	dt	signup_date	active_mins_post	active_mins_pre
0	0	0	2019-02-06	2018-09-24	3.307692	3.333333
1	1	0	2019-02-06	2016-11-07	160.052632	158.330579
2	2	0	2019-02-06	2018-09-17	2.428571	3.700000
3	3	0	2019-02-06	2018-03-04	3.208333	5.684211
4	4	0	2019-02-06	2017-03-09	1.950000	2.357143

In [33]:

```
t123['active_mins_diff'] = t123['active_mins_post'] - t123['active_mins_pre']
```

```
In [34]: stats123 = t123.groupby(['variant_number'])['active_mins_post', 'active_
         mins_pre', 'active_mins_diff'].agg(['mean', 'count', 'std','var'])
         stats123
```

Out[34]:

	active_mins_post				active_mins_pre			
	mean	**count**	**std**	**var**	**mean**	**count**	**std**	**var**
variant_number								
0	8.778037	37313	18.805182	353.634855	9.114909	37313	18.827314	354.46774
1	11.668744	9165	21.145444	447.129817	7.221025	9165	12.555481	157.64009

Difference in difference

For each user, first obtain difference in post-test user average time and pre-test user average time: $\Delta x_1 \equiv x_{1,post} - x_{1,pre}$ and $\Delta x_2 \equiv x_{0,post} - x_{0,pre}$

Then obtain the difference in mean: $\overline{\Delta x_1} - \overline{\Delta x_2}$

Standard Error: $\sqrt{\frac{s^2_{\Delta x_1}}{n_1} + \frac{s^2_{\Delta x_2}}{n_1}}$

```
In [35]: mean_did = stats123.loc[1,'active_mins_diff']['mean'] - stats123.loc[0,
         'active_mins_diff']['mean']

         sigma_did = np.sqrt(stats123.loc[1,'active_mins_diff']['var'] / stats123
         .loc[1,'active_mins_diff']['count'] +
                             stats123.loc[0,'active_mins_diff']['var'] / stats123
         .loc[0,'active_mins_diff']['count'])

         upper = mean_did + (1.96 * sigma_did)
         lower = mean_did - (1.96 * sigma_did)
         print([lower, upper])

         [4.499019292447001, 5.070162945220937]
```

Perform a t-test:

```
In [36]: stats.ttest_ind(t123.active_mins_diff[t123.variant_number==1], t123.acti
         ve_mins_diff[t123.variant_number==0], equal_var =False)

Out[36]: Ttest_indResult(statistic=32.838668686479814, pvalue=2.6124834427323526
         e-226)
```

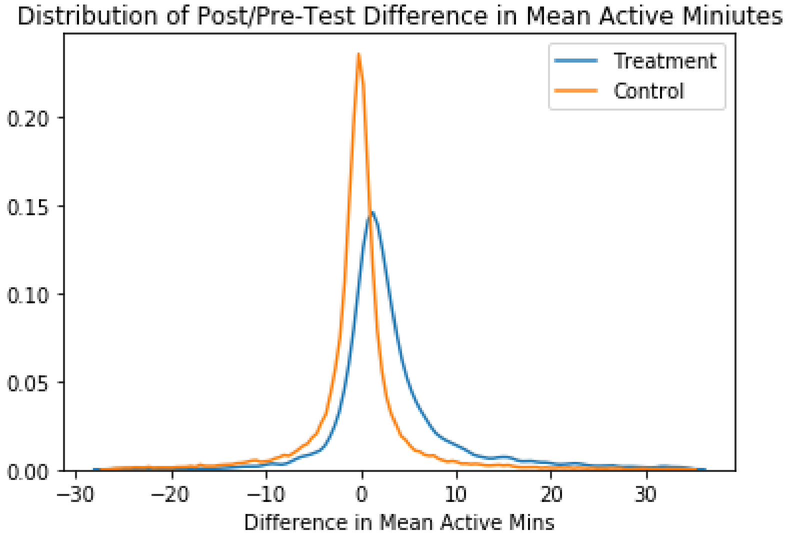
The low p-value suggests high evidence for the new UI design to make a positive impact on total active time per user.

In [37]:

```
def build_dist(df1, df2, col1, col2, xlabel, legend1, legend2, title):
    sns.distplot(df1[col1], hist=False, label = legend1, axlabel = xlabe
l)
    sns.distplot(df2[col2], hist=False, label = legend2, axlabel = xlabe
l)
    plt.title(title)
    #plt.xlim([-1000,1000])
    plt.show()
```

In [111]:

```
build_dist(t123[(t123.variant_number==1) &
                (t123.active_mins_diff < t123.active_mins_diff.quantile(
0.99)) &
                (t123.active_mins_diff > t123.active_mins_diff.quantile(
0.01))],
           t123[(t123.variant_number==0) &
                (t123.active_mins_diff < t123.active_mins_diff.quantile(
0.99)) &
                (t123.active_mins_diff > t123.active_mins_diff.quantile(
0.01))],
           "active_mins_diff","active_mins_diff","Difference in Mean Act
ive Mins", "Treatment","Control",
           title='Distribution of Post/Pre-Test Difference in Mean Activ
e Miniutes')
```

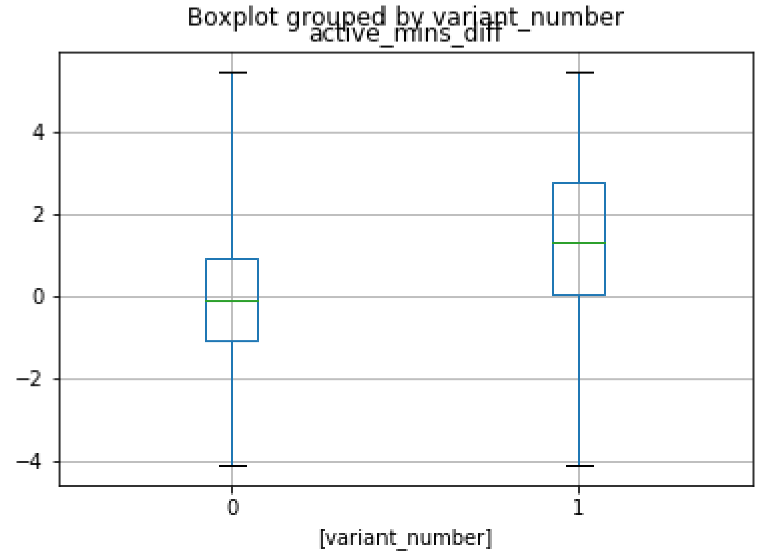


In [49]:

```
t123[(t123.active_mins_diff < t123.active_mins_diff.quantile(0.9)) &
     (t123.active_mins_diff > t123.active_mins_diff.quantile(0.1))][['variant_number','active_mins_diff']].boxplot(by=['variant_number'],whis='range')
```

Out[49]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x134035d30>
```



Add Table 4

In [57]:

```
t4_user_attributes.head()
```

Out[57]:

	uid	gender	user_type
0	0	male	non_reader
1	1	male	reader
2	2	male	non_reader
3	3	male	non_reader
4	4	male	non_reader

In [58]:

```
t1234 = pd.merge(t4_user_attributes, t123, on='uid')
```

In [59]:
```
t1234.head()
```

Out[59]:

	uid	gender	user_type	variant_number	dt	signup_date	active_mins_post	active_mins_pr
0	0	male	non_reader	0	2019-02-06	2018-09-24	3.307692	3.33333
1	1	male	reader	0	2019-02-06	2016-11-07	160.052632	158.33057
2	2	male	non_reader	0	2019-02-06	2018-09-17	2.428571	3.70000
3	3	male	non_reader	0	2019-02-06	2018-03-04	3.208333	5.68421
4	4	male	non_reader	0	2019-02-06	2017-03-09	1.950000	2.35714

In [60]:
```
pd.crosstab(t1234['variant_number'],t1234['gender']).apply(lambda r: r/r
.sum(), axis=1)
```

Out[60]:

gender	female	male	unknown
variant_number			
0	0.286737	0.561493	0.151770
1	0.282815	0.559083	0.158101

In [61]:
```
pd.crosstab(t1234['variant_number'],t1234['user_type']).apply(lambda r:
r/r.sum(), axis=1)
```

Out[61]:

user_type	contributor	new_user	non_reader	reader
variant_number				
0	0.024415	0.061185	0.735240	0.17916
1	0.013857	0.084015	0.764648	0.13748

In [62]:
```
pd.crosstab(t1234['gender'],t1234['user_type']).apply(lambda r: r/r.sum
(), axis=1)
```

Out[62]:

user_type	contributor	new_user	non_reader	reader
gender				
female	0.018734	0.074637	0.745166	0.161463
male	0.025849	0.055801	0.732579	0.185772
unknown	0.016170	0.085208	0.764342	0.134280

Gender

In [63]:

```
stats1234_gender = t1234.groupby(['variant_number','gender'])['active_mi
ns_post', 'active_mins_pre', 'active_mins_diff'].agg(['mean', 'count',
'var'])
stats1234_gender
```

Out[63]:

		active_mins_post			active_mins_pre			active_min
		mean	count	var	mean	count	var	mean
variant_number	gender							
0	female	7.642952	10699	234.200195	8.213522	10699	265.149126	-0.570570
	male	9.704124	20951	424.230905	9.872825	20951	414.573699	-0.168701
	unknown	7.496357	5663	310.960230	8.013876	5663	296.083171	-0.517519
1	female	10.590656	2592	325.528234	6.584398	2592	97.235650	4.006258
	male	12.397131	5124	516.836077	7.553167	5124	187.825623	4.843964
	unknown	11.021509	1449	414.340209	7.185305	1449	158.030374	3.836204

In [64]:

```
def ConfidenceInterval(stats, attribute):
    mean_did = stats.loc[1, attribute]['active_mins_diff', 'mean'] - sta
ts.loc[0,attribute]['active_mins_diff', 'mean']
    SE_did = np.sqrt(stats.loc[1, attribute]['active_mins_diff','var'] /
stats.loc[1, attribute]['active_mins_diff','count'] +
                     stats.loc[0, attribute]['active_mins_diff','var'] /
stats.loc[0, attribute]['active_mins_diff','count'])

    upper = mean_did + (1.96 * SE_did)
    lower = mean_did - (1.96 * SE_did)
    return [lower, upper]
```

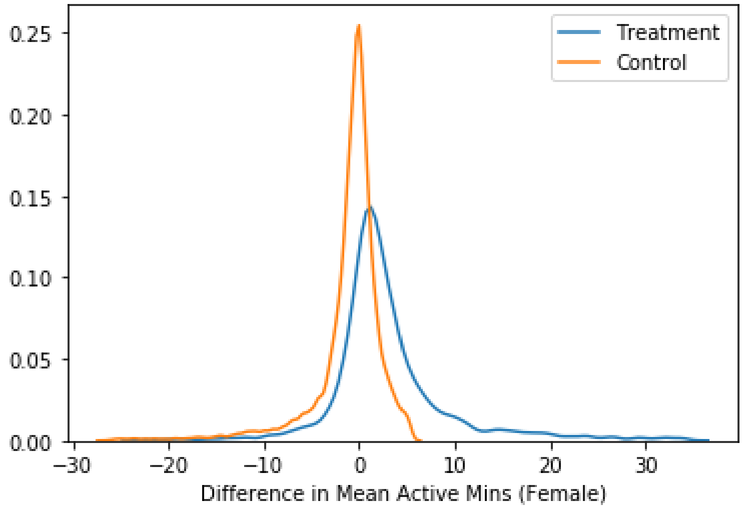
In [65]:

```
intervals = {}
for g in set(t4_user_attributes['gender']):
    intervals[g] = ConfidenceInterval(stats1234_gender, g)
    print(g, ':', ConfidenceInterval(stats1234_gender, g))
```

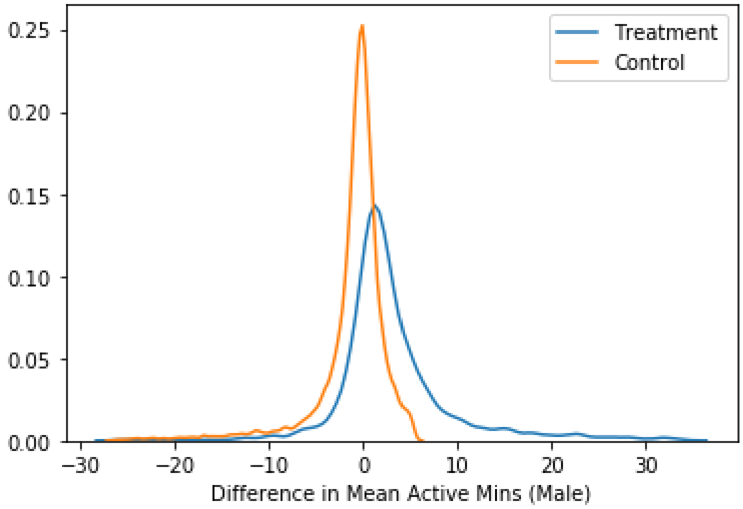
```
female : [4.082668281657273, 5.070986716879877]
male : [4.609718461419328, 5.415611021626094]
unknown : [3.6834453016973905, 5.023999837578068]
```

```
In [112]: for g in set(t4_user_attributes['gender']):
              build_dist(t1234[(t1234.variant_number==1) & (t1234.gender==g) &
                          (t1234.active_mins_diff < t1234.active_mins_diff.quantil
          e(0.99)) &
                          (t1234.active_mins_diff > t1234.active_mins_diff.quantil
          e(0.01))],
                     t1234[(t1234.variant_number==0) & (t1234.gender==g) &
                          (t1234.active_mins_diff < t1234.active_mins_diff.quantil
          e(0.90)) &
                          (t1234.active_mins_diff > t1234.active_mins_diff.quantil
          e(0.01))],
                     "active_mins_diff","active_mins_diff","Difference in Mean Act
          ive Mins " + '(' + g.title() + ')', "Treatment","Control",
                     title='Distribution of Post/Pre-Test Difference in Mean Activ
          e Miniutes ' + '(' + g.title() + ')')
```

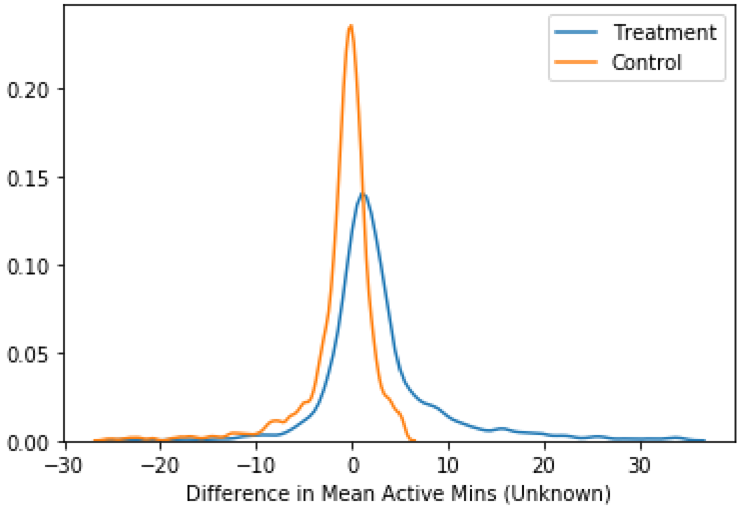
Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Female)



Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Male)



Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Unknown)



User Type

In [68]:

```
stats1234_user_type = t1234.groupby(['variant_number','user_type'])['active_mins_post', 'active_mins_pre', 'active_mins_diff'].agg(['mean', 'count', 'var'])
stats1234_user_type
```

Out[68]:

		active_mins_post			active_mins_pre			active
		mean	count	var	mean	count	var	mean
variant_number	user_type							
0	contributor	53.160145	911	4151.228369	56.656796	911	4184.622372	-3.49
	new_user	3.441046	2283	22.749363	4.483866	2283	32.671914	-1.04
	non_reader	4.447460	27434	29.756430	4.555002	27434	25.243865	-0.10
	reader	22.324380	6685	740.317113	22.930689	6685	702.626548	-0.60
1	contributor	65.306837	127	6568.871801	44.492685	127	2990.703181	20.81
	new_user	6.242814	770	54.127892	4.698840	770	27.781261	1.54
	non_reader	7.508955	7008	68.836087	4.610943	7008	16.323160	2.89
	reader	32.714566	1260	1333.164449	19.522641	1260	406.913170	13.19

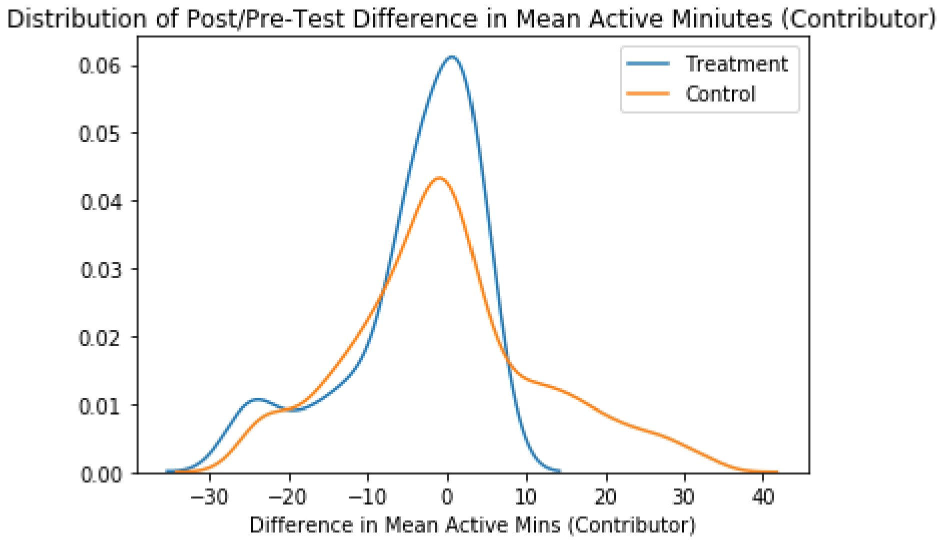
In [69]:

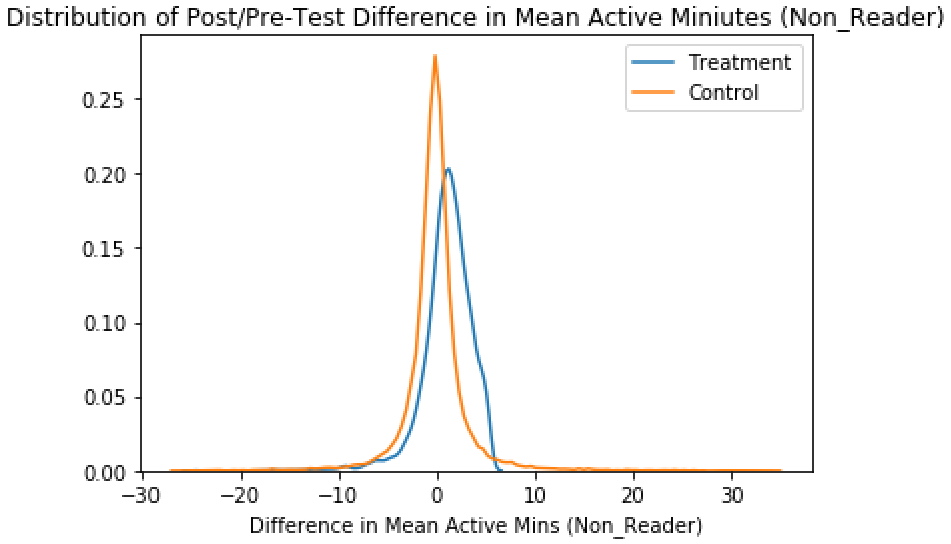
```
for g in set(t4_user_attributes['user_type']):
    intervals[g] = ConfidenceInterval(stats1234_user_type, g)
    print(g, ':', ConfidenceInterval(stats1234_user_type, g))
```

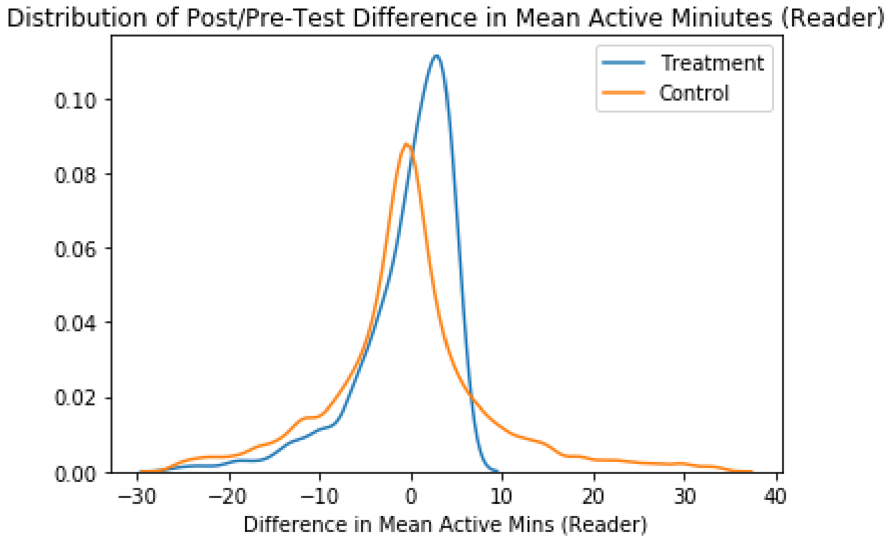
```
contributor : [16.995995749451346, 31.625610810993443]
non_reader : [2.8231488470903954, 3.1879584027166366]
reader : [12.376071976734565, 15.220396987874775]
new_user : [1.9489593821827849, 3.2246293485292896]
```

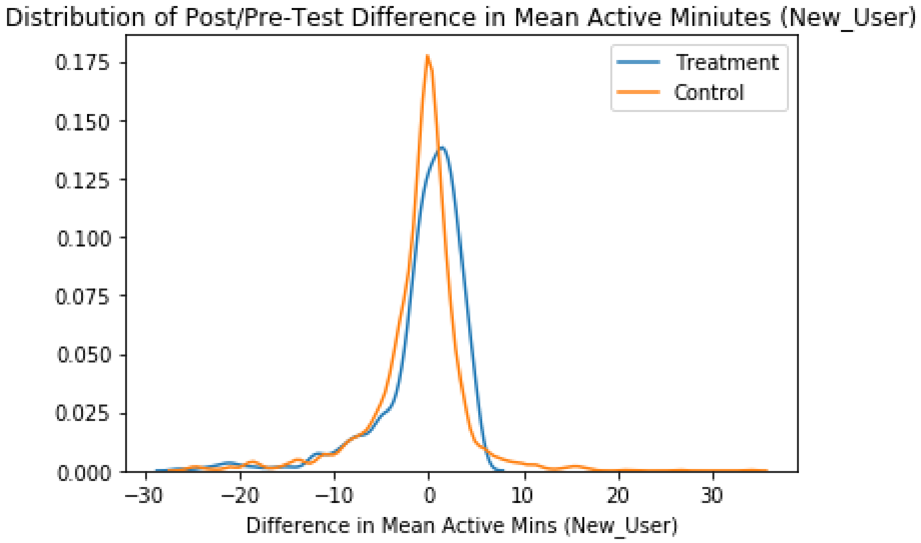
In [113]:

```
for g in set(t4_user_attributes['user_type']):
    build_dist(t1234[(t1234.variant_number==1) & (t1234.user_type==g) &
                     (t1234.active_mins_diff < t1234.active_mins_diff.quantil
e(0.90)) &
                     (t1234.active_mins_diff > t1234.active_mins_diff.quantil
e(0.01))],
               t1234[(t1234.variant_number==0) & (t1234.user_type==g) &
                     (t1234.active_mins_diff < t1234.active_mins_diff.quantil
e(0.99)) &
                     (t1234.active_mins_diff > t1234.active_mins_diff.quantil
e(0.01))],
               "active_mins_diff","active_mins_diff","Difference in Mean Act
ive Mins " + '(' + g.title() + ')', "Treatment","Control",
               title='Distribution of Post/Pre-Test Difference in Mean Activ
e Miniutes ' + '(' + g.title() + ')')
```


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Contributor)


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Non_Reader)


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Reader)



Gender and User Type

In [72]:

```
stats1234 = t1234.groupby(['variant_number','user_type','gender'])['active_mins_post', 'active_mins_pre', 'active_mins_diff'].agg(['mean', 'count', 'var'])
stats1234
```

Out[72]:

			active_mins_post			active_mins_pre		
			mean	count	var	mean	count	var
variant_number	user_type	gender						
0	contributor	female	38.714459	223	2066.235264	46.924005	223	2498.2337
		male	58.017646	592	4452.855001	59.510130	592	4642.2897
		unknown	56.761676	96	6584.499784	61.669696	96	5067.883
	new_user	female	3.306261	732	12.007781	4.266963	732	23.3832
		male	3.476038	1090	17.301803	4.591055	1090	31.1227
		unknown	3.572328	461	52.765472	4.574837	461	51.1318
	non_reader	female	4.353322	7932	41.899578	4.525700	7932	40.1907
		male	4.545964	15175	27.172688	4.592553	15175	20.0565
		unknown	4.274572	4327	16.490376	4.477025	4327	16.0353
	reader	female	19.971238	1812	615.897040	21.187164	1812	655.8153
		male	23.495545	4094	786.285890	23.673501	4094	711.3717
		unknown	21.642918	779	769.423072	23.082419	779	757.3840
1	contributor	female	53.337951	26	3460.047432	34.521595	26	892.4560
		male	72.437659	82	7224.506929	49.235937	82	3690.0442
		unknown	50.910186	19	8008.855289	37.666457	19	2794.9114
	new_user	female	6.033502	260	33.294186	4.900733	260	30.7997
		male	6.098764	365	38.696240	4.404795	365	22.6072
		unknown	6.980743	145	130.679167	5.077011	145	35.3800
	non_reader	female	7.242834	1972	60.681066	4.389482	1972	14.3198
		male	7.719148	3927	76.425539	4.685607	3927	16.264
		unknown	7.237863	1109	56.219090	4.740352	1109	20.0015
	reader	female	30.576685	334	1262.135957	18.679489	334	342.4418
		male	33.391819	750	1364.267695	19.542601	750	407.2959
		unknown	33.885655	176	1338.367584	21.037655	176	528.9380

In [73]:

```
covariates = [[a,b] for a in set(t4_user_attributes['user_type']) for b in set(t4_user_attributes['gender'])]
len(covariates)
```

Out[73]:

```
12
```

In [74]:

```
def ConfidenceInterval_covariate(stats, covariate):
    mean_did = stats.loc[1, covariate[0], covariate[1]]['active_mins_diff', 'mean'] - stats.loc[0, covariate[0], covariate[1]]['active_mins_diff', 'mean']
    SE_did = np.sqrt(stats.loc[1, covariate[0], covariate[1]]['active_mins_diff','var'] / stats.loc[1, covariate[0], covariate[1]]['active_mins_diff','count'] +
                     stats.loc[0, covariate[0], covariate[1]]['active_mins_diff','var'] / stats.loc[0, covariate[0], covariate[1]]['active_mins_diff','count'])

    upper = mean_did + (1.96 * SE_did)
    lower = mean_did - (1.96 * SE_did)
    return [lower, upper]
```

In [75]:

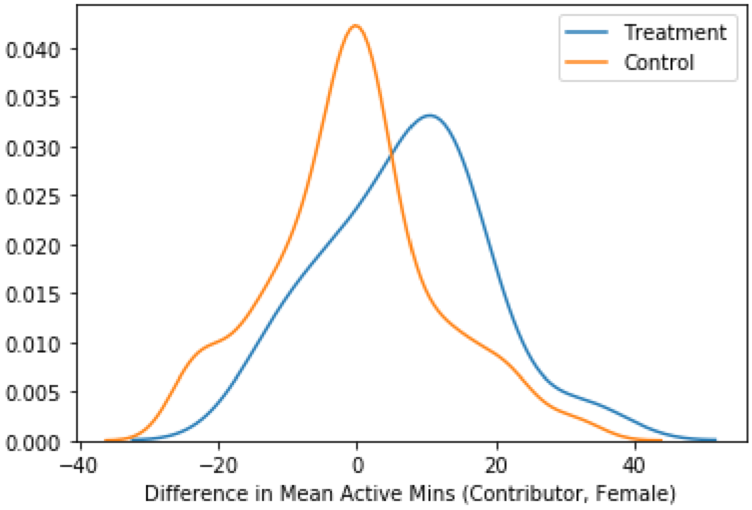
```
for c in covariates:
    intervals['_'.join(c)] = ConfidenceInterval_covariate(stats1234, c)
    print(c, ':', ConfidenceInterval_covariate(stats1234, c))
```

```
['contributor', 'female'] : [12.130744823642814, 41.92105855818548]
['contributor', 'male'] : [15.730777970918131, 33.65763474572131]
['contributor', 'unknown'] : [-5.206375673663516, 41.50987221834224]
['non_reader', 'female'] : [2.700530875954784, 3.350930907232195]
['non_reader', 'male'] : [2.8255329034443393, 3.334727150265173]
['non_reader', 'unknown'] : [2.27444359904215, 3.1254824622100603]
['reader', 'female'] : [10.39133884786024, 15.834906003385534]
['reader', 'male'] : [12.112992512053513, 15.941355559229901]
['reader', 'unknown'] : [11.037669873335581, 17.537331735804244]
['new_user', 'female'] : [1.0973757799120127, 3.0895669986213443]
['new_user', 'male'] : [2.033044325735124, 3.584928320328269]
['new_user', 'unknown'] : [0.7886594445558002, 5.023821684977663]
```

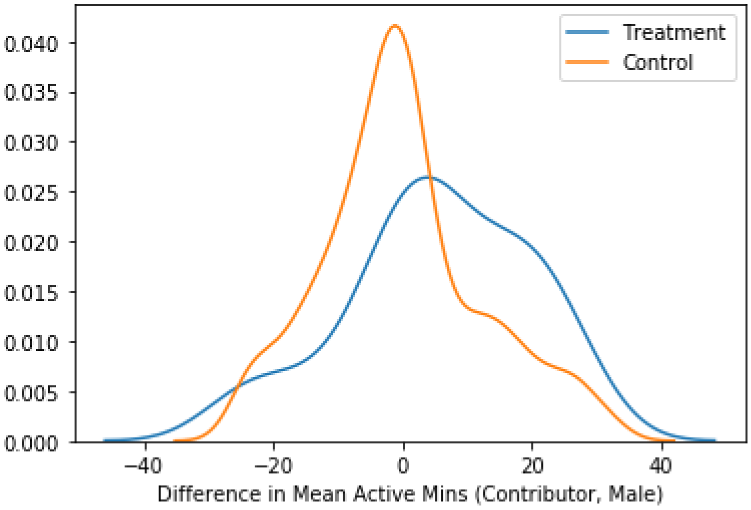
In [76]:

```
df_intervals = pd.DataFrame(intervals)
```

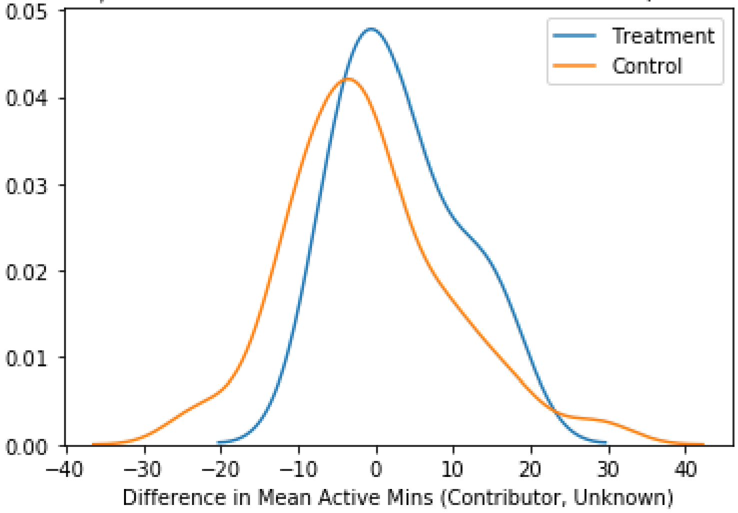
```
In [114]: for c in covariates:
              build_dist(t1234[(t1234.variant_number==1) & (t1234.user_type==c[0])
          & (t1234.gender==c[1]) &
                                (t1234.active_mins_diff < t1234.active_mins_diff.quantil
          e(0.99)) &
                                (t1234.active_mins_diff > t1234.active_mins_diff.quantil
          e(0.01))],
                           t1234[(t1234.variant_number==0) & (t1234.user_type==c[0]) & (
          t1234.gender==c[1]) &
                                (t1234.active_mins_diff < t1234.active_mins_diff.quantil
          e(0.99)) &
                                (t1234.active_mins_diff > t1234.active_mins_diff.quantil
          e(0.01))],
                           "active_mins_diff","active_mins_diff","Difference in Mean Act
          ive Mins " + '(' + c[0].title() + ', ' + c[1].title() + ')', "Treatment"
          ,"Control",
                           title='Distribution of Post/Pre-Test Difference in Mean Activ
          e Miniutes ' + '(' + c[0].title() + ', ' + c[1].title() + ')')
```



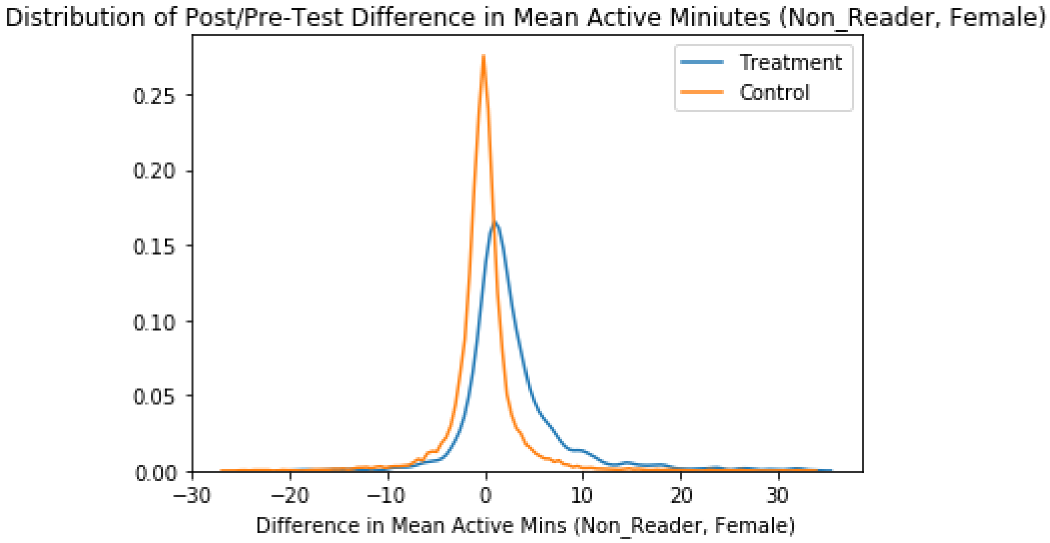
Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Contributor, Female)

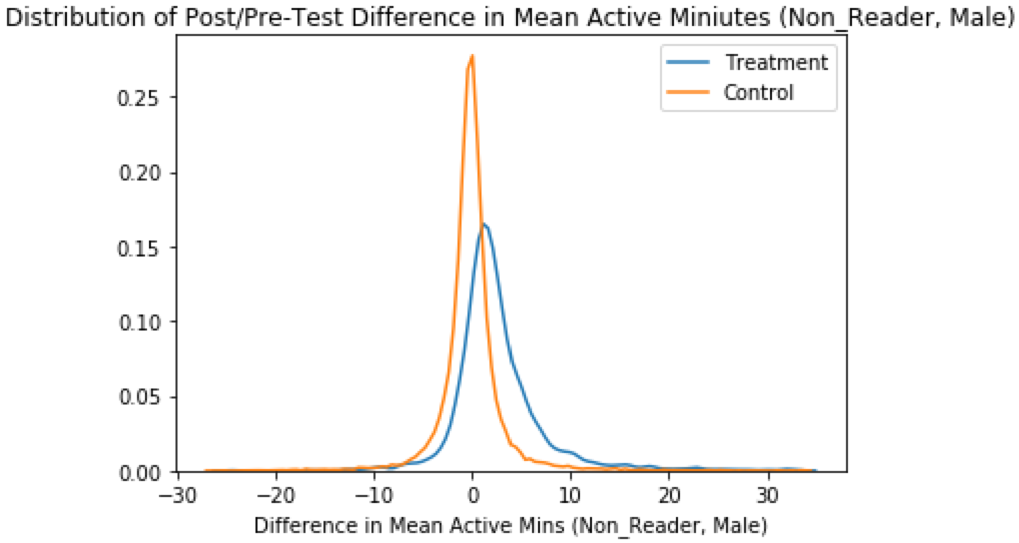


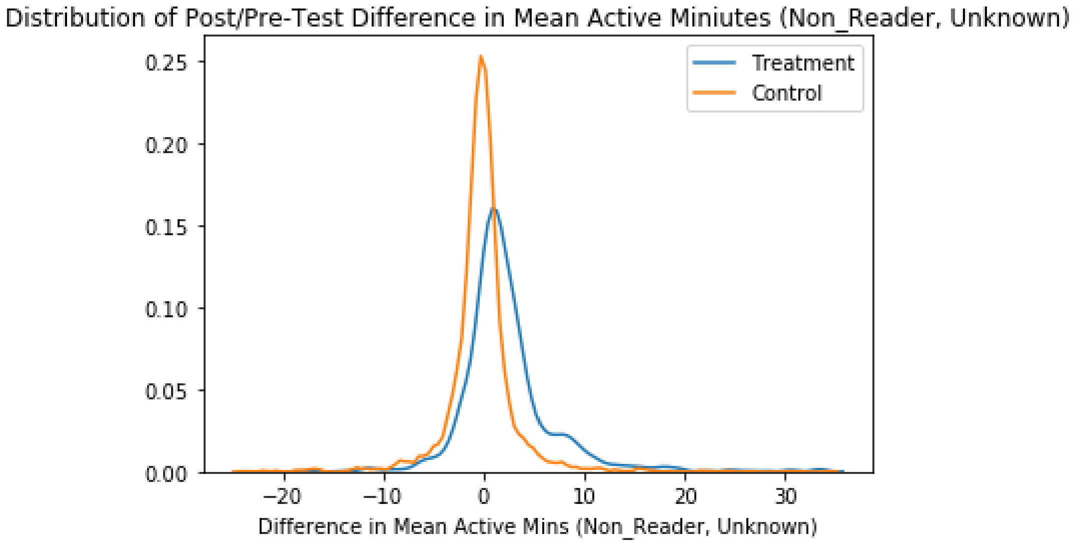
Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Contributor, Male)

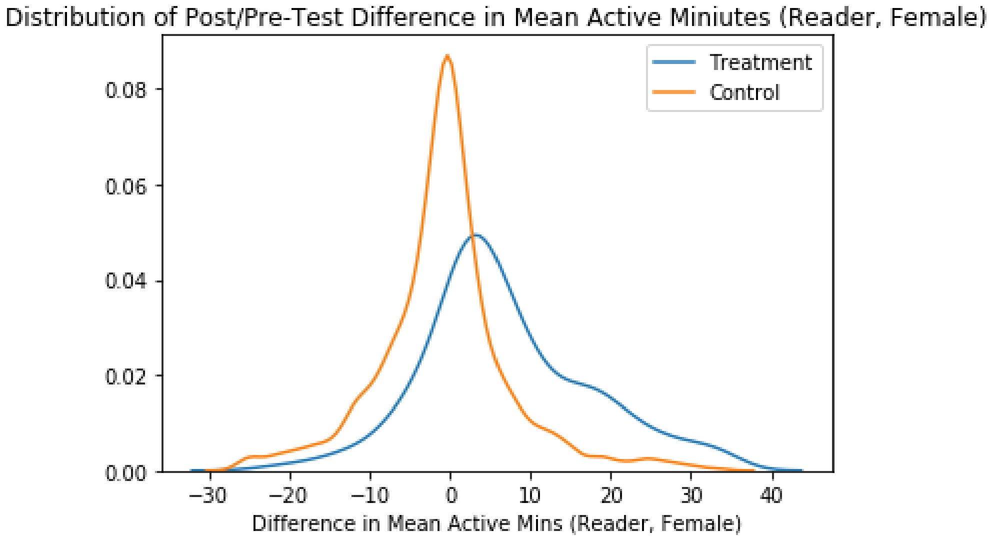


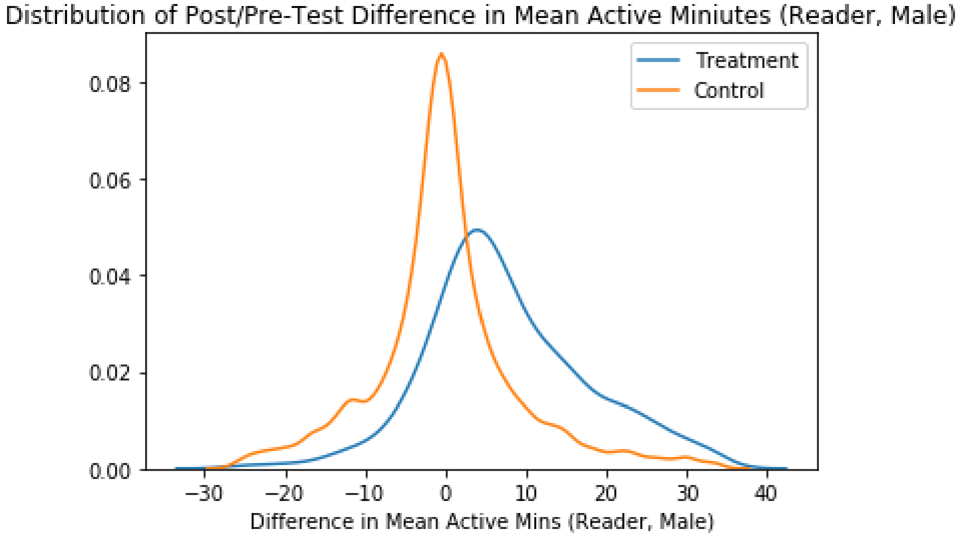
Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Contributor, Unknown)

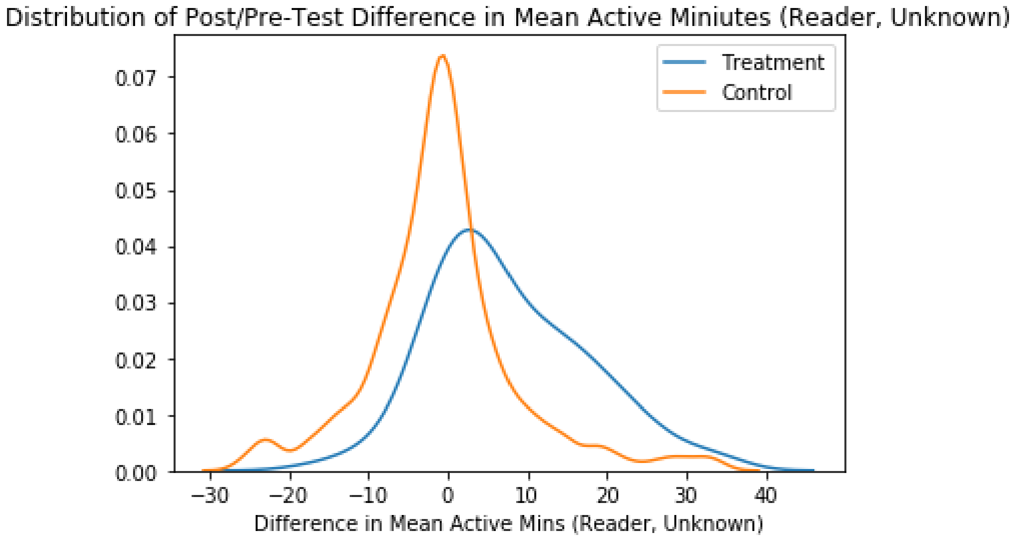

Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Non_Reader, Female)
Treatment
Control
0.25
0.20
0.15
0.10
0.05
0.00
−30
−20
−10
0
10
20
30
Difference in Mean Active Mins (Non_Reader, Female)


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Non_Reader, Male)
Treatment
Control
0.25
0.20
0.15
0.10
0.05
0.00
−30
−20
−10
0
10
20
30
Difference in Mean Active Mins (Non_Reader, Male)

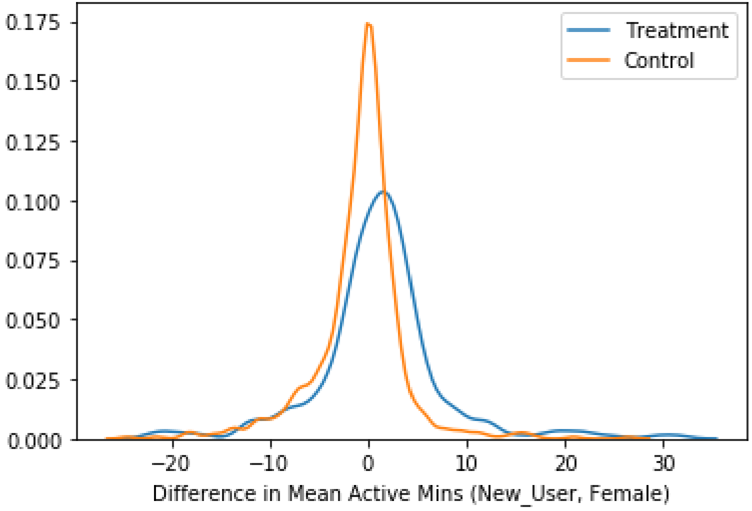

Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Non_Reader, Unknown)
Treatment
Control
0.25
0.20
0.15
0.10
0.05
0.00
−20
−10
0
10
20
30
Difference in Mean Active Mins (Non_Reader, Unknown)


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Reader, Female)
Treatment
Control
0.08
0.06
0.04
0.02
0.00
−30
−20
−10
0
10
20
30
40
Difference in Mean Active Mins (Reader, Female)

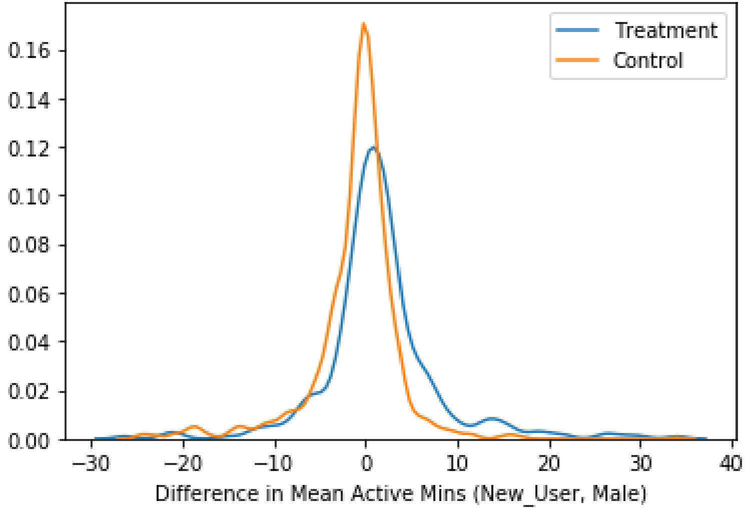

Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Reader, Male)
Treatment
Control
0.08
0.06
0.04
0.02
0.00
−30
−20
−10
0
10
20
30
40
Difference in Mean Active Mins (Reader, Male)


Distribution of Post/Pre-Test Difference in Mean Active Miniutes (Reader, Unknown)
Treatment
Control
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
−30
−20
−10
0
10
20
30
40
Difference in Mean Active Mins (Reader, Unknown)

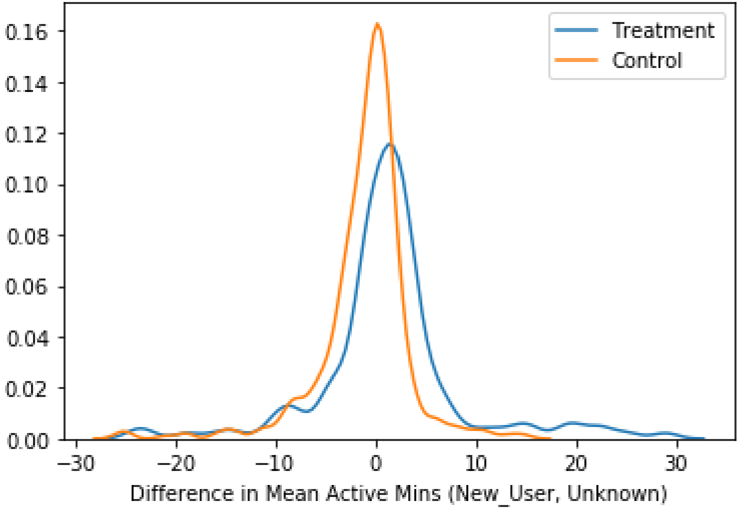
Distribution of Post/Pre-Test Difference in Mean Active Miniutes (New_User, Female)



Distribution of Post/Pre-Test Difference in Mean Active Miniutes (New_User, Male)



Distribution of Post/Pre-Test Difference in Mean Active Miniutes (New_User, Unknown)



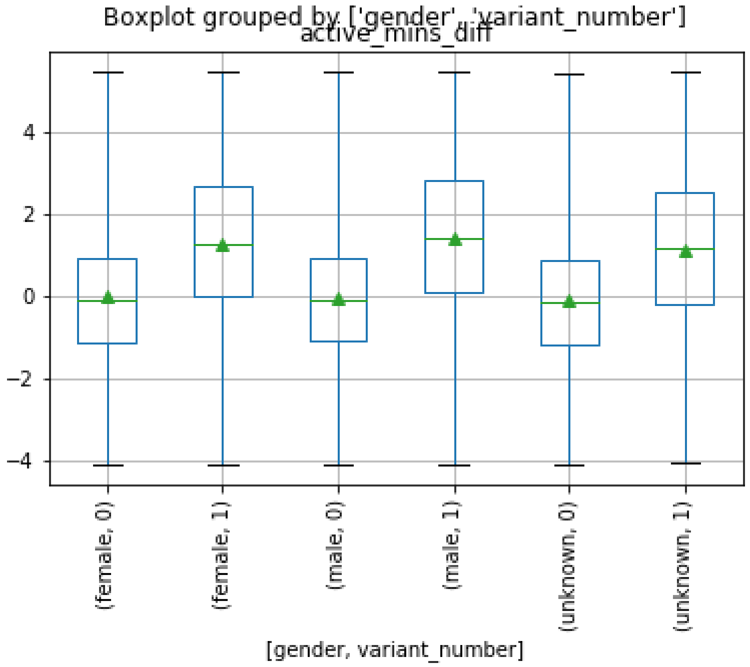
In [103]:

```
t1234[(t1234.active_mins_diff < t1234.active_mins_diff.quantile(0.9)) &
     (t1234.active_mins_diff > t1234.active_mins_diff.quantile(0.1))][[
'variant_number','gender', 'active_mins_diff']].boxplot(by=['gender', 'v
ariant_number'],whis='range',

showmeans=True)
plt.xticks(rotation=90)
```

Out[103]:

```
(array([1, 2, 3, 4, 5, 6]), <a list of 6 Text xticklabel objects>)
```

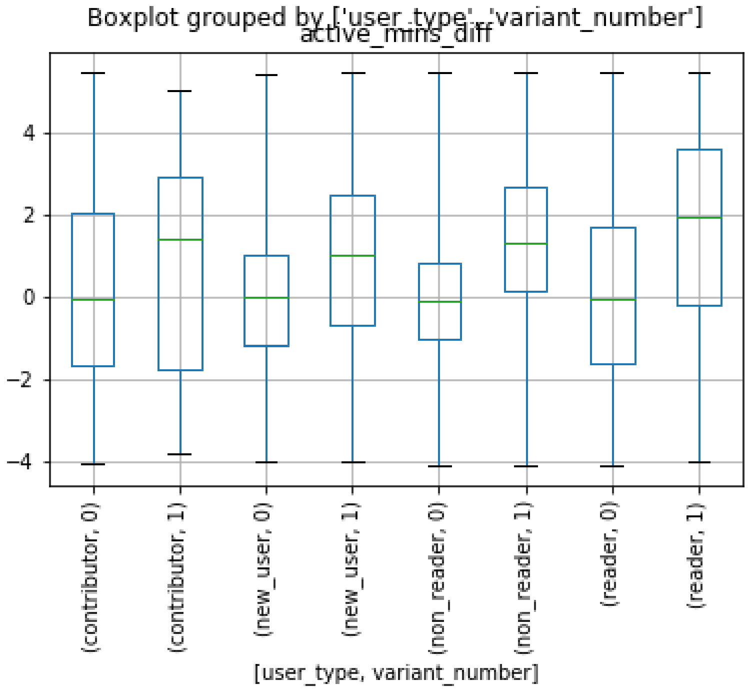


In [104]:

```
t1234[(t1234.active_mins_diff < t1234.active_mins_diff.quantile(0.9)) &
     (t1234.active_mins_diff > t1234.active_mins_diff.quantile(0.1))][[
'variant_number', 'user_type', 'active_mins_diff']].boxplot(by=['user_type', 'variant_number'], whis='range')
plt.xticks(rotation=90)
```

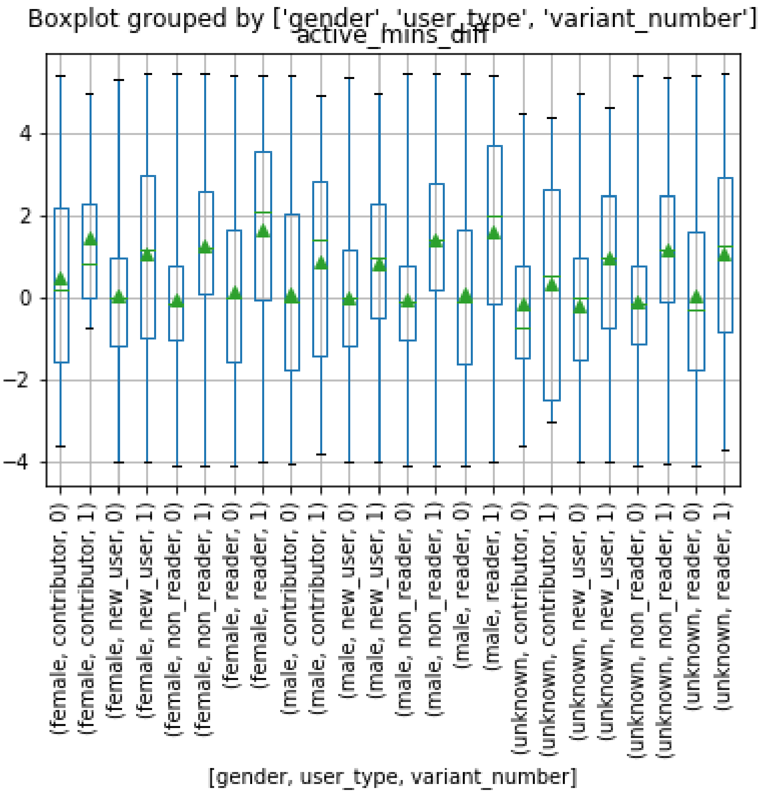
Out[104]:

```
(array([1, 2, 3, 4, 5, 6, 7, 8]), <a list of 8 Text xticklabel objects>)
```



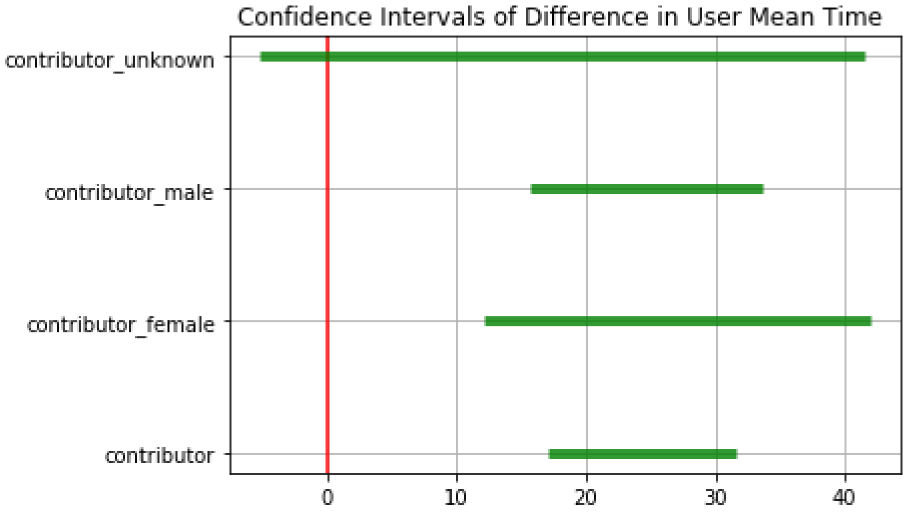
```
In [105]: t1234[(t1234.active_mins_diff < t1234.active_mins_diff.quantile(0.9)) &
               (t1234.active_mins_diff > t1234.active_mins_diff.quantile(0.1))][[
          'variant_number', 'gender', 'user_type', 'active_mins_diff']].boxplot(by
          =['gender', 'user_type', 'variant_number'], showmeans=True, whis='range'
          )
          plt.xticks(rotation=90)
```

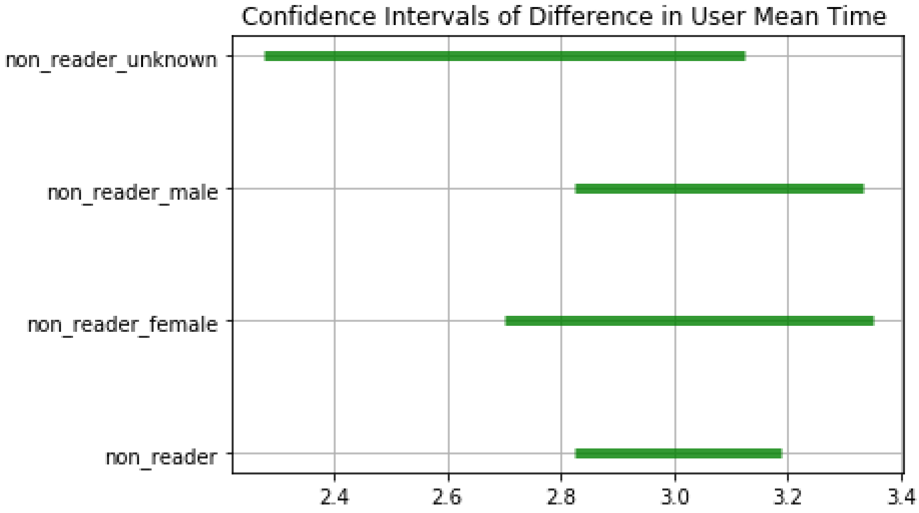
```
Out[105]: (array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
          17,
                  18, 19, 20, 21, 22, 23, 24]), <a list of 24 Text xticklabel obj
          ects>)
```

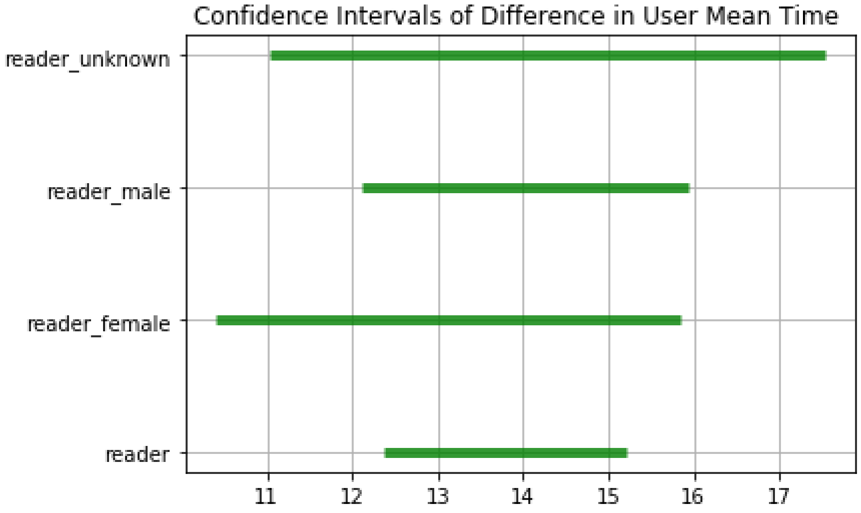


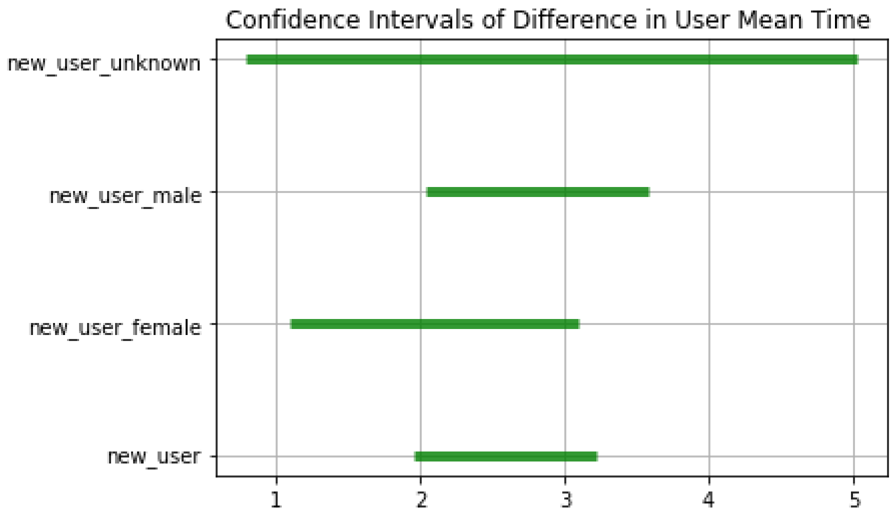
In [107]:

```
for u in set(t4_user_attributes.user_type):
    plt.figure()
    plt.title('Confidence Intervals of Difference in User Mean Time ')
    plt.grid()
    plt.vlines(0, -1, 4, colors='r')
    for i in df_intervals.columns[df_intervals.columns.str.startswith(u
)]:
        plt.hlines(i, df_intervals[i][0], df_intervals[i][1], colors='g'
, alpha=0.8, linestyles='solid', label=i, linewidth=5.0)
```


Confidence Intervals of Difference in User Mean Time
contributor_unknown
contributor_male
contributor_female
contributor
0
10
20
30
40


Confidence Intervals of Difference in User Mean Time
non_reader_unknown
non_reader_male
non_reader_female
non_reader
2.4
2.6
2.8
3.0
3.2
3.4


Confidence Intervals of Difference in User Mean Time
reader_unknown
reader_male
reader_female
reader
11
12
13
14
15
16
17


Confidence Intervals of Difference in User Mean Time
new_user_unknown
new_user_male
new_user_female
new_user
1
2
3
4
5