

### Assignment 3:

The dataset is of energy consumption of several parts of Boston city taken in the interval of 5 minutes for year 2014. There are 72 csv files, each has a data Variables of the dataset is as follows:

1. Account - account number of each building
2. Date - date on which the meter reading is taken
3. Channel - channel of energy supply
4. Units - units of energy consumption
5. Time - interval of every 5 minutes

Setting the working directory and uploading the csv file from local system

```
setwd("C:/Users/Mushtaq/Downloads/ADS/COB Interval data 2014/CY2014 COB Interval data 1")  
BPDhead<-read.csv("COB-BPD.HEADQUARTERS.2014.csv",header=T)
```

### Checking the missing values

```
summary(is.na(BPDhead))
```

##	Account	Date	Channel	Units
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X0.05	X0.10	X0.15	X0.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X0.25	X0.30	X0.35	X0.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1092	FALSE:1095
##	NA's :0	NA's :0	TRUE :3	NA's :0
##			NA's :0	
##	X0.45	X0.50	X0.55	X1.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0

```

##
##      X1.05      X1.10      X1.15      X1.20
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X1.25      X1.30      X1.35      X1.40
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1092    FALSE:1092    FALSE:1095
## NA's :0       TRUE :3       TRUE :3       NA's :0
##              NA's :0       NA's :0
##      X1.45      X1.50      X1.55      X2.00
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X2.05      X2.10      X2.15      X2.20
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1092    FALSE:1092    FALSE:1092    FALSE:1092
## TRUE :3       TRUE :3       TRUE :3       TRUE :3
## NA's :0       NA's :0       NA's :0       NA's :0
##      X2.25      X2.30      X2.35      X2.40
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1092    FALSE:1092    FALSE:1092    FALSE:1092
## TRUE :3       TRUE :3       TRUE :3       TRUE :3
## NA's :0       NA's :0       NA's :0       NA's :0
##      X2.45      X2.50      X2.55      X3.00
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1092    FALSE:1092    FALSE:1092    FALSE:1092
## TRUE :3       TRUE :3       TRUE :3       TRUE :3
## NA's :0       NA's :0       NA's :0       NA's :0
##      X3.05      X3.10      X3.15      X3.20
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095

```

##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X3.25	X3.30	X3.35	X3.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X3.45	X3.50	X3.55	X4.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X4.05	X4.10	X4.15	X4.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X4.25	X4.30	X4.35	X4.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X4.45	X4.50	X4.55	X5.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X5.05	X5.10	X5.15	X5.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X5.25	X5.30	X5.35	X5.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical

##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X5.45	X5.50	X5.55	X6.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X6.05	X6.10	X6.15	X6.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X6.25	X6.30	X6.35	X6.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X6.45	X6.50	X6.55	X7.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X7.05	X7.10	X7.15	X7.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X7.25	X7.30	X7.35	X7.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X7.45	X7.50	X7.55	X8.00

##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X8.05	X8.10	X8.15	X8.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X8.25	X8.30	X8.35	X8.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X8.45	X8.50	X8.55	X9.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X9.05	X9.10	X9.15	X9.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X9.25	X9.30	X9.35	X9.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X9.45	X9.50	X9.55	X10.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				

##	X10.05	X10.10	X10.15	X10.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X10.25	X10.30	X10.35	X10.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X10.45	X10.50	X10.55	X11.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X11.05	X11.10	X11.15	X11.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X11.25	X11.30	X11.35	X11.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X11.45	X11.50	X11.55	X12.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X12.05	X12.10	X12.15	X12.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0

```

##
##      X12.25      X12.30      X12.35      X12.40
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X12.45      X12.50      X12.55      X13.00
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X13.05      X13.10      X13.15      X13.20
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X13.25      X13.30      X13.35      X13.40
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X13.45      X13.50      X13.55      X14.00
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X14.05      X14.10      X14.15      X14.20
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095
## NA's :0       NA's :0       NA's :0       NA's :0
##
##      X14.25      X14.30      X14.35      X14.40
## Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095    FALSE:1095    FALSE:1095    FALSE:1095

```

##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X14.45	X14.50	X14.55	X15.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X15.05	X15.10	X15.15	X15.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X15.25	X15.30	X15.35	X15.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X15.45	X15.50	X15.55	X16.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X16.05	X16.10	X16.15	X16.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X16.25	X16.30	X16.35	X16.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X16.45	X16.50	X16.55	X17.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical



##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X17.05	X17.10	X17.15	X17.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X17.25	X17.30	X17.35	X17.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X17.45	X17.50	X17.55	X18.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X18.05	X18.10	X18.15	X18.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X18.25	X18.30	X18.35	X18.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X18.45	X18.50	X18.55	X19.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X19.05	X19.10	X19.15	X19.20

##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X19.25	X19.30	X19.35	X19.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X19.45	X19.50	X19.55	X20.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X20.05	X20.10	X20.15	X20.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X20.25	X20.30	X20.35	X20.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X20.45	X20.50	X20.55	X21.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X21.05	X21.10	X21.15	X21.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				

##	X21.25	X21.30	X21.35	X21.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X21.45	X21.50	X21.55	X22.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X22.05	X22.10	X22.15	X22.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X22.25	X22.30	X22.35	X22.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X22.45	X22.50	X22.55	X23.00
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X23.05	X23.10	X23.15	X23.20
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0
##				
##	X23.25	X23.30	X23.35	X23.40
##	Mode :logical	Mode :logical	Mode :logical	Mode :logical
##	FALSE:1095	FALSE:1095	FALSE:1095	FALSE:1095
##	NA's :0	NA's :0	NA's :0	NA's :0

```
##
##      X23.45      X23.50      X23.55      X24.00
##  Mode :logical  Mode :logical  Mode :logical  Mode :logical
## FALSE:1095     FALSE:1095     FALSE:1095     FALSE:1095
## NA's :0        NA's :0        NA's :0        NA's :0
##
```

Taking only the time columns and the transpose of it

```
BPDheadTime<-BPDhead[,c(5:292)]
BPDheadTranspose<-t(BPDheadTime)
```

Transforming the unit values for every 5 minutes.

```
a<-0

for(i in seq(from=1,to=ncol(BPDheadTranspose),by=3)){
  b<-i
  c<-i+2
  a<-rbind(a,BPDheadTranspose[,b:c])
}

a=a[-1,]
head(a)

##      [,1]      [,2]      [,3]
## X0.05 50.52 0.939320 18.45
## X0.10 50.64 0.942246 18.00
## X0.15 47.64 0.936813 17.79
## X0.20 46.92 0.937213 17.46
## X0.25 50.79 0.945337 17.52
## X0.30 50.91 0.944542 17.70
```

Assigning the names to all 4 columns.

```
write.csv(a,file="mushtaq36.csv")
xyz<-read.csv("mushtaq36.csv",header=FALSE)
head(xyz)
```

```
##      V1      V2      V3      V4
## 1      V1      V2      V3
## 2 X0.05      50.52  0.93932 18.449999
## 3 X0.10 50.639999 0.942246      18
## 4 X0.15 47.639999 0.936813 17.789999
## 5 X0.20 46.919998 0.937213 17.459999
## 6 X0.25 50.789997 0.945337      17.52

xyz=xyz[-1,]
colnames(xyz)<-c("Time", "KWh", "powerfactor", "KVARh")
head(xyz)
```

##	Time	KWh	powerfactor	KVARh
## 2	X0.05	50.52	0.93932	18.449999
## 3	X0.10	50.639999	0.942246	18
## 4	X0.15	47.639999	0.936813	17.789999
## 5	X0.20	46.919998	0.937213	17.459999
## 6	X0.25	50.789997	0.945337	17.52
## 7	X0.30	50.91	0.944542	17.699999

Splitting the time column values in two other columns of hour and minute

```
library(stringr)
out<-str_split_fixed(xyz$Time, ".", 2)
head(out)
```

##	[,1]	[,2]
## [1,]	"	"0.05"
## [2,]	"	"0.10"
## [3,]	"	"0.15"
## [4,]	"	"0.20"
## [5,]	"	"0.25"
## [6,]	"	"0.30"

```
head(xyz)
```

##	Time	KWh	powerfactor	KVARh
## 2	X0.05	50.52	0.93932	18.449999
## 3	X0.10	50.639999	0.942246	18
## 4	X0.15	47.639999	0.936813	17.789999

```
## 5 X0.20 46.919998 0.937213 17.459999
## 6 X0.25 50.789997 0.945337 17.52
## 7 X0.30 50.91 0.944542 17.699999
```

```
xyz<-cbind(xyz,out)
```

```
head(xyz)
```

```
##      Time      KWh powerfactor      KVARh 1      2
## 2 X0.05      50.52      0.93932 18.449999 0.05
## 3 X0.10 50.639999      0.942246      18 0.10
## 4 X0.15 47.639999      0.936813 17.789999 0.15
## 5 X0.20 46.919998      0.937213 17.459999 0.20
## 6 X0.25 50.789997      0.945337      17.52 0.25
## 7 X0.30      50.91      0.944542 17.699999 0.30
```

```
xyz<-xyz[, -5]
```

```
head(xyz)
```

```
##      Time      KWh powerfactor      KVARh      2
## 2 X0.05      50.52      0.93932 18.449999 0.05
## 3 X0.10 50.639999      0.942246      18 0.10
## 4 X0.15 47.639999      0.936813 17.789999 0.15
## 5 X0.20 46.919998      0.937213 17.459999 0.20
## 6 X0.25 50.789997      0.945337      17.52 0.25
## 7 X0.30      50.91      0.944542 17.699999 0.30
```

```
colnames(xyz)<-c("Time", "KWh", "powerfactor", "KVARh", "time")
```

```
xyz$time<-as.character(xyz$time)
```

```
library(reshape)
```

```
xyz=transform(xyz,time=colsplit(time,split="\\. ",names=c('hour', 'minute')))
```

```
head(xyz)
```

```
##      Time      KWh powerfactor      KVARh time.hour time.minute
## 2 X0.05      50.52      0.93932 18.449999      0      5
## 3 X0.10 50.639999      0.942246      18      0      10
## 4 X0.15 47.639999      0.936813 17.789999      0      15
## 5 X0.20 46.919998      0.937213 17.459999      0      20
## 6 X0.25 50.789997      0.945337      17.52      0      25
## 7 X0.30      50.91      0.944542 17.699999      0      30
```

```
h<-xyz[,c(5)]
hour<-h[,c(1)]
minute<-h[,c(2)]
xyz<-xyz[, -5]
xyz<-cbind(xyz, hour, minute)
head(xyz)
```

##	Time	KWh	powerfactor	KVARh	hour	minute
## 2	X0.05	50.52	0.93932	18.449999	0	5
## 3	X0.10	50.639999	0.942246	18	0	10
## 4	X0.15	47.639999	0.936813	17.789999	0	15
## 5	X0.20	46.919998	0.937213	17.459999	0	20
## 6	X0.25	50.789997	0.945337	17.52	0	25
## 7	X0.30	50.91	0.944542	17.699999	0	30

Now coming back to main file, taking only two columns- Account and date

```
BPDheadAccount<-BPDhead[,c(1:2)]
head(BPDheadAccount)
```

##	Account	Date
## 1	26429921005	1/1/2014
## 2	26429921005	1/1/2014
## 3	26429921005	1/1/2014
## 4	26429921005	1/2/2014
## 5	26429921005	1/2/2014
## 6	26429921005	1/2/2014

Making the set of 3 columns and changing it to 1 row for each 5 day interval of 1 day

```
library(splitstackshape)
## Warning: package 'splitstackshape' was built under R version 3.2.4
## Loading required package: data.table
## Warning: package 'data.table' was built under R version 3.2.4
##
## Attaching package: 'data.table'
## The following object is masked from 'package:reshape':
##
```

```
##      melt
a<-0
for(i in seq(from=1,to=nrow(BPDheadAccount),by=3))
{
  b<-i
  a<-rbind(a,expandRows(BPDheadAccount[c(b),],288,count.is.col = FALSE))
}

## Warning in `[<-.factor`(`*tmp*`, ri, value = structure(c(NA, 1L, 1L, 1L, :
## invalid factor level, NA generated

head(a)
##      Account      Date
## 1          0    <NA>
## 11 26429921005 1/1/2014
## 1.1 26429921005 1/1/2014
## 1.2 26429921005 1/1/2014
## 1.3 26429921005 1/1/2014
## 1.4 26429921005 1/1/2014

a=a[-1,]
head(a)
##      Account      Date
## 11 26429921005 1/1/2014
## 1.1 26429921005 1/1/2014
## 1.2 26429921005 1/1/2014
## 1.3 26429921005 1/1/2014
## 1.4 26429921005 1/1/2014
## 1.5 26429921005 1/1/2014
```

Splitting the date column in month, day and year.

```
datetxt<-a[,c(2)]
datetxt<-as.character(datetxt)
datetxt<-as.Date(datetxt,"%m/%d/%Y")

class(datetxt)
## [1] "Date"
```



```
df <- data.frame(month = as.numeric(format(datetxt, format = "%m")),
                 day = as.numeric(format(datetxt, format = "%d")),
                 year = as.numeric(format(datetxt, format = "%Y")))
```

```
head(df)
```

```
##   month day year
## 1     1   1 2014
## 2     1   1 2014
## 3     1   1 2014
## 4     1   1 2014
## 5     1   1 2014
## 6     1   1 2014
```

### Combining all the sections of a csv file and exporting it as a local file

```
finalBPDhead<-cbind(a,df)
```

```
finalBPDhead<-cbind(finalBPDhead,xyz)
```

```
head(finalBPDhead)
```

```
##           Account      Date month day year   Time      KWh powerfactor
## 11  26429921005 1/1/2014      1   1 2014 X0.05    50.52    0.93932
## 1.1 26429921005 1/1/2014      1   1 2014 X0.10  50.639999    0.942246
## 1.2 26429921005 1/1/2014      1   1 2014 X0.15  47.639999    0.936813
## 1.3 26429921005 1/1/2014      1   1 2014 X0.20  46.919998    0.937213
## 1.4 26429921005 1/1/2014      1   1 2014 X0.25  50.789997    0.945337
## 1.5 26429921005 1/1/2014      1   1 2014 X0.30    50.91    0.944542
```

```
##           KVARh hour minute
```

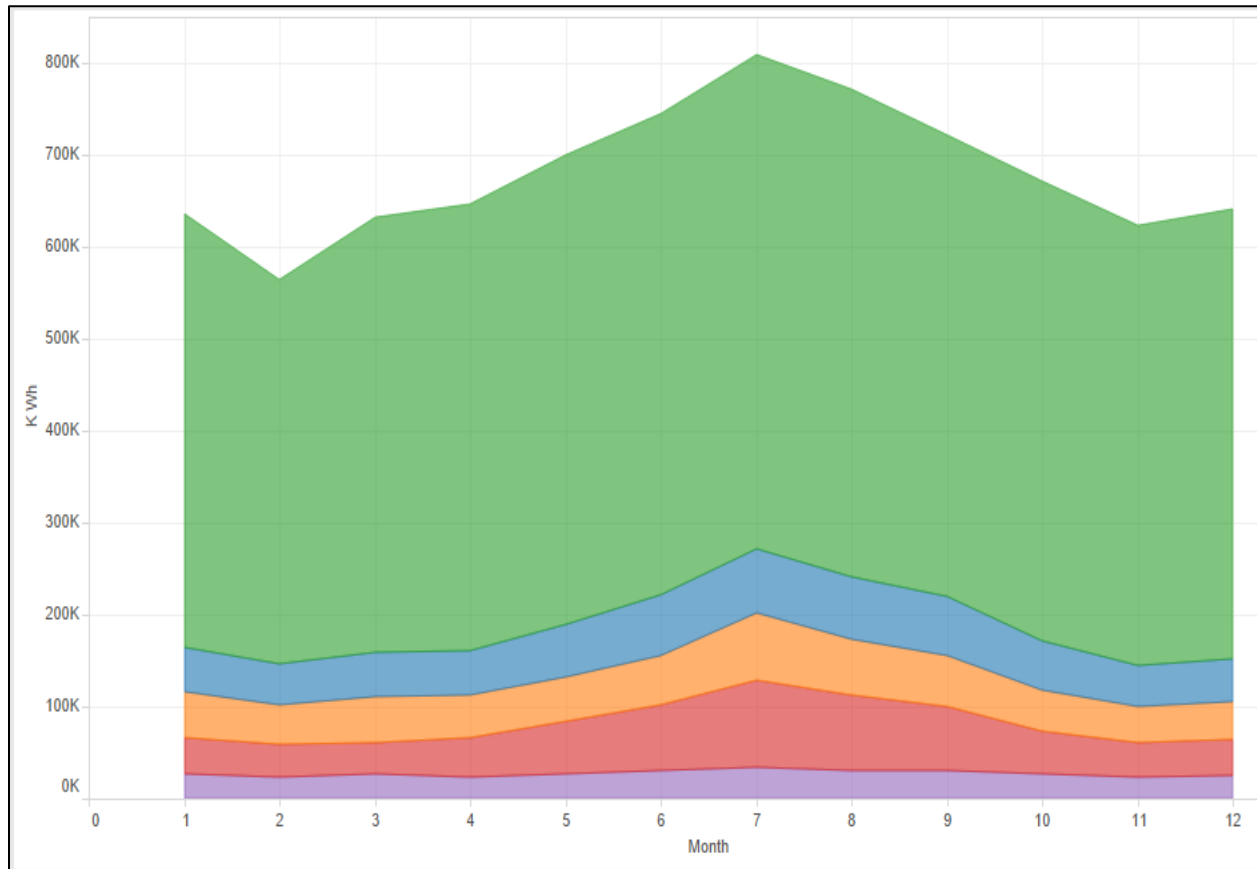
```
## 11  18.449999    0      5
## 1.1      18    0     10
## 1.2 17.789999    0     15
## 1.3 17.459999    0     20
## 1.4   17.52    0     25
## 1.5 17.699999    0     30
```

```
write.csv(finalBPDhead,file="finalBPLhead.csv")
```

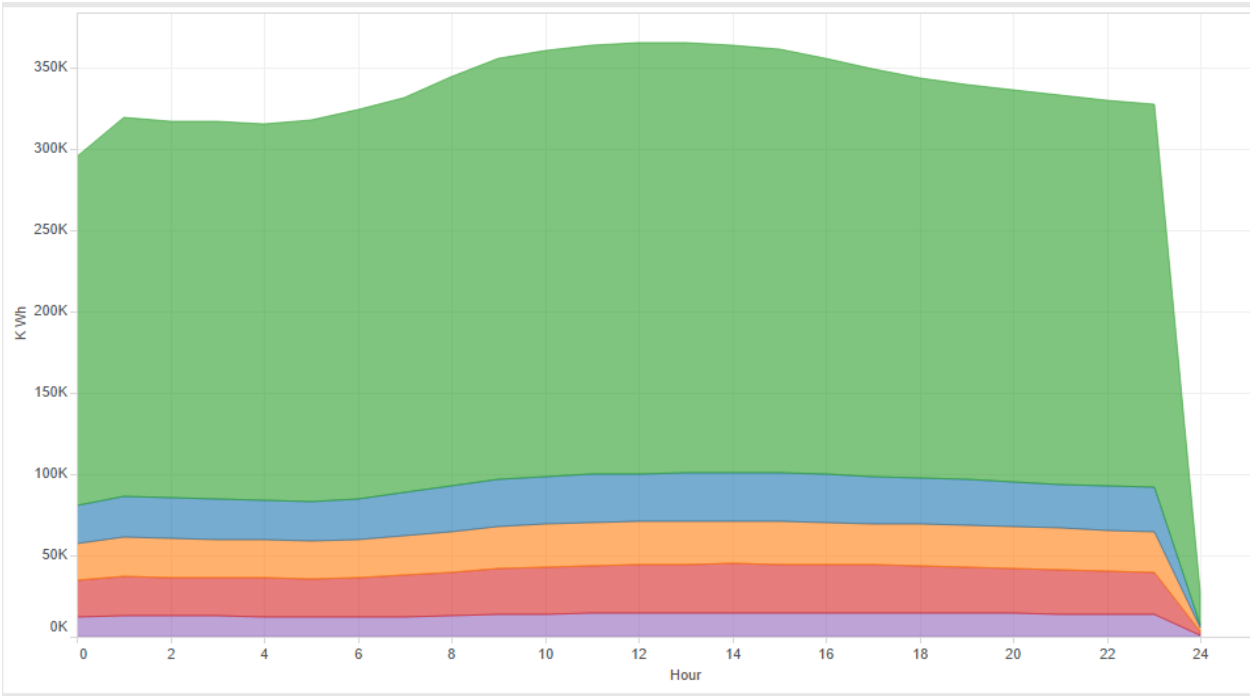
### Visualization and Data Exploratory analysis:

As explained above, we merged the data for Boston Police Department from different locations of Boston. The locations included are Head Quarters mentioned as BPD\_Head, District mentioned as BPD\_District, Dudley as BPD\_Dudley, Head Quarter Department as BPD\_HeadD and Station Landing as BPD\_Station. The all have the energy consumption data for the year 2014 and is split into yearly, monthly, daily and hourly and minutes wise.

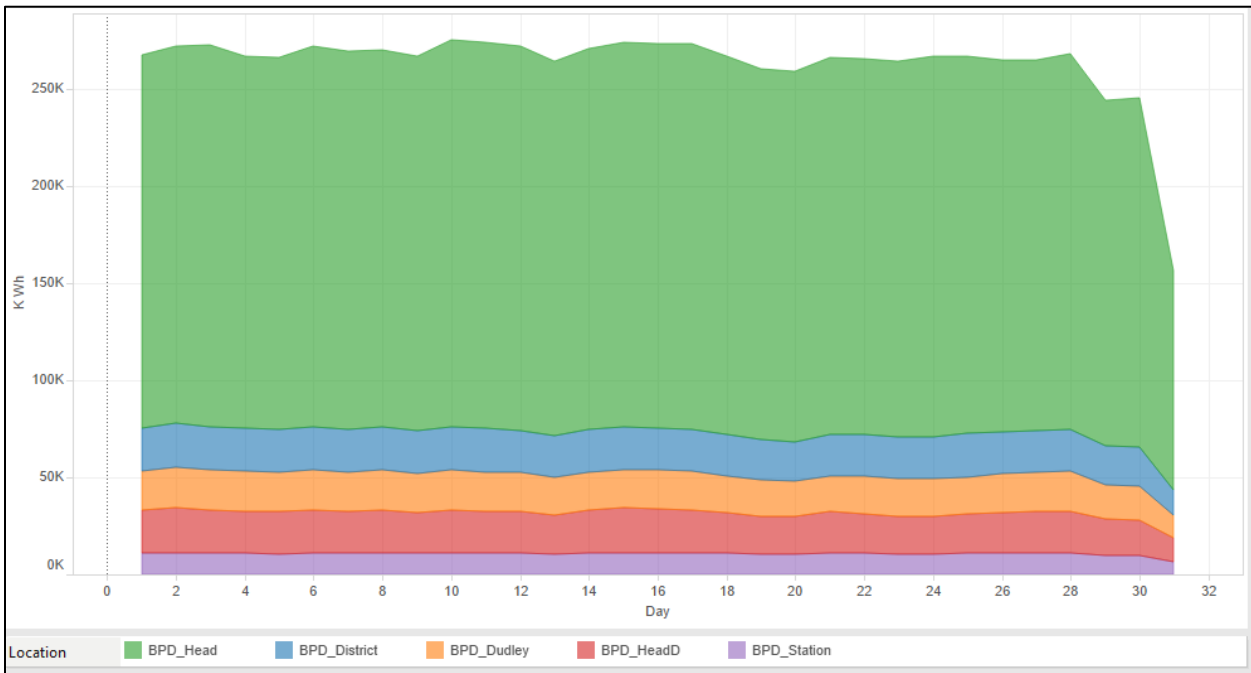
Power consumption of different places over the year:



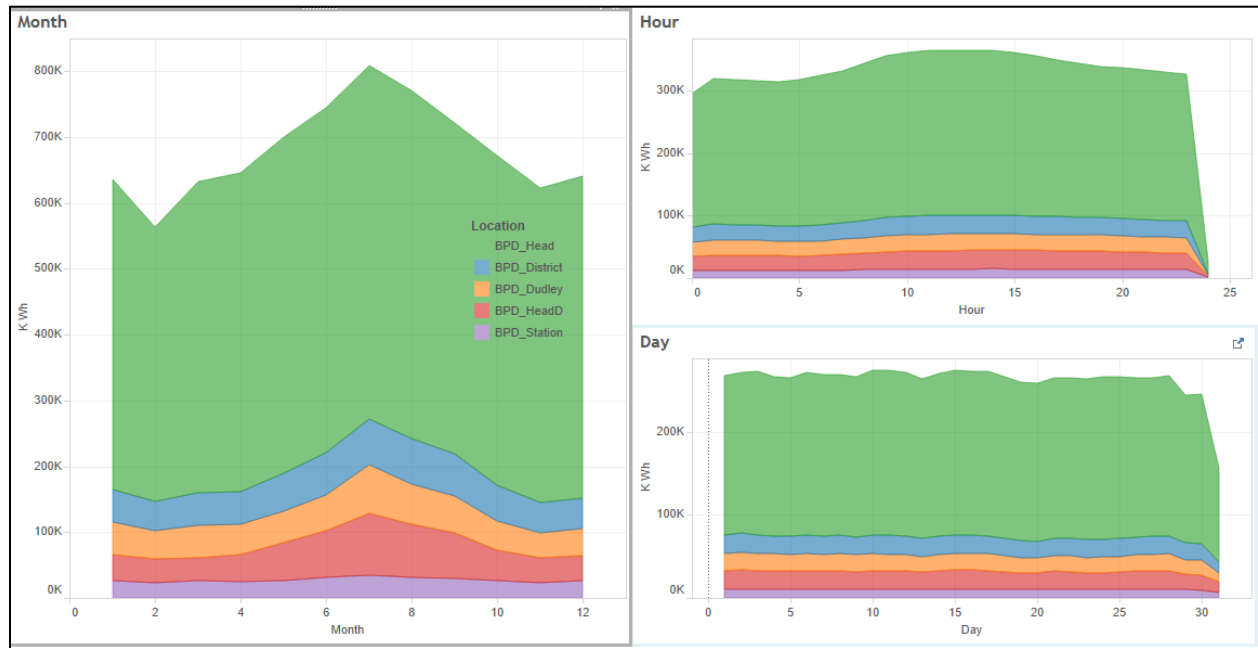
Power consumption of different places in hourly basis:



Average Power consumption of different places over the months:



## Dashboard:



We did a time-series analysis to calculate the trend of power consumption over the year. Like most time-series data, the best way to represent the data is by doing a line chart. Few observations made from the charts above are:

1. The highest power consumption for all the location was found to be in the 7<sup>th</sup> month of the year.
2. BPD\_Head has the highest power consumption all-round the year.
3. The hourly graph shows that power consumption goes up in the graph from 6<sup>th</sup> hour and starts to drop from the 18<sup>th</sup> hour.
4. Daily graph shows the trend where weekdays have more power consumption compared to the ones on weekends as seen in the graph.

Similar dashboards are made for all the other departments present in Boston and similar graphs has been plotted to visualize the power consumption over the year, month and day. This helps us predict the trend for the future and help the business learn more about which has the highest power consumption and which needs to further cut down on their power consumption.

## Building predictive model in Microsoft Azure:

### Boosted Tree Decision Algorithm:

**Importing dataset:** The Dataset contains energy consumption details of Boston Police Department at various locations. The fields are Account, Date, Month, Day, Year, Time, KWh, PowerFactor, KVARh, hour, minute and location. We get very granular information regarding the energy consumption with respect to 5 minute intervals. The location attribute specifies the location at which the BPD is present.

### Visualization:

BPDfinal > finalBPD2.csv > dataset

rows: 525600, columns: 14

view as:

Column 0	X	Account	Date	month	day	year	Time	KWh	powerfactor	KVARh	hour	minute	location
1	11	26434931007	2014-01-01T00:00:00	1	1	2014	X0.05	2.736	0.925935	1.116	0	5	BPD_S
2	1.1	26434931007	2014-01-01T00:00:00	1	1	2014	X0.10	2.736	0.919458	1.17	0	10	BPD_S
3	1.2	26434931007	2014-01-01T00:00:00	1	1	2014	X0.15	2.736	0.915049	1.206	0	15	BPD_S
4	1.3	26434931007	2014-01-01T00:00:00	1	1	2014	X0.20	2.736	0.923795	1.134	0	20	BPD_S
5	1.4	26434931007	2014-01-01T00:00:00	1	1	2014	X0.25	2.79	0.922194	1.17	0	25	BPD_S
6	1.5	26434931007	2014-01-01T00:00:00	1	1	2014	X0.30	2.79	0.926401	1.134	0	30	BPD_S
7	1.6	26434931007	2014-01-01T00:00:00	1	1	2014	X0.35	2.754	0.926796	1.116	0	35	BPD_S
8	1.7	26434931007	2014-01-01T00:00:00	1	1	2014	X0.40	2.646	0.925848	1.08	0	40	BPD_S
9	1.8	26434931007	2014-01-01T00:00:00	1	1	2014	X0.45	2.574	0.919809	1.098	0	45	BPD_S
10	1.9	26434931007	2014-01-01T00:00:00	1	1	2014	X0.50	2.574	0.915126	1.134	0	50	BPD_S
11	1.1	26434931007	2014-01-01T00:00:00	1	1	2014	X0.55	2.556	0.916453	1.116	0	55	BPD_S
			2014-01-										

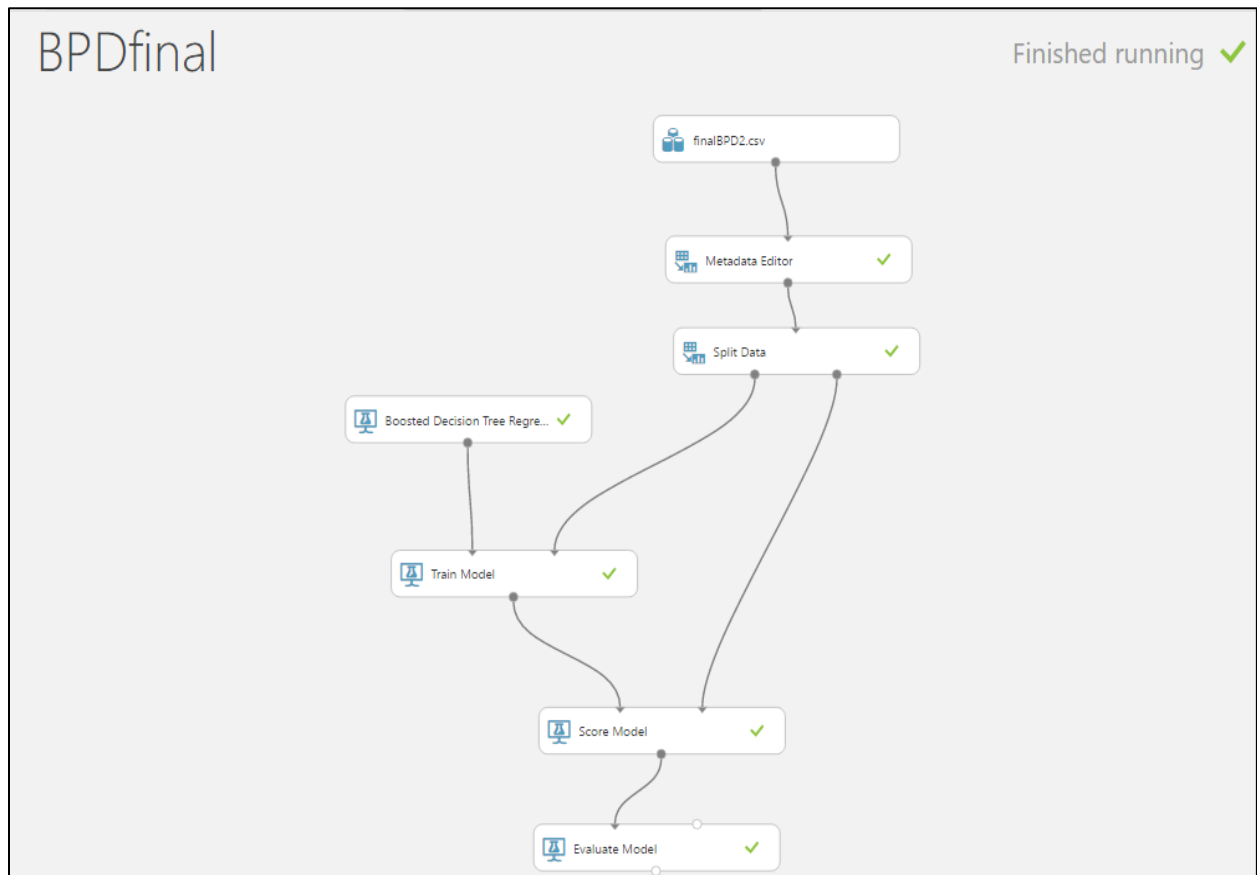
**Split Data:** The module was incorporated to divide the dataset into training dataset and validation dataset. Training dataset consisted 70% of the actual data.

**Boosted Decision Tree Algorithm:** The Boosted Decision Tree Algorithm was chosen to train the dataset.

**Train Module:** This module is to train the dataset. The training data along with boosted decision tree algorithm trains the training data to build a model.

**Score Model:** The model built with Boosted Decision Tree Algorithm should be tested on new data. Score module tests the built model on validation data which was obtained from split data. Score Module helps understand correctness of the built model.

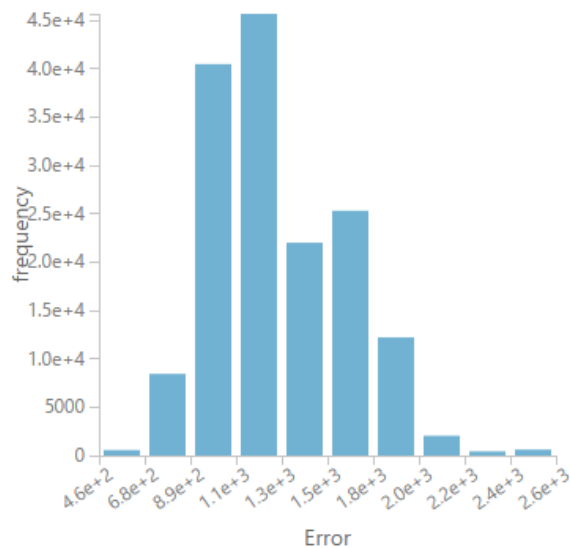
**Evaluate Model:** Evaluate Model helps understand the RMS Error, and other error parameters like coefficient of determination.



### Metrics

Mean Absolute Error	1306.434861
Root Mean Squared Error	1344.536808
Relative Absolute Error	9.642065
Relative Squared Error	69.345879
Coefficient of Determination	-68.345879

### Error Histogram

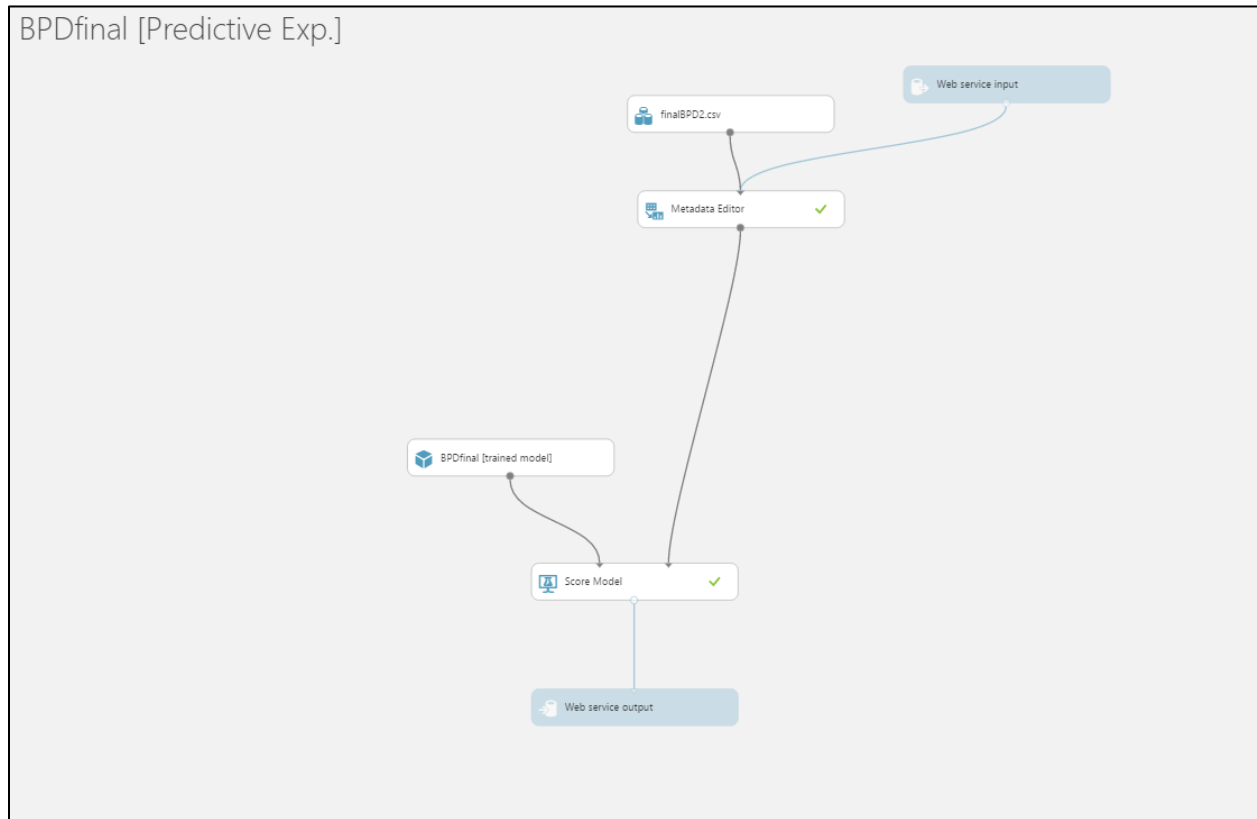


### Prediction:

While building the training model, parameter KWh is set as the output variable. The model predicts the value of KWh for any parameter entered. The model should predict the value of KWh for parameters entered by user consuming the deployed web service.

### Set up web service:

Setting up the web service inserts web service input and web service output modules thus changing the structure of the model.



### Deploying web service:

Deploying web service generates API key which is used in the Authorization Request Header to call the model from form. It generates URI so that it can be entered in AJAX call.



Microsoft Azure Machine Learning

bpdfinal [predictive exp.]

DASHBOARD CONFIGURATION

General

Published experiment

View snapshot View latest

Description

No description provided for this web service.

API key

Default Endpoint

API HELP PAGE TEST APPS LAST UPDATED

REQUEST/RESPONSE	TEST	APPS	LAST UPDATED
	Test	Excel 2013 or later Excel 2010 or earlier workbook	4/17/2016 8:43:02 PM
BATCH EXECUTION		Excel 2013 or later workbook	4/17/2016 8:43:02 PM

+ "BPDFinal [Predictive Exp.] test returned [{"0","0","26429921005","1/1/2015 12:00:00 AM","1","1","2014","5",null,"0.8","12","1","5","BPD\_HEAD","697.479431152344"}]...

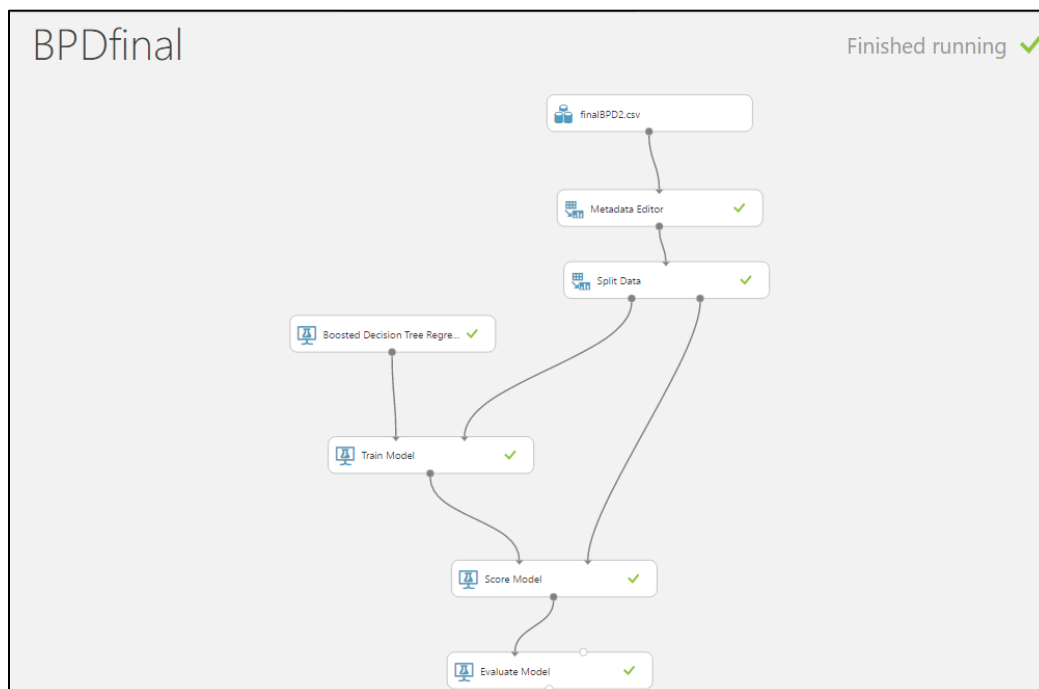
Result: [{"Results":{"output1":{"type":"table","value":{"ColumnNames":["Column 0","1","2","Account","Date","month","day","year","Time","KWh","powerfactor","KVAh","hour","minute","location","Scored Labels"],"ColumnTypes":["int32","Double","int64","DateTime","int32","int32","String","String","String","String","Double"],"Values":["0","0","26429921005","1/1/2015 12:00:00 AM","1","1","2014","5",null,"0.8","12","1","5","BPD\_HEAD","697.479431152344"]}}}]

NEW

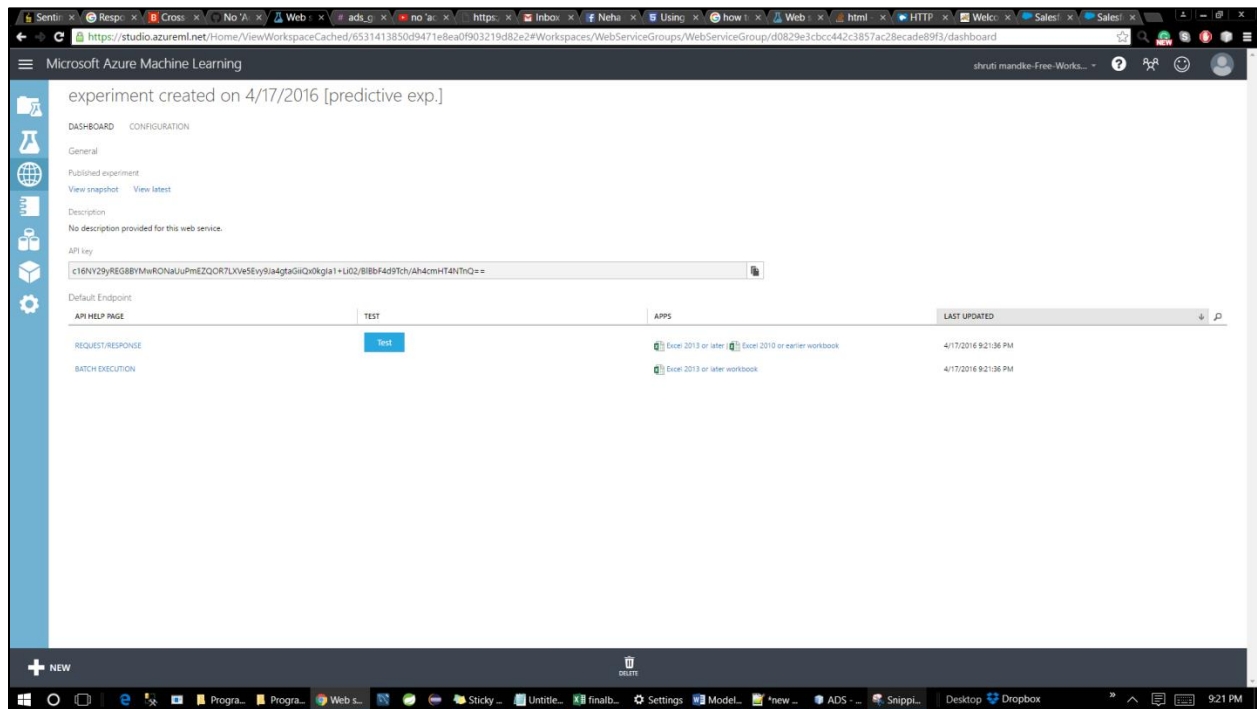
9:20 PM

Random Forest Regression:

Model:



API Key:



The dataset can be applied different prediction models to obtain a trained and scored model. The best model can be chosen based on Root Mean Square Error and other error parameters.

Links for Tableau Public for BPD and BPL:

- [https://public.tableau.com/views/BPD-Boston-Day/Day?:embed=y&:display\\_count=yes&:showTabs=y](https://public.tableau.com/views/BPD-Boston-Day/Day?:embed=y&:display_count=yes&:showTabs=y)
- [https://public.tableau.com/views/BPL-Boston-dashboard/Dashboard1?:embed=y&:display\\_count=yes&:showTabs=y](https://public.tableau.com/views/BPL-Boston-dashboard/Dashboard1?:embed=y&:display_count=yes&:showTabs=y)