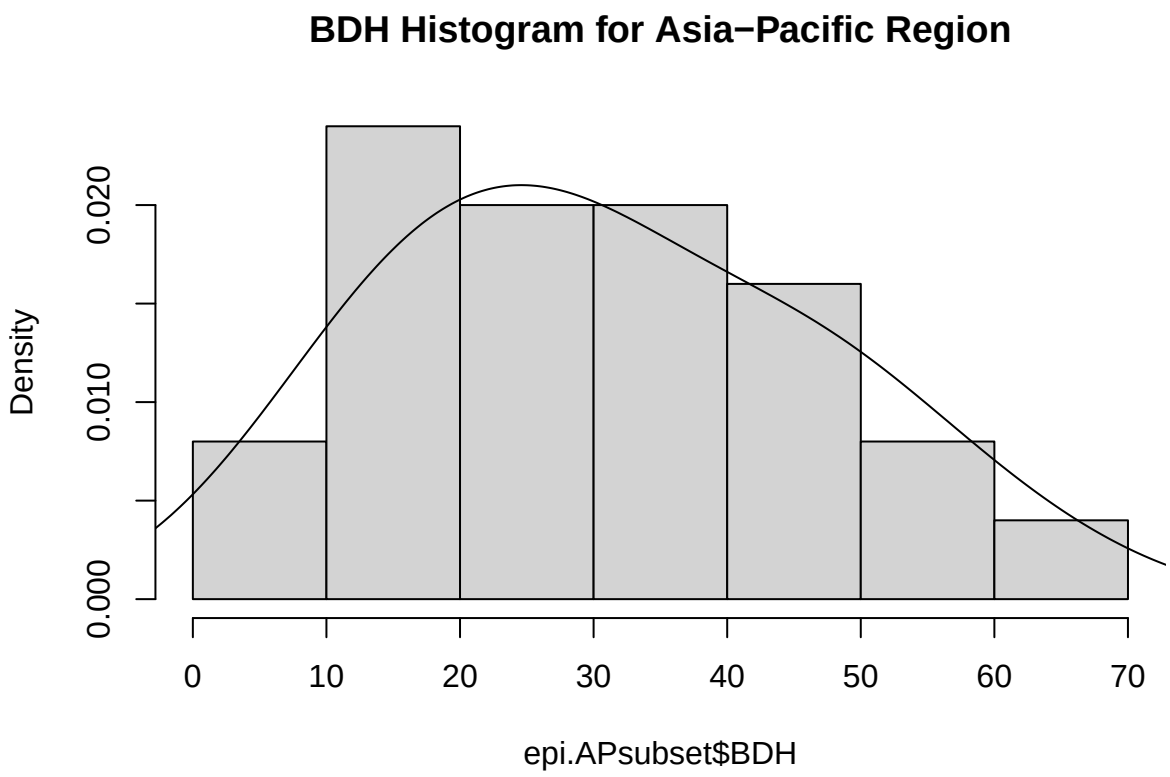
Lab3

Beatrice Dang

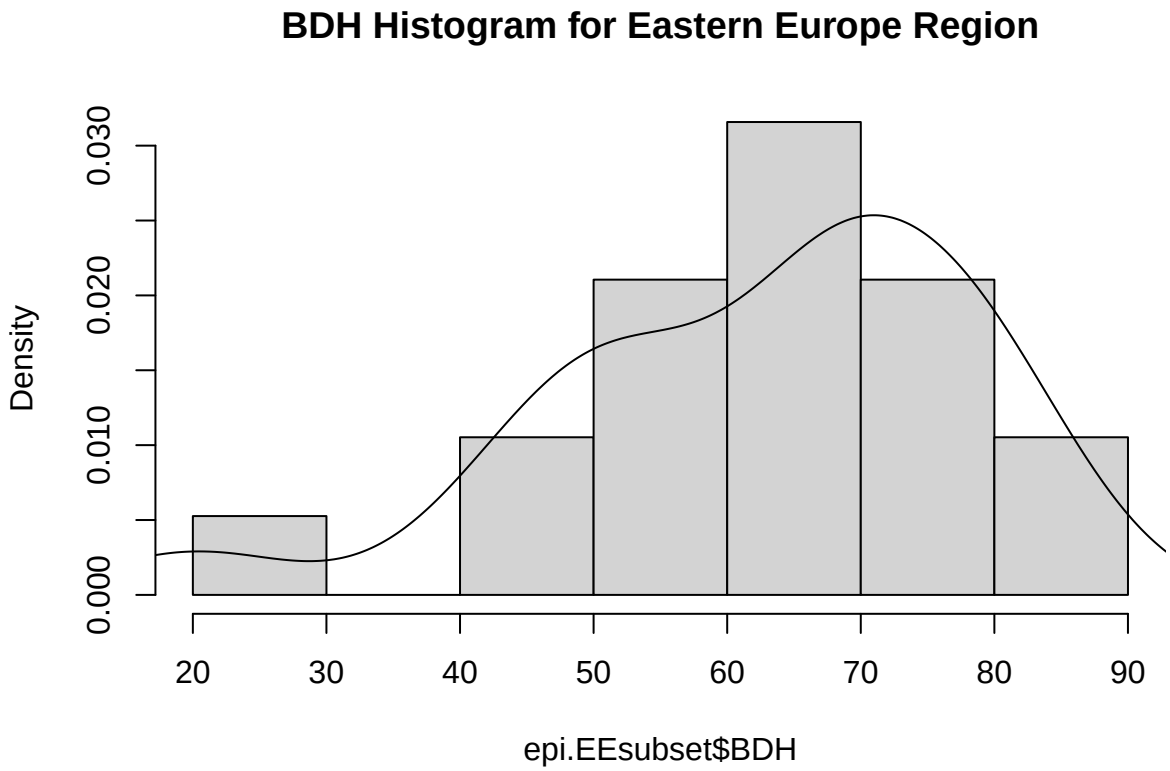
2024-10-04

PART 1: VARIABLE DISTIBUTIONS

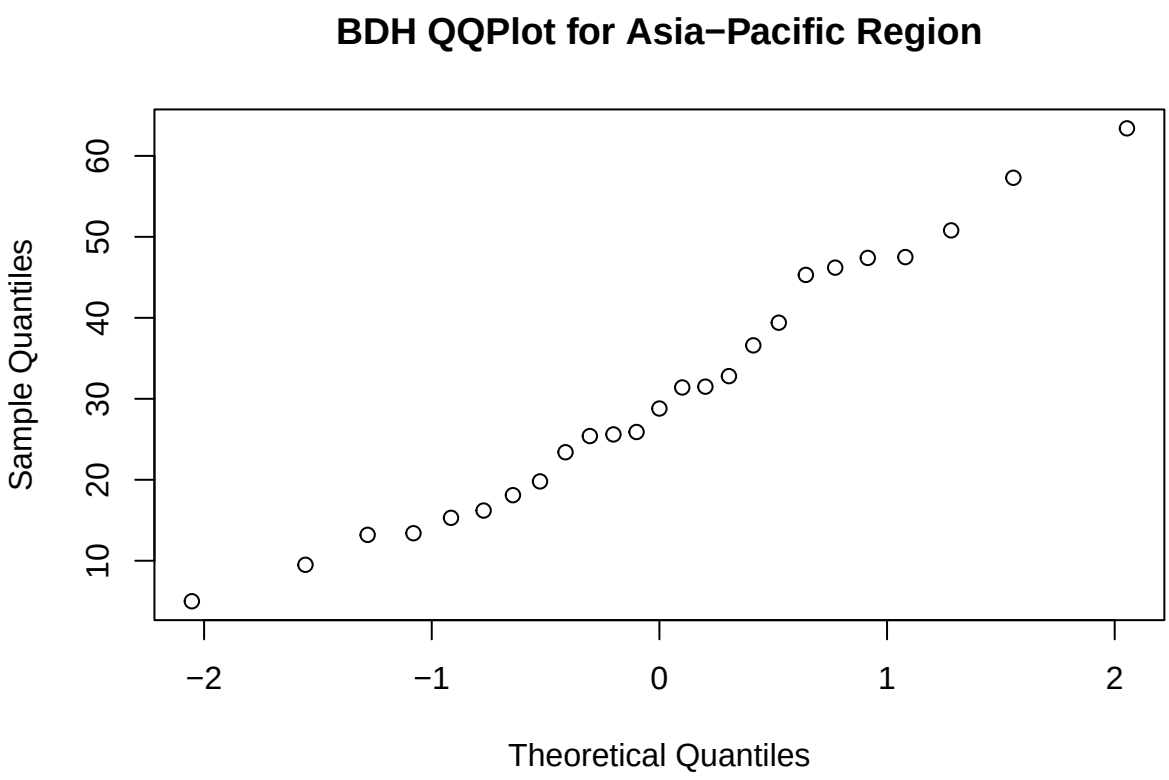
```
EPI_data <- read.csv("C:\\Users\\bmd\\Downloads\\epi2024results_DA_F24_lab03.csv", header=TRUE)

epi <- EPI_data
attach(epi)

##asian-pacific, eastern europe -- BDH
epi.APsubset <- epi[epi$region == 'Asia-Pacific', ]
epi.EEsubset <- epi[epi$region == 'Eastern Europe',]

hist(epi.APsubset$BDH, main='BDH Histogram for Asia-Pacific Region', freq = FALSE)
lines(density(epi.APsubset$BDH,na.rm=TRUE,bw="SJ"))
```



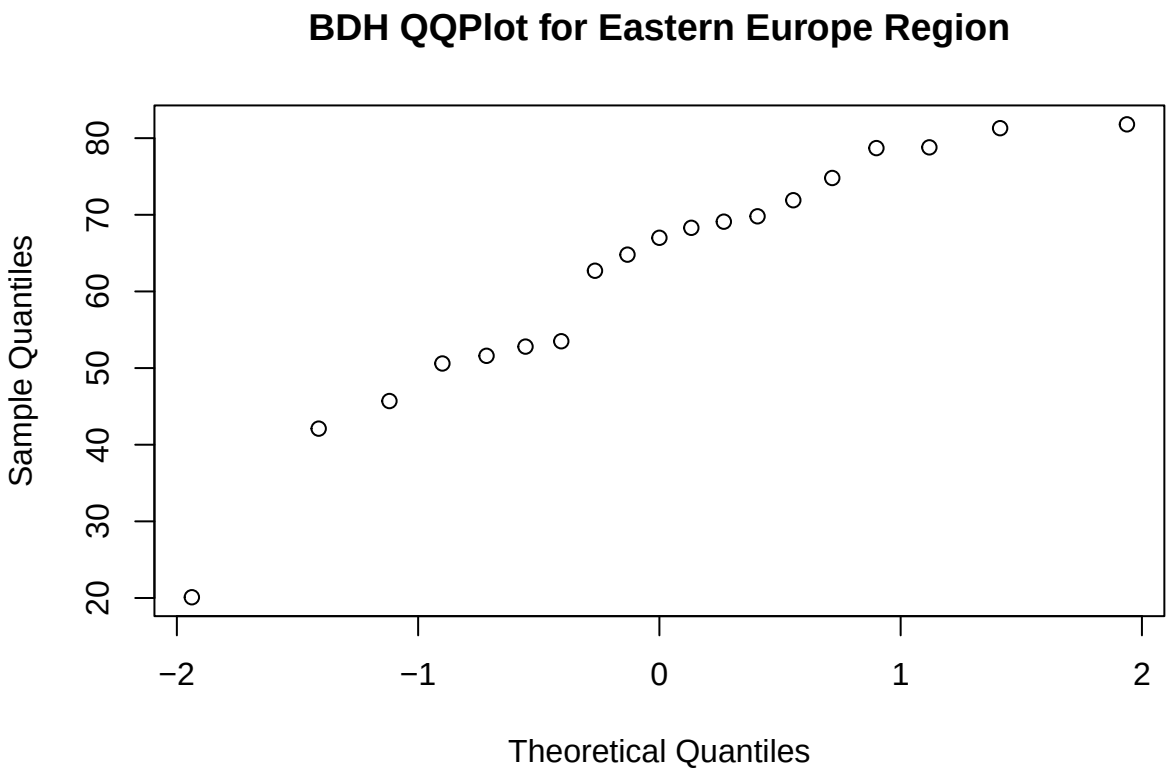
```r
hist(epi.EEsubset$BDH, main='BDH Histogram for Eastern Europe Region', freq = FALSE)
lines(density(epi.EEsubset$BDH,na.rm=TRUE,bw="SJ"))
```




```r
qqnorm(epi.APsubset$BDH, main='BDH QQPlot for Asia-Pacific Region')
```




```
qqnorm(epi.EEsubset$BDH, main='BDH QQPlot for Eastern Europe Region')
```



PART TWO: LINEAR MODELS

```
library(ggplot2)
lin.mod.epi <- lm(EPI~BDH+ECO+MKP+MHP+MPE, data=epi)
summary(lin.mod.epi)
```

```
##
## Call:
## lm(formula = EPI ~ BDH + ECO + MKP + MHP + MPE, data = epi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.344  -3.374  -0.437   3.280  13.413
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.462659   2.707696   1.648    0.102
## BDH         -0.379631   0.064737  -5.864 4.01e-08 ***
## ECO          1.213033   0.085455  14.195  < 2e-16 ***
## MKP         -0.016450   0.019082  -0.862    0.390
## MHP          0.021618   0.027973   0.773    0.441
## MPE         -0.004785   0.017685  -0.271    0.787
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.029 on 121 degrees of freedom
```

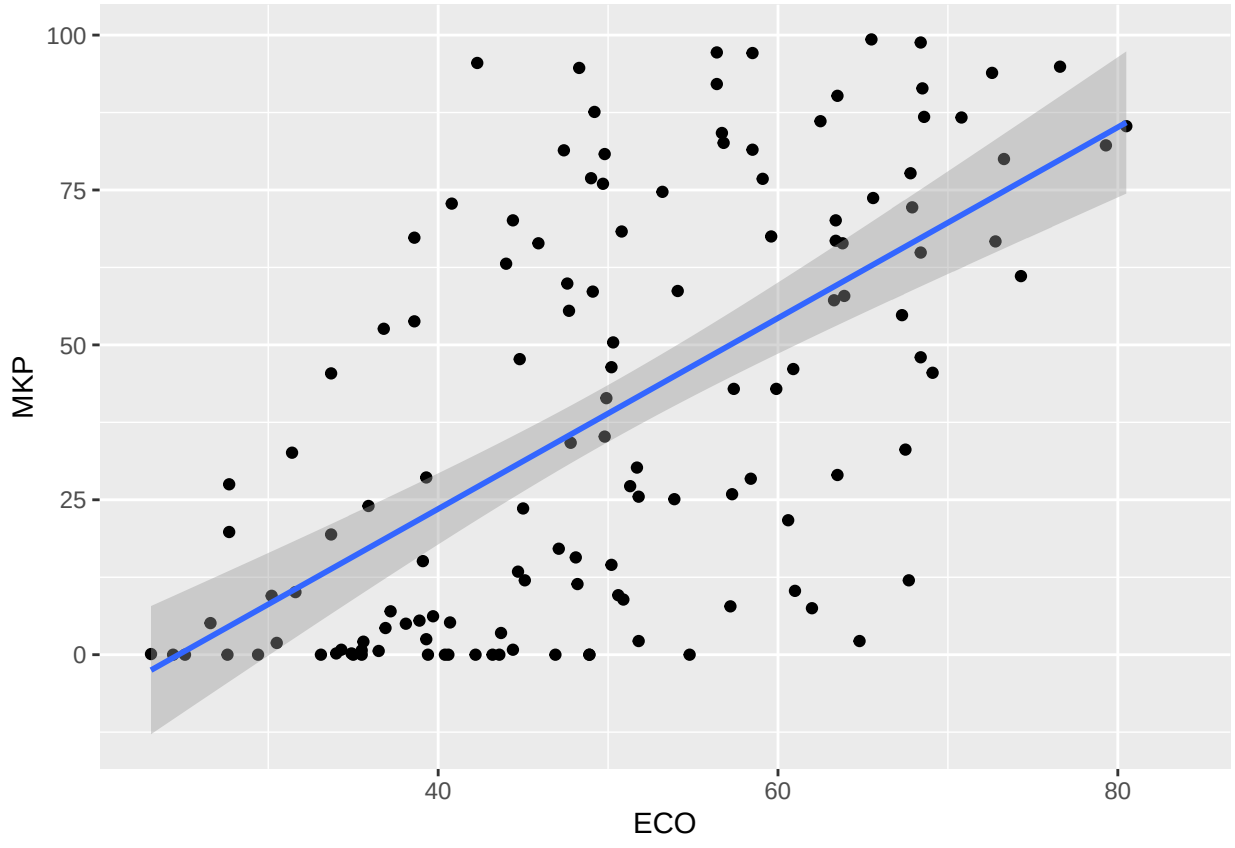
```
##   (53 observations deleted due to missingness)
## Multiple R-squared:  0.813,  Adjusted R-squared:  0.8053
## F-statistic: 105.2 on 5 and 121 DF,  p-value: < 2.2e-16
```

```
##from the summary, found that ECO variable has the smallest p-value, therefore most significantly infl

ggplot(epi, aes(x = ECO, y = MKP)) +
  geom_point() +
  stat_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 48 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 48 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
##linear model for Asia-Pacific region
lin.mod.APsubset <- lm(EPI~BDH+ECO+MKP+MHP+MPE, data=epi.APsubset)
summary(lin.mod.APsubset)
```

```
##
## Call:
```

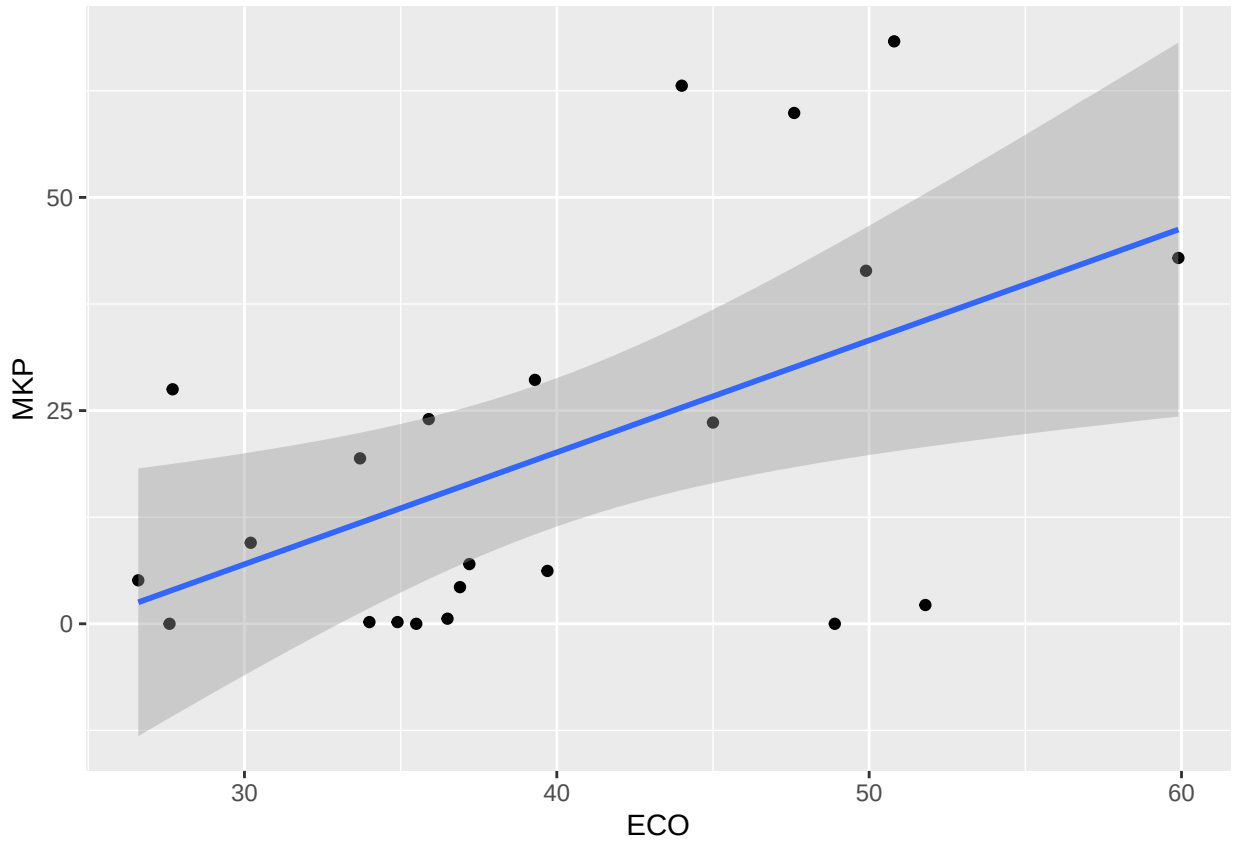
```
## lm(formula = EPI ~ BDH + ECO + MKP + MHP + MPE, data = epi.APsubset)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.8565 -1.7783 -0.1404  2.6285  6.0523
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.63065    6.84769  -0.238  0.81480
## BDH         -0.35425    0.11979  -2.957  0.00927 **
## ECO          1.21289    0.18997   6.385 9.04e-06 ***
## MKP         -0.06437    0.06507  -0.989  0.33726
## MHP          0.12553    0.10991   1.142  0.27022
## MPE          0.06232    0.04242   1.469  0.16116
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.121 on 16 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.8299, Adjusted R-squared:  0.7767
## F-statistic: 15.61 on 5 and 16 DF,  p-value: 1.161e-05
```

```
##from the summary, found that ECO most significantly influences EPI
ggplot(epi.APsubset, aes(x = ECO, y = MKP)) +
  geom_point() +
  stat_smooth(method = "lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
##compare RSE for both models
summary(lin.mod.epi)$sigma
```

```
## [1] 5.028606
```

```
summary(lin.mod.APsubset)$sigma
```

```
## [1] 4.120778
```

(Part 2) Which model is a better fit?

By calculating the RSE (residual standard error) for each model – the standard deviation of residuals – we can determine which model is a better fit. Because the linear model for the Asia-Pacific subset has a smaller RSE (less error), it is the better fit.

PART 3: CLASSFICATION

```
library(class)

##filter dataset for 3 regions
epi.subset1 <- epi[epi$region %in% c('Asia-Pacific', 'Eastern Europe', 'Sub-Saharan Africa'), ]

epi.subset1 <- na.omit(epi.subset1)

##generate a sample of the filtered subset
```

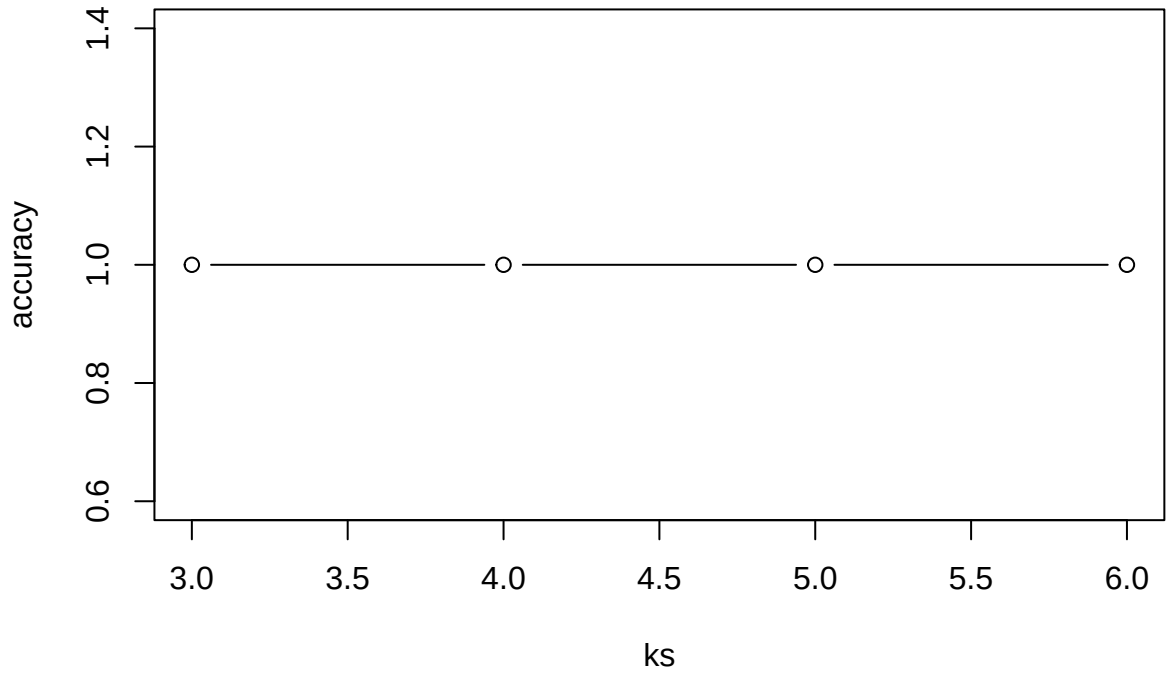
```r
epi.subset1.sample <- sample(90, 63)

epi.subset1.train <- epi.subset1[epi.subset1.sample,]
epi.subset1.test <- epi.subset1[-epi.subset1.sample,]

epi.subset1.train <- na.omit(epi.subset1.train)
epi.subset1.test <- na.omit(epi.subset1.test)

k=3

KNNpred <- knn(train = epi.subset1.train[6:10], test = epi.subset1.test[6:10], cl = epi.subset1.train$r

contingency.table <- table(KNNpred, epi.subset1.test$region)

contingency.matrix = as.matrix(contingency.table)

print(contingency.matrix)
```

```
##
## KNNpred              Sub-Saharan Africa
##   Asia-Pacific                        1
##   Sub-Saharan Africa                  0
```

```r
sum(diag(contingency.matrix))/length(epi.subset1.test$region)
```

```
## [1] 1
```

```r
accuracy <- c()
ks <- c(3,4,5,6)

for (k in ks) {

  KNNpred <- knn(train = epi.subset1.train[6:10], test = epi.subset1.test[6:10], cl = epi.subset1.train
  cm = as.matrix(table(Actual=KNNpred, Predicted = epi.subset1.test$region, dnn=list('predicted','actua

  accuracy <- c(accuracy,sum(diag(cm))/length(epi.subset1.test$region))

}

plot(ks,accuracy,type = "b")
```




```
##filter dataset for 3 regions
epi.subset2 <- epi[epi$region %in% c('Global West', 'Greater Middle East', 'Latin America & Caribbean')

epi.subset2 <- na.omit(epi.subset2)
##generate a sample of the filtered subset
epi.subset2.sample <- sample(70, 49)

epi.subset2.train <- epi.subset2[epi.subset2.sample,]
epi.subset2.test <- epi.subset2[-epi.subset2.sample,]

epi.subset2.train <- na.omit(epi.subset2.train)
epi.subset2.test <- na.omit(epi.subset2.test)

k=3

KNNpred <- knn(train = epi.subset2.train[6:10], test = epi.subset2.test[6:10], cl = epi.subset2.train$r

contingency.table <- table(KNNpred, epi.subset2.test$region)

contingency.matrix = as.matrix(contingency.table)

print(contingency.matrix)
```

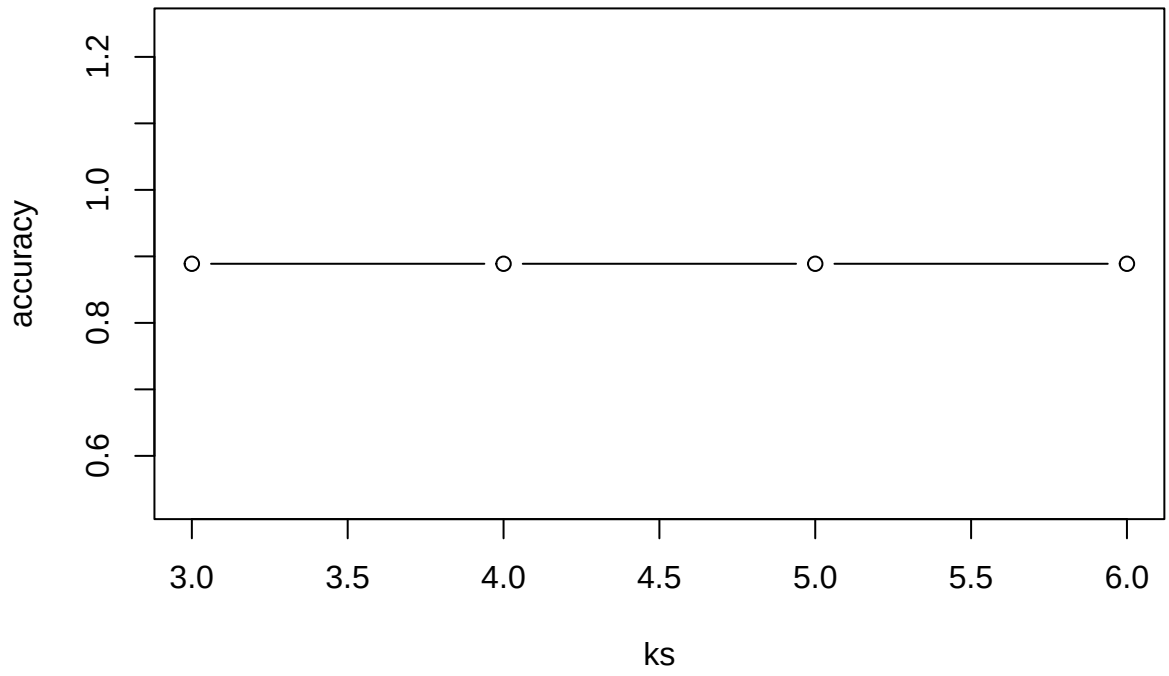
```
##
## KNNpred                     Global West Latin America & Caribbean
##   Global West                         0                         0
```

```
##   Latin America & Caribbean           1                         8
```

```
sum(diag(contingency.matrix))/length(epi.subset2.test$region)
```

```
## [1] 0.8888889
```

```
accuracy <- c()
ks <- c(3,4,5,6)

for (k in ks) {

  KNNpred <- knn(train = epi.subset2.train[6:10], test = epi.subset2.test[6:10], cl = epi.subset2.train$
  cm = as.matrix(table(Actual=KNNpred, Predicted = epi.subset2.test$region, dnn=list('predicted','actual

  accuracy <- c(accuracy,sum(diag(cm))/length(epi.subset2.test$region))

}

plot(ks,accuracy,type = "b")
```



(Part 3) Which model is better?

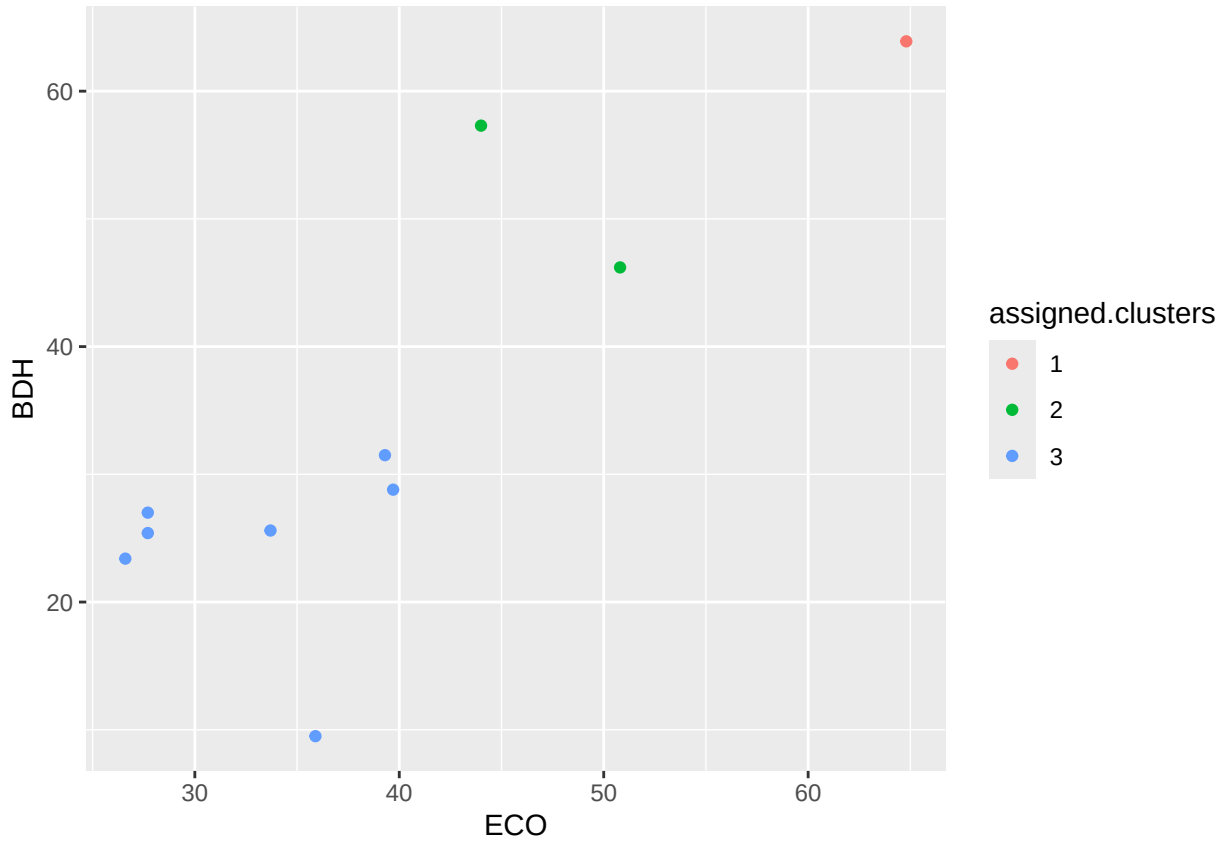
After testing each model across multiple k values, the model for subset 1 has a higher percent accuracy and therefore appears to be a better fit.

PART 4: CLUSTERING

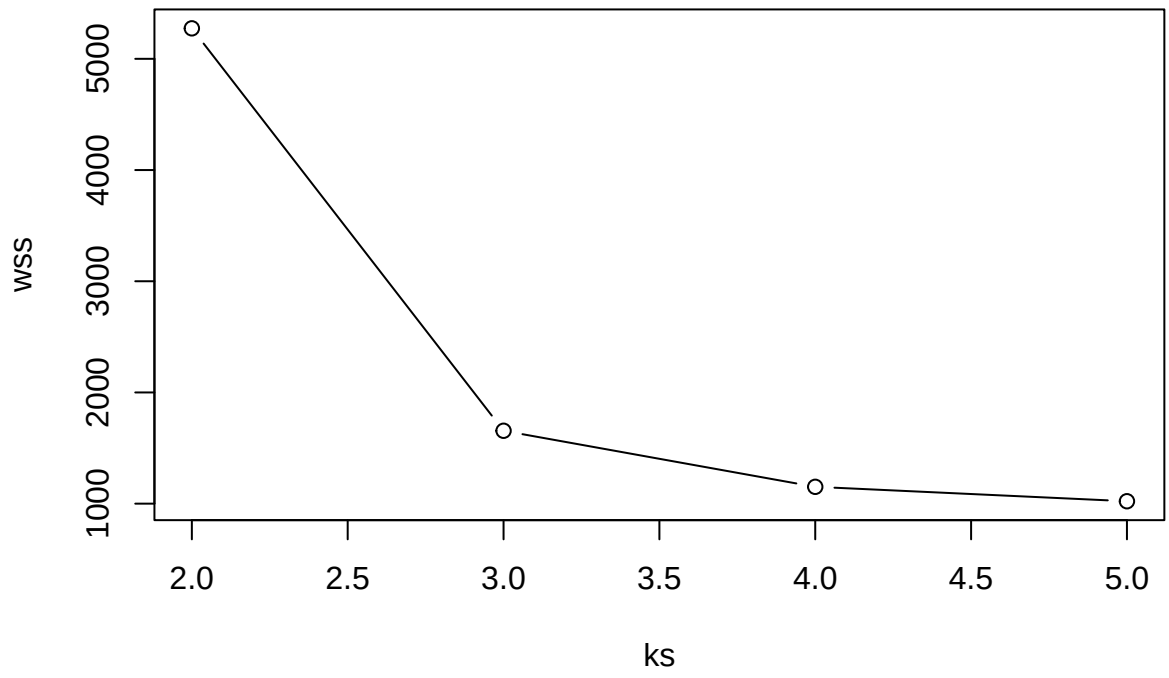
```r
##kmeans for subset1
set.seed(123)
epi.km1 <- kmeans(epi.subset1[6:10], centers = 3)
print(epi.km1)
```

```
## K-means clustering with 3 clusters of sizes 1, 2, 7
##
## Cluster means:
##        EPI      ECO      BDH      MKP       MHP
## 1 53.10000 64.80000 63.90000  2.20000 50.000000
## 2 38.20000 47.40000 51.75000 65.70000 23.100000
## 3 31.97143 32.94286 24.45714 18.65714  9.185714
##
## Clustering vector:
##  29  35  60  76  97  99 113 129 161 178
##   2   3   1   3   3   3   3   3   2   3
##
## Within cluster sum of squares by cluster:
## [1]    0.000  262.425 1392.854
##  (between_SS / total_SS =  84.9 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```r
assigned.clusters <- as.factor(epi.km1$cluster)

ggplot(epi.subset1, aes(x = ECO, y = BDH, colour = assigned.clusters)) + geom_point()
```



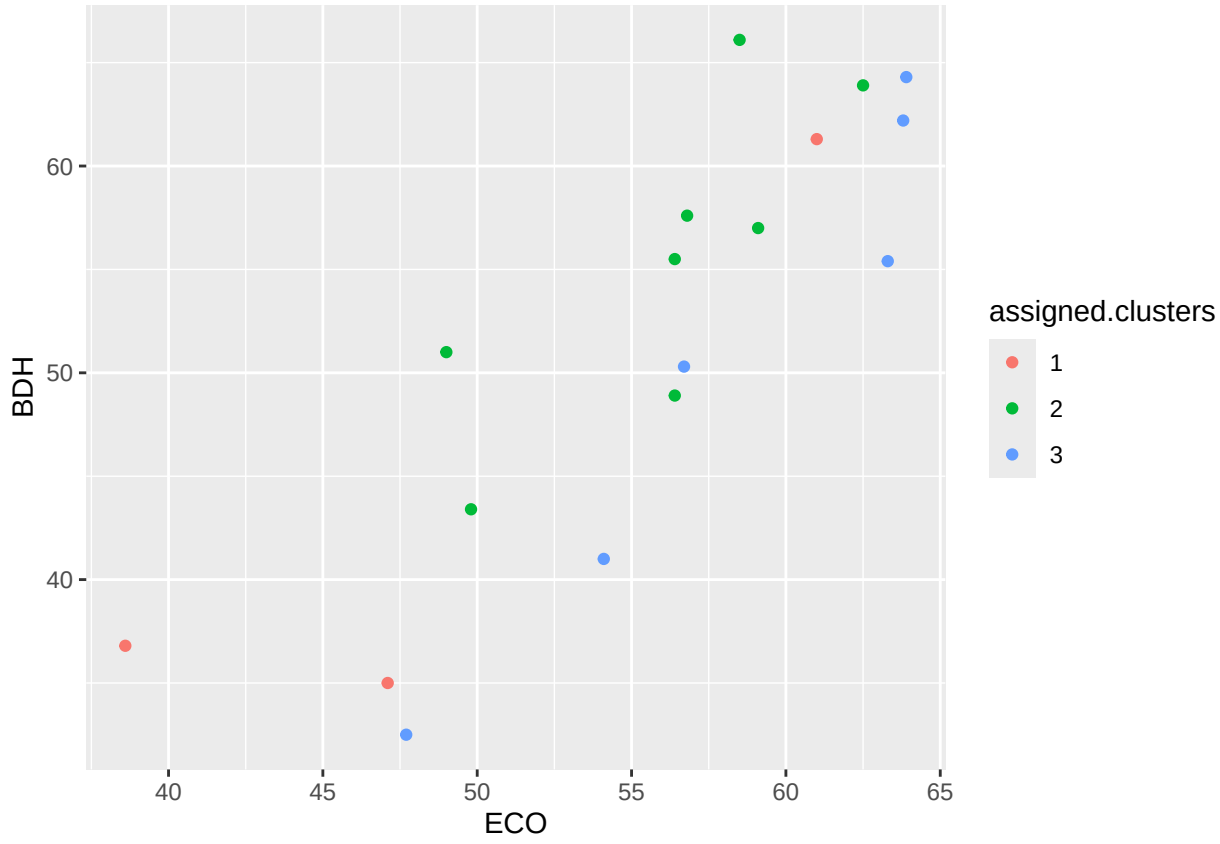

```
wss <- c()
ks <- c(2,3,4,5)
for (k in ks) {
  epi.km1 <- kmeans(epi.subset1[6:10], centers = k)
  wss <- c(wss,epi.km1$tot.withinss)
}
plot(ks,wss,type = "b")
```



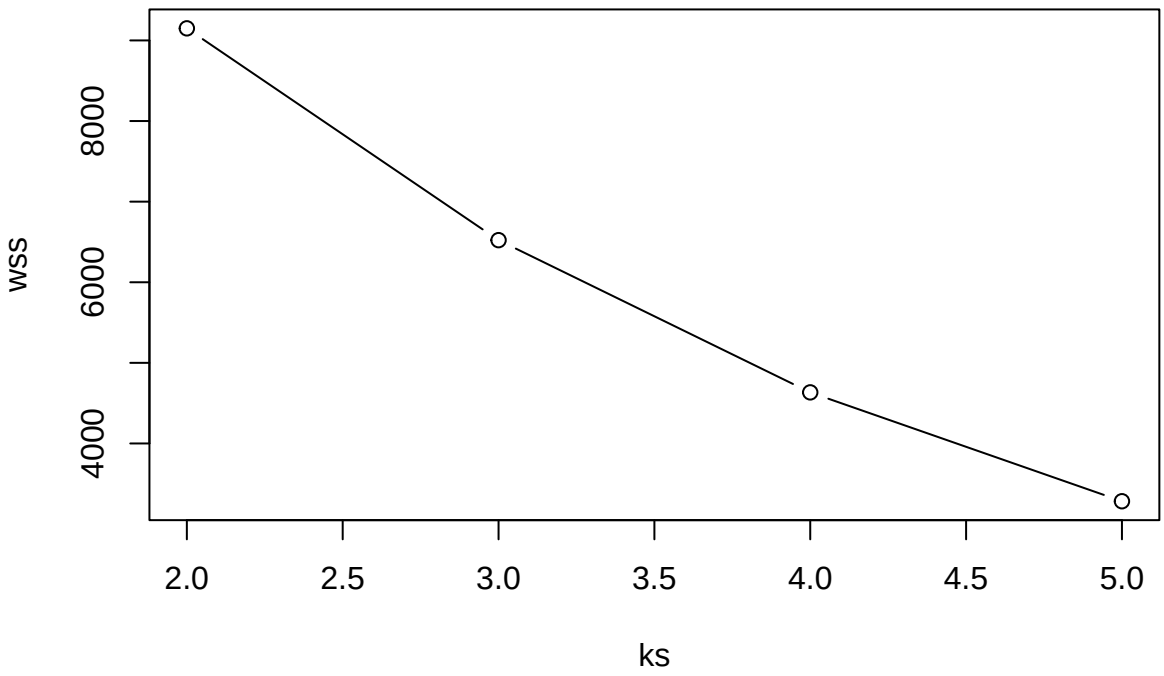

```
##kmeans for subset2
epi.km2 <- kmeans(epi.subset2[6:10], centers = 3)
print(epi.km2)
```

```
## K-means clustering with 3 clusters of sizes 3, 8, 6
##
## Cluster means:
##        EPI     ECO      BDH      MKP      MHP
## 1 44.16667 48.9000 44.36667 27.06667 22.36667
## 2 48.98750 56.0625 55.42500 86.20000 34.02500
## 3 54.30000 58.2500 50.95000 63.31667 47.80000
##
## Clustering vector:
##   6   8  23  36  38  41  48  49  67  72 106 118 125 128 155 173 177
##   1   3   3   2   2   2   2   3   1   2   3   2   2   2   3   3   1
##
## Within cluster sum of squares by cluster:
## [1] 2181.967 2112.697 2227.438
##  (between_SS / total_SS =  60.3 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
assigned.clusters <- as.factor(epi.km2$cluster)

ggplot(epi.subset2, aes(x = ECO, y = BDH, colour = assigned.clusters)) + geom_point()
```




```
wss <- c()
ks <- c(2,3,4,5)
for (k in ks) {
  epi.km2 <- kmeans(epi.subset2[6:10], centers = k)
  wss <- c(wss,epi.km2$tot.withinss)
}
plot(ks,wss,type = "b")
```



(Part 4) Which model is better?

The model with lower WCSS value across different k-values is a better fit; looking at the elbow plots, we can tell that the model for subset 1 contains lower values of WCSS across k-values, and is therefore a better model.