

```
In [1]: import pandas as pd
import numpy as np
import re
```

Data Source:

https://www.dropbox.com/s/lfsbhwy8elgjo15/UCI_HAR_Dataset.zip?dl=0
[\(https://www.dropbox.com/s/lfsbhwy8elgjo15/UCI_HAR_Dataset.zip?dl=0\)](https://www.dropbox.com/s/lfsbhwy8elgjo15/UCI_HAR_Dataset.zip?dl=0)

Importing features

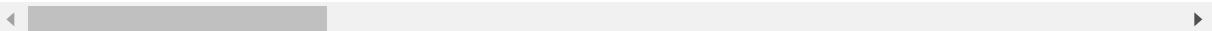
```
In [2]: features = pd.read_csv('features.txt', sep= " ", header = None)
features = features.drop(features.columns[0], axis =1)
```

```
In [3]: features = features.T
features.head()
```

Out[3]:

	0	1	2	3	4	5	6	7	
1	tBodyAcc-mean()-X	tBodyAcc-mean()-Y	tBodyAcc-mean()-Z	tBodyAcc-std()-X	tBodyAcc-std()-Y	tBodyAcc-std()-Z	tBodyAcc-mad()-X	tBodyAcc-mad()-Y	tBodyAcc-mad()-Z

1 rows × 561 columns



```
In [4]: header = features.iloc[0]
```

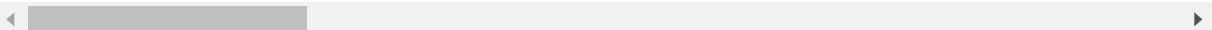
```
In [5]: features = features.rename(columns = header)
```

```
In [6]: features
```

Out[6]:

	tBodyAcc-mean()-X	tBodyAcc-mean()-Y	tBodyAcc-mean()-Z	tBodyAcc-std()-X	tBodyAcc-std()-Y	tBodyAcc-std()-Z	tBodyAcc-mad()-X	tBodyAcc-mad()-Y	tBodyAcc-mad()-Z
1	tBodyAcc-mean()-X	tBodyAcc-mean()-Y	tBodyAcc-mean()-Z	tBodyAcc-std()-X	tBodyAcc-std()-Y	tBodyAcc-std()-Z	tBodyAcc-mad()-X	tBodyAcc-mad()-Y	tBodyAcc-mad()-Z

1 rows × 561 columns



```
In [7]: A = []
for i in features:

    j = re.sub("-", "|", i.lower())
    z = i, j

    A.append(j)
```

```
In [8]: A = pd.DataFrame(A)
A=A.T
```

```
In [9]: A
```

```
Out[9]:
```

	0	1	2	3	4	5	
0	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz	tbodyac

1 rows × 561 columns



Lets import X_train and merge the file with features

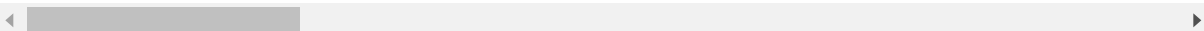
```
In [10]: X_train = pd.read_fwf('X_train.txt', sep=" ", header=None)
```

```
In [11]: X_train = pd.concat([A, X_train], ignore_index=True, sort=False, axis=0)
X_train.head()
```

```
Out[11]:
```

	0	1	2	3	4	5	
0	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz	tbodyac
1	0.288585	-0.0202942	-0.132905	-0.995279	-0.983111	-0.913526	-0.
2	0.278419	-0.0164106	-0.12352	-0.998245	-0.9753	-0.960322	-0.!
3	0.279653	-0.0194672	-0.113462	-0.99538	-0.967187	-0.978944	-0
4	0.279174	-0.0262006	-0.123283	-0.996091	-0.983403	-0.990675	-0.!

5 rows × 561 columns



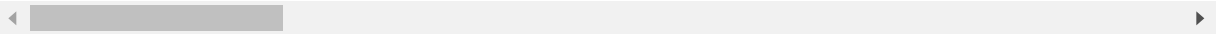
```
In [12]: header = X_train.iloc[0]
X_train = X_train.rename(columns = header )
X_train = X_train.drop([0])
```

In [13]: `X_train.head()`

Out[13]:

	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz
1	0.288585	-0.0202942	-0.132905	-0.995279	-0.983111	-0.913526
2	0.278419	-0.0164106	-0.12352	-0.998245	-0.9753	-0.960322
3	0.279653	-0.0194672	-0.113462	-0.99538	-0.967187	-0.978944
4	0.279174	-0.0262006	-0.123283	-0.996091	-0.983403	-0.990675
5	0.276629	-0.0165697	-0.115362	-0.998139	-0.980817	-0.990482

5 rows × 561 columns



In [14]: `X_train.shape`

Out[14]: (7352, 561)

lets import X_test and merge with features file

In [15]: `X_test = pd.read_fwf('X_test.txt', sep=" ", header=None)`

In [16]: `X_test = pd.concat([A, X_test], ignore_index=True, sort=False, axis=0)`

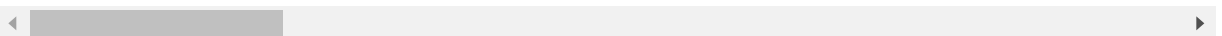
In [17]: `header = X_test.iloc[0]`
`X_test = X_test.rename(columns = header)`
`X_test = X_test.drop([0])`

In [18]: `X_test.head()`

Out[18]:

	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz
1	0.257178	-0.0232852	-0.0146538	-0.938404	-0.920091	-0.667683
2	0.286027	-0.0131634	-0.119083	-0.975415	-0.967458	-0.944958
3	0.275485	-0.0260504	-0.118152	-0.993819	-0.969926	-0.962748
4	0.270298	-0.0326139	-0.11752	-0.994743	-0.973268	-0.967091
5	0.274833	-0.0278478	-0.129527	-0.993852	-0.967445	-0.978295

5 rows × 561 columns



In [19]: `X_test.shape`

Out[19]: (2947, 561)

lets import y train and test

```
In [20]: y_train= pd.read_fwf('y_train.txt', sep=" ", header= None)  
y_train.head()
```

```
Out[20]:  
      0  
-----  
0  5  
1  5  
2  5  
3  5  
4  5
```

```
In [21]: y_train.shape
```

```
Out[21]: (7352, 1)
```

```
In [22]: y_train.nunique()
```

```
Out[22]: 0      6  
dtype: int64
```

```
In [23]: y_test = pd.read_fwf('y_test.txt', sep=" ", header=None)  
y_test.head()
```

```
Out[23]:  
      0  
-----  
0  5  
1  5  
2  5  
3  5  
4  5
```

```
In [24]: y_test.nunique()
```

```
Out[24]: 0      6  
dtype: int64
```

```
In [25]: P= pd.concat([y_train, y_test], ignore_index=True, sort=False, axis=0)
P.head()
```

```
Out[25]:
```

	0
0	5
1	5
2	5
3	5
4	5

```
In [26]: len(P)
```

```
Out[26]: 10299
```

lets import Activity labels

```
In [27]: activity_labels = pd.read_fwf('activity_labels.txt', sep=" ", header=None)
activity_labels.head(6)
```

```
Out[27]:
```

	0	1
0	1	WALKING
1	2	WALKING_UPSTAIRS
2	3	WALKING_DOWNSTAIRS
3	4	SITTING
4	5	STANDING
5	6	LAYING

lets merge the two table (y and activity labels)

```
In [28]: P = pd.merge(P,
                        activity_labels,
                        left_on =0,
                        right_on = 0,
                        how = 'left')
```

```
In [29]: P = P.drop(0, axis =1)
P.head()
```

Out[29]:

	1
0	STANDING
1	STANDING
2	STANDING
3	STANDING
4	STANDING

lets import Subject (train and test)

```
In [30]: Subject_test = pd.read_fwf('Subject_test.txt', sep=" ", header=None)
Subject_test.head()
```

Out[30]:

	0
0	2
1	2
2	2
3	2
4	2

```
In [31]: Subject_test.shape
```

Out[31]: (2947, 1)

```
In [32]: Subject_train = pd.read_fwf('Subject_train.txt', sep=" ", header=None)
Subject_train.head()
```

Out[32]:

	0
0	1
1	1
2	1
3	1
4	1

```
In [33]: Q= pd.concat([Subject_train, Subject_test], ignore_index=True, sort=False, axis=0)
Q.head()
```

```
Out[33]:
```

	0
0	1
1	1
2	1
3	1
4	1

```
In [34]: len(Q)
```

```
Out[34]: 10299
```

Lets merge P and Q

```
In [35]: R= pd.concat([P, Q], sort=False, axis=1, ignore_index=True)
R.head()
```

```
Out[35]:
```

	0	1
0	STANDING	1
1	STANDING	1
2	STANDING	1
3	STANDING	1
4	STANDING	1

```
In [36]: R.shape
```

```
Out[36]: (10299, 2)
```

Here,

0-index is for: P --> y(train and test)

1- index is for: Q --> Subject(train and test)

lets rename this

```
In [37]: R.rename(columns={0 : 'activity_y',
                           1: 'Subject'}, inplace =True)
R.head()
```

Out[37]:

	activity_y	Subject
0	STANDING	1
1	STANDING	1
2	STANDING	1
3	STANDING	1
4	STANDING	1

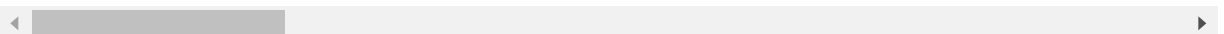
lets merge X_train and X_test

```
In [38]: myfile1 = pd.concat([X_train, X_test], ignore_index=True, sort=False, axis=0)
myfile1.head()
```

Out[38]:

	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz
0	0.288585	-0.0202942	-0.132905	-0.995279	-0.983111	-0.913526
1	0.278419	-0.0164106	-0.12352	-0.998245	-0.9753	-0.960322
2	0.279653	-0.0194672	-0.113462	-0.99538	-0.967187	-0.978944
3	0.279174	-0.0262006	-0.123283	-0.996091	-0.983403	-0.990675
4	0.276629	-0.0165697	-0.115362	-0.998139	-0.980817	-0.990482

5 rows × 561 columns



```
In [39]: myfile1.shape
```

Out[39]: (10299, 561)

```
In [40]: B =[]
for i in myfile1:
    colus = re.findall("[mM]ean|[Ss]td", i)
    if len(colus) !=0:
        B.append(i)

#[mean\W]
```

```
In [41]: myfile1 = myfile1[B]
```

```
In [42]: myfile1.shape
```

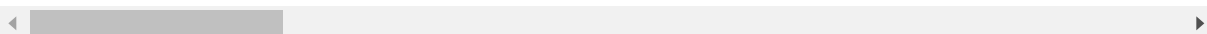
Out[42]: (10299, 86)

In [43]: myfile1.head()

Out[43]:

	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz
0	0.288585	-0.0202942	-0.132905	-0.995279	-0.983111	-0.913526
1	0.278419	-0.0164106	-0.12352	-0.998245	-0.9753	-0.960322
2	0.279653	-0.0194672	-0.113462	-0.99538	-0.967187	-0.978944
3	0.279174	-0.0262006	-0.123283	-0.996091	-0.983403	-0.990675
4	0.276629	-0.0165697	-0.115362	-0.998139	-0.980817	-0.990482

5 rows × 86 columns



In [44]: myfile1.shape

Out[44]: (10299, 86)

In [45]: R.shape

Out[45]: (10299, 2)

In [46]: myfile1.reset_index(drop=True, inplace=True)
R.reset_index(drop=True, inplace=True)

In [47]: result = pd.concat([myfile1, R], ignore_index=False, axis=1)

In [48]: result.shape

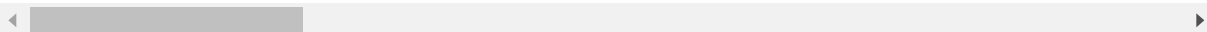
Out[48]: (10299, 88)

In [49]: result.head()

Out[49]:

	tbodyaccmeanx	tbodyaccmeany	tbodyaccmeanz	tbodyaccstdx	tbodyaccstdy	tbodyaccstdz
0	0.288585	-0.0202942	-0.132905	-0.995279	-0.983111	-0.913526
1	0.278419	-0.0164106	-0.12352	-0.998245	-0.9753	-0.960322
2	0.279653	-0.0194672	-0.113462	-0.99538	-0.967187	-0.978944
3	0.279174	-0.0262006	-0.123283	-0.996091	-0.983403	-0.990675
4	0.276629	-0.0165697	-0.115362	-0.998139	-0.980817	-0.990482

5 rows × 88 columns



In [50]: result.shape

Out[50]: (10299, 88)

```
In [51]: result.to_csv('myfile_final.csv')
```

```
In [ ]:
```

```
In [ ]:
```