Project

Aummul Baneen Manasawala

4/28/2018

Medicare by regions Decoded

Data Introduction

The dataset to be analysed is from Centers for Medicare and Medicaid Services (https://www.cms.gov/research-statistics-data-and-systems/statistics-trends-and-reports/medicare-provider-charge-data/inpatient.html (https://www.cms.gov/research-statistics-data-and-systems/statistics-trends-and-reports/medicare-provider-charge-data/inpatient.html)). **Medicare** is the US program that assists in covering the costs of health expenses for people who are 65 or older, as well as some younger people with disabilities. The dataset we will be using gives the cumulative charges for procedures billed to Medicare for more than 3,000 U.S. hospitals for Fiscal Year 2011(Fiscal Year: the 12-month period ending on 30 September of that year, having begun on 1 October of theprevious calendar year). The dataset is intended to help Medicare recipients to have a sense of the costs at different institutes or for different procedures (Medicare beneficiaries still will have remaining out-of-pocketcosts after the federal government pays its portion).The dataset would be analysed to gain insights using probability distributions, visualization of distribu-tions, and group comparison methods.

The dataset is setup so that each combination of diagnosis and hospital is a separate entry. So the first entry gives information regarding the total costs at Fairbanks Memorial Hospital in Arkansas (the hospital) for the diagnosis of simple pneumonia and pleurisy with complications. A diagnosis/hospital combination is only included if the hospital has charges related to the particular diagnosis. Keep in mind that 'hospital' covers a range of institutes of different sizes. Some hospitals might be big organizations, while some might be more local clinics that do not handle specialized conditions or procedures.

Goal

The overarching goal of this project is to evaluate how the the amount a patient has to pay (i.e. after Medicare) change for different regions or urban/rural areas of the country and different diseases/health conditions. In making this comparison, we will focus on the following diagnoses that cover a range of different conditions that have different implications as to the hospitalstay required and the procedures necessary:
•Chronic Obstructive Pulmonary Disease (COPD) : This is an umbrella term used to describe progressivelung diseases, characterized by increasing breathlessness.192 - CHRONIC OBSTRUCTIVE PULMONARY DISEASE W/O CC/MCC
•Heart failure293 - HEART FAILURE & SHOCK W/O CC/MCC
•Hip/Pelvis fractures536 - FRACTURES OF HIP & PELVIS W/O MCC
•Diabetes638 - DIABETES W CC

1. Data Entry

```
#reading the data file
medicare_all <- read.csv("combinedData.csv")
```

```
#subsetting the data frame for the 4 issues to be analysed
medicare_sub <- medicare_all[medicare_all$DRG.Definition=="192 - CHRONIC OBSTRUCTIVE PUL
MONARY DISEASE W/O CC/MCC"|
medicare_all$DRG.Definition=="293 - HEART FAILURE & SHOCK W/O CC/MCC"|
medicare_all$DRG.Definition=="536 - FRACTURES OF HIP & PELVIS W/O MCC"|
medicare_all$DRG.Definition=="638 - DIABETES W CC", ]

#Defining a vector for names of the four diagnosis

name_diag <- c("COPD", "Heart Failure", "Hip Fracture", "Diabetes")
```

```
# making Urban and Regions factors


medicare_sub$Urban <- factor(medicare_sub$Urban,
                             levels = c(0:5),
                             labels = c("mix", "only_rural", "rural_urbanclusters", "onl
y_urbanclusters", "urbanclusters_urbanarea", "only_urbanarea")
                             )
levels(medicare_sub$Urban)
```

```
## [1] "mix"                     "only_rural"
## [3] "rural_urbanclusters"     "only_urbanclusters"
## [5] "urbanclusters_urbanarea" "only_urbanarea"
```

Creating new variables that define the following: the absolute amount the patient pays (PatientPays) and the percentage of the payment that is paid by the patient(PctPatientPays).

```
medicare_sub[ , 13] <- medicare_sub$Average.Total.Payments - medicare_sub$Average.Medica
re.Payments
medicare_sub[, 14] <- (medicare_sub[ , 13]/medicare_sub$Average.Total.Payments)*100

#Naming the columns
names(medicare_sub)[13] <- "PatientPays"
names(medicare_sub)[14] <- "PctPatientPays"
```

Creating a factor variable urbanByRegions in data.frame that gives the cross of Urban and regions(e.g. rural, South, rural Northeast, etc.)

```
medicare_sub$UrbanByRegions <- medicare_sub$Urban:medicare_sub$regions
levels(medicare_sub$UrbanByRegions)
```

```
##  [1] "mix:midwest"
##  [2] "mix:northeast"
##  [3] "mix:south"
##  [4] "mix:west"
##  [5] "only_rural:midwest"
##  [6] "only_rural:northeast"
##  [7] "only_rural:south"
##  [8] "only_rural:west"
##  [9] "rural_urbanclusters:midwest"
## [10] "rural_urbanclusters:northeast"
## [11] "rural_urbanclusters:south"
## [12] "rural_urbanclusters:west"
## [13] "only_urbanclusters:midwest"
## [14] "only_urbanclusters:northeast"
## [15] "only_urbanclusters:south"
## [16] "only_urbanclusters:west"
## [17] "urbanclusters_urbanarea:midwest"
## [18] "urbanclusters_urbanarea:northeast"
## [19] "urbanclusters_urbanarea:south"
## [20] "urbanclusters_urbanarea:west"
## [21] "only_urbanarea:midwest"
## [22] "only_urbanarea:northeast"
## [23] "only_urbanarea:south"
## [24] "only_urbanarea:west"
```

```
table(medicare_sub$UrbanByRegions)
```

```
##
##                       mix:midwest                       mix:northeast
##                               663                                 535
##                         mix:south                         mix:west
##                              1239                                 511
##                 only_rural:midwest               only_rural:northeast
##                                13                                  13
##                   only_rural:south                   only_rural:west
##                                75                                   3
##       rural_urbanclusters:midwest     rural_urbanclusters:northeast
##                               464                                 199
##         rural_urbanclusters:south          rural_urbanclusters:west
##                              1010                                 154
##        only_urbanclusters:midwest      only_urbanclusters:northeast
##                                 0                                   2
##          only_urbanclusters:south           only_urbanclusters:west
##                                 4                                   0
##   urbanclusters_urbanarea:midwest urbanclusters_urbanarea:northeast
##                                 0                                   0
##     urbanclusters_urbanarea:south      urbanclusters_urbanarea:west
##                                 0                                   0
##            only_urbanarea:midwest          only_urbanarea:northeast
##                               612                                 625
##              only_urbanarea:south              only_urbanarea:west
##                               784                                 499
```

```
medicare_sub <- droplevels(medicare_sub)
table(medicare_sub$UrbanByRegions)
```

```
##
##                     mix:midwest                     mix:northeast
##                             663                               535
##                       mix:south                       mix:west
##                            1239                               511
##              only_rural:midwest             only_rural:northeast
##                              13                                13
##                only_rural:south                only_rural:west
##                              75                                 3
##   rural_urbanclusters:midwest rural_urbanclusters:northeast
##                             464                               199
##     rural_urbanclusters:south      rural_urbanclusters:west
##                            1010                               154
##  only_urbanclusters:northeast      only_urbanclusters:south
##                               2                                 4
##         only_urbanarea:midwest      only_urbanarea:northeast
##                             612                               625
##           only_urbanarea:south            only_urbanarea:west
##                             784                               499
```

```
summary(medicare_sub)
```

```
##  Provider.State
##  CA     : 596
##  TX     : 592
##  FL     : 519
##  NY     : 459
##  PA     : 399
##  IL     : 389
##  (Other):4947
##                                                DRG.Definition
##  192 - CHRONIC OBSTRUCTIVE PULMONARY DISEASE W/O CC/MCC:2593
##  293 - HEART FAILURE & SHOCK W/O CC/MCC                :2443
##  536 - FRACTURES OF HIP & PELVIS W/O MCC               :1103
##  638 - DIABETES W CC                                   :1762
##
##
##
##   Provider.Id                        Provider.Name       Provider.City
##  Min.   : 10001   GOOD SAMARITAN HOSPITAL :  29   CHICAGO     :  81
##  1st Qu.:110186   MERCY MEDICAL CENTER    :  20   BALTIMORE   :  42
##  Median :250069   ST JOSEPH HOSPITAL      :  20   BROOKLYN    :  41
##  Mean   :257006   ST JOSEPH MEDICAL CENTER:  20   HOUSTON     :  41
##  3rd Qu.:390016   MERCY HOSPITAL          :  19   PHILADELPHIA:  40
##  Max.   :670071   ST FRANCIS HOSPITAL     :  15   SPRINGFIELD :  37
##                   (Other)                 :7778   (Other)     :7619
##  Total.Discharges Average.Covered.Charges Average.Total.Payments
##  Min.   : 11.00   Min.   :  3134          Min.   : 3144
##  1st Qu.: 17.00   1st Qu.: 11352          1st Qu.: 4212
##  Median : 25.00   Median : 15544          Median : 4711
##  Mean   : 33.47   Mean   : 18368          Mean   : 5072
##  3rd Qu.: 41.00   3rd Qu.: 22070          3rd Qu.: 5532
##  Max.   :326.00   Max.   :130690          Max.   :19512
##
##  Average.Medicare.Payments Provider.Zip.Code      regions
##  Min.   : 2182             Min.   : 1040     midwest  :1842
##  1st Qu.: 3255             1st Qu.:27565     northeast:1469
##  Median : 3723             Median :44112     south    :3357
##  Mean   : 4093             Mean   :47812     west     :1233
##  3rd Qu.: 4517             3rd Qu.:72342
##  Max.   :18613             Max.   :99701
##
##                  Urban       PatientPays      PctPatientPays
##  mix                :2948   Min.   :  261.2   Min.   : 3.898
##  only_rural         : 104   1st Qu.:  770.2   1st Qu.:15.297
##  rural_urbanclusters:1827   Median :  898.4   Median :19.230
##  only_urbanclusters :   6   Mean   :  979.3   Mean   :19.949
##  only_urbanarea     :2520   3rd Qu.: 1059.3   3rd Qu.:23.569
##  NA's               : 496   Max.   :10676.8   Max.   :74.215
##
##                    UrbanByRegions
##  mix:south                :1239
##  rural_urbanclusters:south:1010
##  only_urbanarea:south     : 784
##  mix:midwest              : 663
```

```
##  only_urbanarea:northeast : 625
##  (Other)                  :3084
##  NA's                     : 496
```

2. Basic Summaries

```
# Basic Summaries
basic_summary_patientPays <- summary(medicare_sub$PatientPays)
basic_summary_patientPays
```

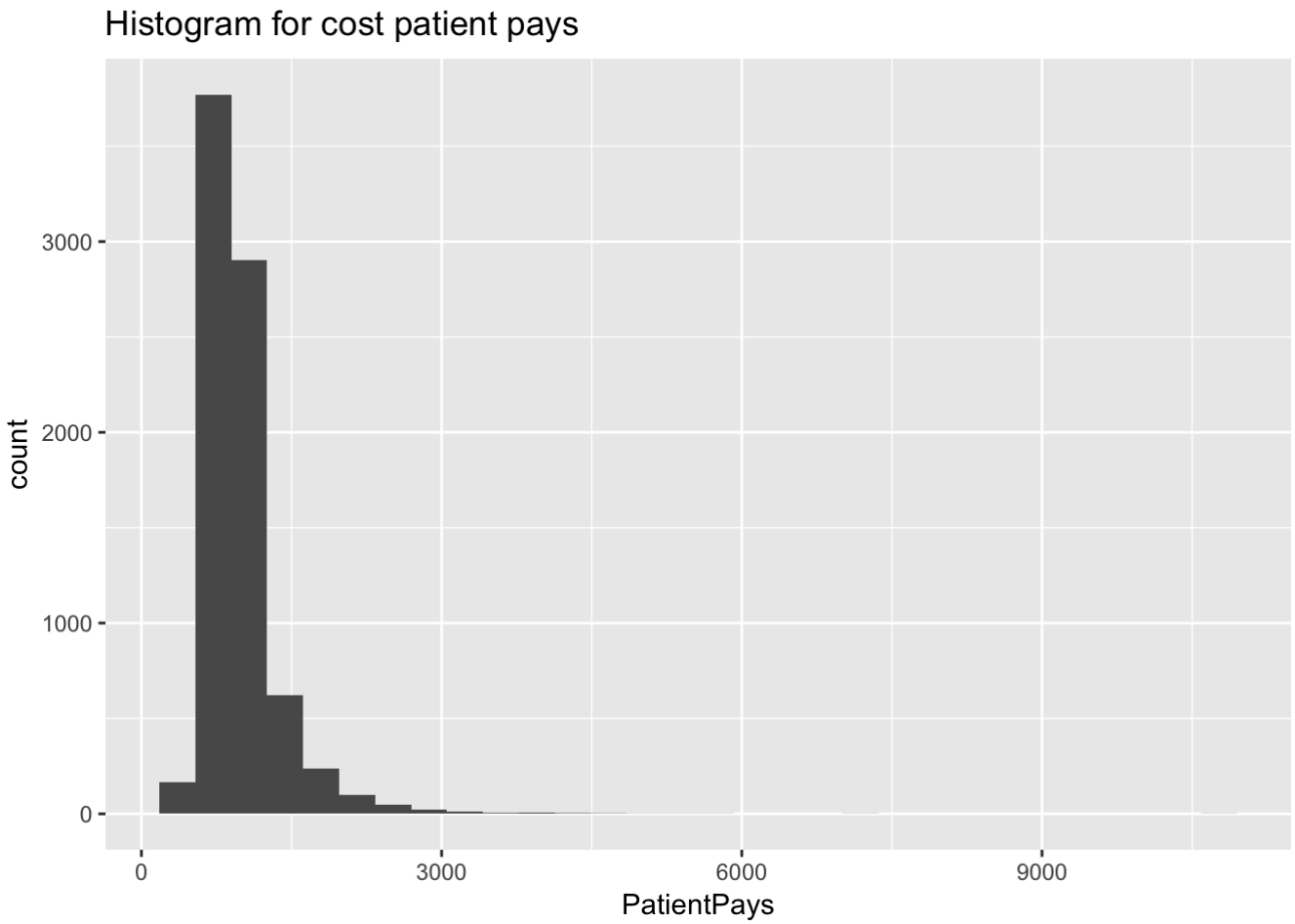
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   261.2   770.2   898.4   979.3  1059.3 10676.8
```

```
basic_summary_pctPatientPays <- summary(medicare_sub$PctPatientPays)
basic_summary_pctPatientPays
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.898  15.297  19.230  19.949  23.569  74.215
```

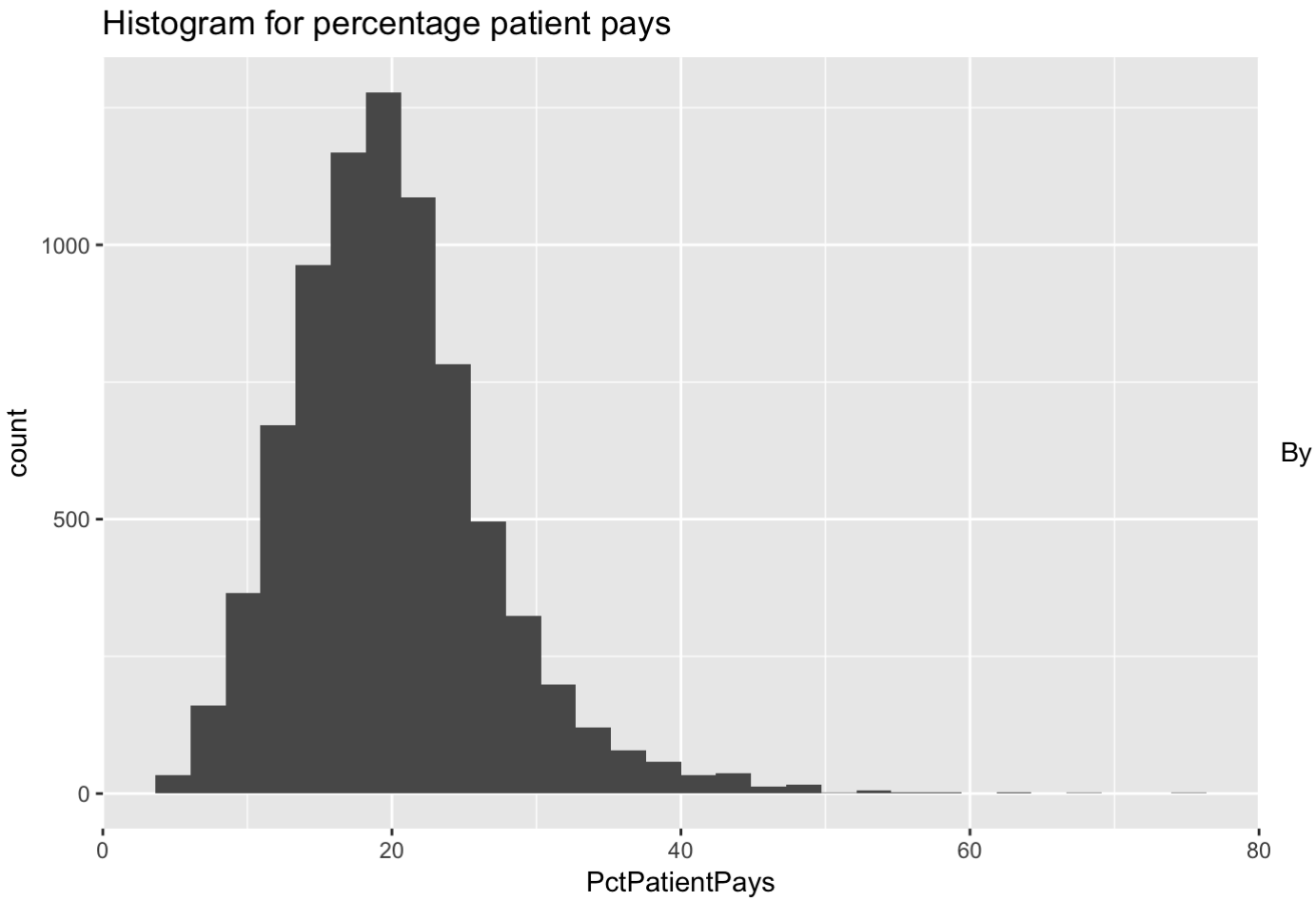
```
#Plotting Histograms
ggplot(data = medicare_sub)+
  geom_histogram(aes(x = PatientPays))+
  labs(title = "Histogram for cost patient pays")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data = medicare_sub)+
  geom_histogram(aes(x = PctPatientPays))+
  labs(title = "Histogram for percentage patient pays")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

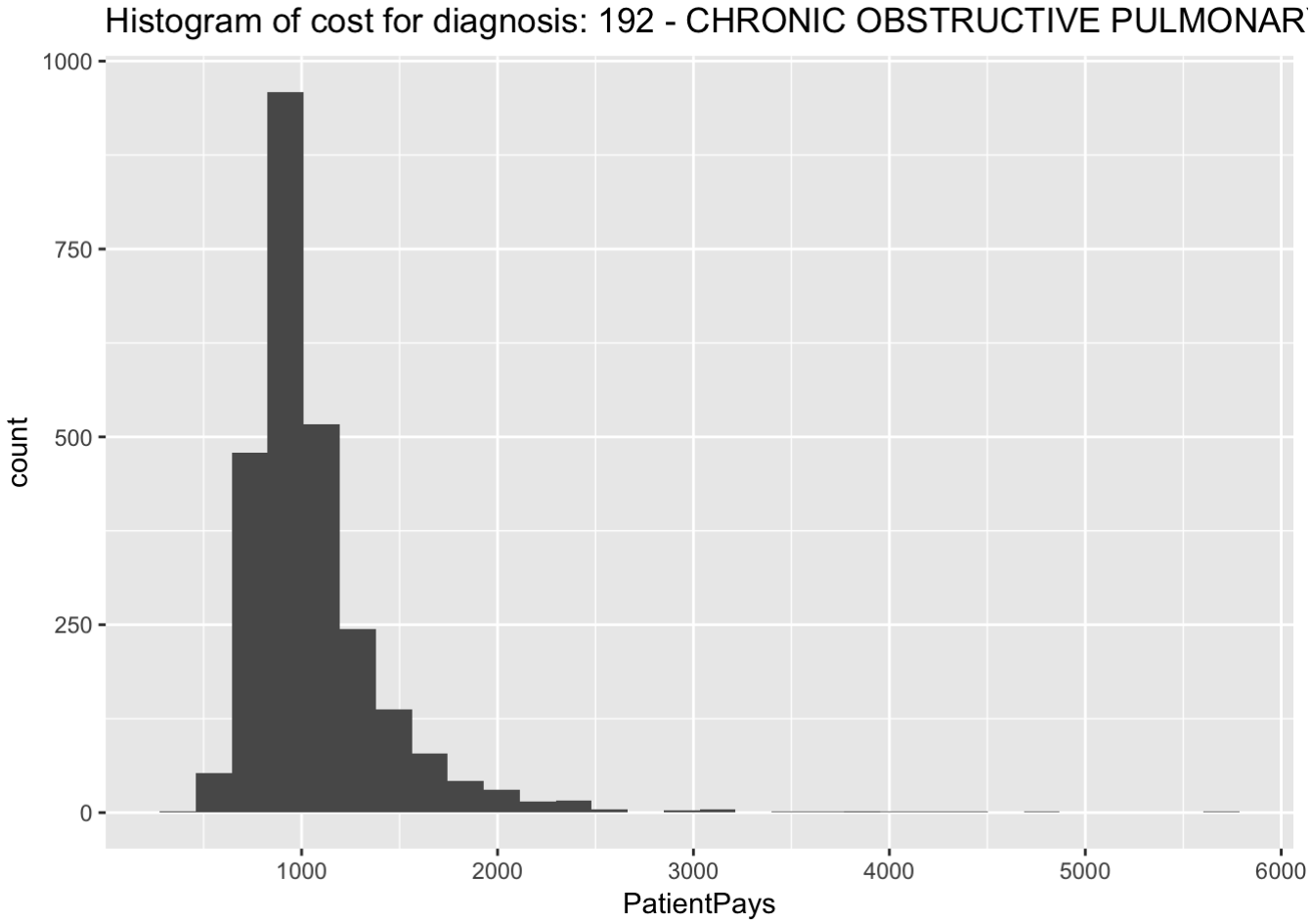


By looking at the two histograms, while the histogram for percentage pay looks good, the histogram for the absolute cost that the patient pays from pocket appears to be a bit skewed. When we have skewed data, it can be difficult to compare the distributions because so much of the data is bunched up on one end, but our axes stretch to cover the large values that make up a relatively small proportion of the data. This is also means that our eye focuses on those values too. In current situation, data is all positive, yet take on values close to zero. In this case, there are many data points bunched up by zero (because they can't go lower) with a definite right skew. The data can be nicely spread out for visualization purposes by either the log or square-root transformations.

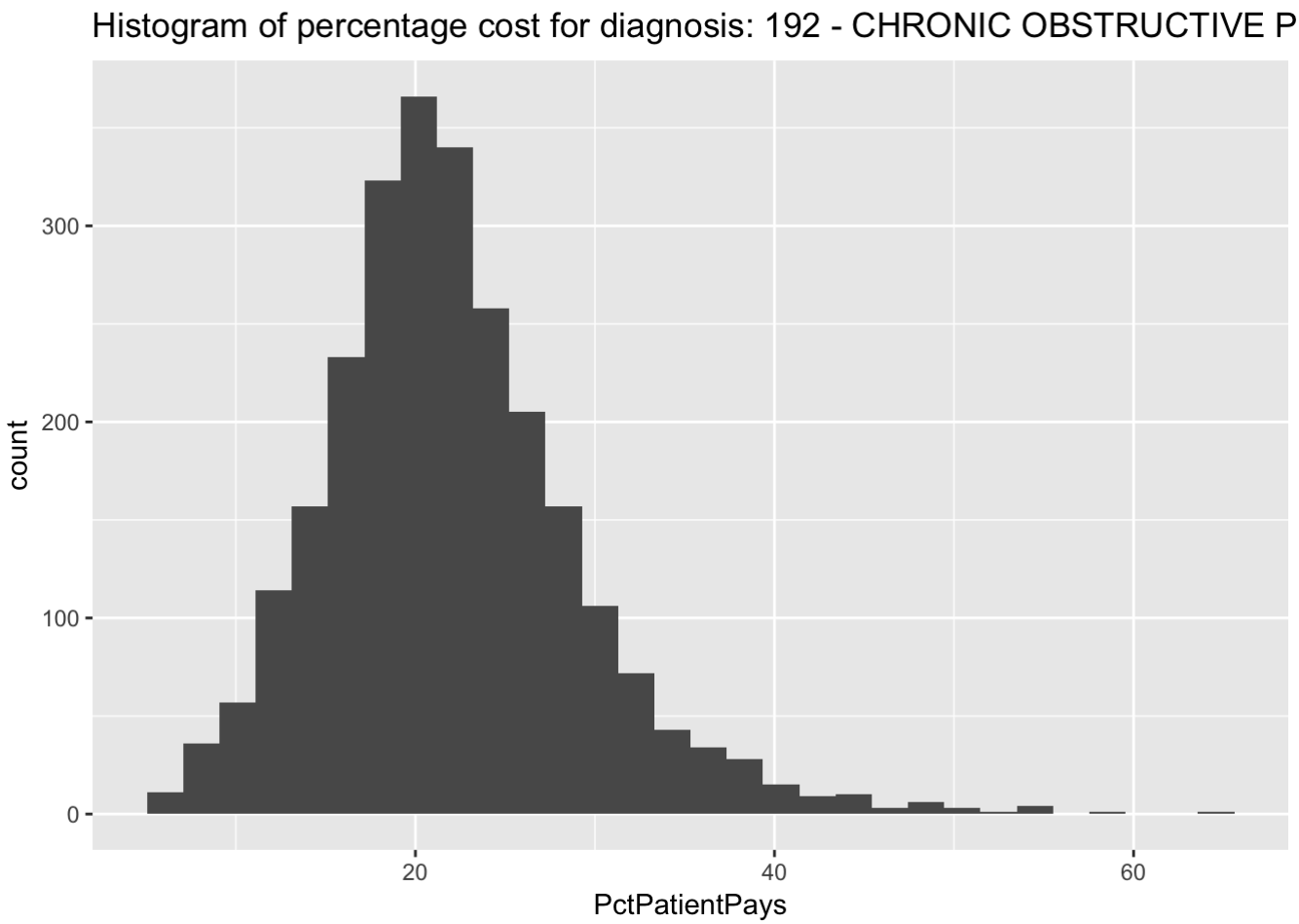
Let us look at the distribution of the response variable for each of the four diagnoses to decide which amongst them would need a log transformation

```
par(mfrow = c(4, 2))
for (i in c(1:4)) {
  title_abs <- paste0("Histogram of cost for diagnosis: ", levels(medicare_sub$DRG.Defin
ition)[i])
  title_pct <- paste0("Histogram of percentage cost for diagnosis: ", levels(medicare_su
b$DRG.Definition)[i])
  list_wrt_diagnosis <- medicare_sub[medicare_sub$DRG.Definition==levels(medicare_sub$DR
G.Definition)[i], ]
  print(ggplot(data = list_wrt_diagnosis)+geom_histogram(aes(x = PatientPays))+labs(titl
e = title_abs))
  print(ggplot(data = list_wrt_diagnosis)+geom_histogram(aes(x = PctPatientPays))+labs(t
itle = title_pct))
}
```

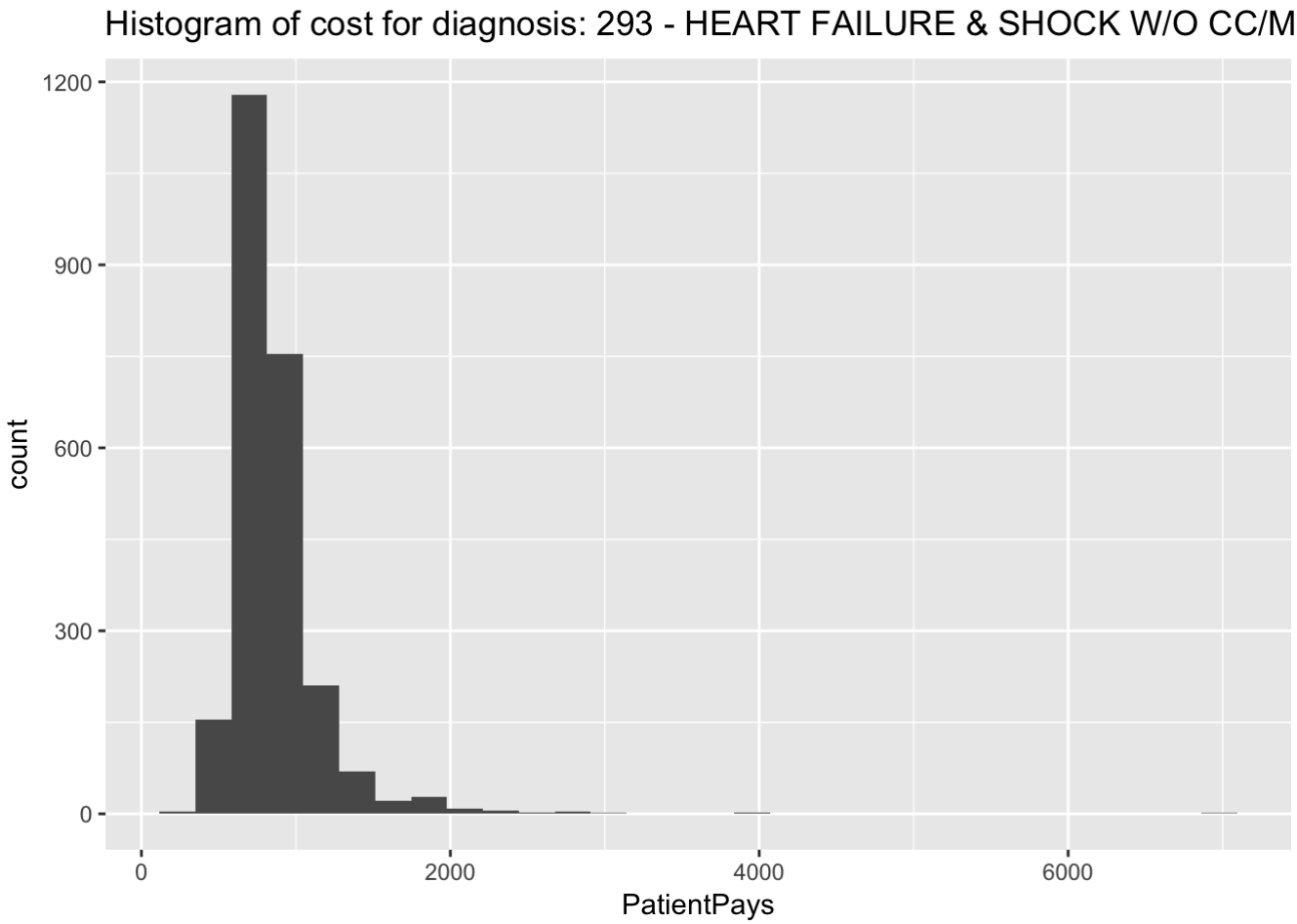
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



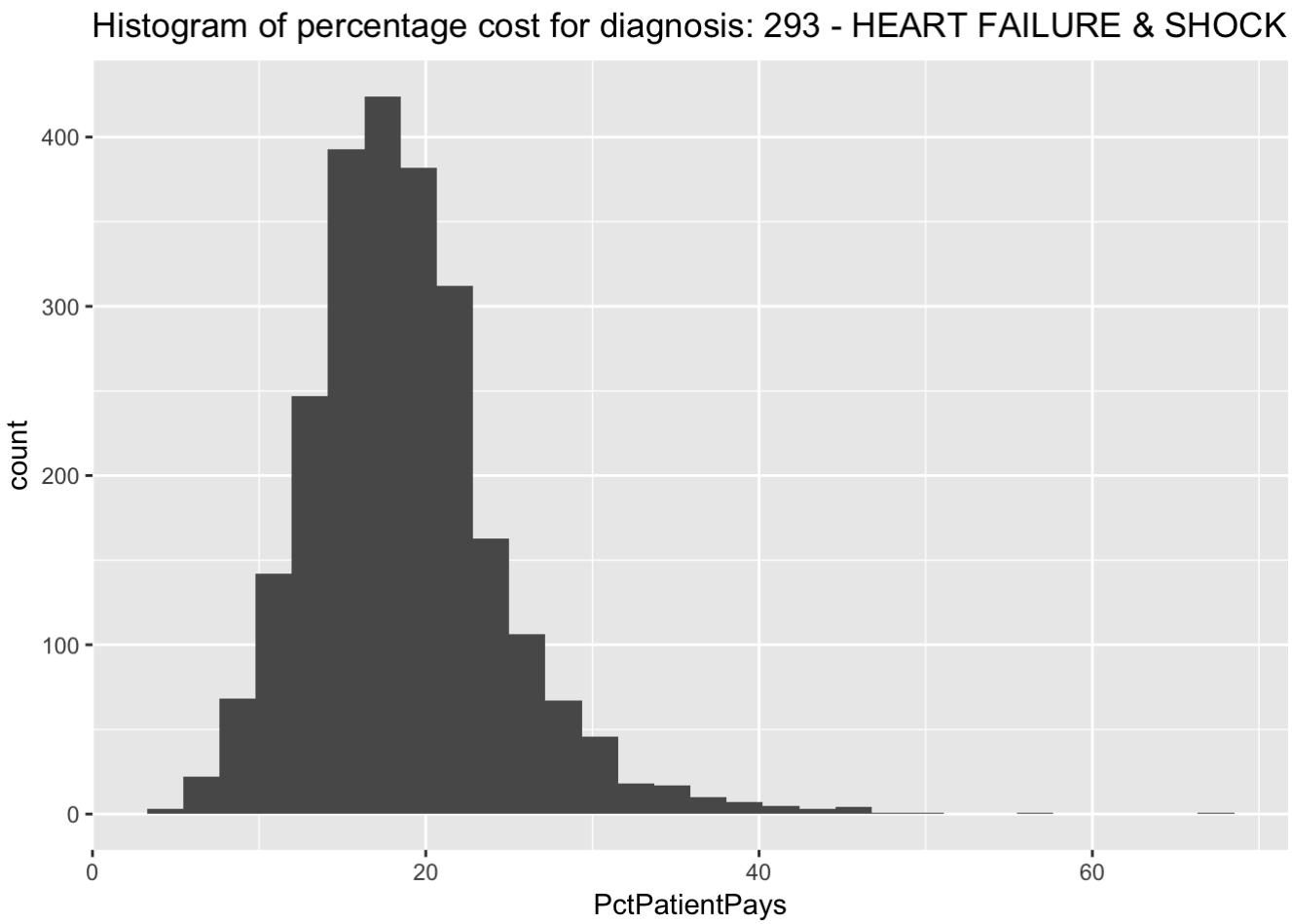

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



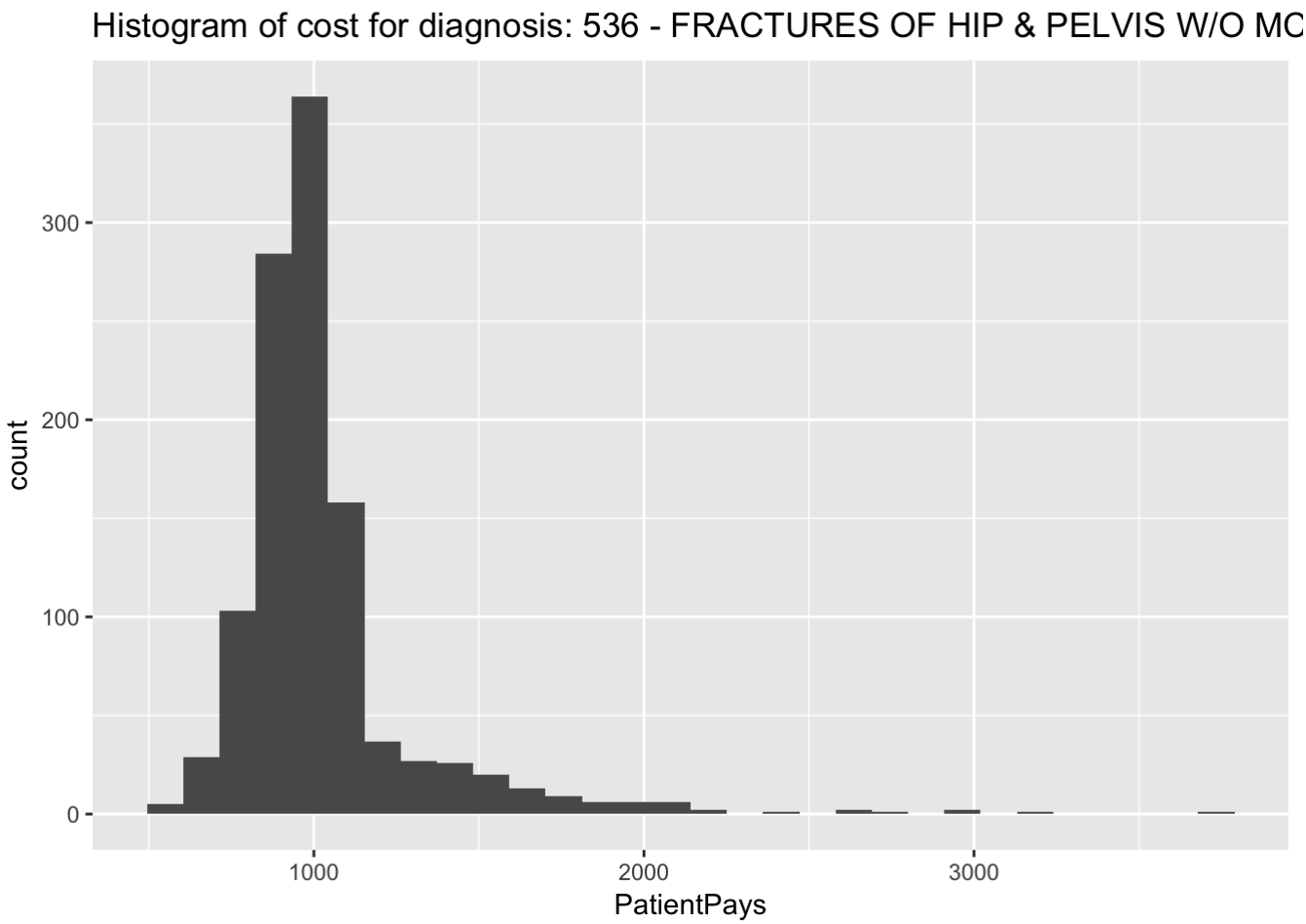

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



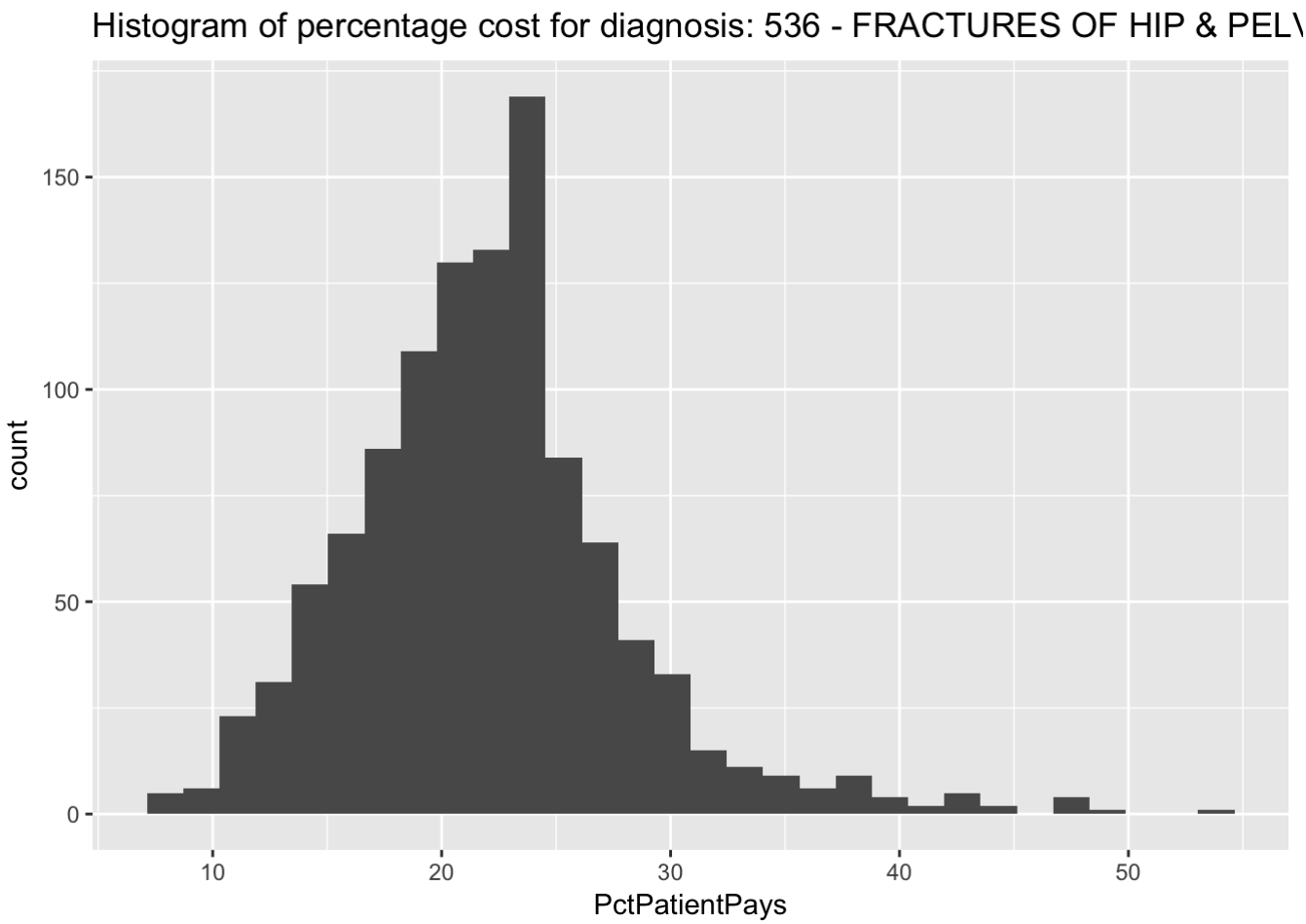

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



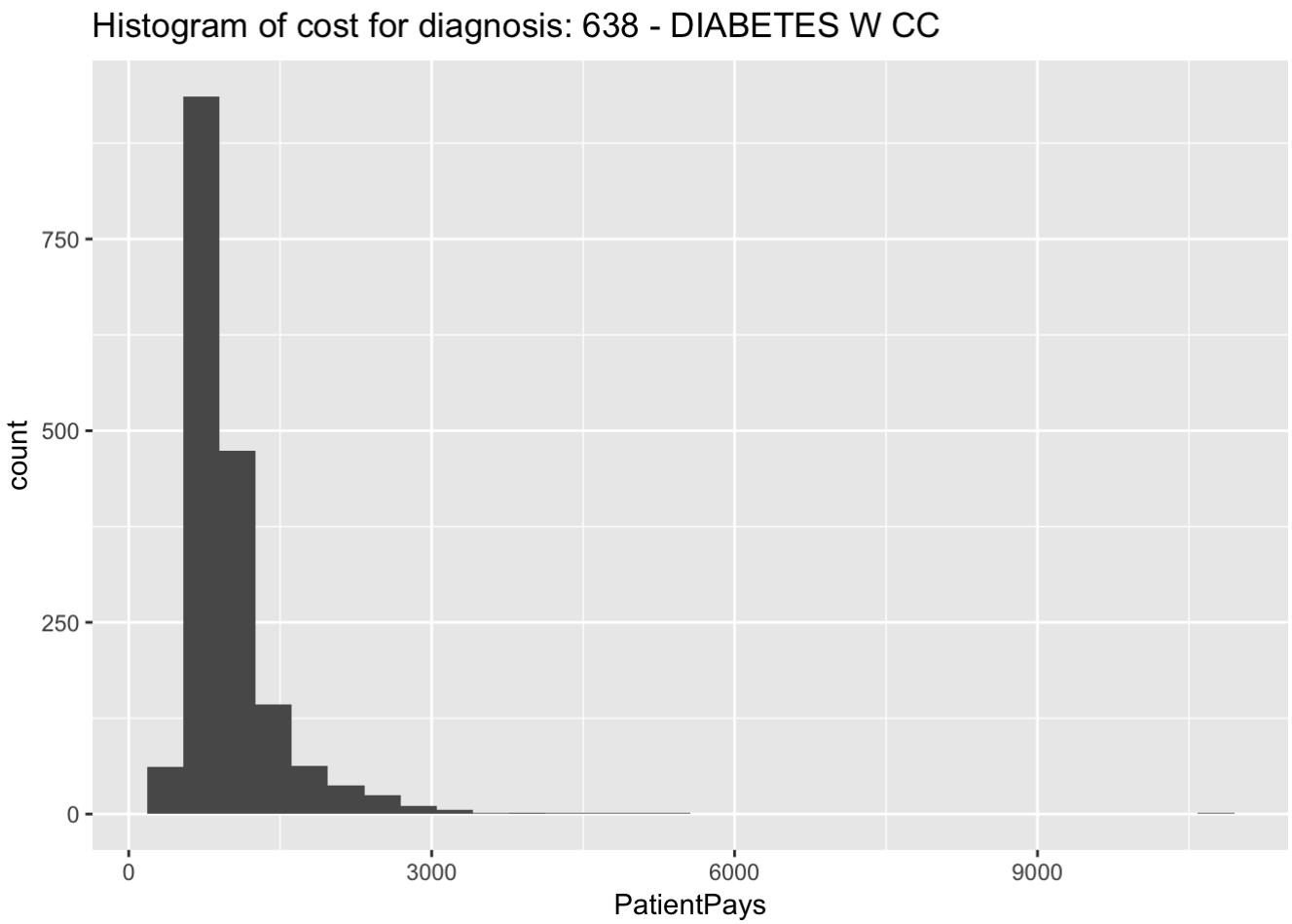

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



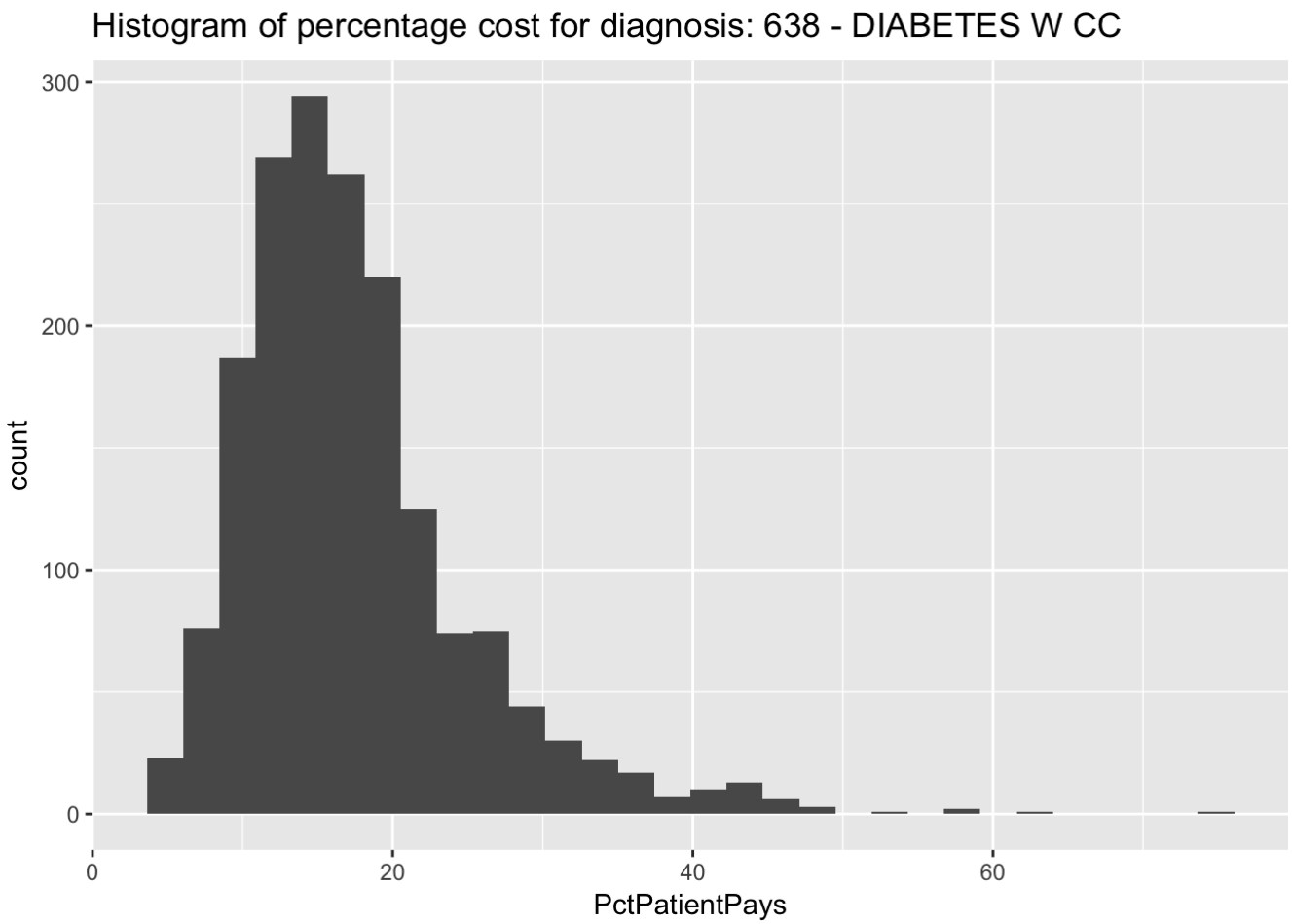

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



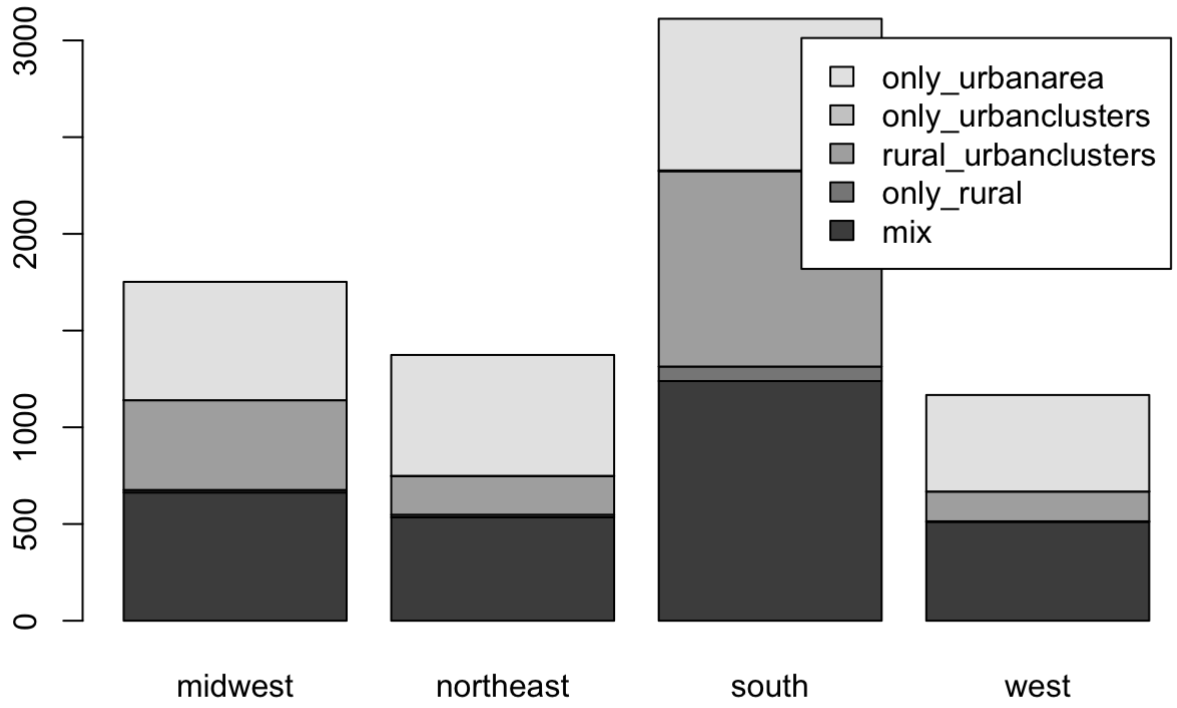
For each of the four diagnosis, we find highly skewed distribution of absolute cost paid by the patient and would recommend a log transformation. Since we have two categorical variables, then the natural way to summarize their relationship is to cross-tabulate the values of the levels. Following is the contingency table of Urbans and regions.

```
cont_urbans_regions <- with(medicare_sub, table(Urban, regions))
cont_urbans_regions
```

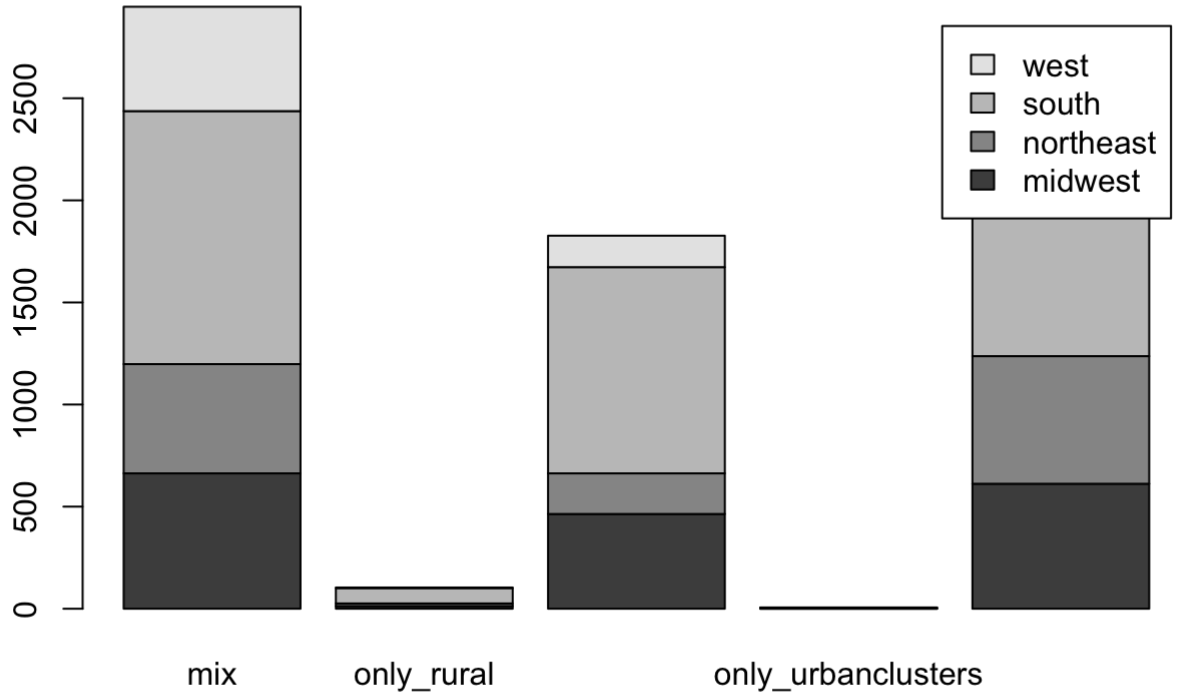
```
##                     regions
## Urban                midwest northeast south west
##   mix                    663       535  1239  511
##   only_rural              13        13    75    3
##   rural_urbanclusters    464       199  1010  154
##   only_urbanclusters       0         2     4    0
##   only_urbanarea         612       625   784  499
```

We can similarly make barplots to demonstrate these relationships.

```
barplot(cont_urbans_regions, legend = TRUE)
```

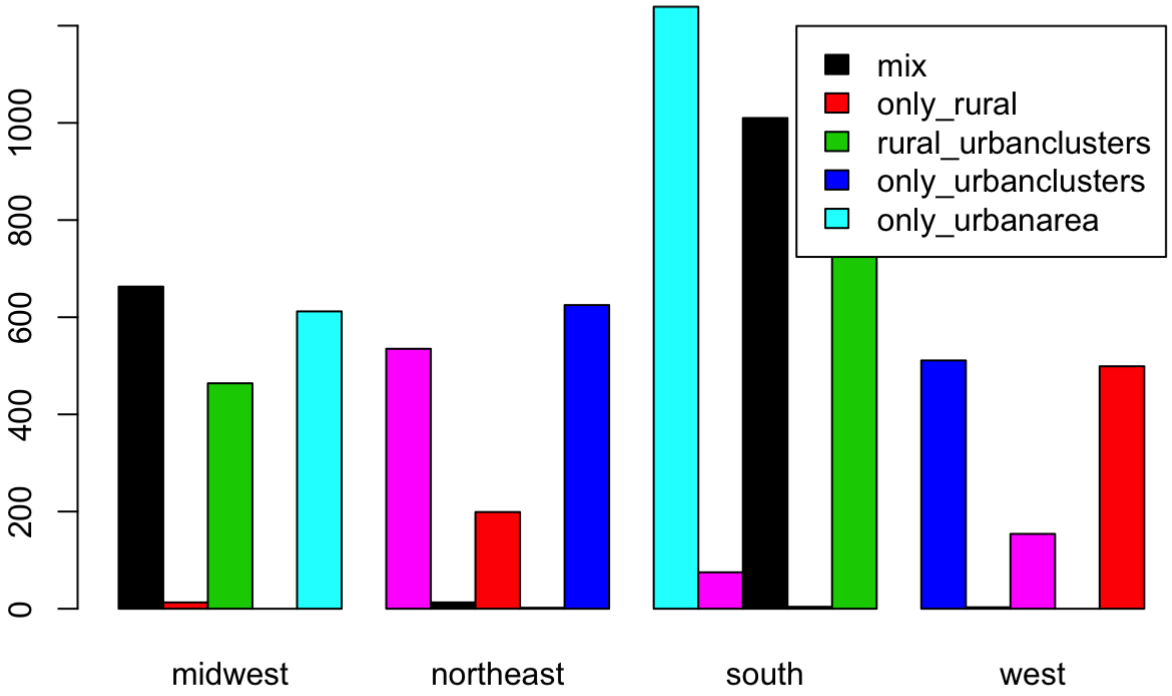



```
barplot(t(cont_urbans_regions), legend = TRUE)
```



It can also be helpful to separate out the other variables, rather than stacking them, and to change the colors.

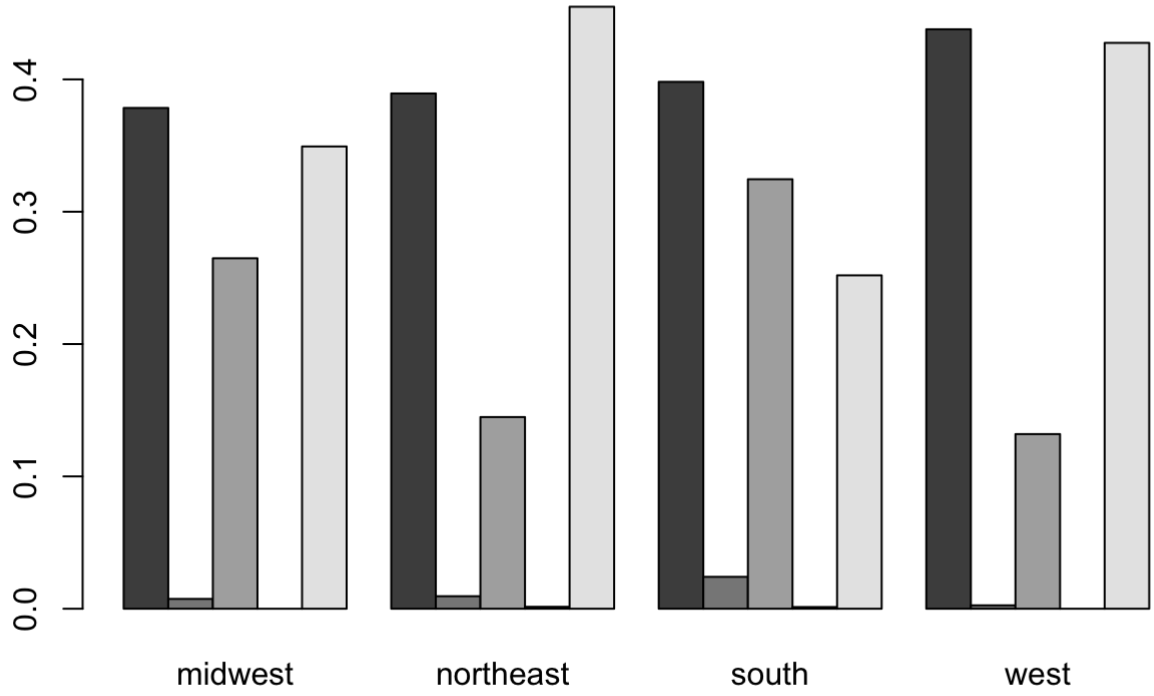
```
barplot(cont_urbans_regions, beside = TRUE, legend = TRUE,
col = palette()[1:6])
```



When we look at the contingency table, a natural question we ask is whether the distribution of the data changes across the different categories. For example, for patients from “south” region, there is a distribution of urbana/rural areas and similarly for each region. We can get these by making the counts into proportions within each category.

```
prop_cont_urban_region <- prop.table(with(medicare_sub, table(Urban, regions)), margin =
 2)
```

```
barplot(prop.table(prop_cont_urban_region, margin = 2), beside = TRUE)
```

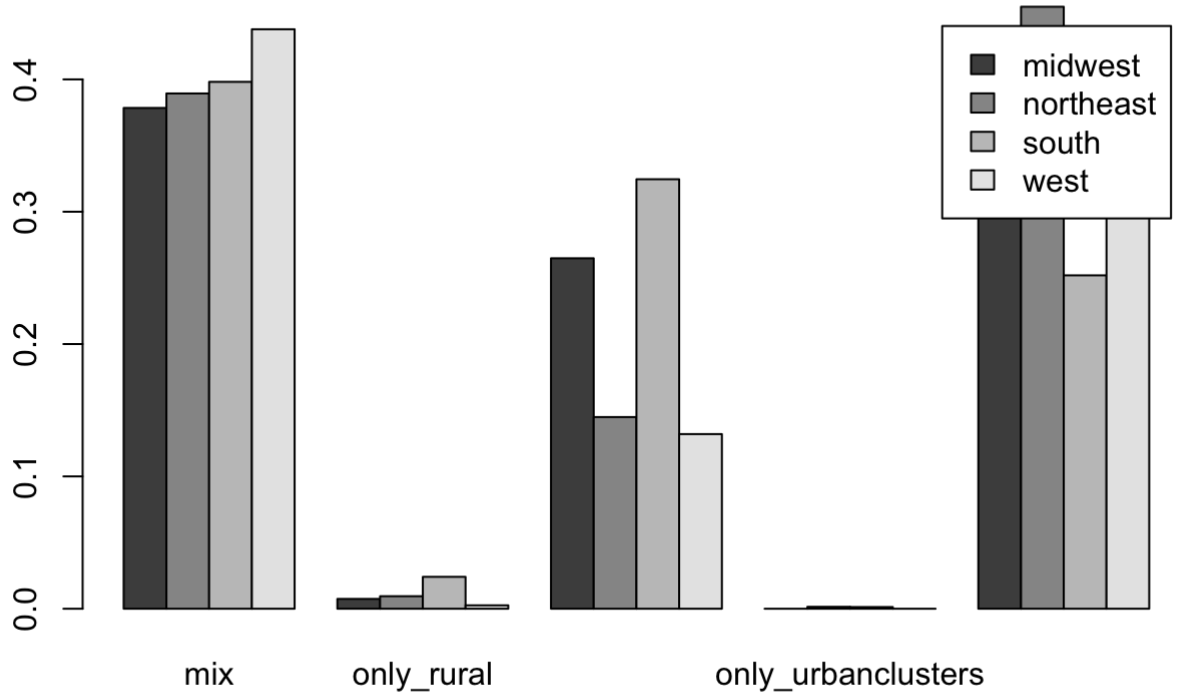


Flipping the coin to get idea on the other aspect.

```
prop_cont_region_urban <- prop.table(with(medicare_sub, table(regions, Urban)), margin =
 1)
prop_cont_region_urban
```

```
##           Urban
## regions             mix  only_rural rural_urbanclusters only_urbanclusters
##   midwest   0.378424658 0.007420091         0.264840183        0.000000000
##   northeast 0.389374090 0.009461426         0.144832606        0.001455604
##   south     0.398136247 0.024100257         0.324550129        0.001285347
##   west      0.437874893 0.002570694         0.131962296        0.000000000
##           Urban
## regions     only_urbanarea
##   midwest      0.349315068
##   northeast    0.454876274
##   south        0.251928021
##   west         0.427592117
```

```
barplot((prop.table(prop_cont_region_urban, margin = 1)), beside = TRUE,
legend = TRUE)
```



These plots show that the data points in the bucket of Urban 1, 3 and 4 are quite few compared to other levels of the factor Urban. We could ignore these buckets for better comparision and insights in the other buckets that majorly contributes the data.

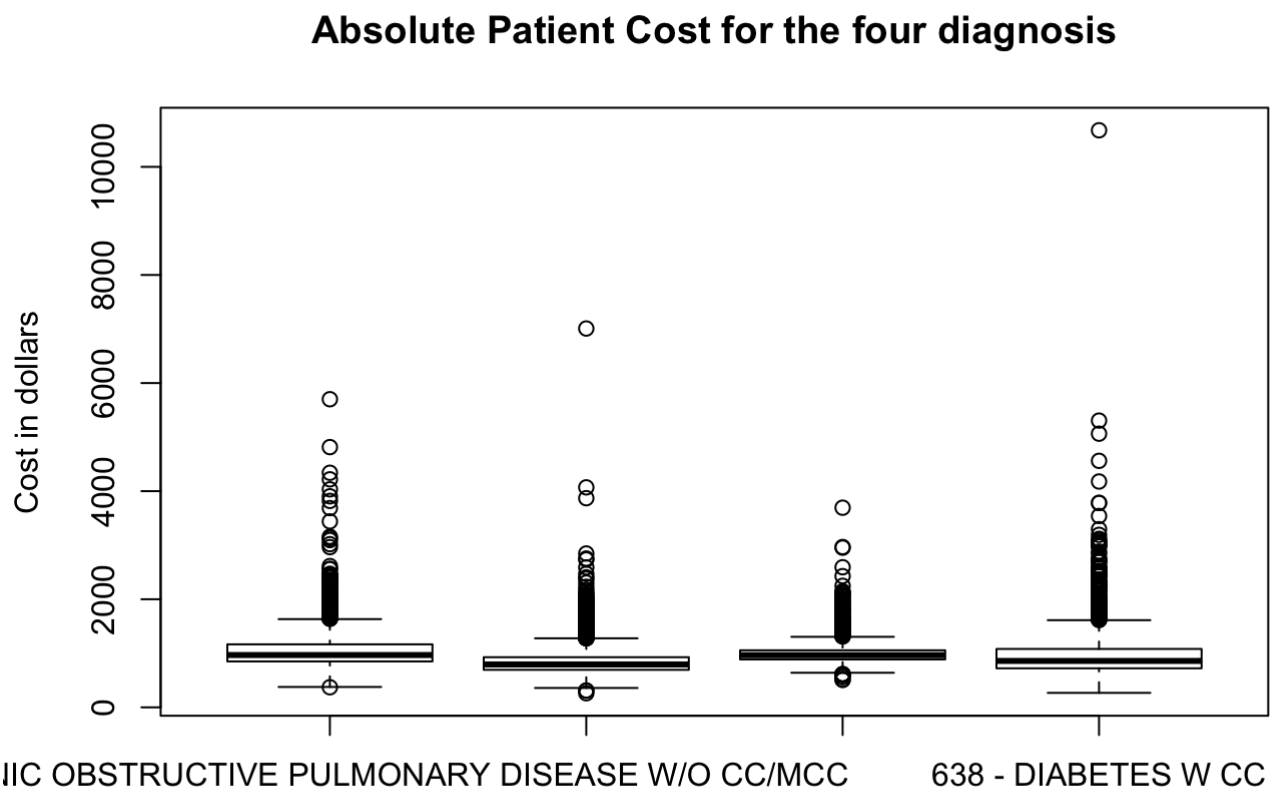
3. Exploration of Distributions in groups

In order to get a better idea about the distributions of the two response variables as a function of being in any one of the group of the categorical variable urbanByRegions, we would start with plotting the box plots to visualise the data for each of the four diagnosis. Since, we noted previously that the values in the group of Urban 1, 3 and 4 were rare and not substantial enough to do analysis on, we would remove those from now on.

```
#Removing the categories 1, 3, 4 of Urban from our analysis
dff <- medicare_sub[medicare_sub$Urban!="only_rural"&medicare_sub$Urban!="only_urbanclus
ters"& medicare_sub$Urban!="urbanclusters_urbanarea", ]
dff <- droplevels(dff)
```

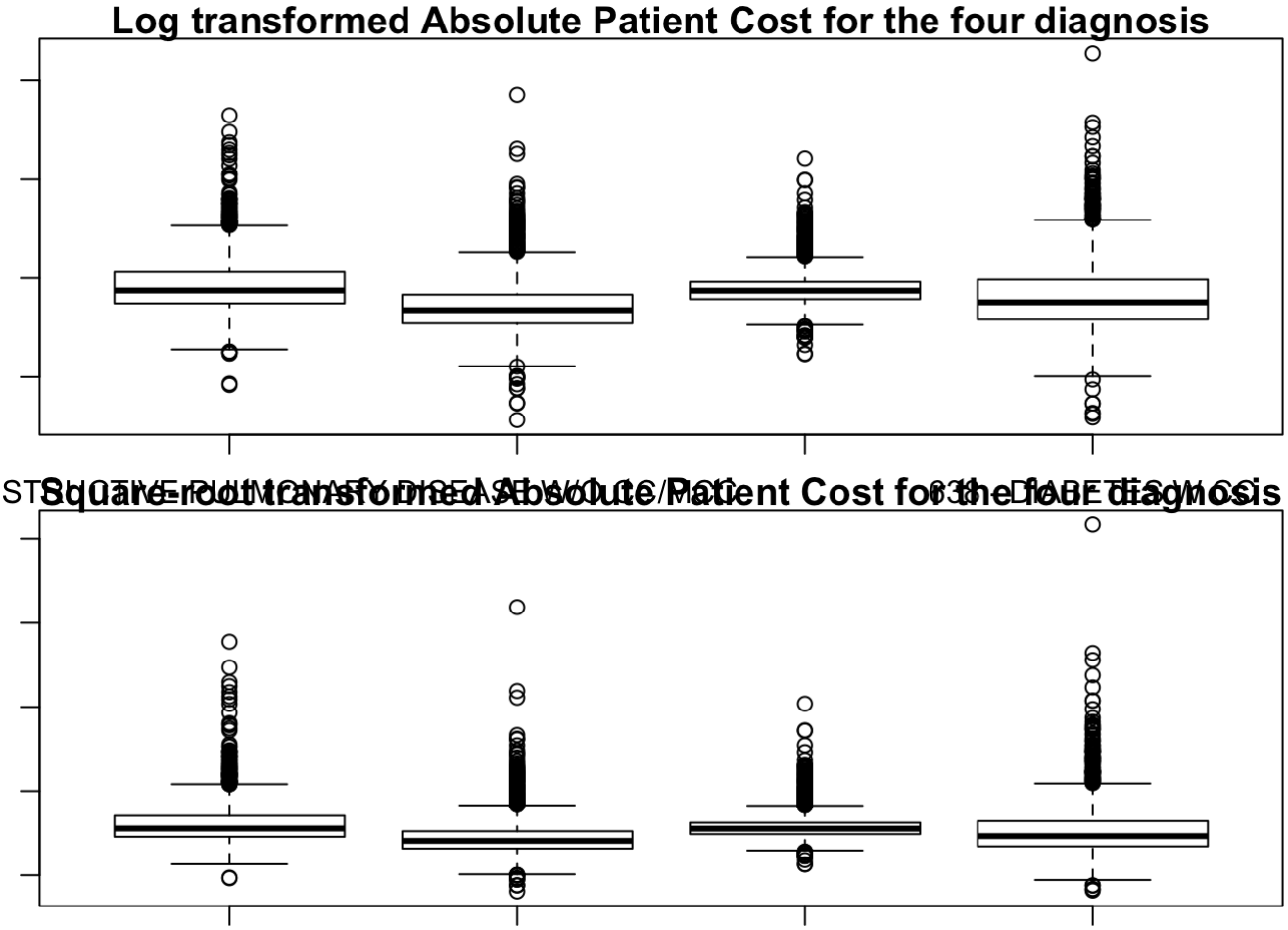
3.1 Analysing the absolute cost for Patients of distinct urbanity, region and diagnosis

```
boxplot(dff$PatientPays~dff$DRG.Definition,
        main = "Absolute Patient Cost for the four diagnosis", ylab = "Cost in dollars")
```



According to our previous decision we would take the log transform of the variable PatientPays to be able to have better visualisation. Here is the boxplot distribution after the transform and the square-root transform.

```
par(mfrow = c(2, 1))
par(mar=c(1,1,1,1))

boxplot(log(dff$PatientPays)~dff$DRG.Definition,
        main = "Log transformed Absolute Patient Cost for the four diagnosis", ylab = "l
og(Cost)")

boxplot(sqrt(dff$PatientPays)~dff$DRG.Definition,
        main = "Square-root transformed Absolute Patient Cost for the four diagnosis", y
lab = "srt(Cost)")
```



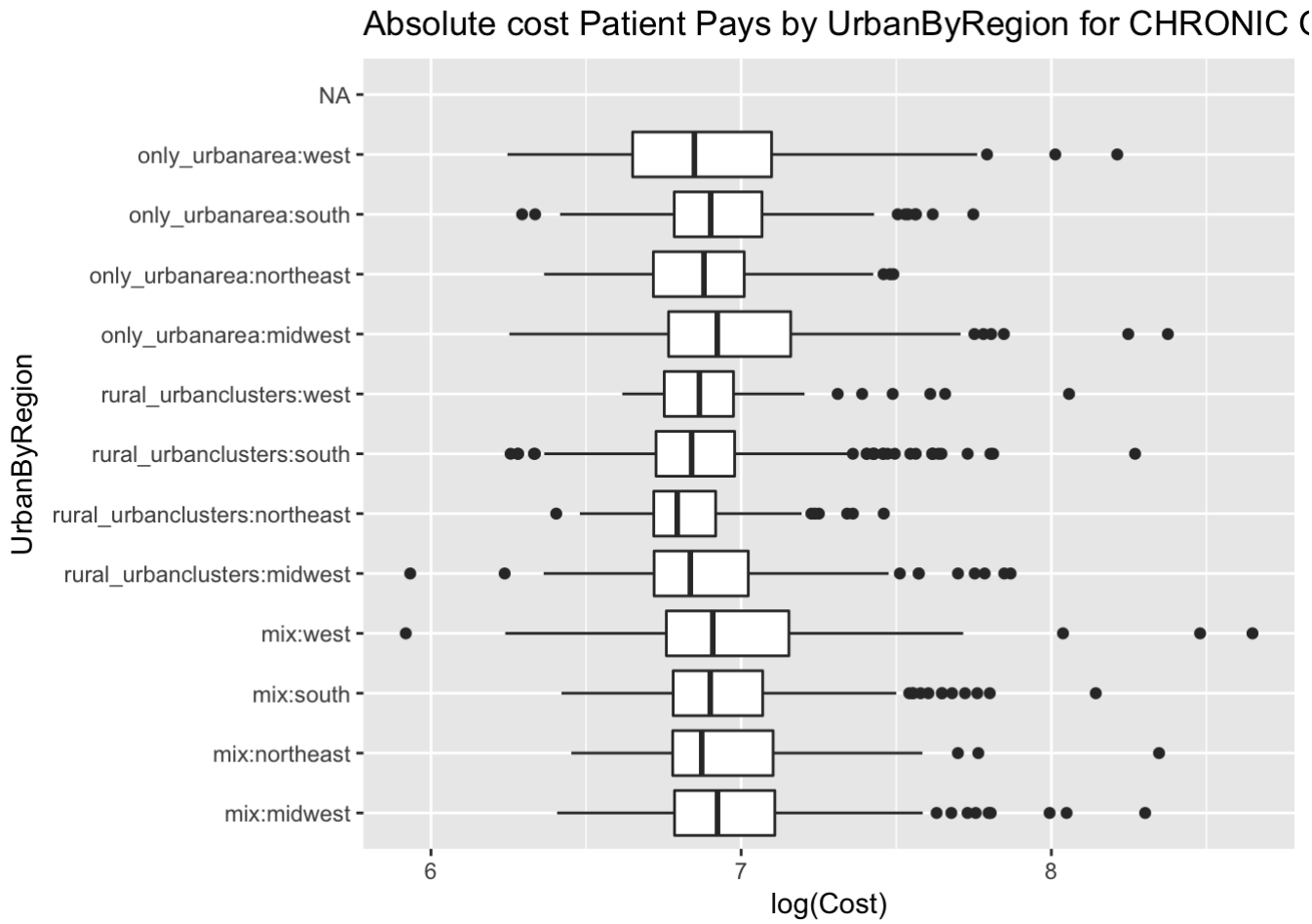
From the analysis of absolute cost paid by the patient for the 4 diseases, it is clear that the average cost for 92 - CHRONIC OBSTRUCTIVE PULMONARY DISEASE W/O CC/MCC and 536 - FRACTURES OF HIP & PELVIS W/O MCC is higher than the other two diagnosis and the range is highest for the 638 - DIABETES W CC that could cost from few hundreds to more than $10,000 to the patient depending on the personalised condition, severity and treatment. The range for 536 - FRACTURES OF HIP & PELVIS W/O MCC is the shortest and hence we have more definite prediction for th cost that would incure upon the patient diagnosed this which might be because the standard treatment across all hospitals and all regions as well as less variants in the condition of the patient.

Since we realise that the cost is highly distinct for the four different diagnosis, we would proceed to analyse the distribution by the urban and reional factors for the four conditions seperately. Let's have a look at how the absolute cost that the patient has to pay vary by region and urbanity for our 4 cases.

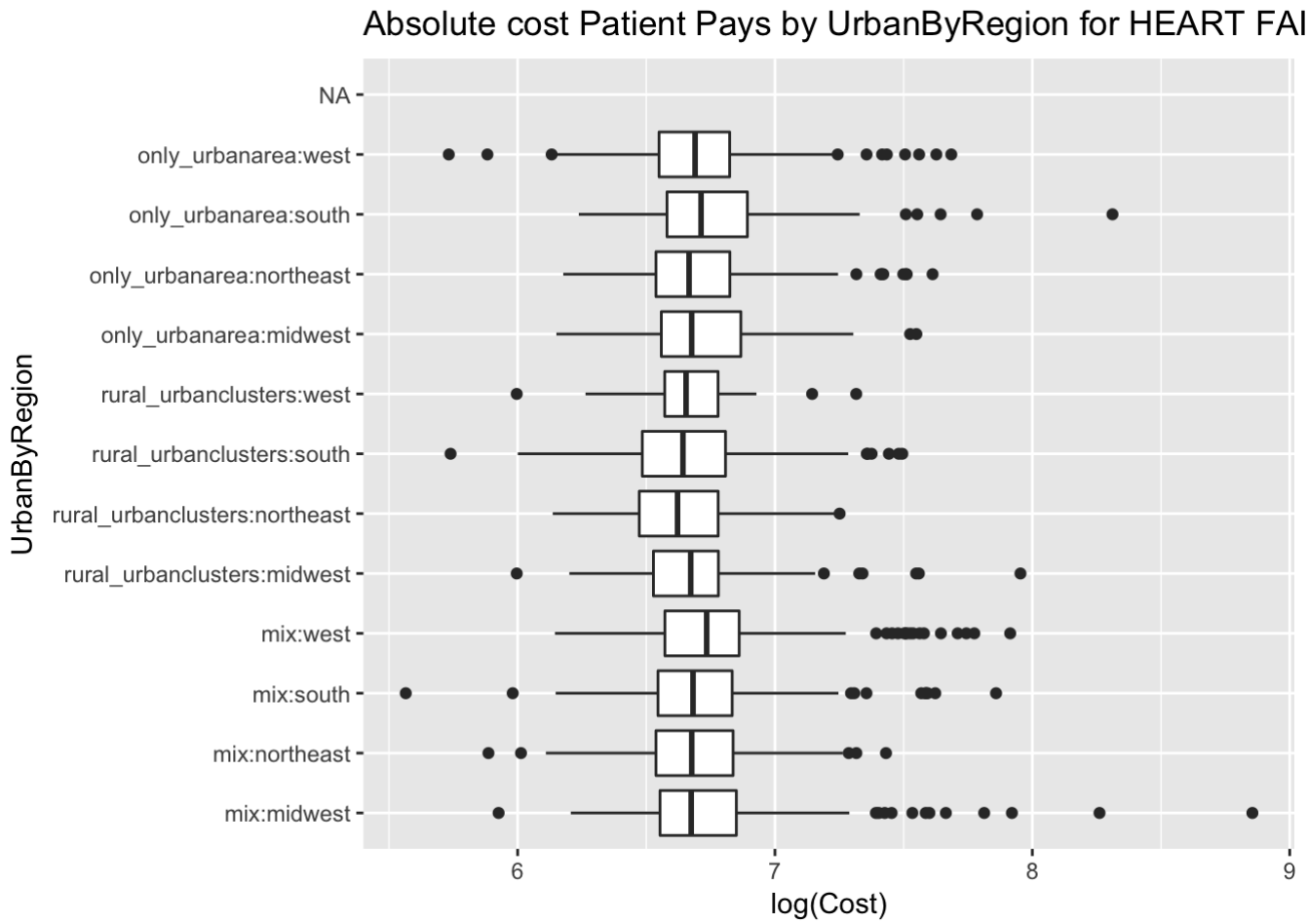
3.1.1 Box Plots for PatientPays

```
#par(mfrow = c(4, 1))
for (i in c(1:4)) {
  new_dataframe <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  print(ggplot(data = new_dataframe)+geom_boxplot(aes(y = log(new_dataframe$PatientPay
s), x =  new_dataframe$UrbanByRegions))+labs(title = paste0("Absolute cost Patient Pays
 by UrbanByRegion for ",str_sub(levels(dff$DRG.Definition)[i], start = 7 )), y = paste(
"log(Cost)"), x = "UrbanByRegion")+ coord_flip())

  }
```

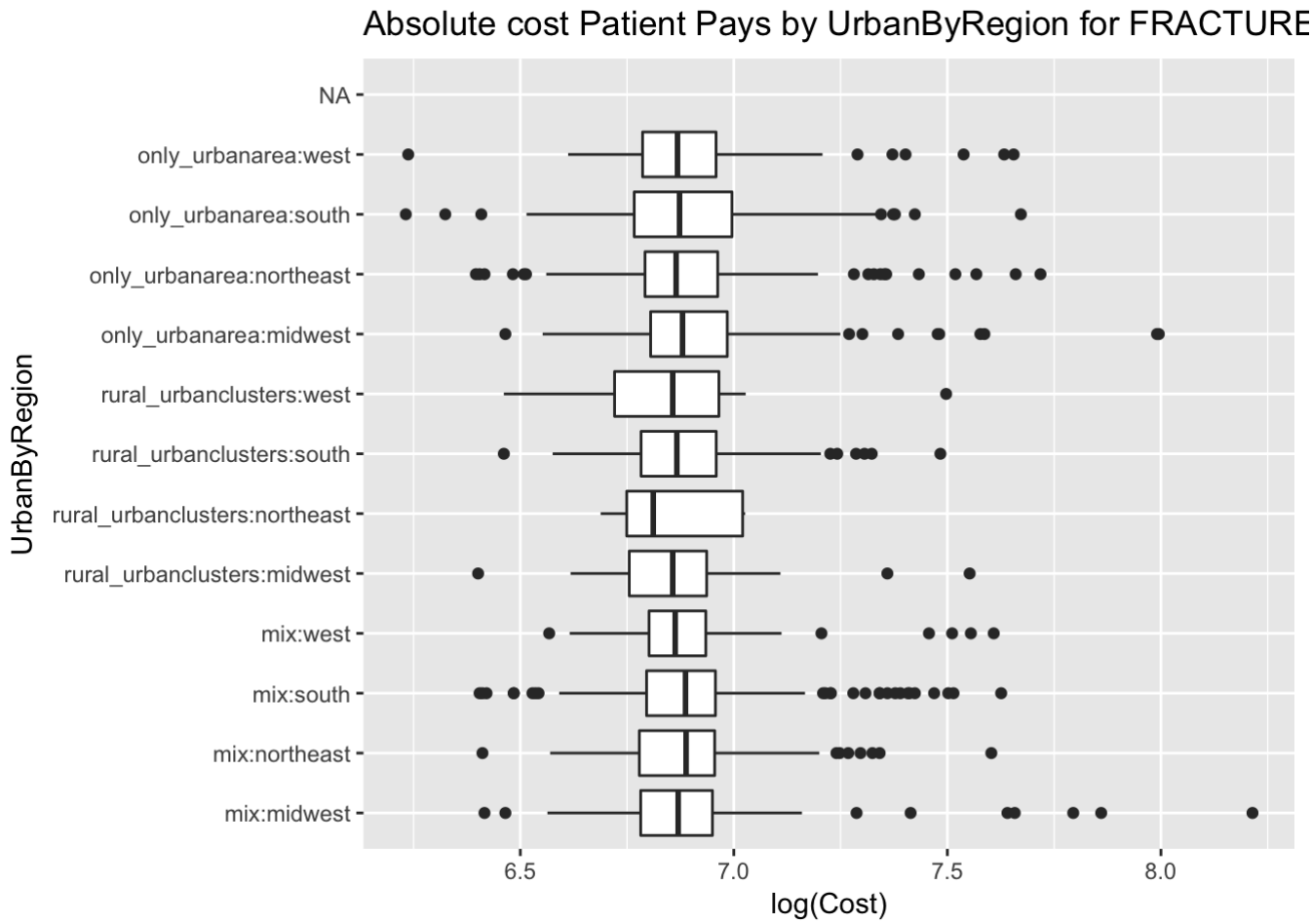
```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```



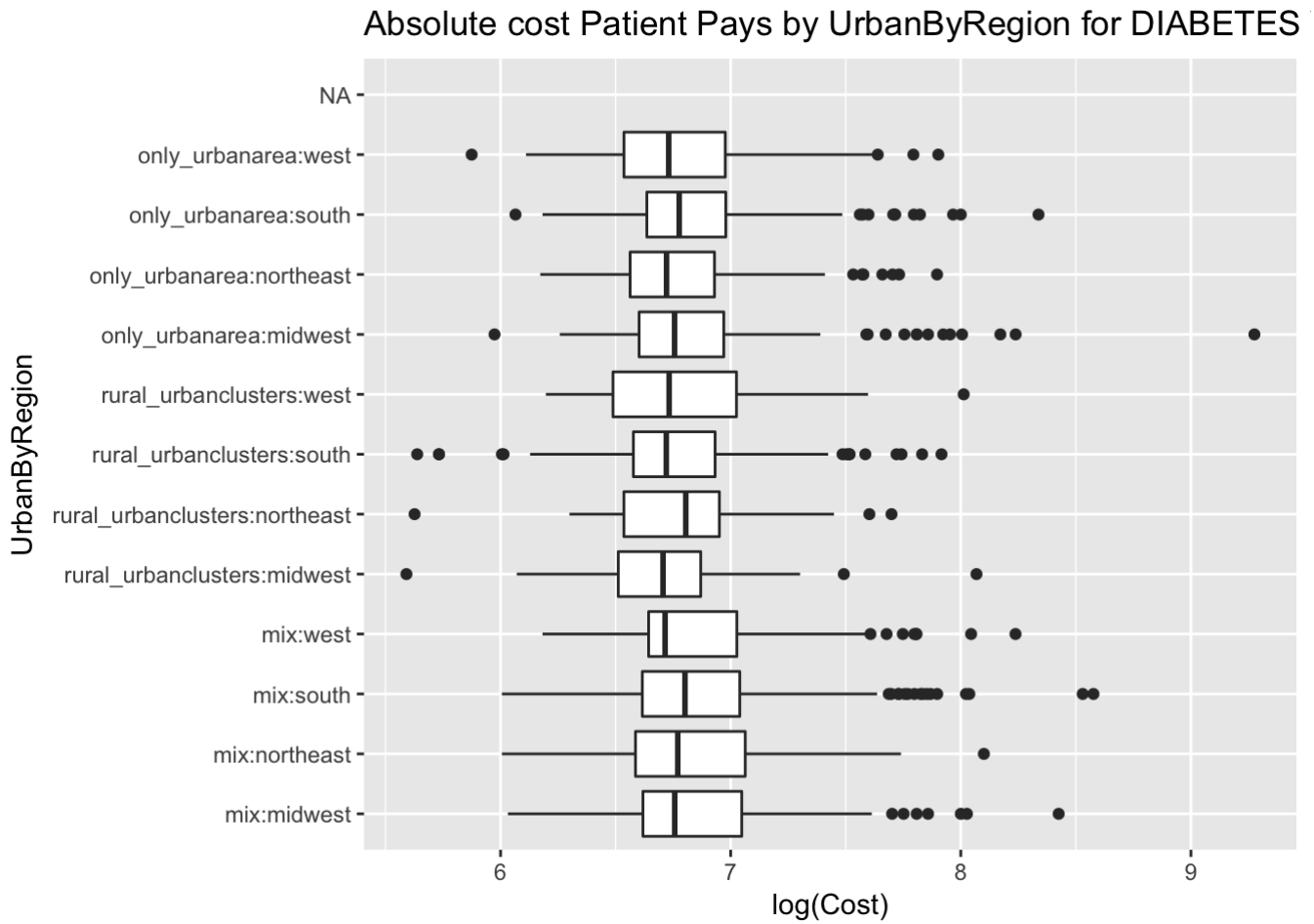

```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```



Observations : 1. Chronic Obstructive Pulmonary Disease (COPD) : We observe that although the median cost is almost similar for all the groups of Urbanity and regions, we find that areas with a mix of Urbanised Areas and Rural(and perhaps also Urban Clusters) in the zipcode have higher whiskers (cost at 1.5 IQR) than the other areas and the west region has a slightly lower median values than the other regions in all the kinds of urban settings.

2. Heart failure : Unlike the case of COPD, for heart faiures we observe that although the median cost is almost similar for all the groups of Urbanity and regions, the areas with combination of rural and Urban Clusters in the zipcode have lower cost especially at 1.5 IQR than the other areas.
3. Hip/Pelvis fractures : While the median is approximately the same, with an exception of lower median in the rural and Urban Clusters in the zipcode of North Eastern region. Also, the range covered by the areas of rural and Urban Clusters in the zipcode of all region is lower compared to the rest of urbanity levels for this diagnosis.
4. Diabetes : The median is approximately the same for all 12 categories, except lower median for mix of Urbanized Areas and Rural (and perhaps also Urban Clusters) in the zipcode areas of West region.

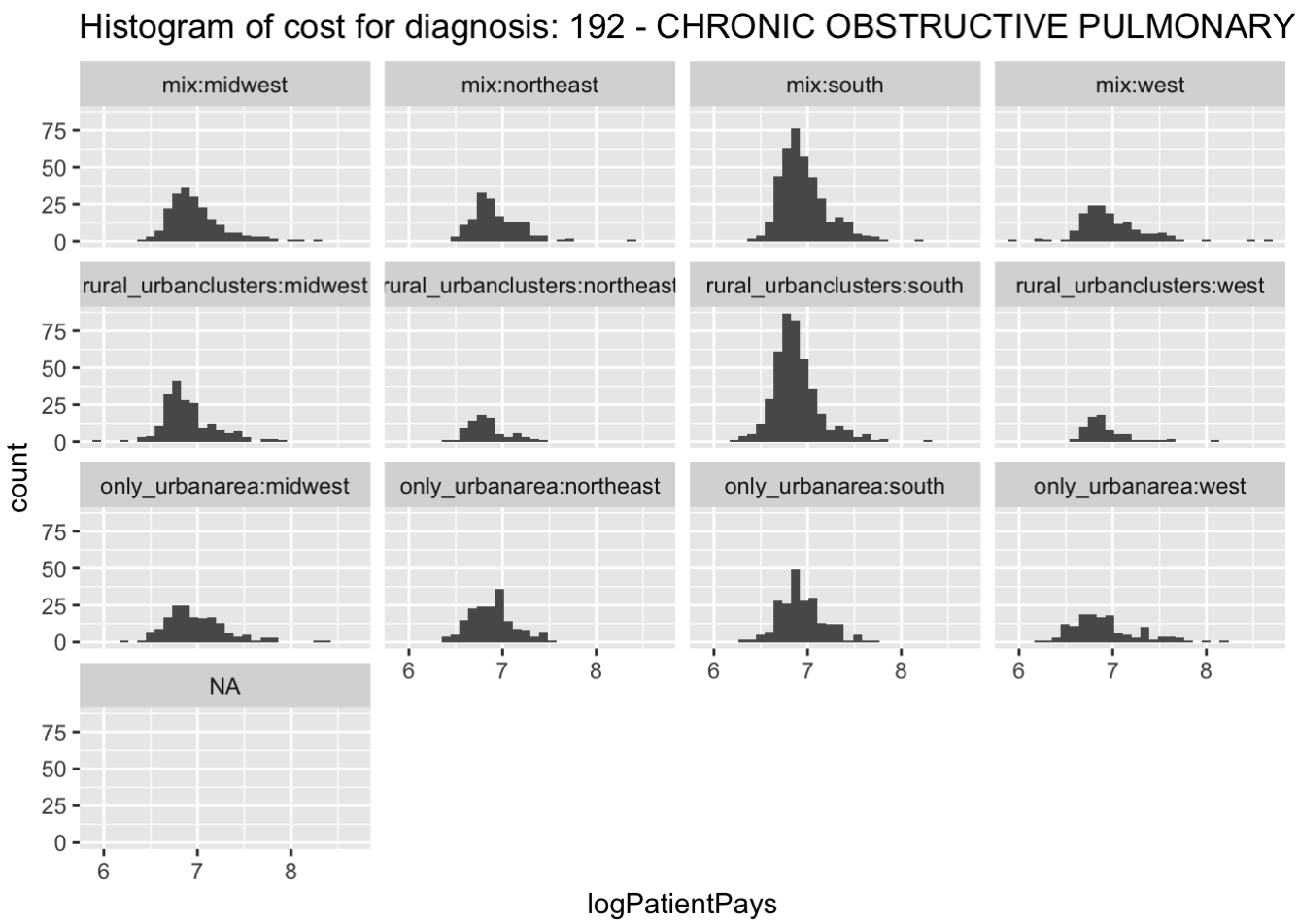
3.1.2 Histograms for PatientPays

Although we could get some idea on how the distribution of the cost patient pays in each disease vary with the regions but it was from a very high level and we donot yet have much idea about the actual distribution, its shape and other characteristics. Lets take a look at the histograms to do indepth study of their distributions.

```
for (i in c(1:4)) {
  title_abs <- paste0("Histogram of cost for diagnosis: ", levels(dff$DRG.Definition)
[i])
  list_wrt_diagnosis<-dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i],]
  list_wrt_diagnosis[, 16] <- log(list_wrt_diagnosis$PatientPays)
  names(list_wrt_diagnosis)[16] <- "logPatientPays"
  print(ggplot(data = list_wrt_diagnosis)+geom_histogram(aes(x = logPatientPays))+facet_
wrap(~UrbanByRegions)+labs(title = title_abs))
}
```

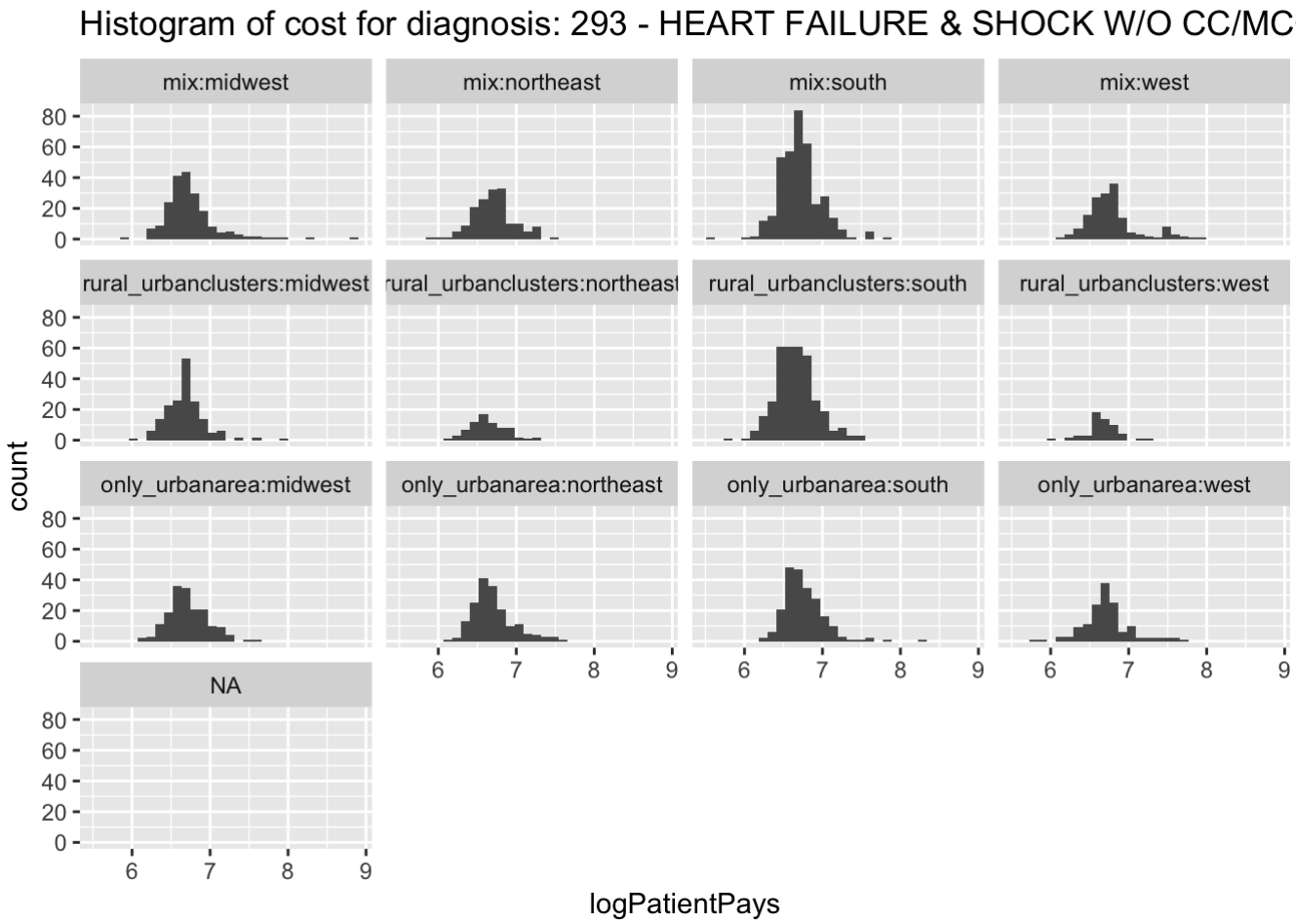
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



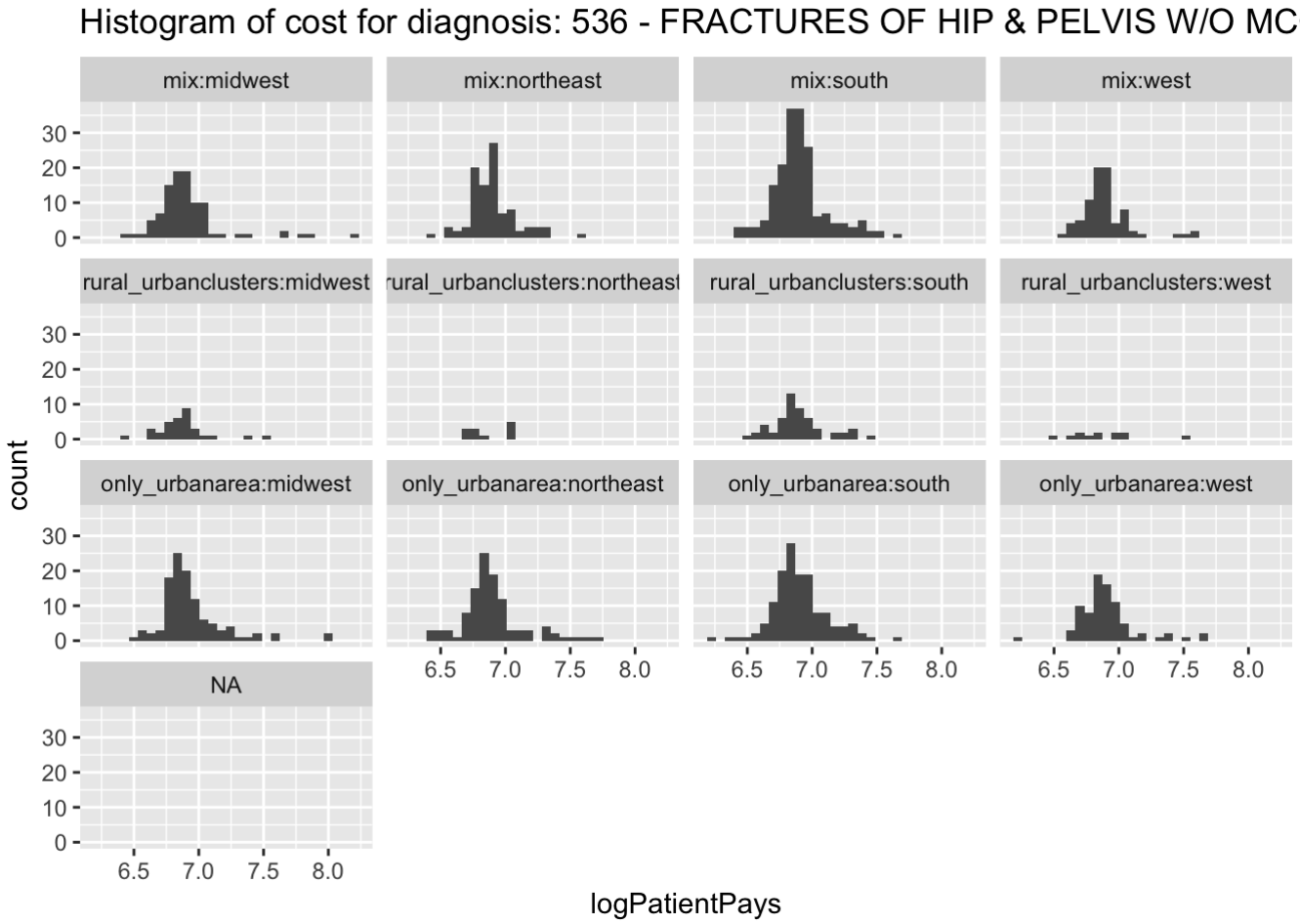

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



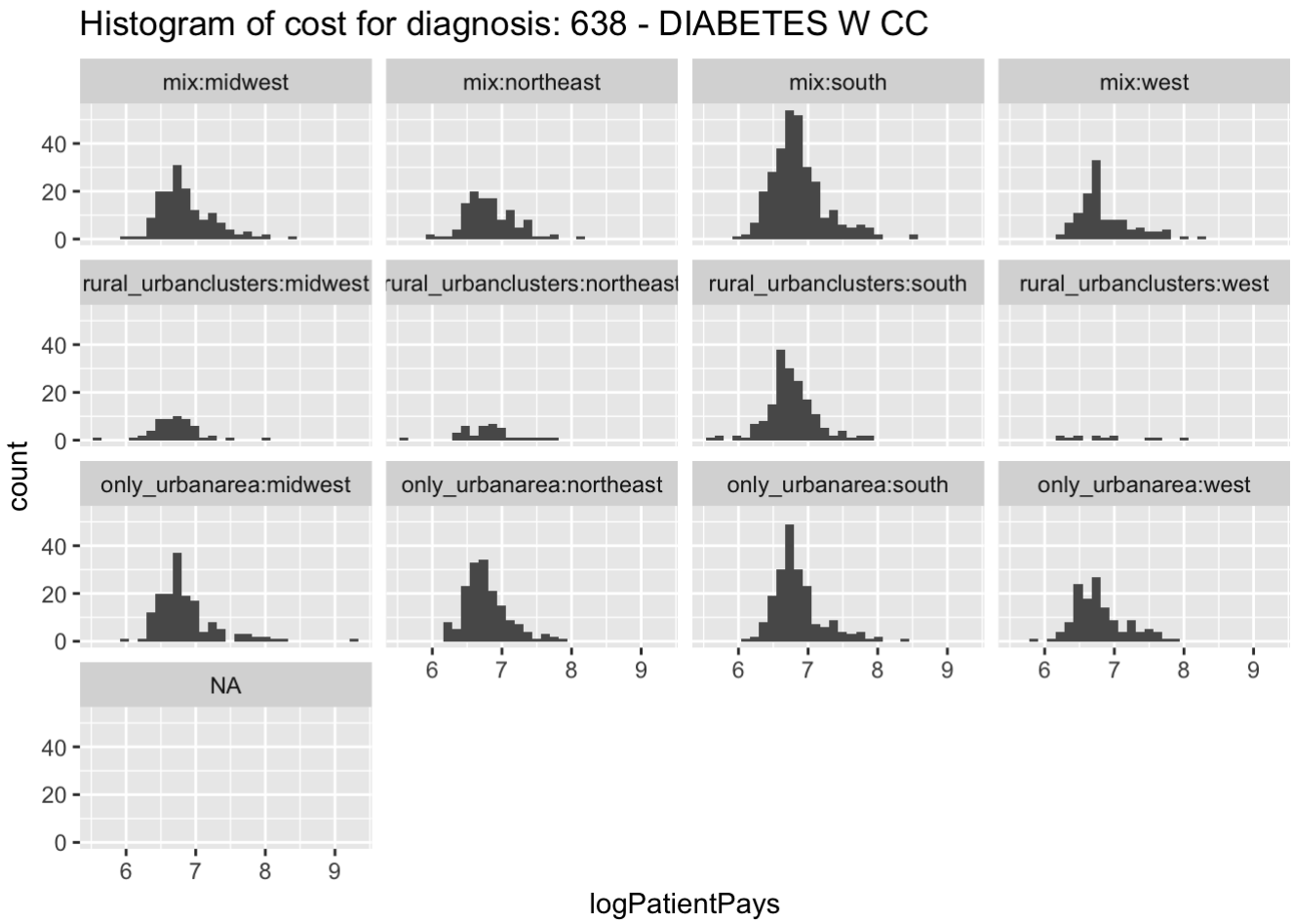

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



Observations : 1. Chronic Obstructive Pulmonary Disease (COPD) : We observe that although the median cost is almost similar for all the groups of Urbanity and regions, we find that areas with a mix of Urbanised Areas and Rural(and perhaps also Urban Clusters) in the zipcode have higher whiskers (cost at 1.5 IQR) than the other areas and the west region has a slightly lower median values than the other regions in all the kinds of urban settings.

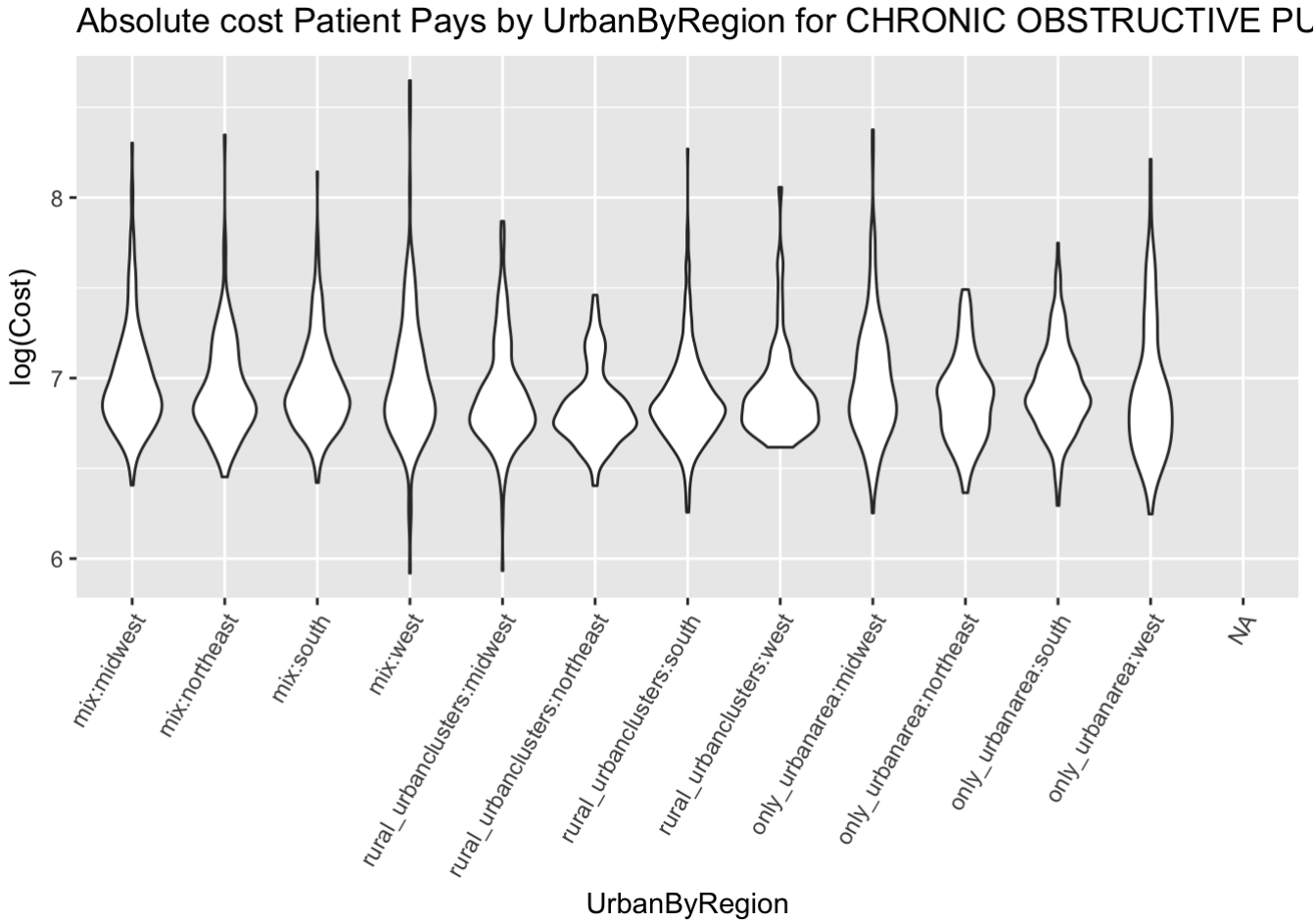
2. Heart failure : Unlike the case of COPD, for heart faiures we observe that although the median cost is almost similar for all the groups of Urbanity and regions, the areas with combination of rural and Urban Clusters in the zipcode have lower cost especially at 1.5 IQR than the other areas.
3. Hip/Pelvis fractures : While the median is approximately the same, with an exception of lower median in the rural and Urban Clusters in the zipcode of North Eastern region. Also, the range covered by the areas of rural and Urban Clusters in the zipcode of all region is lower compared to the rest of urbanity levels for this diagnosis.
4. Diabetes : The median is approximately the same for all 12 categories, except lower median for mix of Urbanized Areas and Rural (and perhaps also Urban Clusters) in the zipcode areas of West region.

3.1.3 Violin Plots for PatientPays

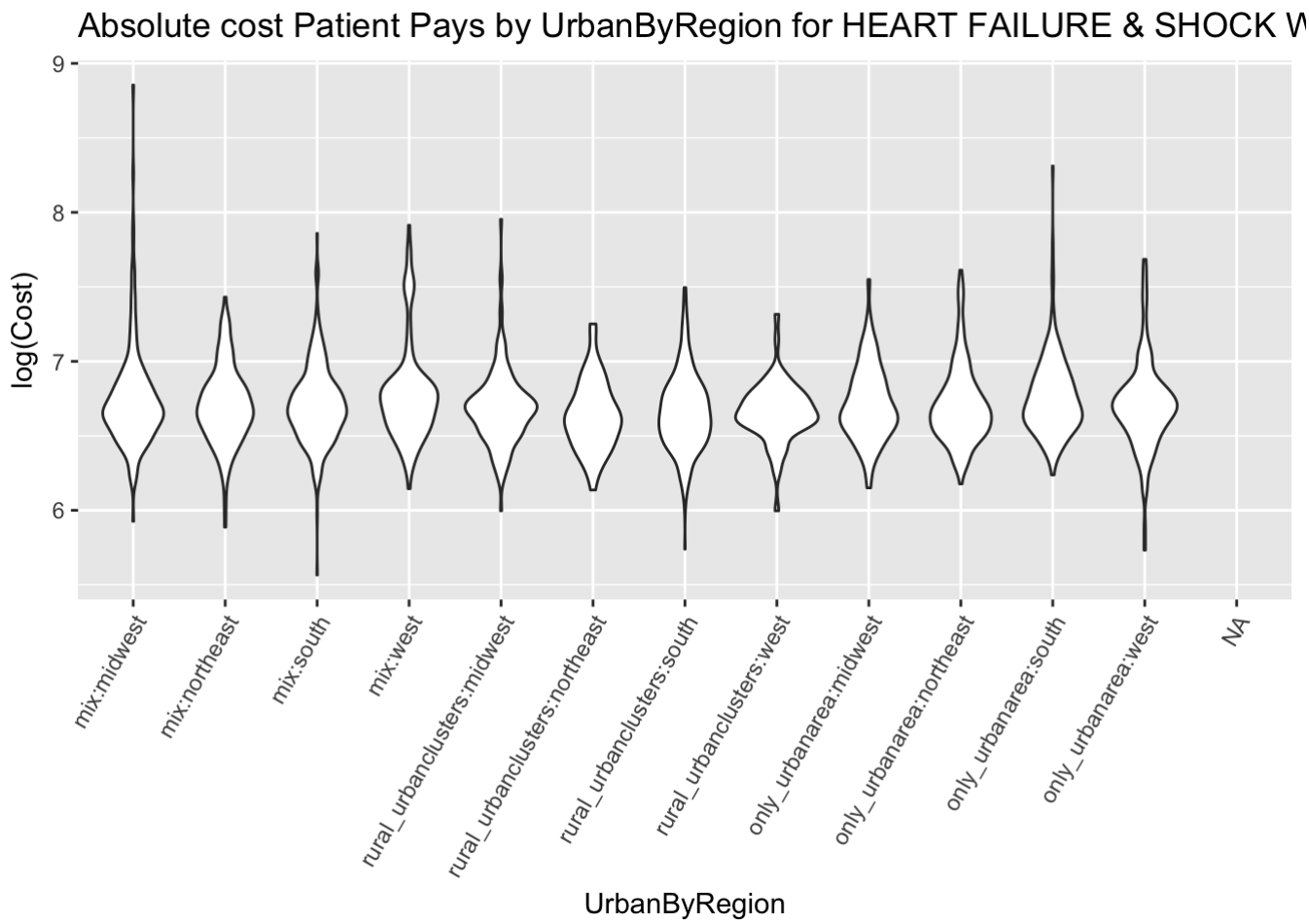
We can combine the idea of density plots and boxplots to get a ‘violin plot’. This is basically just turning the density estimate on its side and putting it next to the boxplot so that we can get finer-grain information about the distribution. Like boxplots, this allows us to compare many groups.

```
for (i in c(1:4)) {
  new_dataframe <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  print(ggplot(data = new_dataframe)+geom_violin(aes(y = log(new_dataframe$PatientPays),
 x =  new_dataframe$UrbanByRegions))+labs(title = paste0("Absolute cost Patient Pays by
 UrbanByRegion for ",str_sub(levels(dff$DRG.Definition)[i], start = 7 )), y = paste("log
(Cost)"), x = "UrbanByRegion")+ theme(axis.text.x = element_text(angle = 60, hjust = 1
)))

  }
```

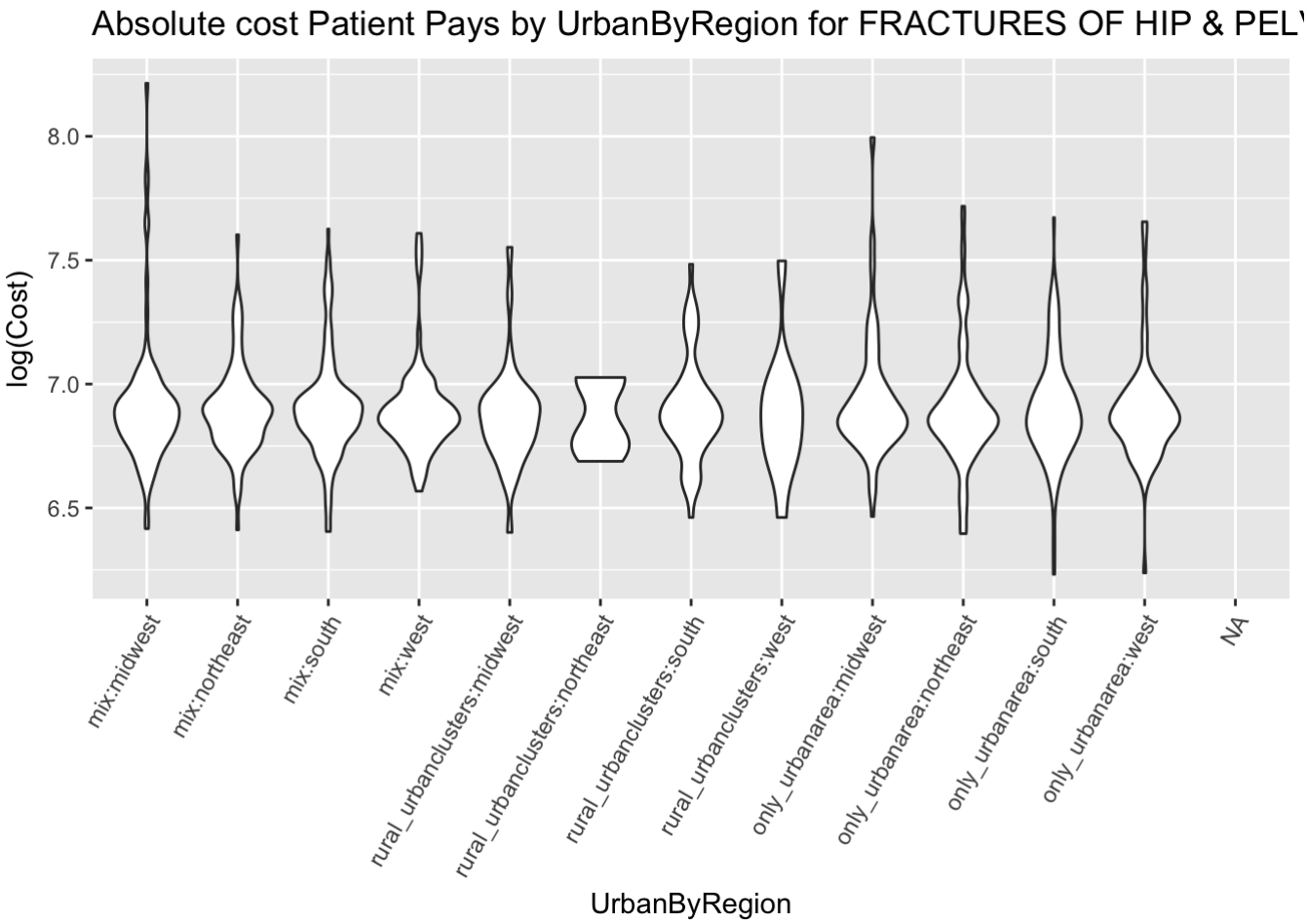
```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```



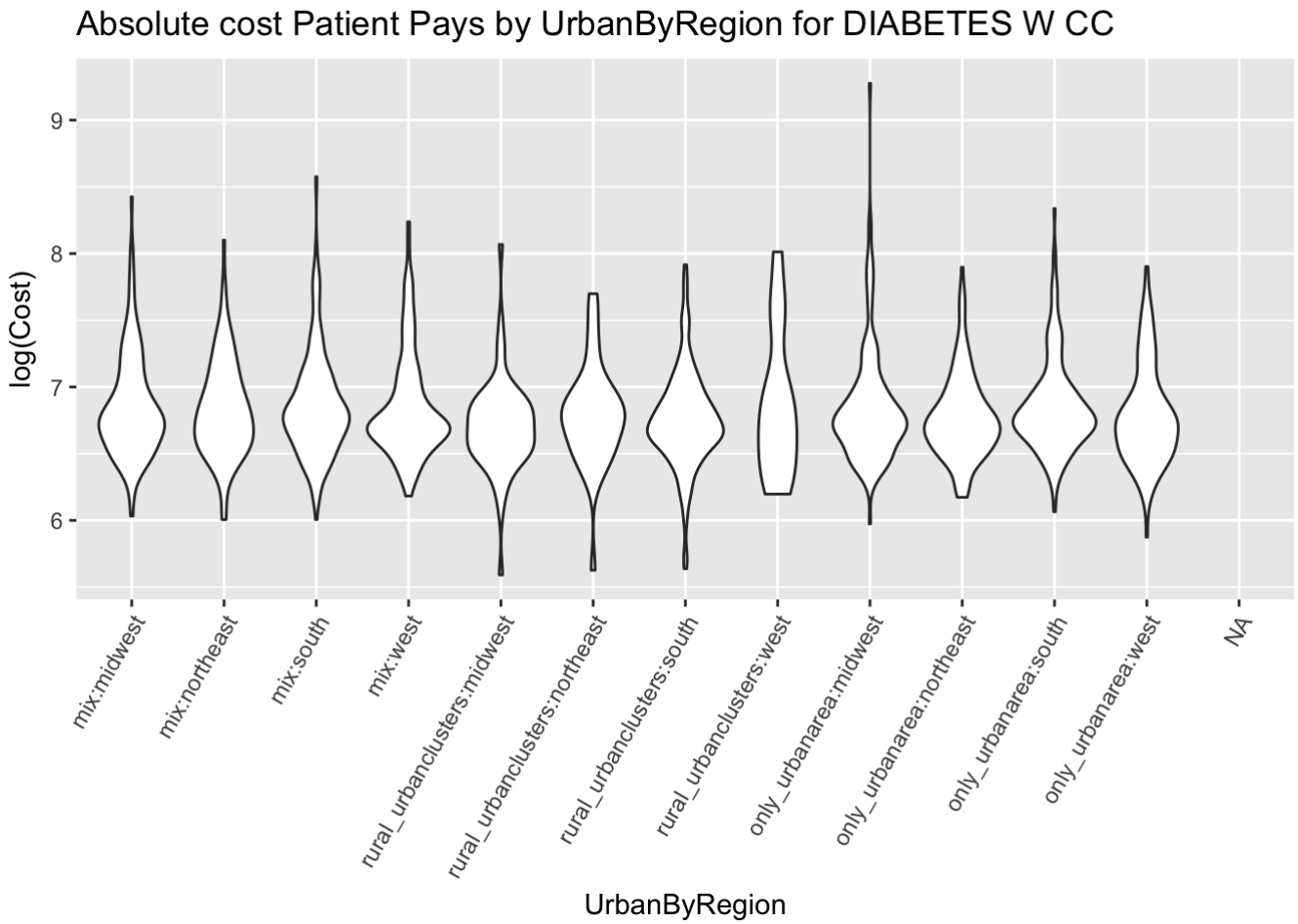

```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```

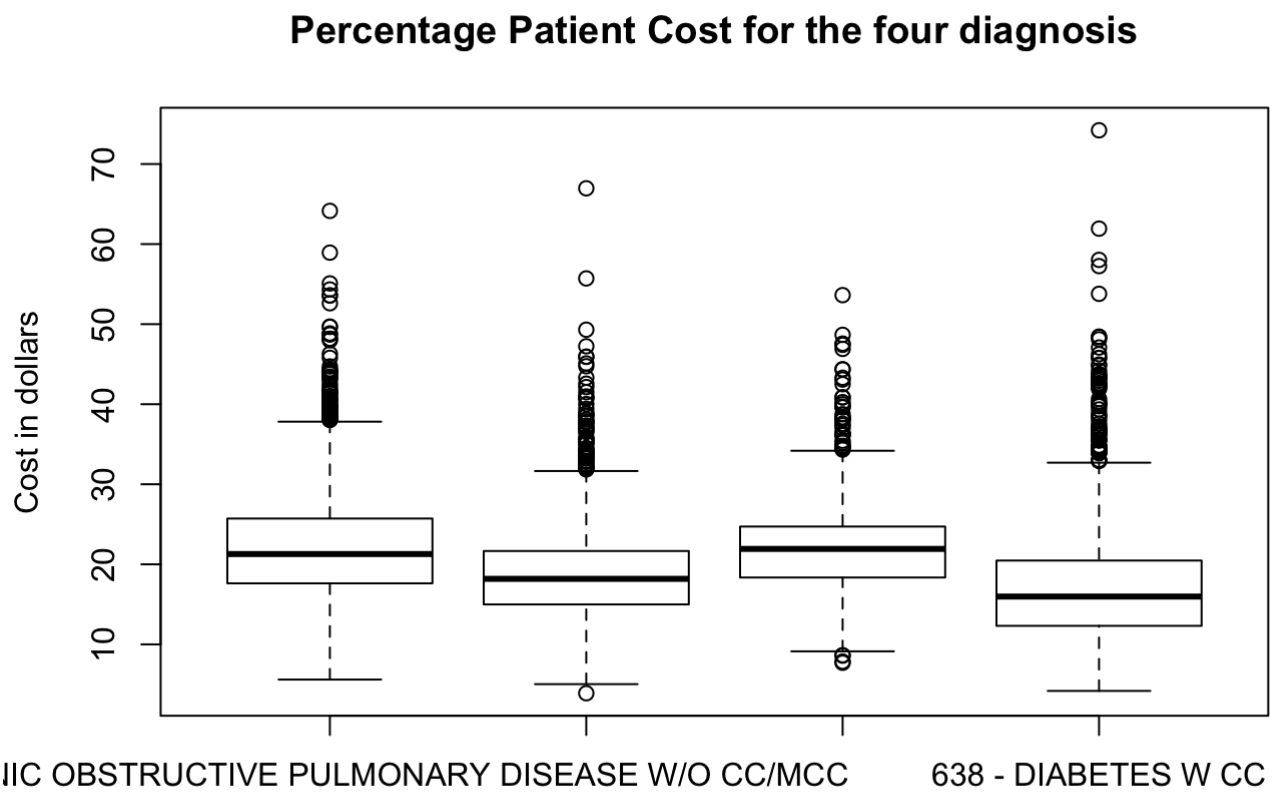



```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```



3.2 Analysing the percentage bill paid by Patients of distinct urbanity, region and diagnosis

```
boxplot(dff$PctPatientPays~dff$DRG.Definition,
        main = "Percentage Patient Cost for the four diagnosis", ylab = "Cost in dollar
s")
```



According to our previous decision we would not take the log transform of the variable PctPatientPays as the distribution is by default well spread out.

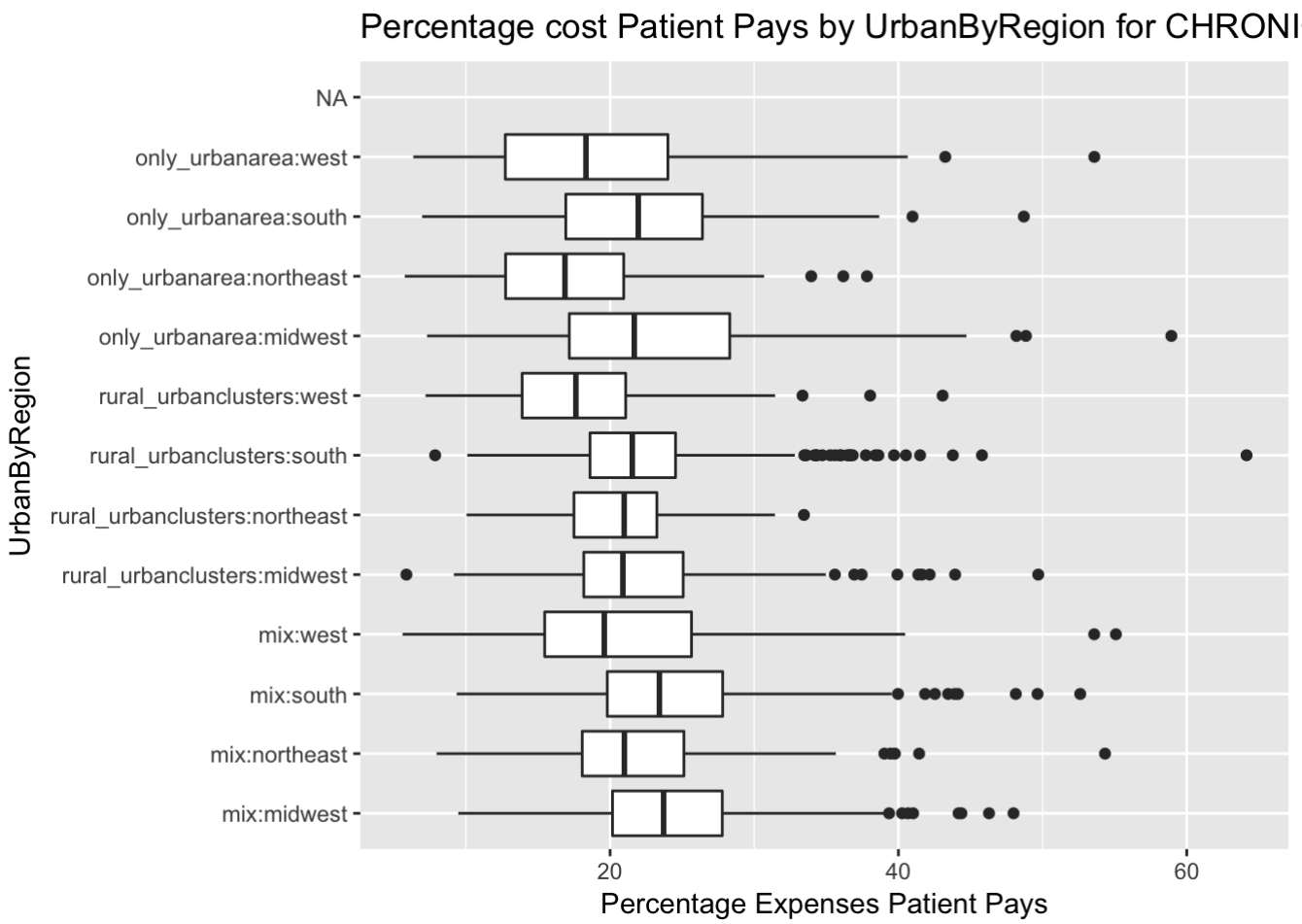
From the analysis of percentage cost paid by the patient for the 4 diseases, it is clear that the average cost for 92 - CHRONIC OBSTRUCTIVE PULMONARY DISEASE W/O CC/MCC and 536 - FRACTURES OF HIP & PELVIS W/O MCC is higher than the other two diagnosis and the range is highest for the 638 - DIABETES W CC that could be from couple of percent to more than 70% to the patient depending on the personalised condition, severity and treatment. The range for 536 - FRACTURES OF HIP & PELVIS W/O MCC is the shortest and hence we have more definite prediction for the percentage cost that would incure upon the patient diagnosed.

Since we realise that the cost is highly distinct for the four different diagnosis, we would proceed to analyse this distribution of percentage paid by the urban and regional factors for the four conditions seperately. Let's have a look at how the percentage cost that the patient has to pay vary by region and urbanity for our 4 cases.

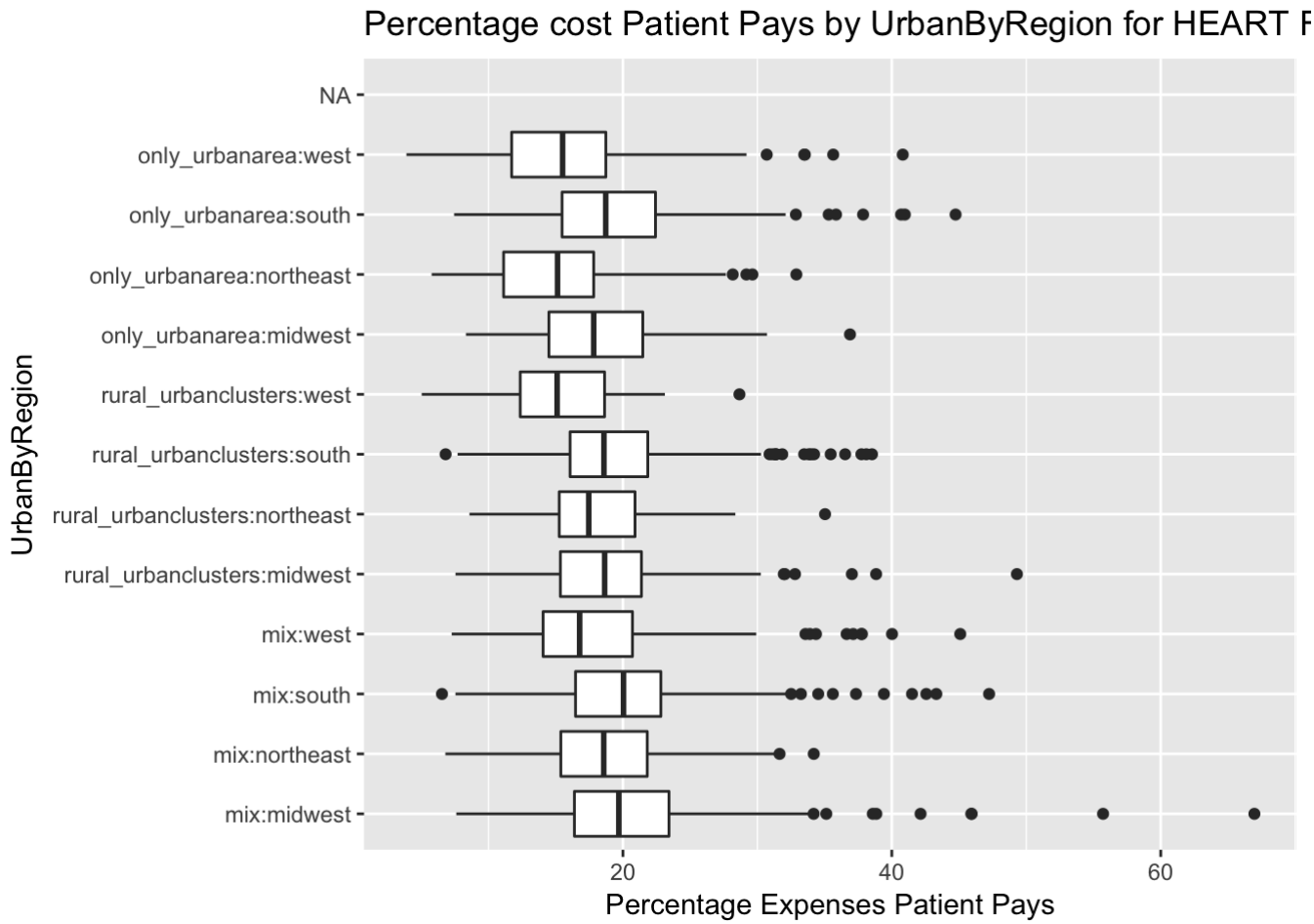
3.2.1 Box Plots for variable PctPatientPays

```
#par(mfrow = c(4, 1))
for (i in c(1:4)) {
  new_dataframe <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  print(ggplot(data = new_dataframe)+geom_boxplot(aes(y = (new_dataframe$PctPatientPay
s), x =  new_dataframe$UrbanByRegions))+labs(title = paste0("Percentage cost Patient Pay
s by UrbanByRegion for ",str_sub(levels(dff$DRG.Definition)[i], start = 7 )), y = paste(
"Percentage Expenses Patient Pays"), x = "UrbanByRegion")+ coord_flip())

  }
```

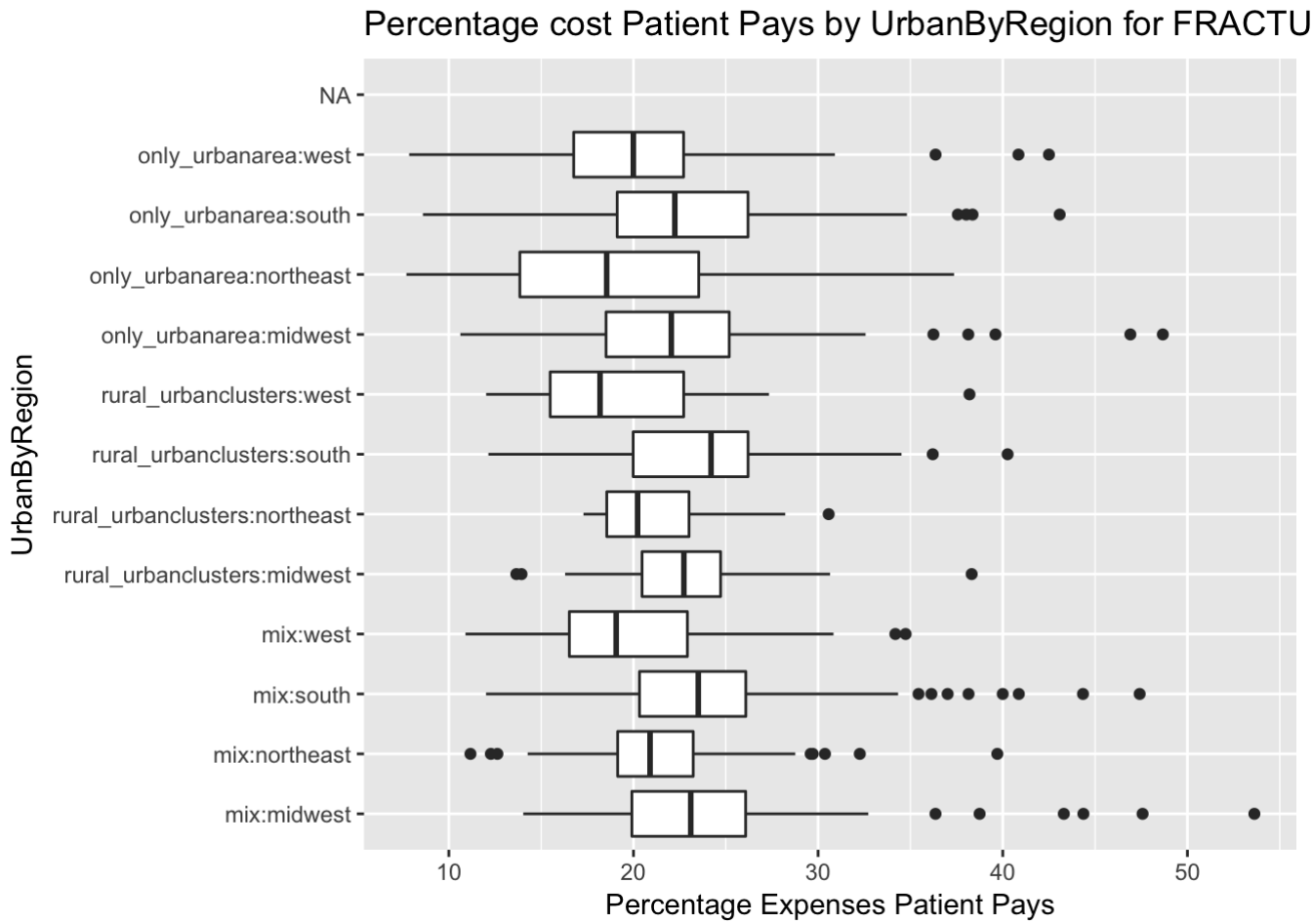
```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```



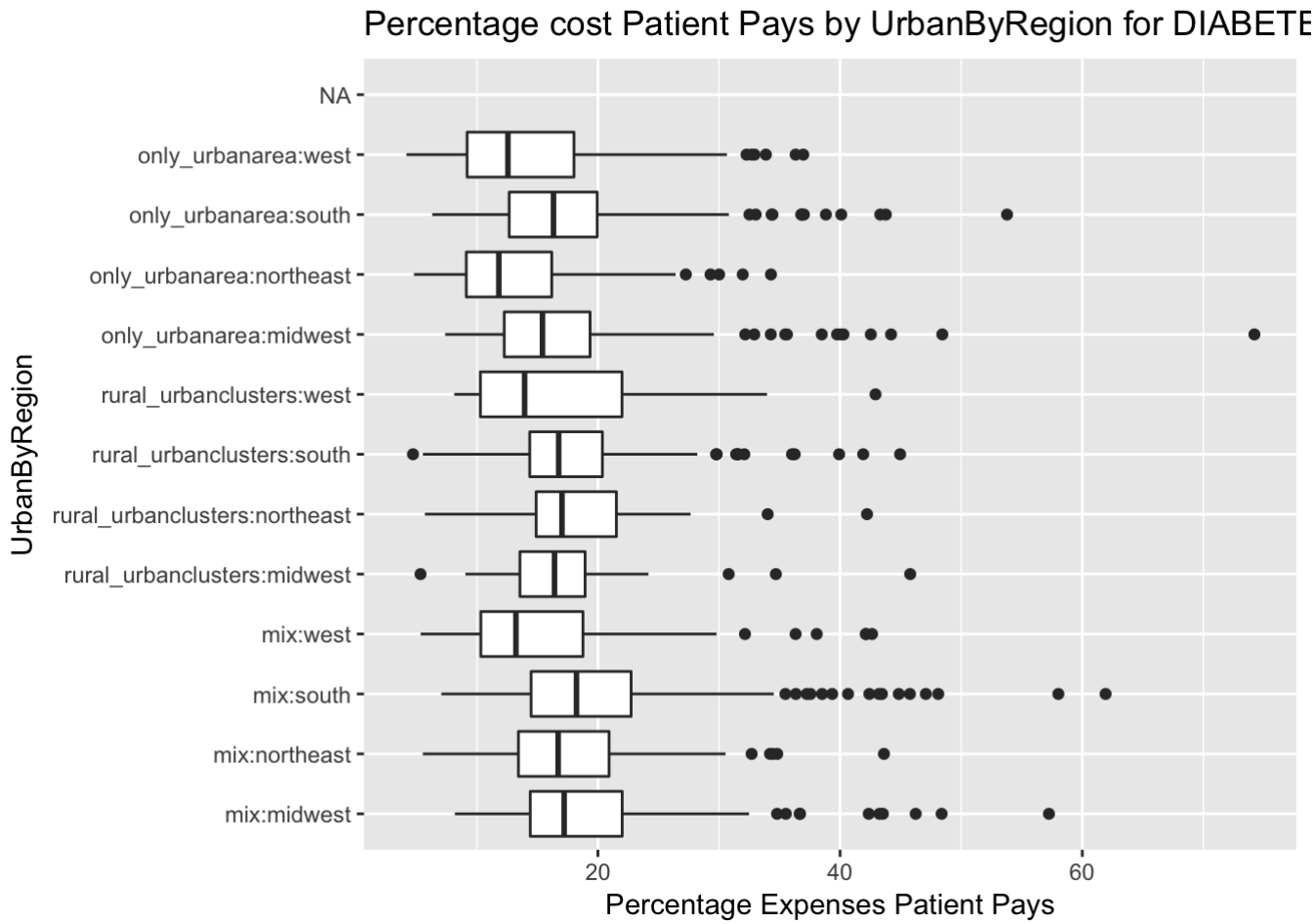

```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_boxplot).
```



Observations : 1. Chronic Obstructive Pulmonary Disease (COPD) : We observe that although the median cost is almost similar for all the groups of Urbanity and regions, we find that areas with a mix of Urbanised Areas and Rural(and perhaps also Urban Clusters) in the zipcode have higher whiskers (cost at 1.5 IQR) than the other areas and the west region has a slightly lower median values than the other regions in all the kinds of urban settings.

2. Heart failure : Unlike the case of COPD, for heart faiures we observe that although the median cost is almost similar for all the groups of Urbanity and regions, the areas with combination of rural and Urban Clusters in the zipcode have lower cost especially at 1.5 IQR than the other areas.

3. Hip/Pelvis fractures : While the median is approximately the same, with an exception of lower median in the rural and Urban Clusters in the zipcode of North Eastern region. Also, the range covered by the areas of rural and Urban Clusters in the zipcode of all region is lower compared to the rest of urbanity levels for this diagnosis.

4. Diabetes : The median is approximately the same for all 12 categories, except lower median for mix of Urbanized Areas and Rural (and perhaps also Urban Clusters) in the zipcode areas of West region.

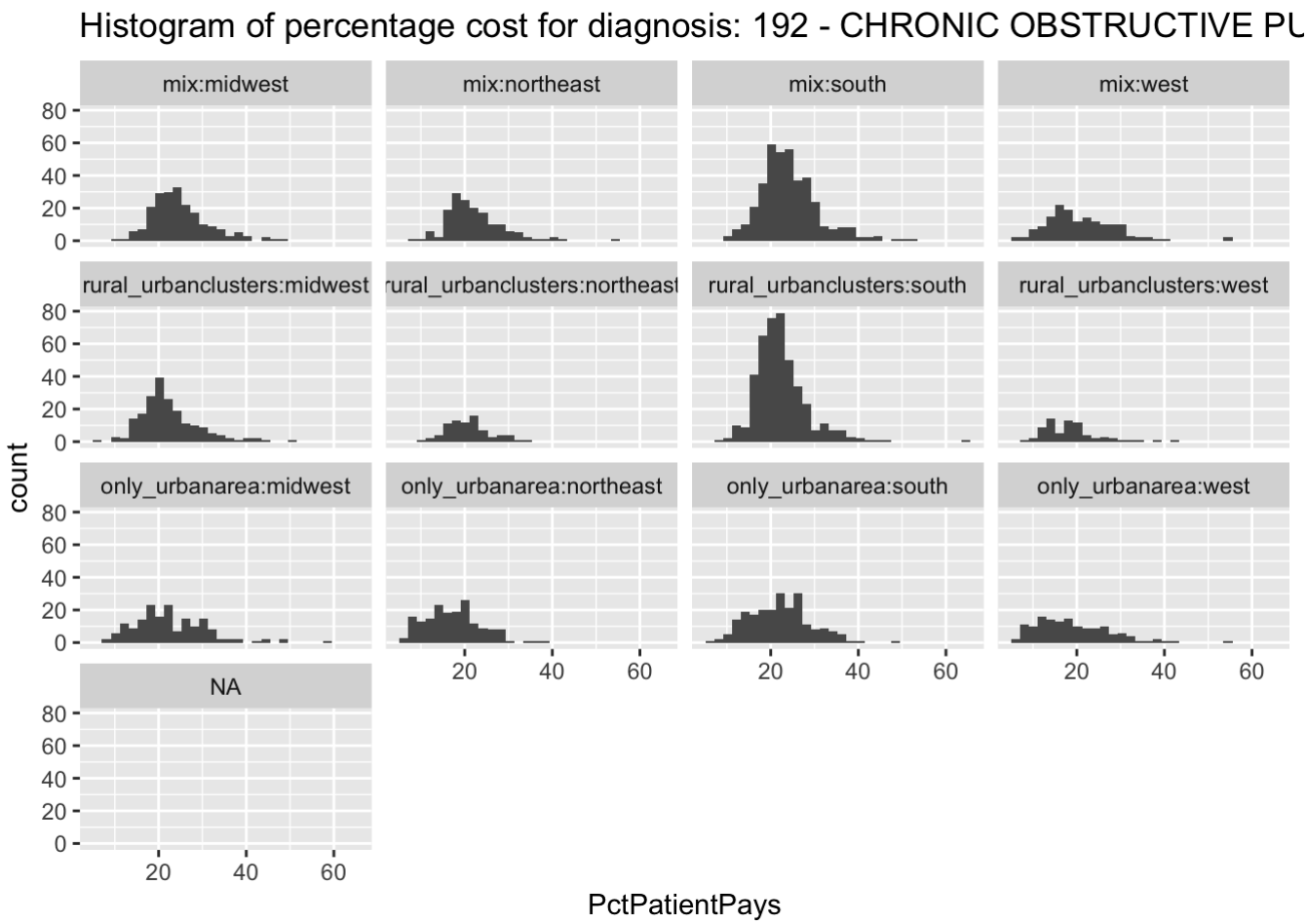
3.2.2 Histograms for variable PctPatientPays

Although we could get some idea on how the distribution of the cost patient pays in each disease vary with the regions but it was from a very high level and we donot yet have much idea about the actual distribution, its shape and other characteristics. Lets take a look at the histograms to do indepth study of their distributions.

```
for (i in c(1:4)) {
  title_abs <- paste0("Histogram of percentage cost for diagnosis: ", levels(dff$DRG.Def
inition)[i])
  list_wrt_diagnosis<-dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i],]
  print(ggplot(data = list_wrt_diagnosis)+geom_histogram(aes(x = PctPatientPays))+facet_
wrap(~UrbanByRegions)+labs(title = title_abs))
}
```

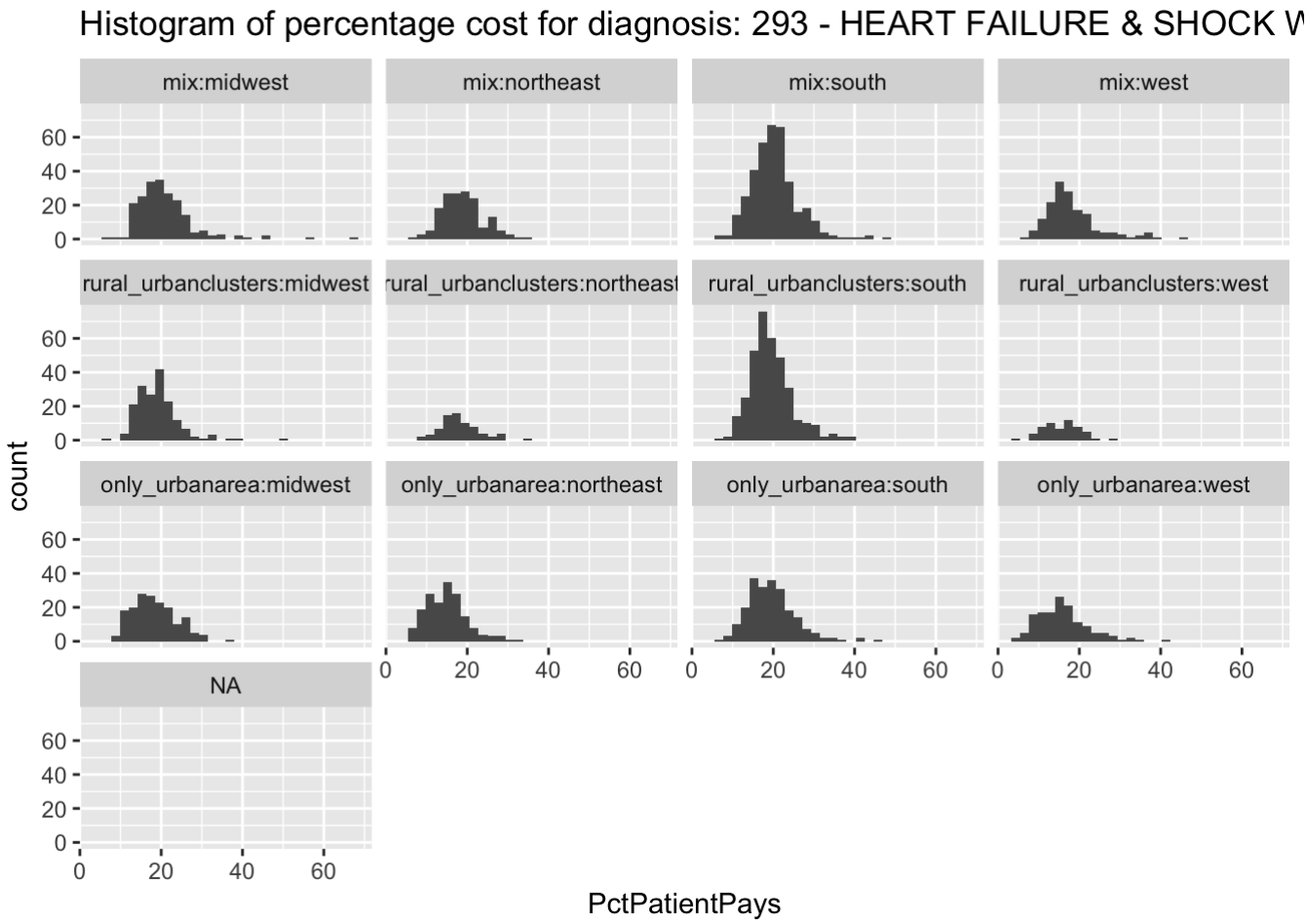
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



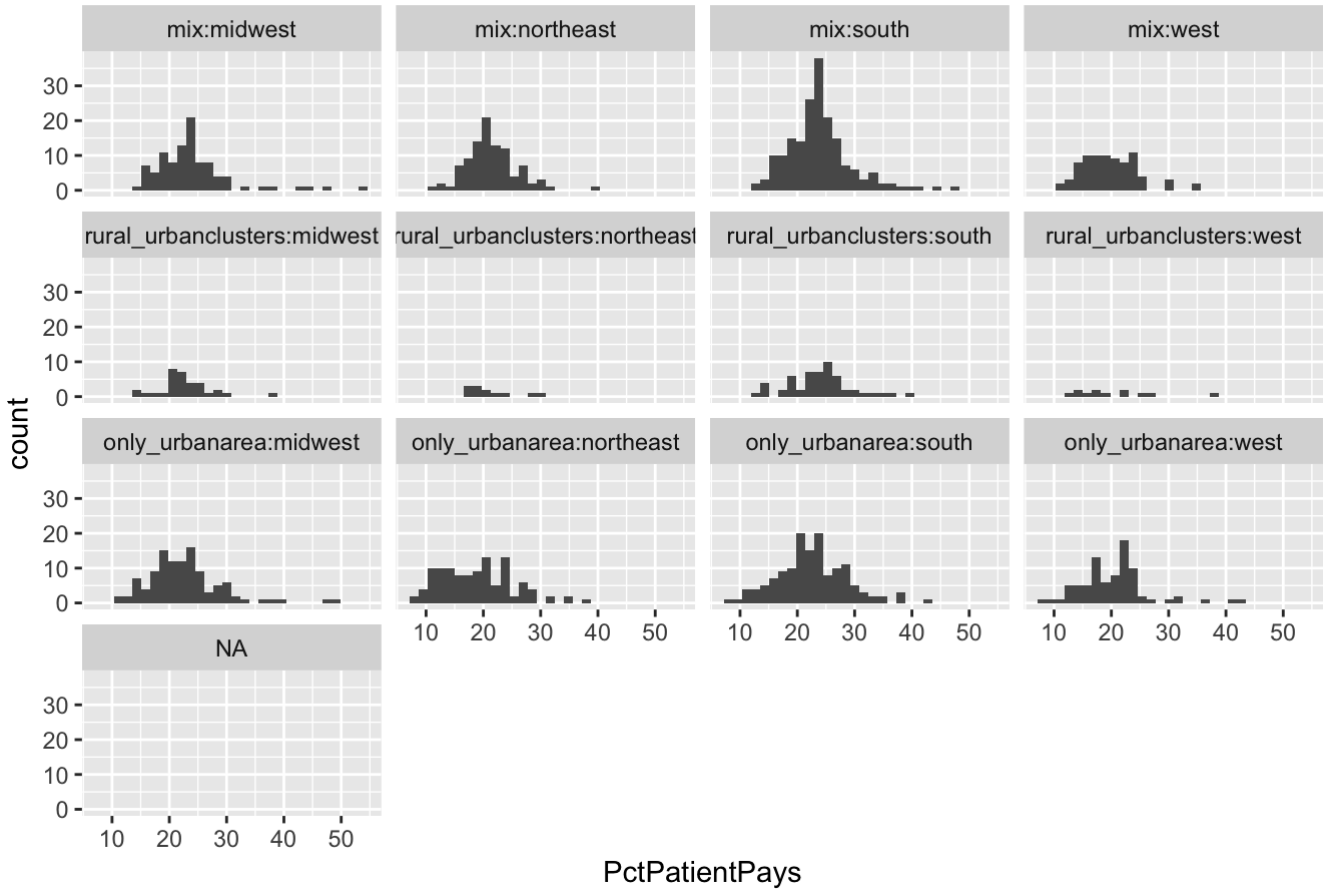

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



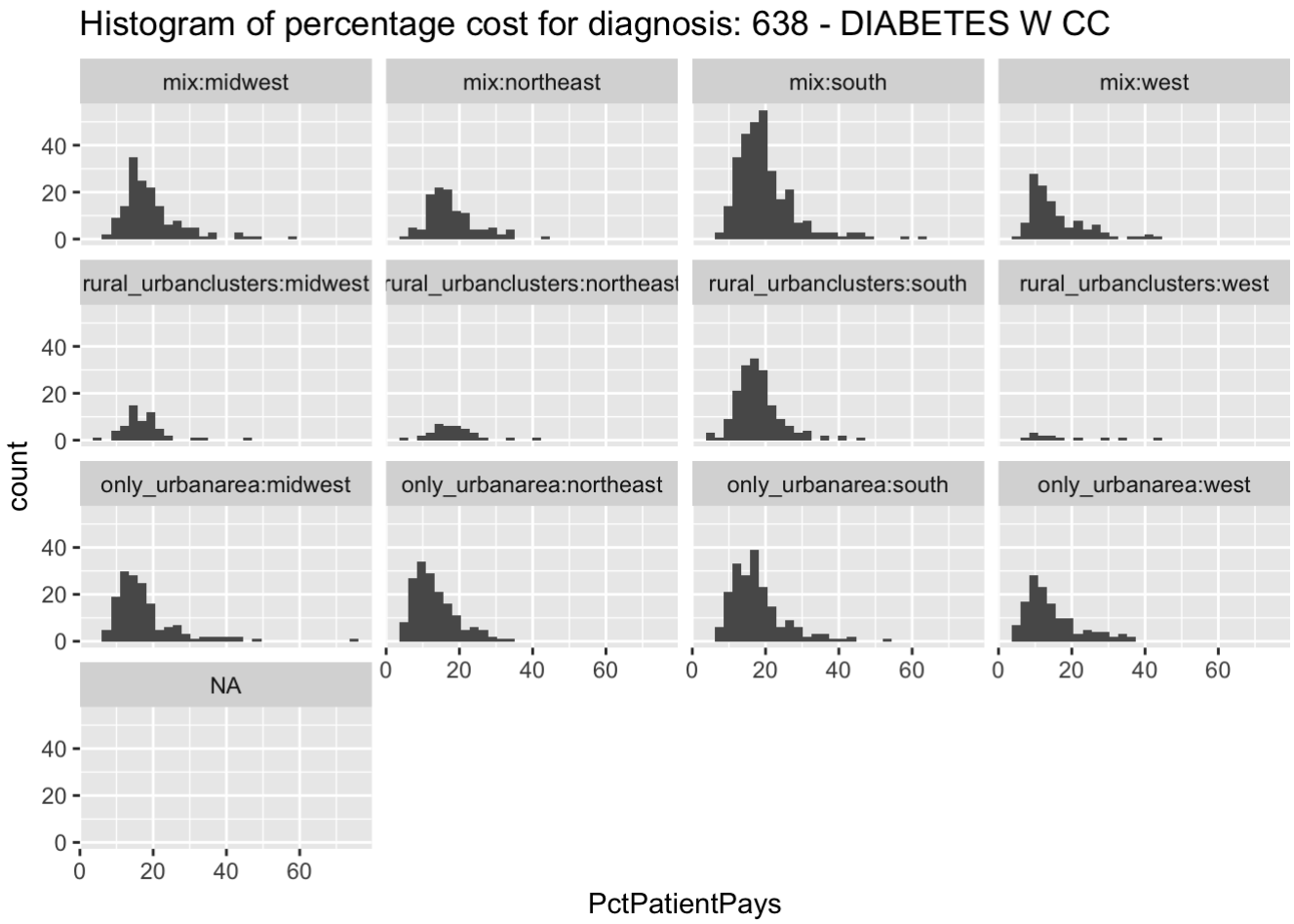

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 496 rows containing non-finite values (stat_bin).
```



Observations : 1. Chronic Obstructive Pulmonary Disease (COPD) : We observe that although the median cost is almost similar for all the groups of Urbanity and regions, we find that areas with a mix of Urbanised Areas and Rural(and perhaps also Urban Clusters) in the zipcode have higher whiskers (cost at 1.5 IQR) than the other areas and the west region has a slightly lower median values than the other regions in all the kinds of urban settings.

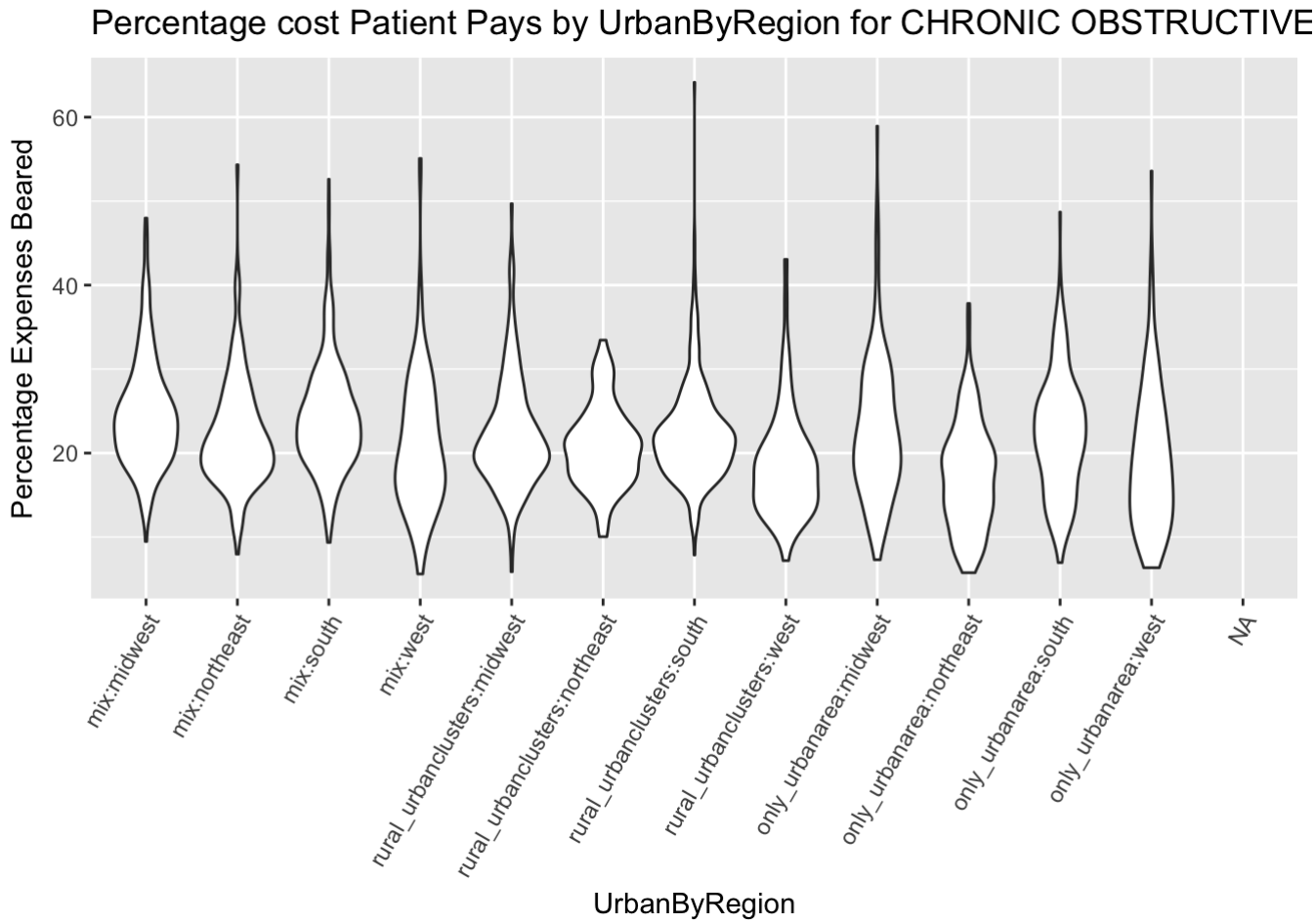
2. Heart failure : Unlike the case of COPD, for heart faiures we observe that although the median cost is almost similar for all the groups of Urbanity and regions, the areas with combination of rural and Urban Clusters in the zipcode have lower cost especially at 1.5 IQR than the other areas.
3. Hip/Pelvis fractures : While the median is approximately the same, with an exception of lower median in the rural and Urban Clusters in the zipcode of North Eastern region. Also, the range covered by the areas of rural and Urban Clusters in the zipcode of all region is lower compared to the rest of urbanity levels for this diagnosis.
4. Diabetes : The median is approximately the same for all 12 categories, except lower median for mix of Urbanized Areas and Rural (and perhaps also Urban Clusters) in the zipcode areas of West region.

3.2.3 Violin Plots for variable PctPatientPays

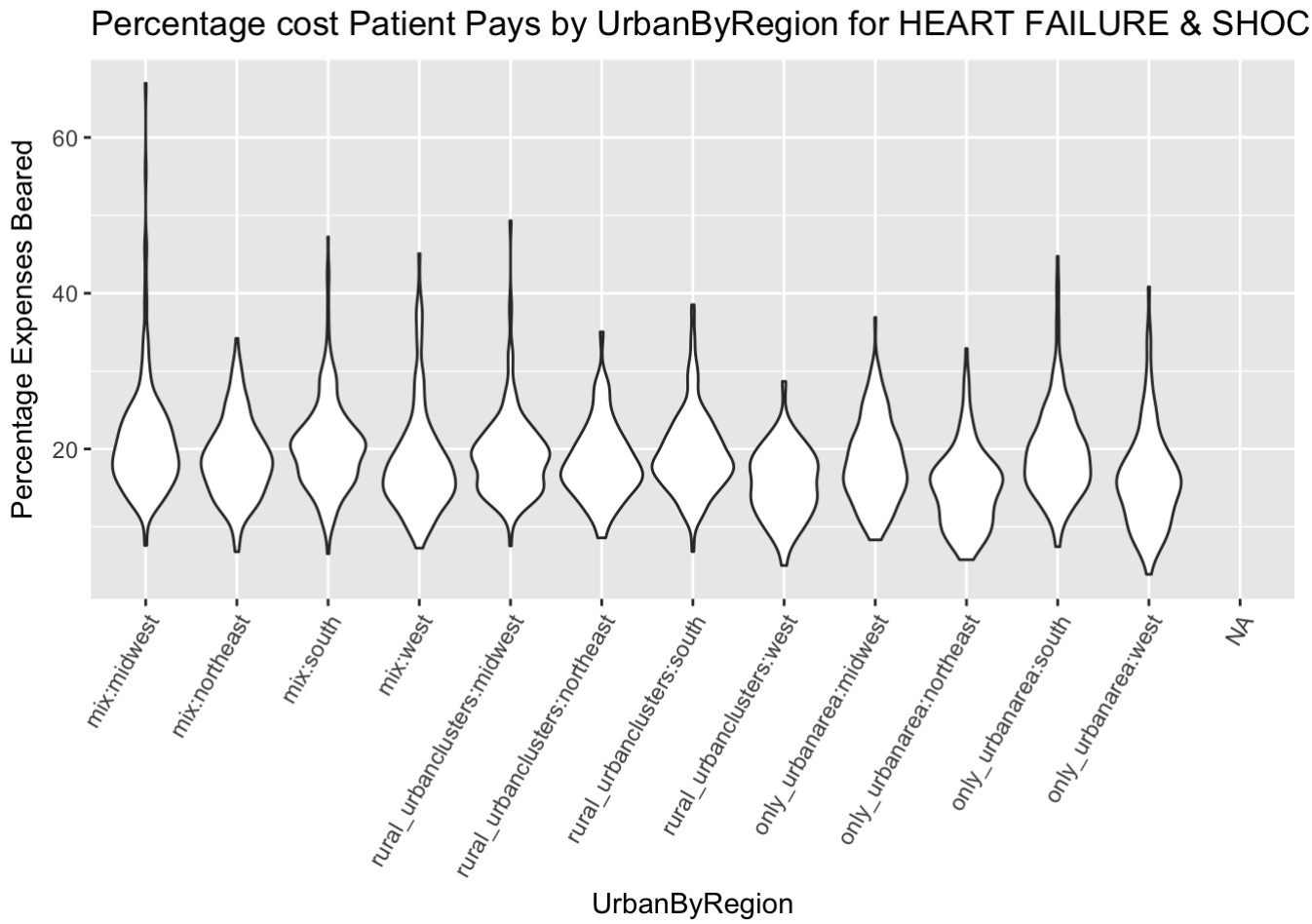
We can combine the idea of density plots and boxplots to get a ‘violin plot’. This is basically just turning the density estimate on its side and putting it next to the boxplot so that we can get finer-grain information about the distribution. Like boxplots, this allows us to compare many groups.

```
for (i in c(1:4)) {
  new_dataframe <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  print(ggplot(data = new_dataframe)+geom_violin(aes(y = new_dataframe$PctPatientPays, x
 =  new_dataframe$UrbanByRegions))+labs(title = paste0("Percentage cost Patient Pays by
 UrbanByRegion for ",str_sub(levels(dff$DRG.Definition)[i], start = 7 )), y = paste("Per
centage Expenses Beared"), x = "UrbanByRegion")+ theme(axis.text.x = element_text(angle
 = 60, hjust = 1)))
  }
```

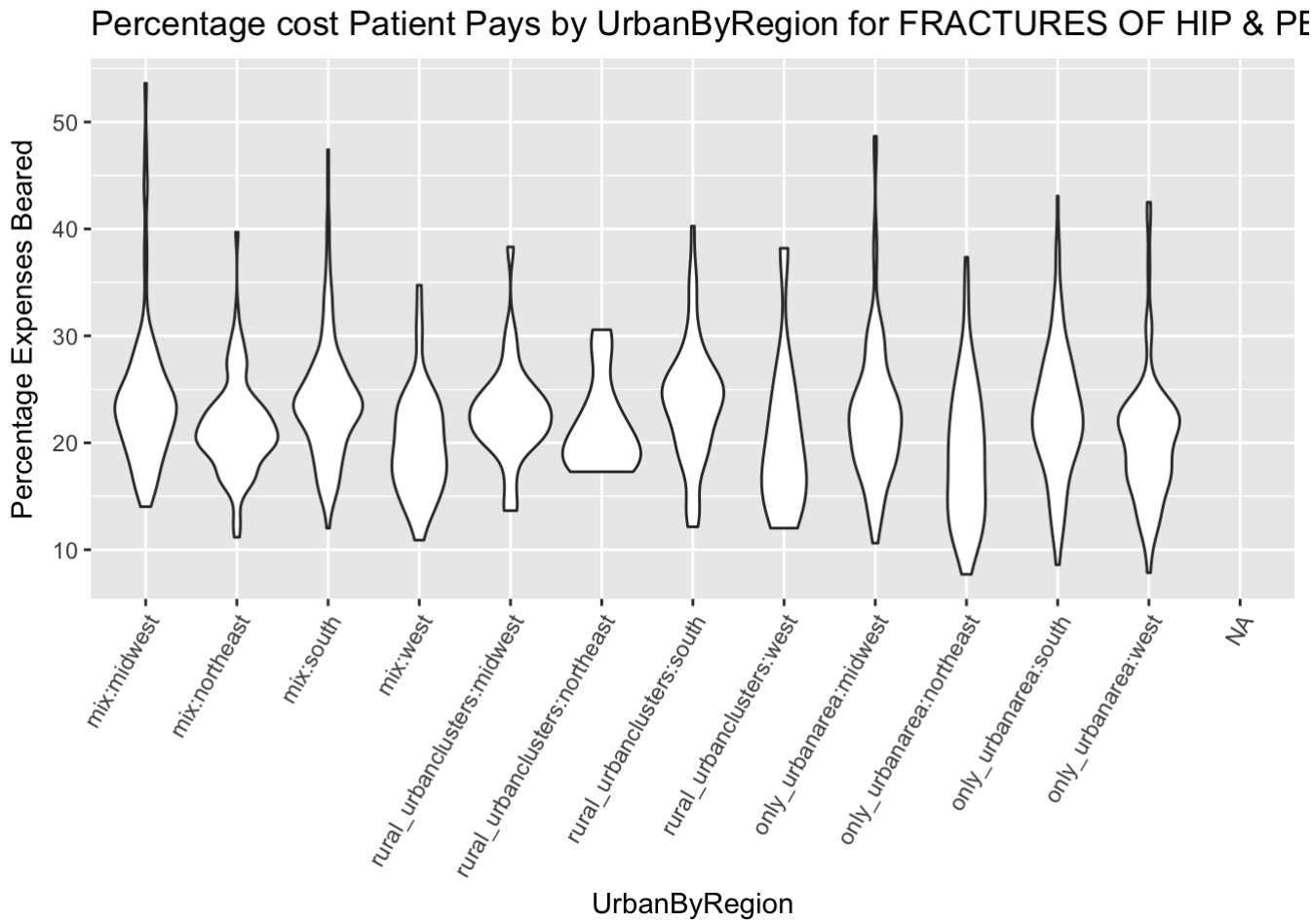
```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```



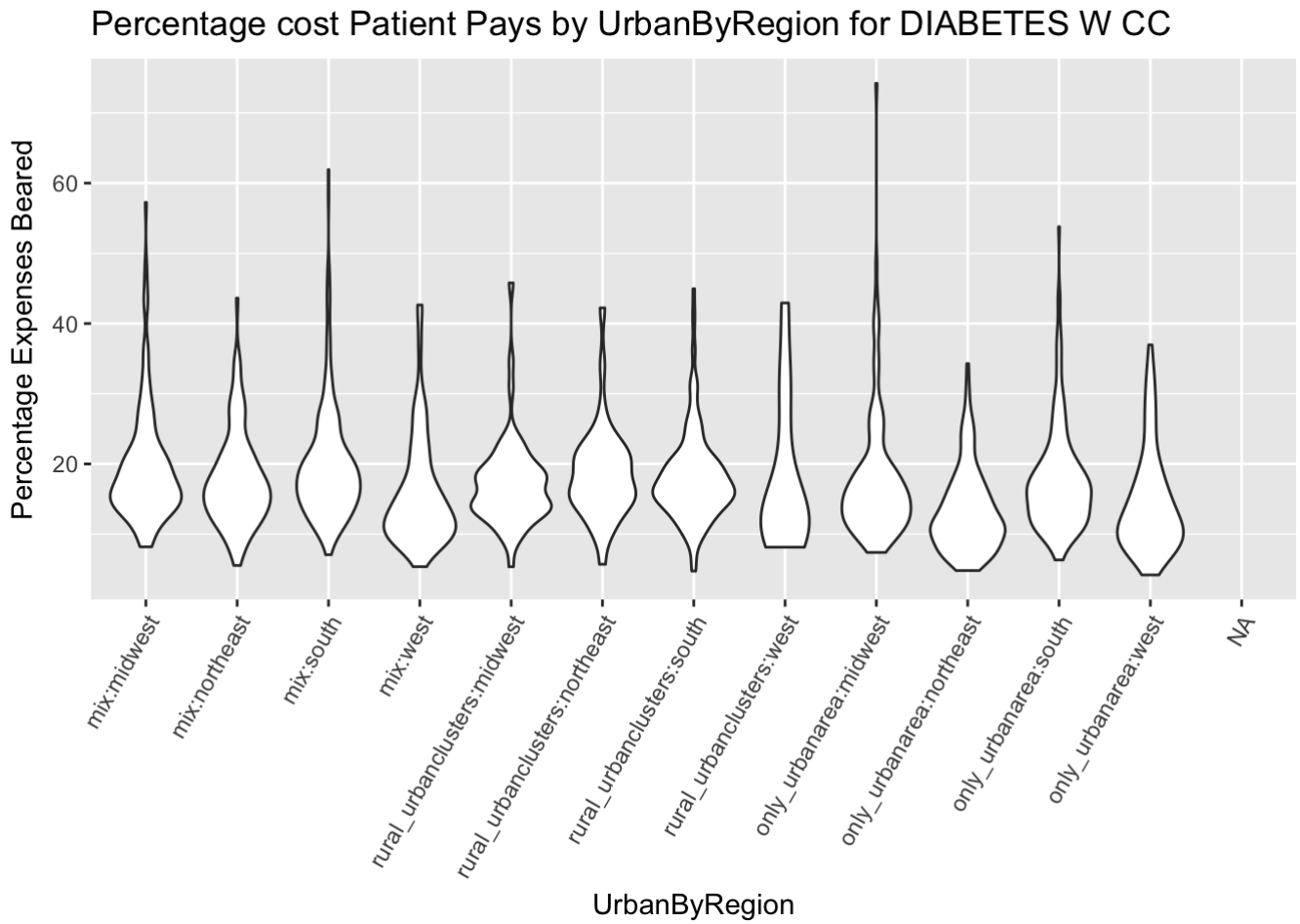

```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```




```
## Warning: Removed 496 rows containing non-finite values (stat_ydensity).
```



4.

Inference to Evaluate the difference between the groups

In the previous section, we mainly reviewed about informally comparing the distribution of data in different groups relying on the visualisations. Now we want to review the tools using statistics to make this more formal – specifically to quantify whether the differences we see are due to natural variability or something deeper.

The Question Whether the distribution of PatientPays and PctPatientPays is different for different categories of UrbanByRegion for each of the different diagnosis. If so, how could we quantify the difference between them using various statistics(any function of the input data sample).

Inference is the process of using statistical tools to evaluate whether the statistic observed indicates some kind of actual difference, or whether we could see such a value due to random chance even if there was no difference. Therefore, to use the tools of statistics – to say something about the generating process – we must have be able to define a random process that we posit created the data.

In this section we would apply pairwise t.test, permutation test and calculate the confidence intervals in order to make inference about which groups amongst all in UrbanByRegions are significantly different and which differences are only due to chance. This will help us conclude the variations of cost of a certain diagnosis that the patient undergoes being in a certain region of the country.

4.1 Drawing Inference of the difference due to UrbanByRegion in PatientPays Variable

4.1.1 Parametric Testing

Parametric test assumes that the data is normally distributed which is a pretty good assumption in our case because as we say in the histograms in the previous section, our data is very close to normal.

We need to compare all the categories of UrbanBy Regions and test whether any differences we see in the two response variables is due to chance. Let's start by comparing them all parametrically using t.test for all four diagnosis. The following a calculation to determine the number of pairs of UrbanByRegion groups possible.

```
UBRgroups<-levels(as.factor(dff$UrbanByRegions))
nUBRgroups<-length(UBRgroups)
npairs<-choose(nUBRgroups,2)
cat("number of pairs:",npairs,"\n")
```

```
## number of pairs: 66
```

To find all of the pairs of UrbanByRegion groups, and calculate a t-test for each pair. The function `combn` gives all the combinations of 2 (`m=2`); the result is a matrix with 2 columns, with each combination a row.

```
pairsOfUBR<-combn(x=UBRgroups,m=2) #get all combinations of length 2, ie all pairs
```

I will again work with the transformed data, so lets create this variable again in my data.frame (before it was saved as a temporary value)

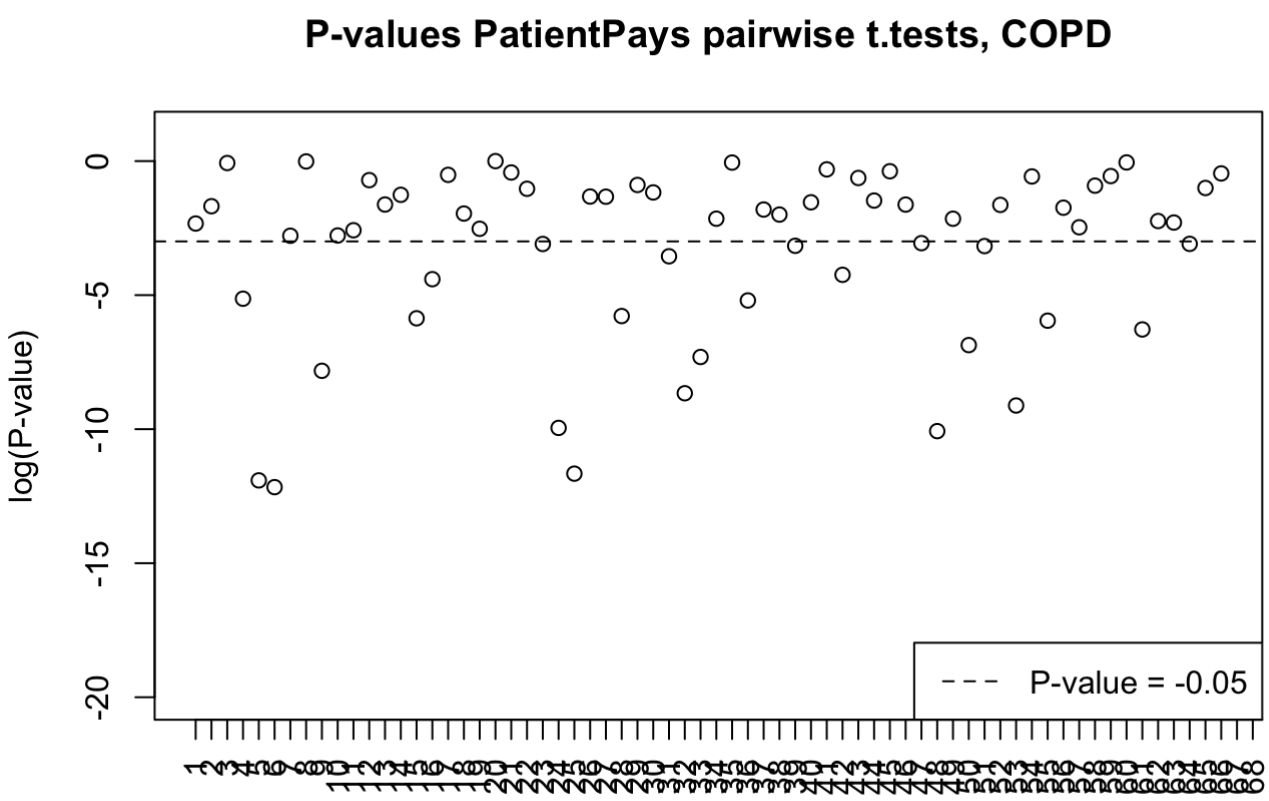
```
dff$logPatientPays<-log(dff$PatientPay)
```

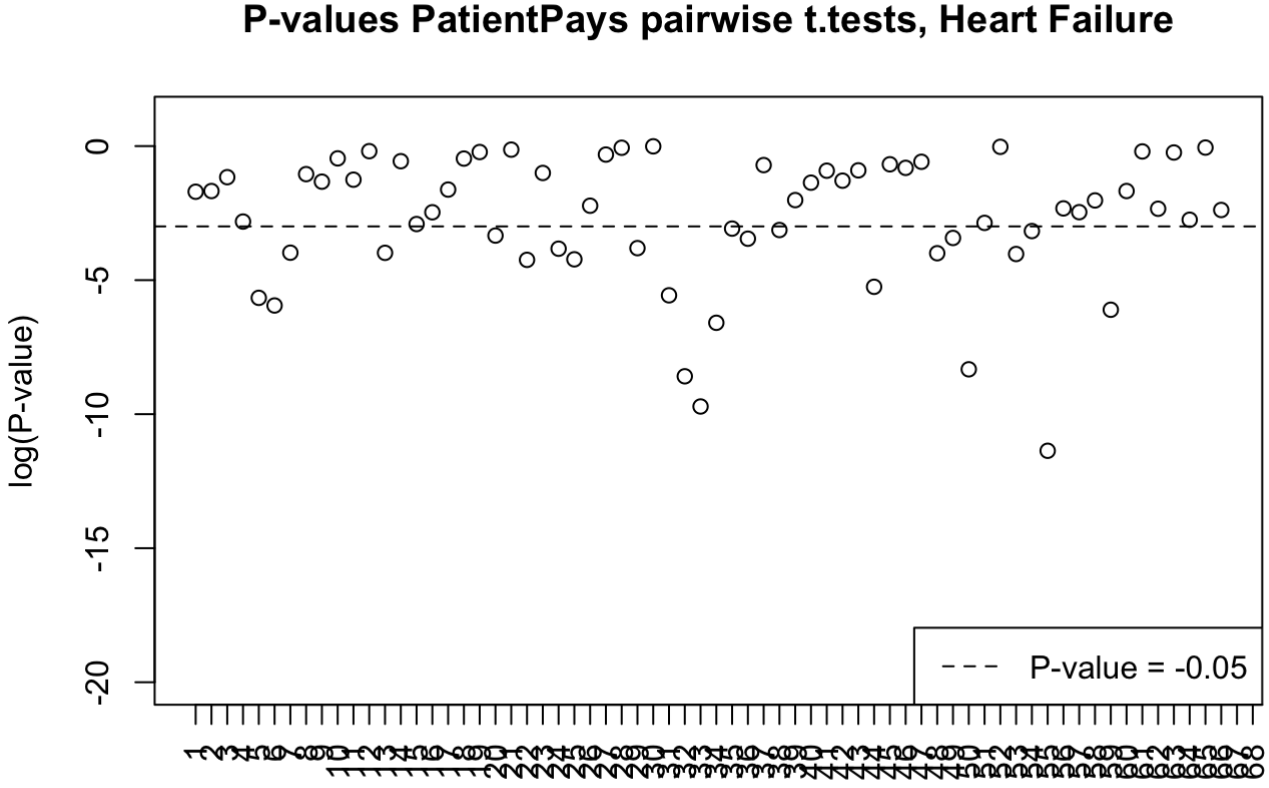
Calculating the pairwise T-Test fr all the pairs of the 4 diagnosis

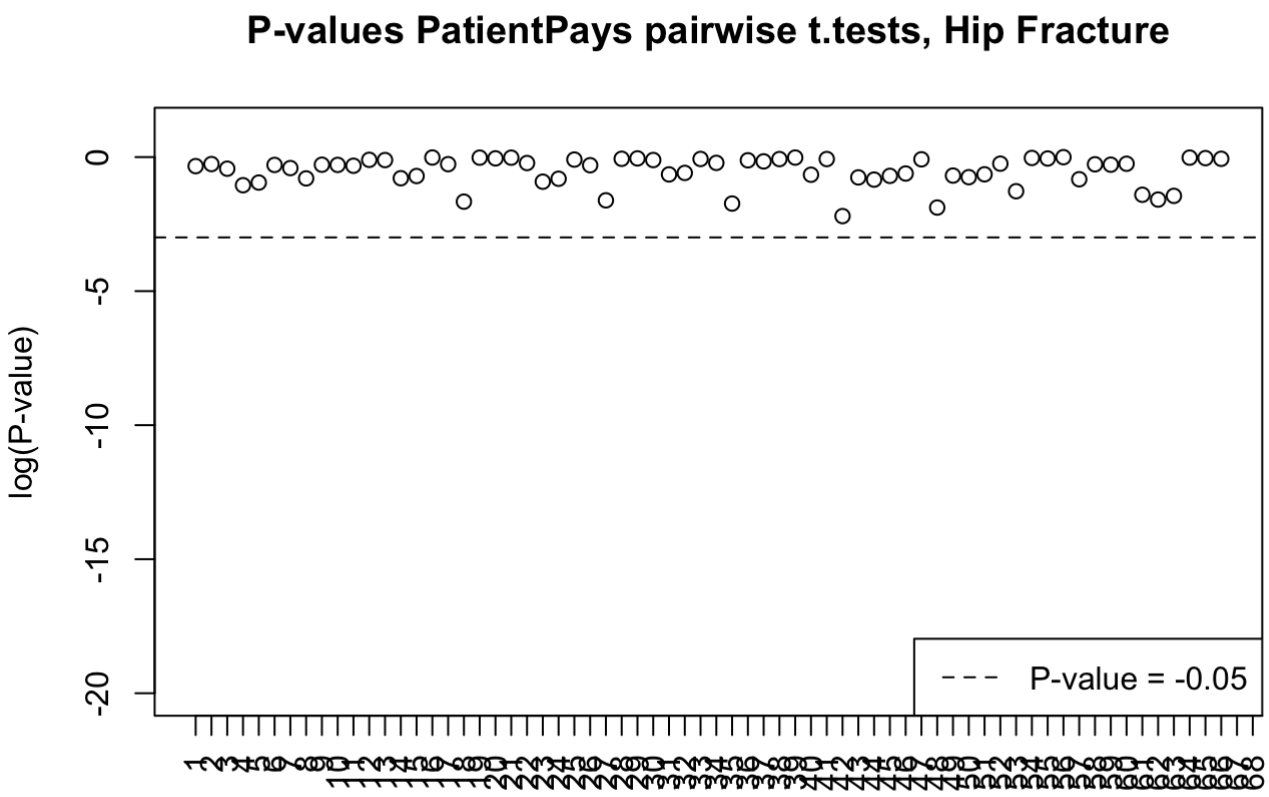
```
t.testPairsUBR <- list()

ttestFun<-function(x,variableName, dff_temp){
    tout<-t.test(dff_temp$logPatientPays[dff_temp[,variableName] == x[1]],dff_temp$logPa
tientPays[dff_temp[,variableName] == x[2]])
    unlist(tout[c("statistic","p.value", "conf.int", "estimate")]) #unlist makes it a ve
ctor rather than list
}

for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.testPairsUBR[[i]]<-combn(x=UBRgroups,m=2, FUN=ttestFun, variableName="UrbanByRegion
s", dff_temp = dff_temporary)
}
```

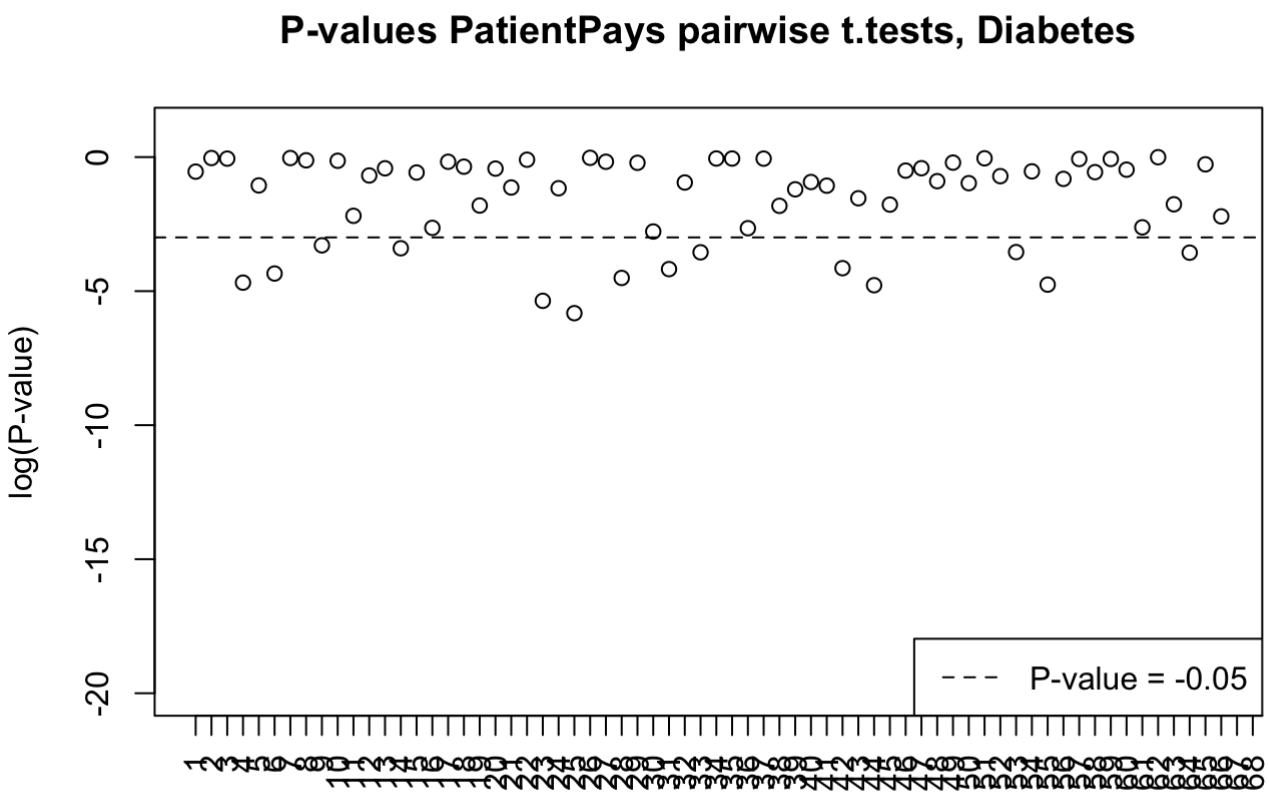
We can plot these p-values to get an idea of their value.

```
for (i in c(1:4)) {
  plot(log(t.testPairsUBR[[i]][2, ]),ylab="log(P-value)",main= paste0("P-values PatientP
ays pairwise t.tests, ", name_diag[i]),xaxt="n",xlab="", ylim = c(-20, 1))
abline(h=log(0.05),lty=2)
legend("bottomright",legend="P-value = -0.05",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


P-values PatientPays pairwise t.tests, COPD
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05

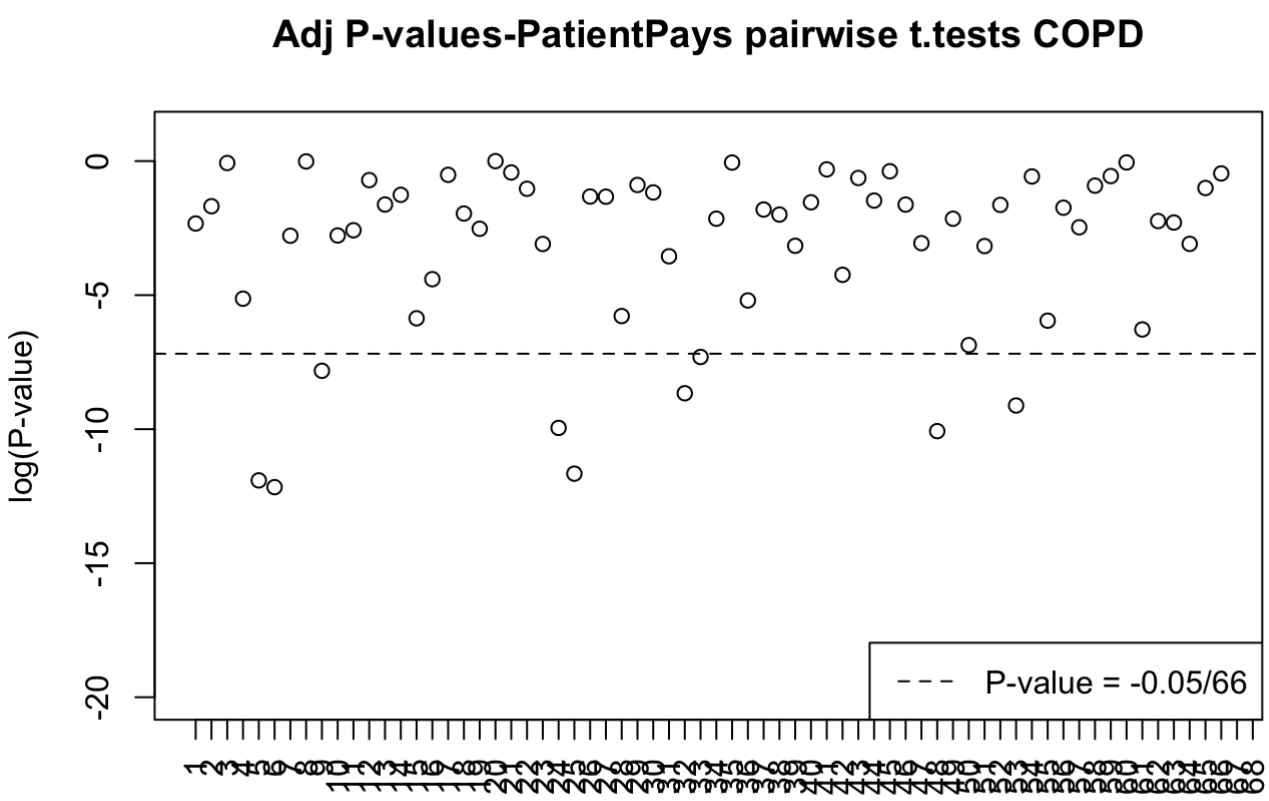

P-values PatientPays pairwise t.tests, Heart Failure
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05

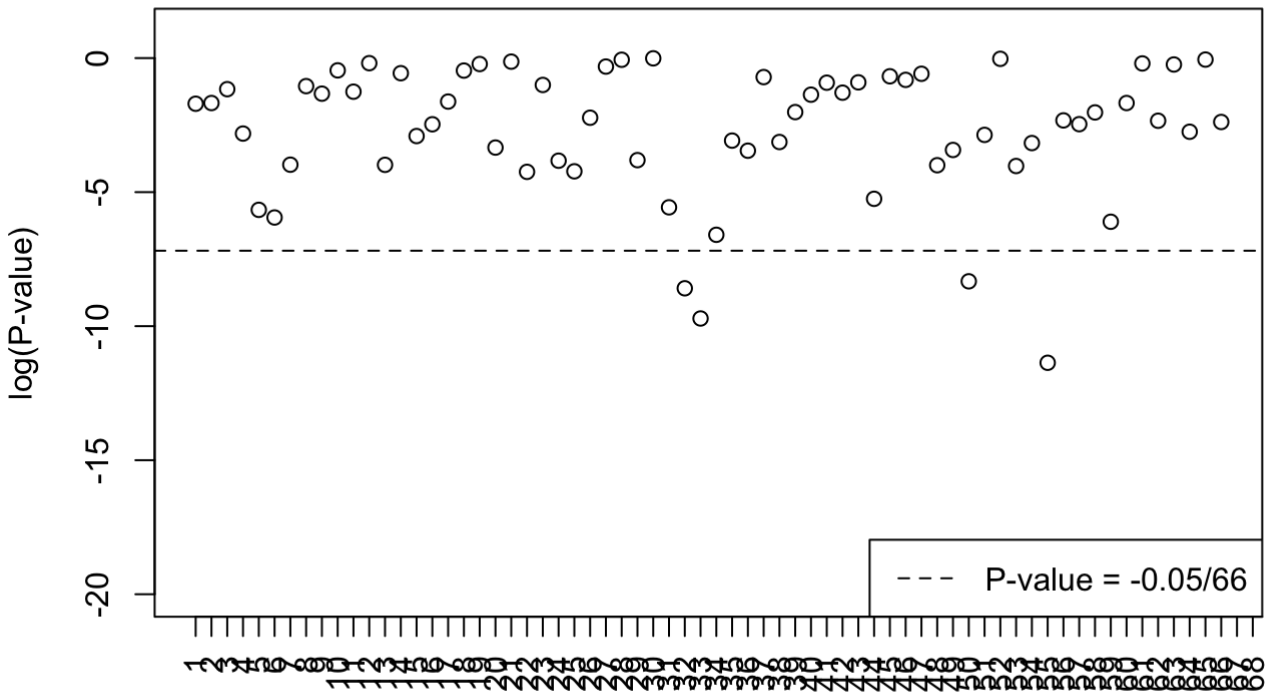


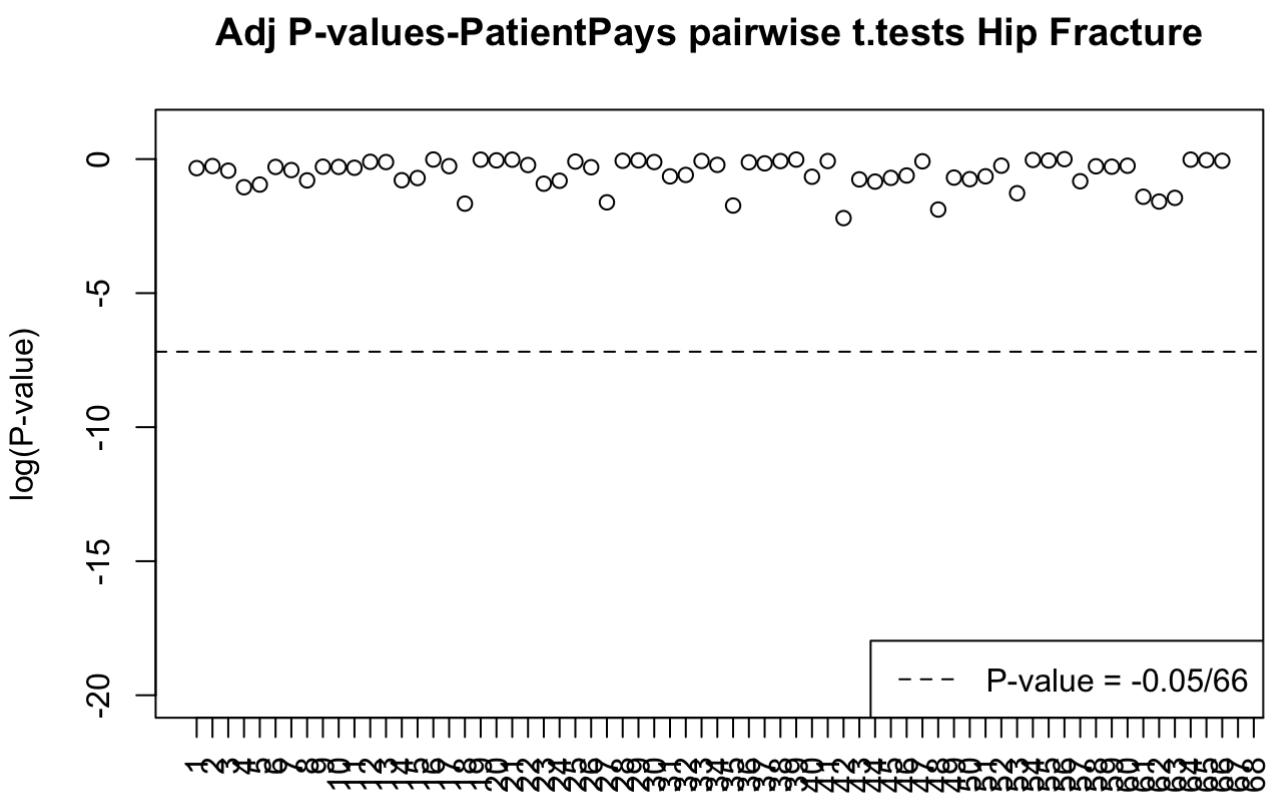


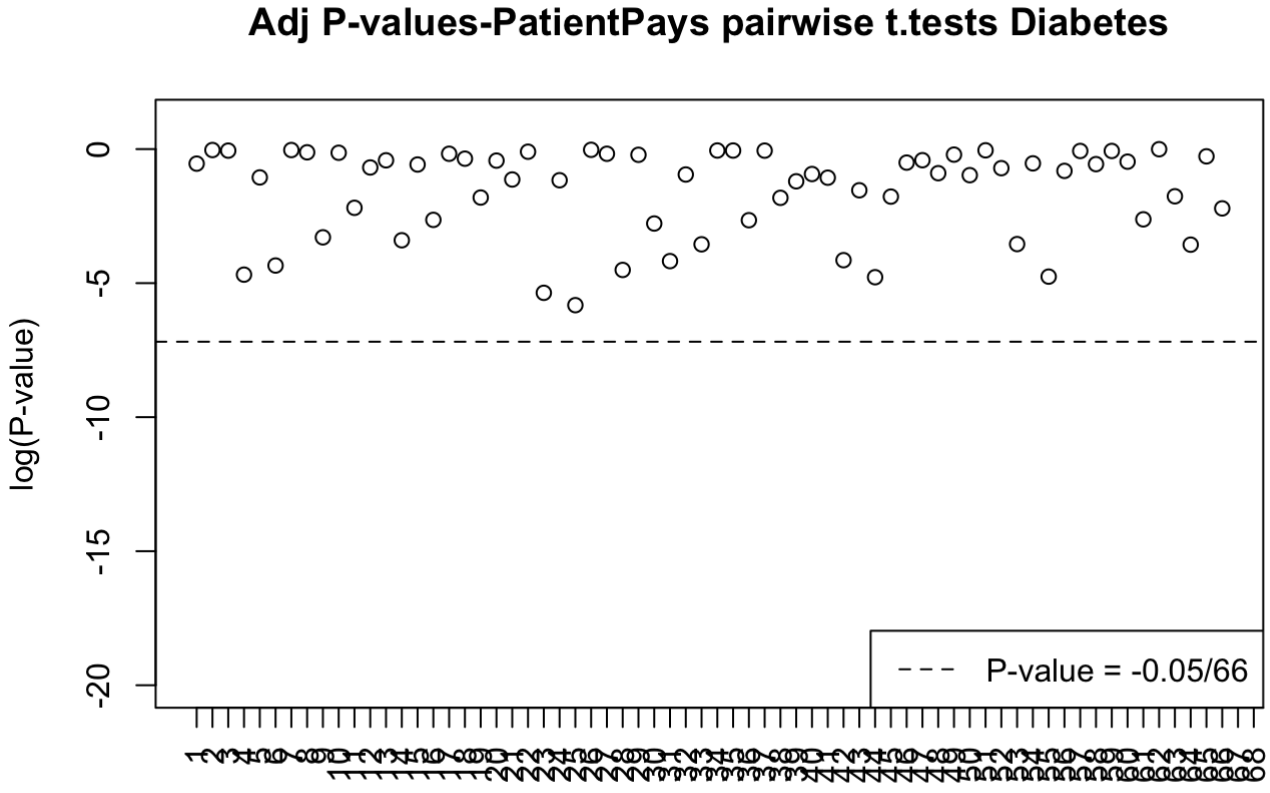
Applying Bonferroni Correction

```
for (i in c(1:4)) {
  plot(log(t.testPairsUBR[[i]][2, ]),ylab="log(P-value)",main= paste0("Adj P-values-Pati
entPays pairwise t.tests ", name_diag[i]),xaxt="n",xlab="", ylim = c(-20, 1))
abline(h=log(0.05/npairs),lty=2)
legend("bottomright",legend="P-value = -0.05/66",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


Adj P-values-PatientPays pairwise t.tests COPD
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05/66


Adj P-values-PatientPays pairwise t.tests Heart Failure
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05/66





Now,

let’s collect and analyse the pairs that were significantly different for the PatientPays.

```
significantly_different_pairs <- list()
for (i in c(1:4)) {
  significantly_different_pairs[[i]] <- pairsOfUBR[,which(log(t.testPairsUBR[[i]][2, ])
 < log(0.05/66))]
}

significantly_different_pairs[[1]]
```

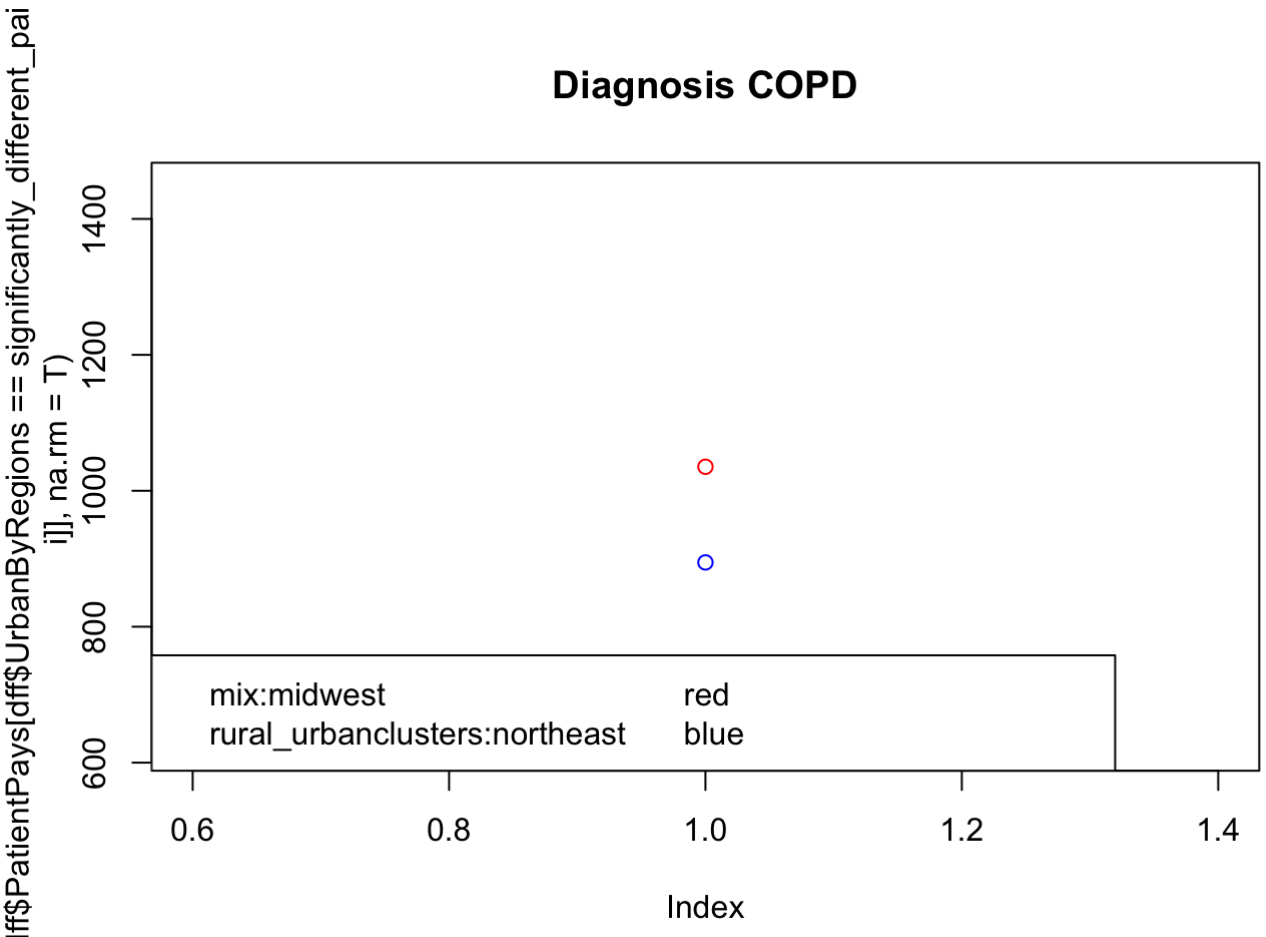
```
##      [,1]                           [,2]
## [1,] "mix:midwest"                  "mix:midwest"
## [2,] "rural_urbanclusters:northeast" "rural_urbanclusters:south"
##      [,3]                        [,4]
## [1,] "mix:midwest"               "mix:south"
## [2,] "only_urbanarea:northeast" "rural_urbanclusters:northeast"
##      [,5]                         [,6]
## [1,] "mix:south"                  "mix:west"
## [2,] "rural_urbanclusters:south" "rural_urbanclusters:northeast"
##      [,7]                         [,8]
## [1,] "mix:west"                   "rural_urbanclusters:northeast"
## [2,] "rural_urbanclusters:south" "only_urbanarea:midwest"
##      [,9]
## [1,] "rural_urbanclusters:south"
## [2,] "only_urbanarea:midwest"
```

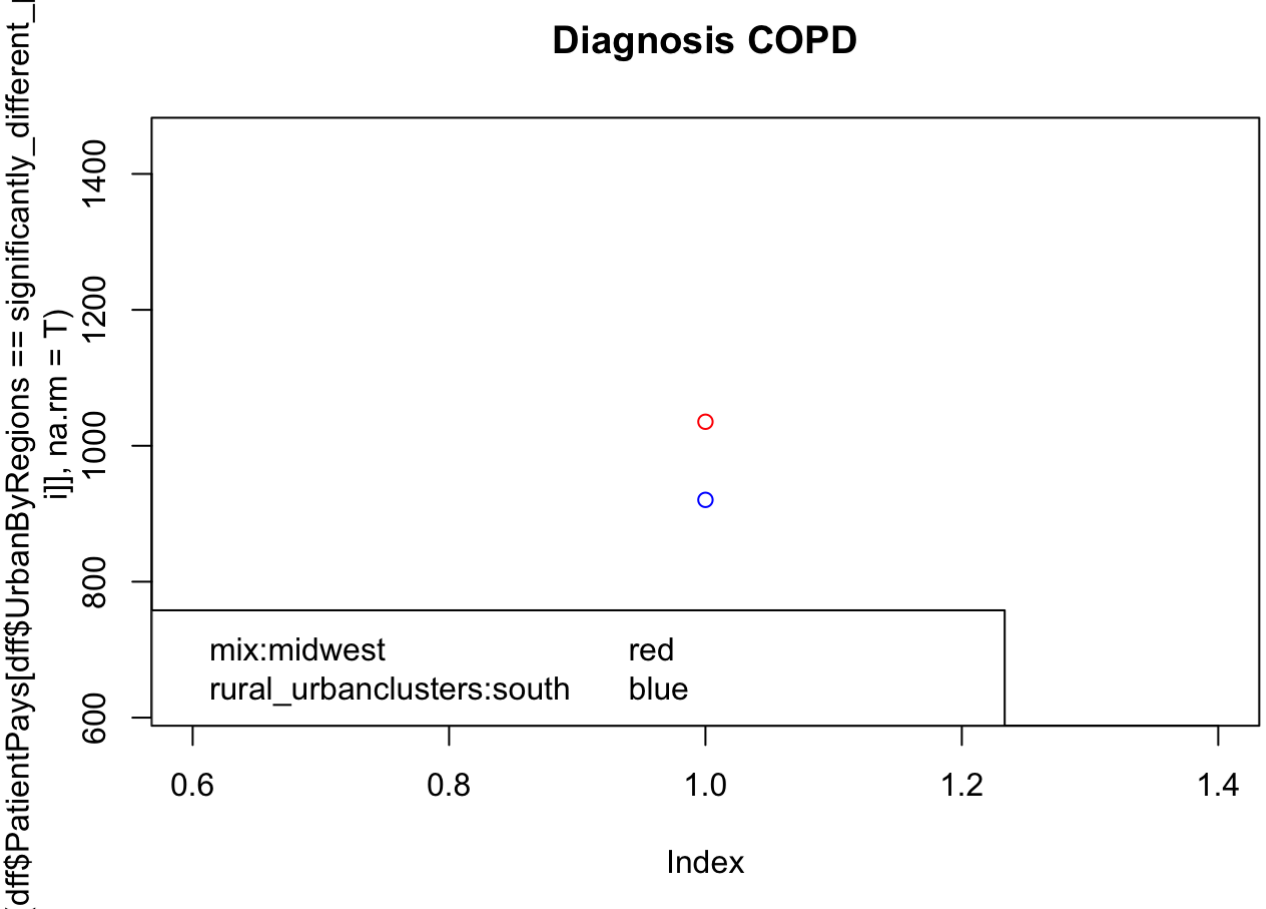
We see that for the diagnosis of chronic obstructive pulminary disease (copd), the absolute cost that the patient pays is significantly higher for the urban areas and areas of a mix of Urbanized Areas and Rural (and perhaps also Urban Clusters) in the zipcode in the region of midwest, west and south compared to combined rural and urban clusters of north east and south.

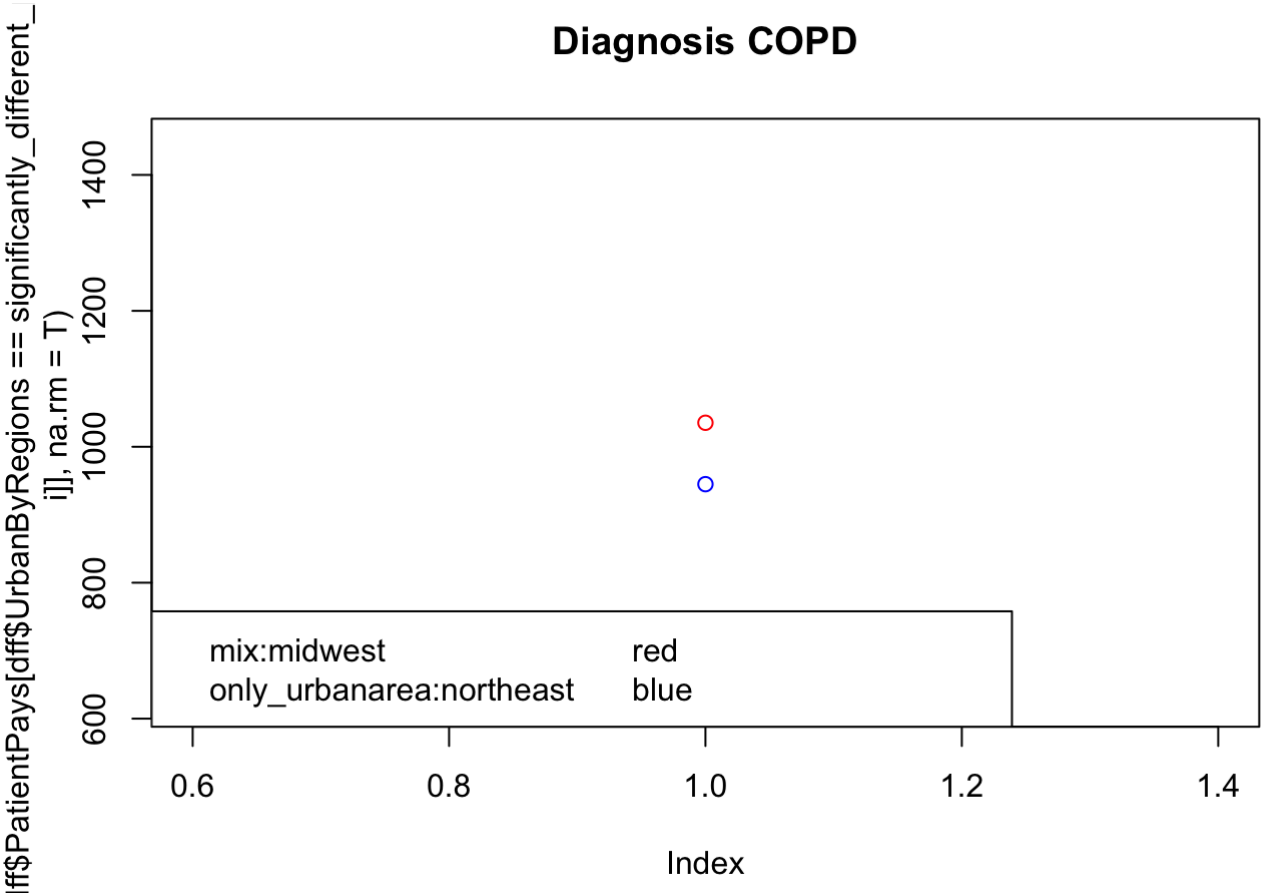
```
par(mfrow <- c(4, 4))
```

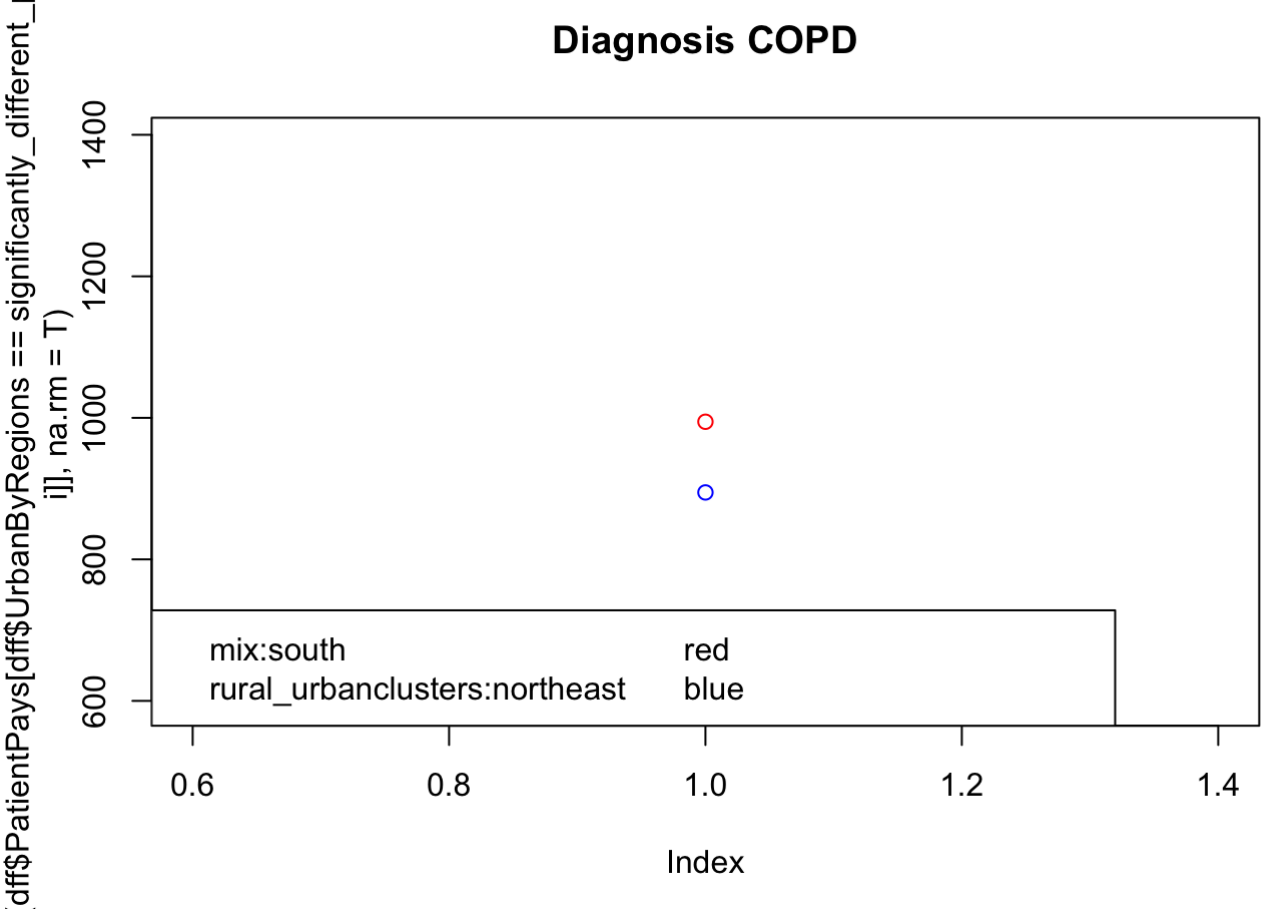
```
## NULL
```

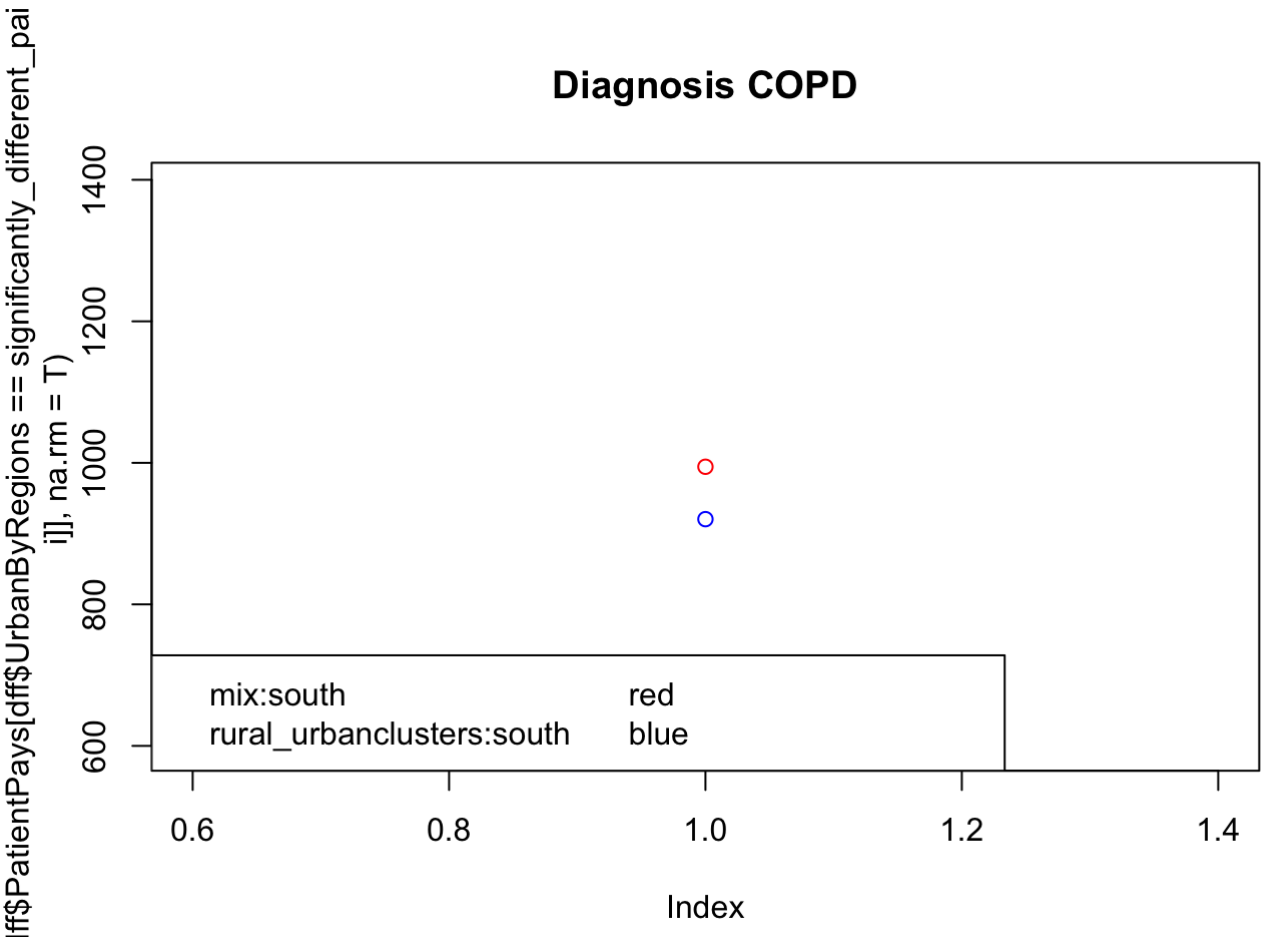
```
for (j in c(1:2)) {
 for (i in c(1:ncol(significantly_different_pairs[[j]]))) {
  plot(mean(dff$PatientPays[dff$UrbanByRegions == significantly_different_pairs[[j]][1,
 i]], na.rm = T), main = paste0("Diagnosis ",name_diag[j]), col = "red")
   legend("bottomleft", legend = c(significantly_different_pairs[[j]][1, i],significantl
y_different_pairs[[j]][2, i] , fill= c("red", "blue")), ncol = 2)
   points(mean(dff$PatientPays[dff$UrbanByRegions == significantly_different_pairs[[j]][
2, i]], na.rm = T), type = "p", col = "blue")
}
}
```

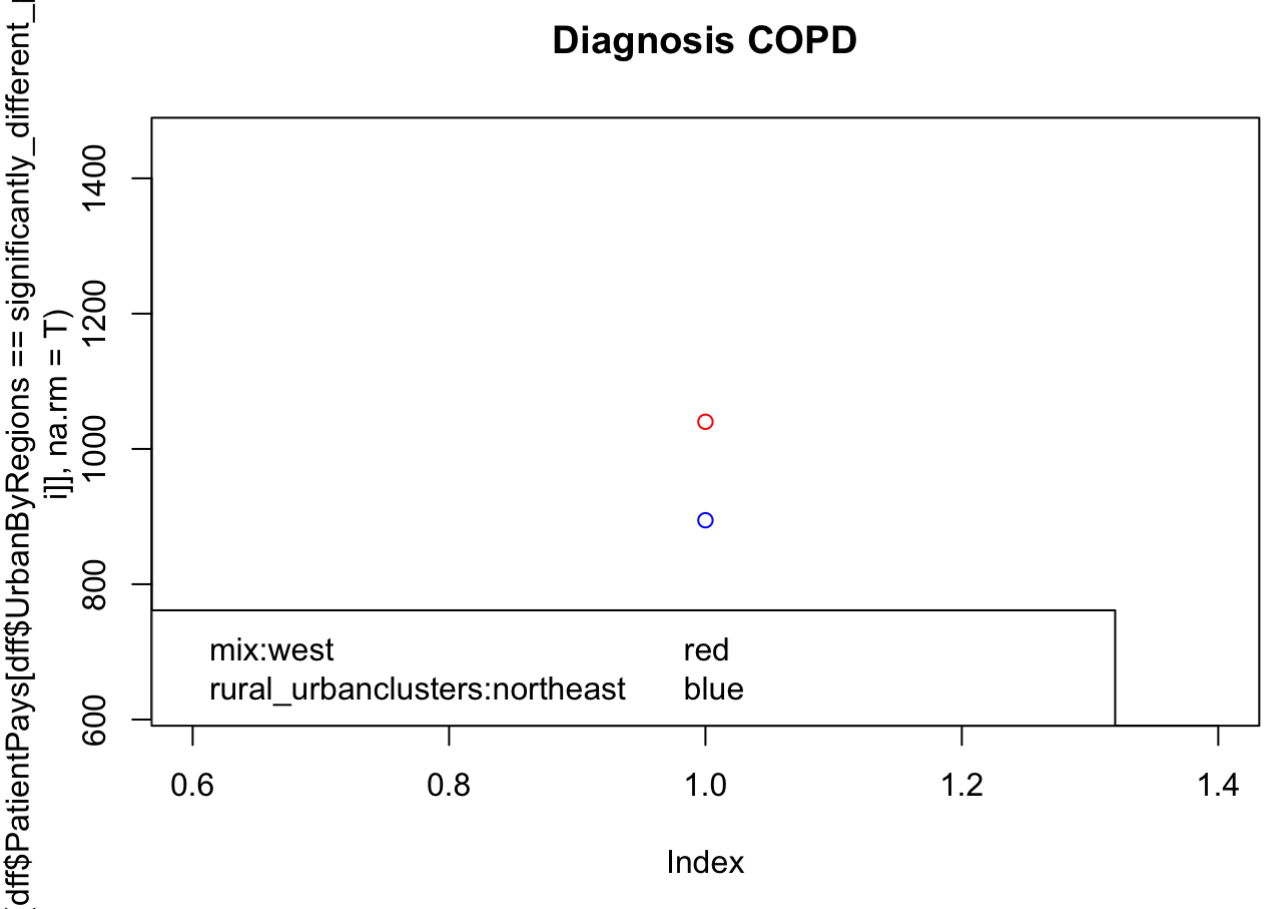

Diagnosis COPD
pai[dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:midwest
rural_urbanclusters:northeast
red
blue

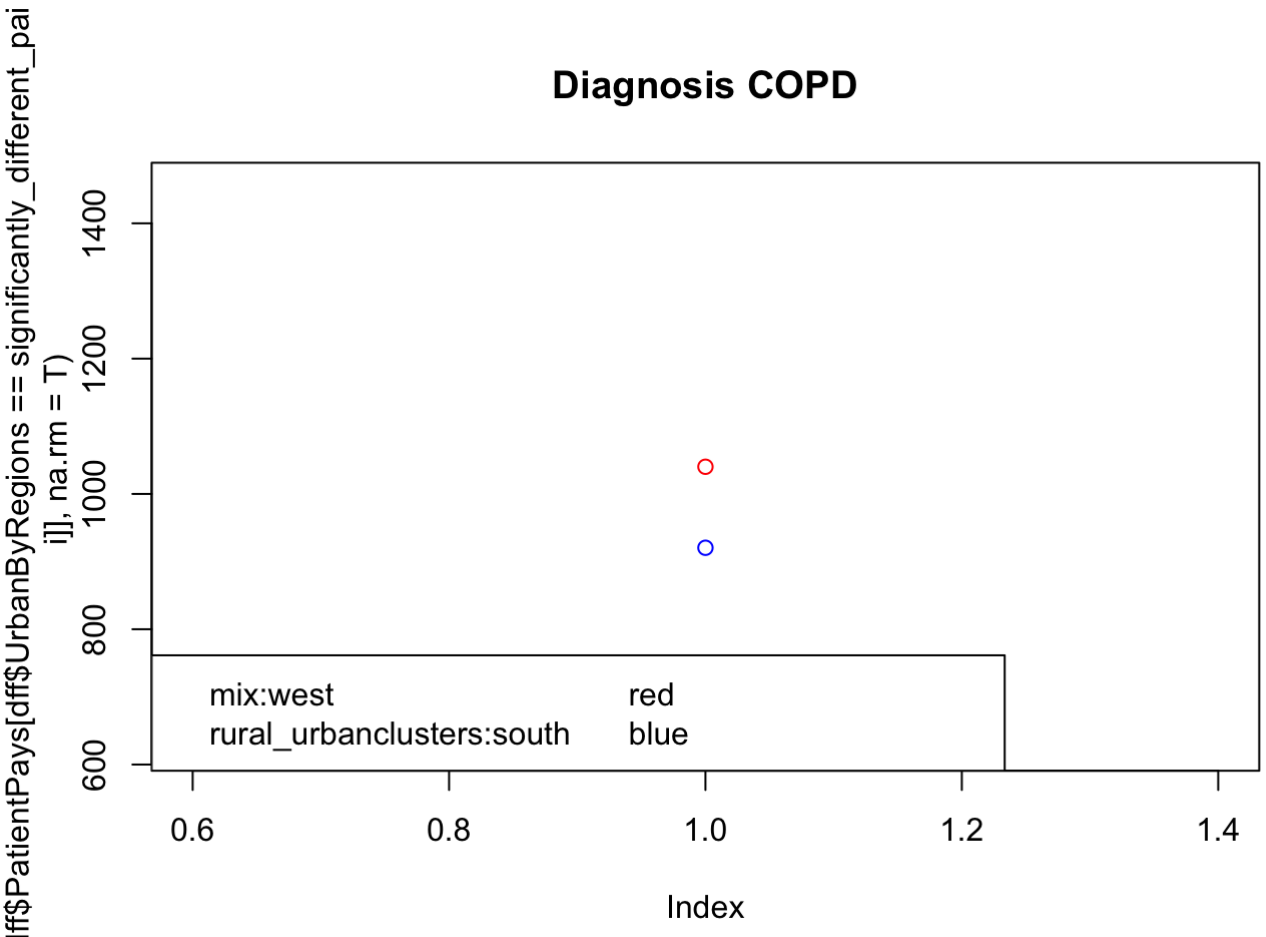

Diagnosis COPD
pai[dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:midwest
rural_urbanclusters:south
red
blue

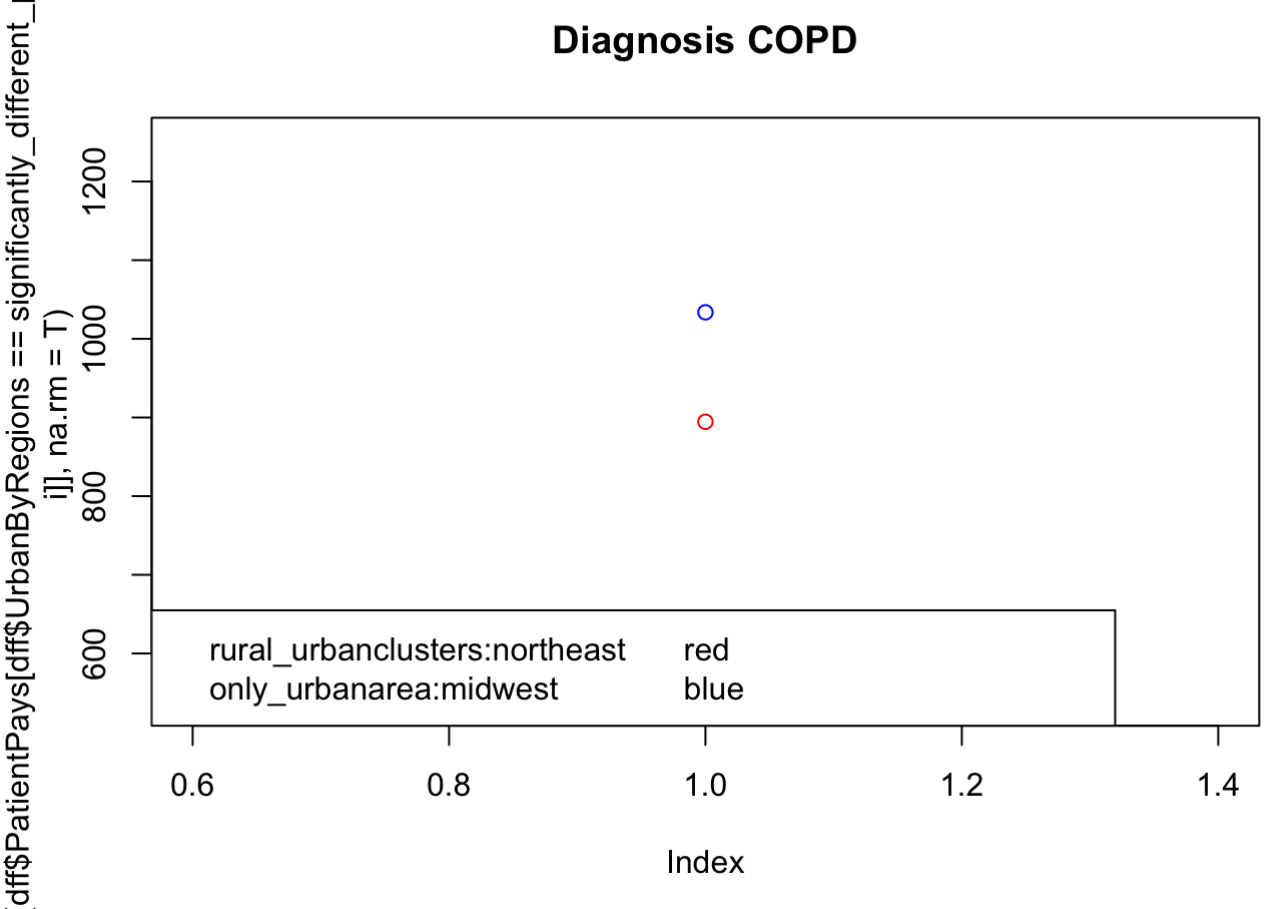

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:midwest red
only_urbanarea:northeast blue

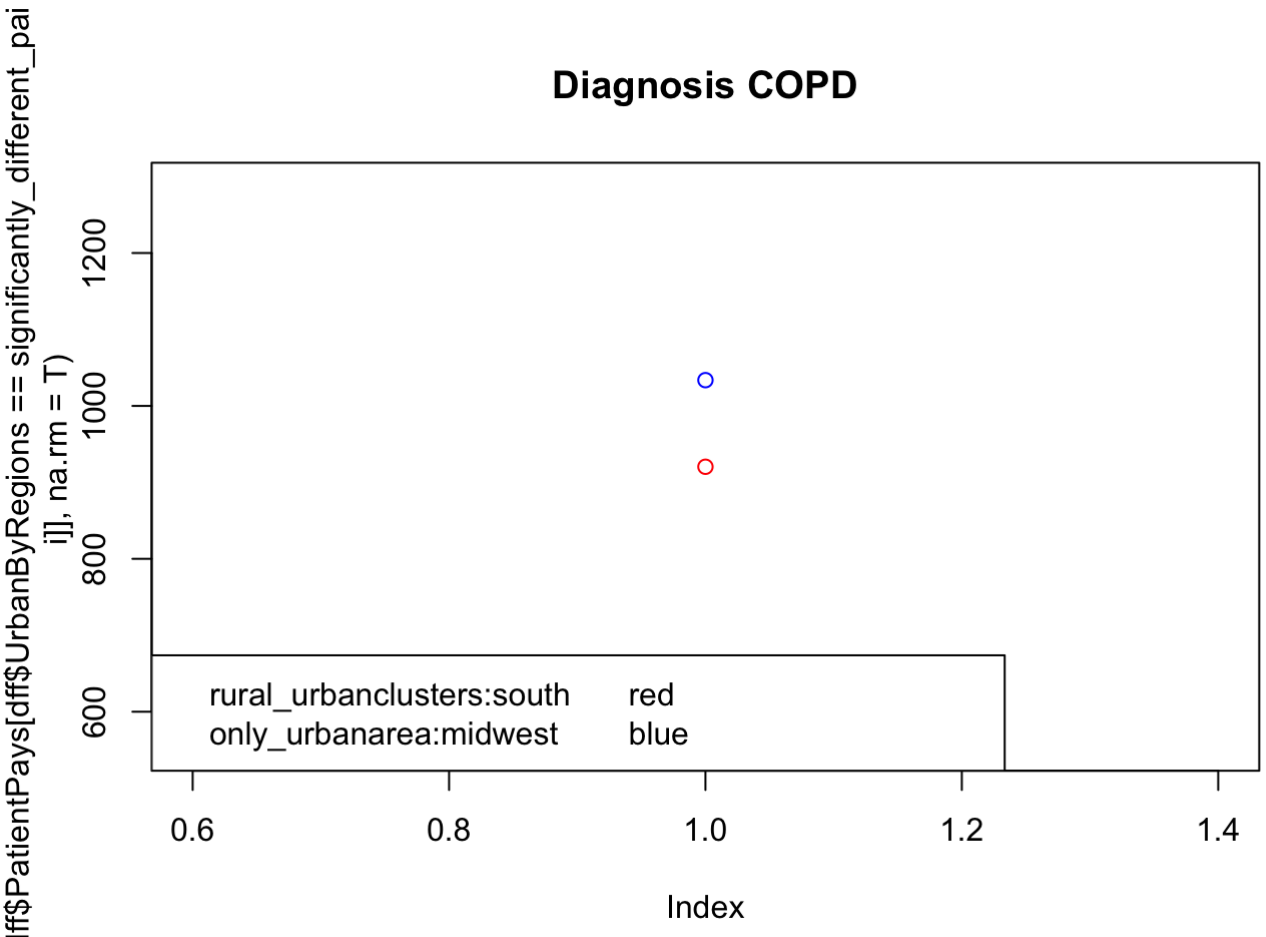

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:south red
rural_urbanclusters:northeast blue

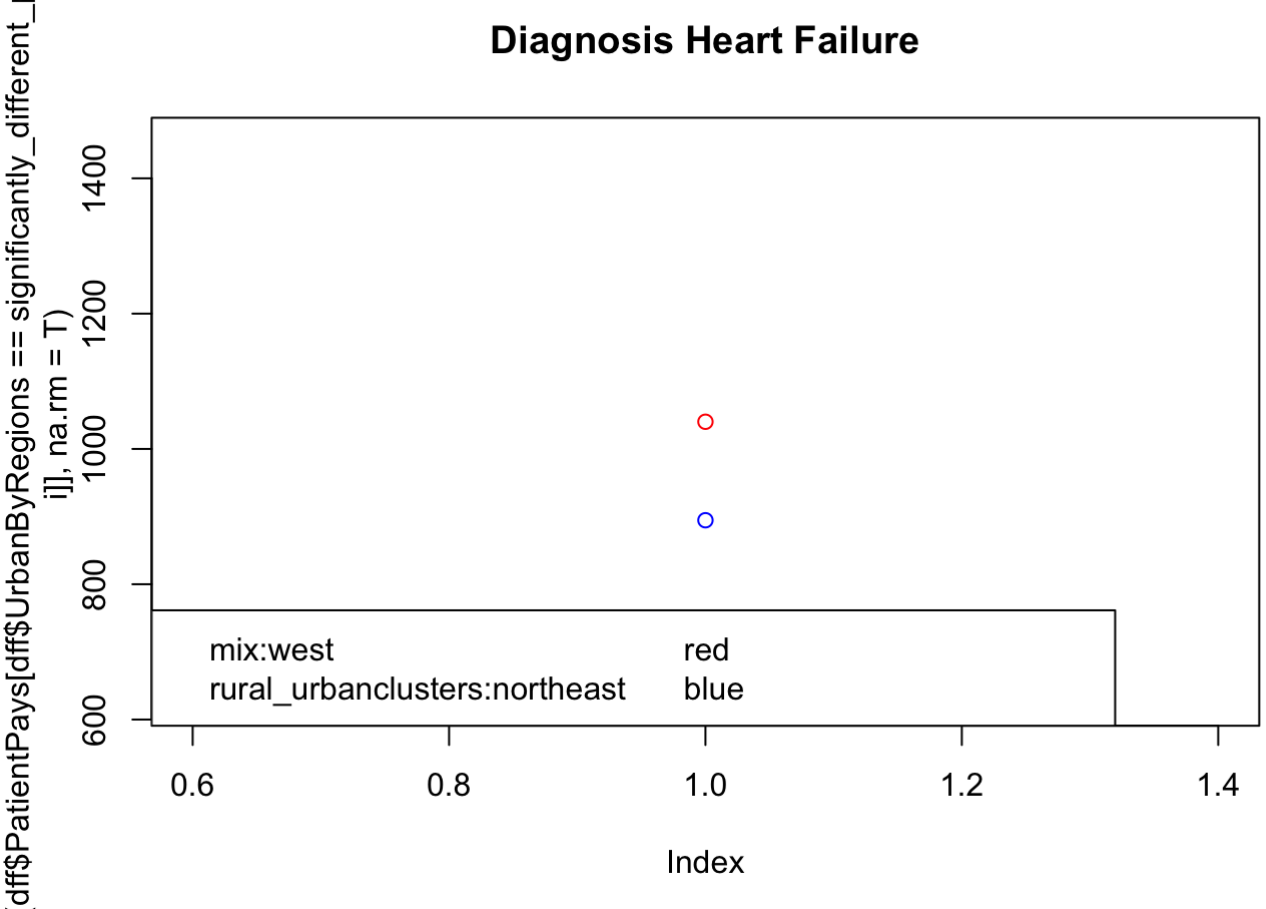

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
mix:south
red
rural_urbanclusters:south
blue
0.6
0.8
1.0
1.2
1.4
Index

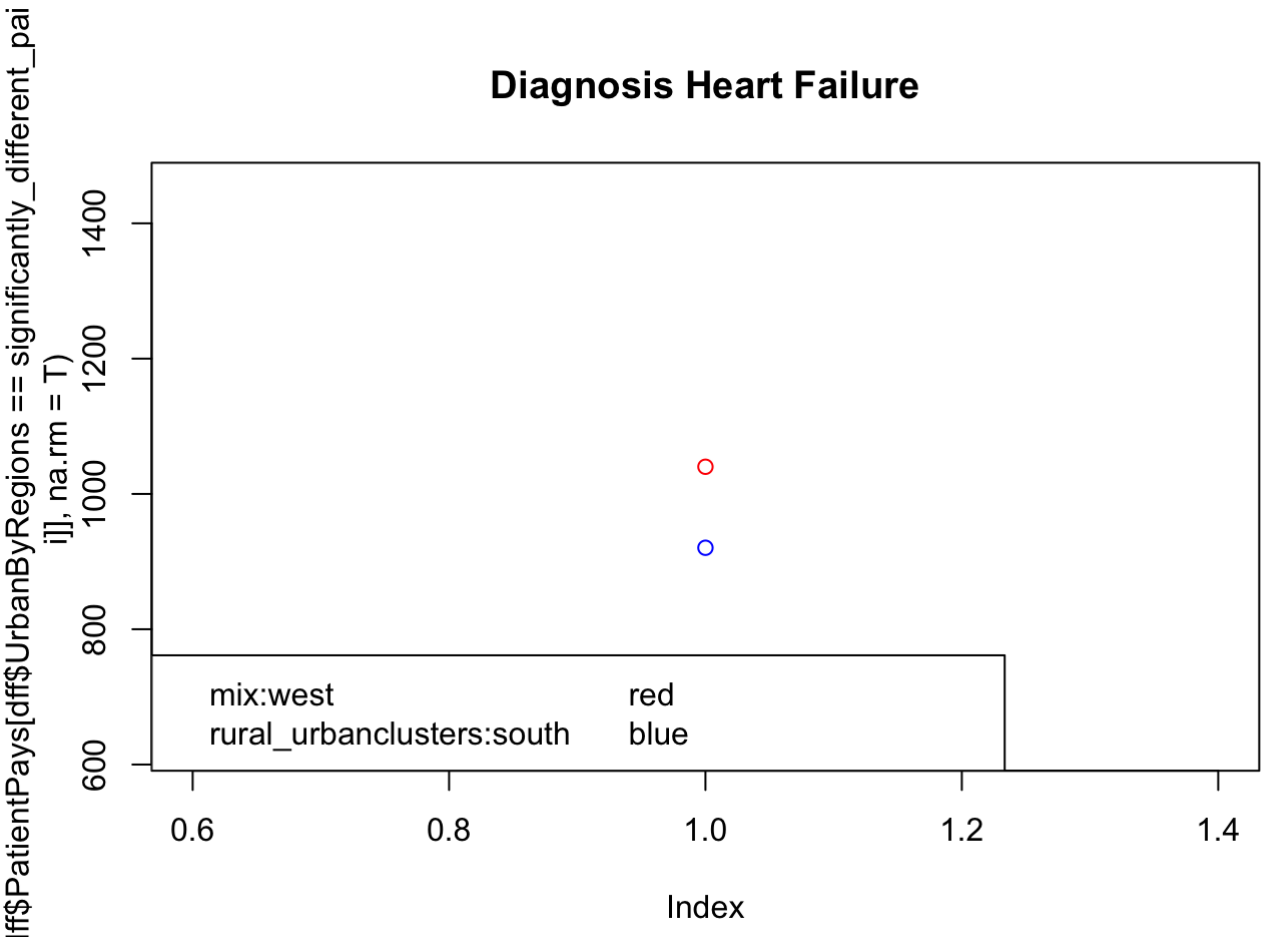

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
mix:west
red
rural_urbanclusters:northeast
blue
0.6
0.8
1.0
1.2
1.4
Index

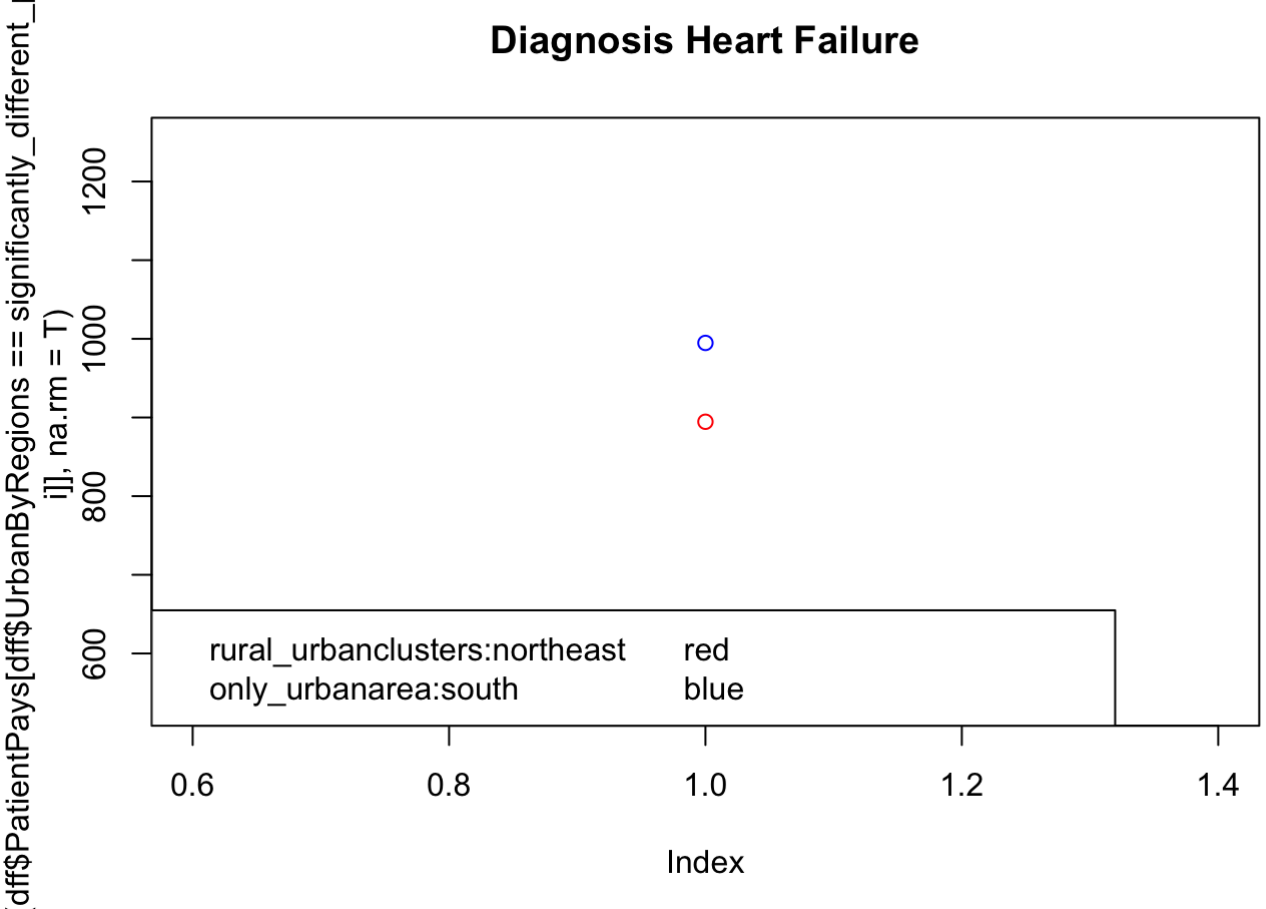

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:west red
rural_urbanclusters:south blue

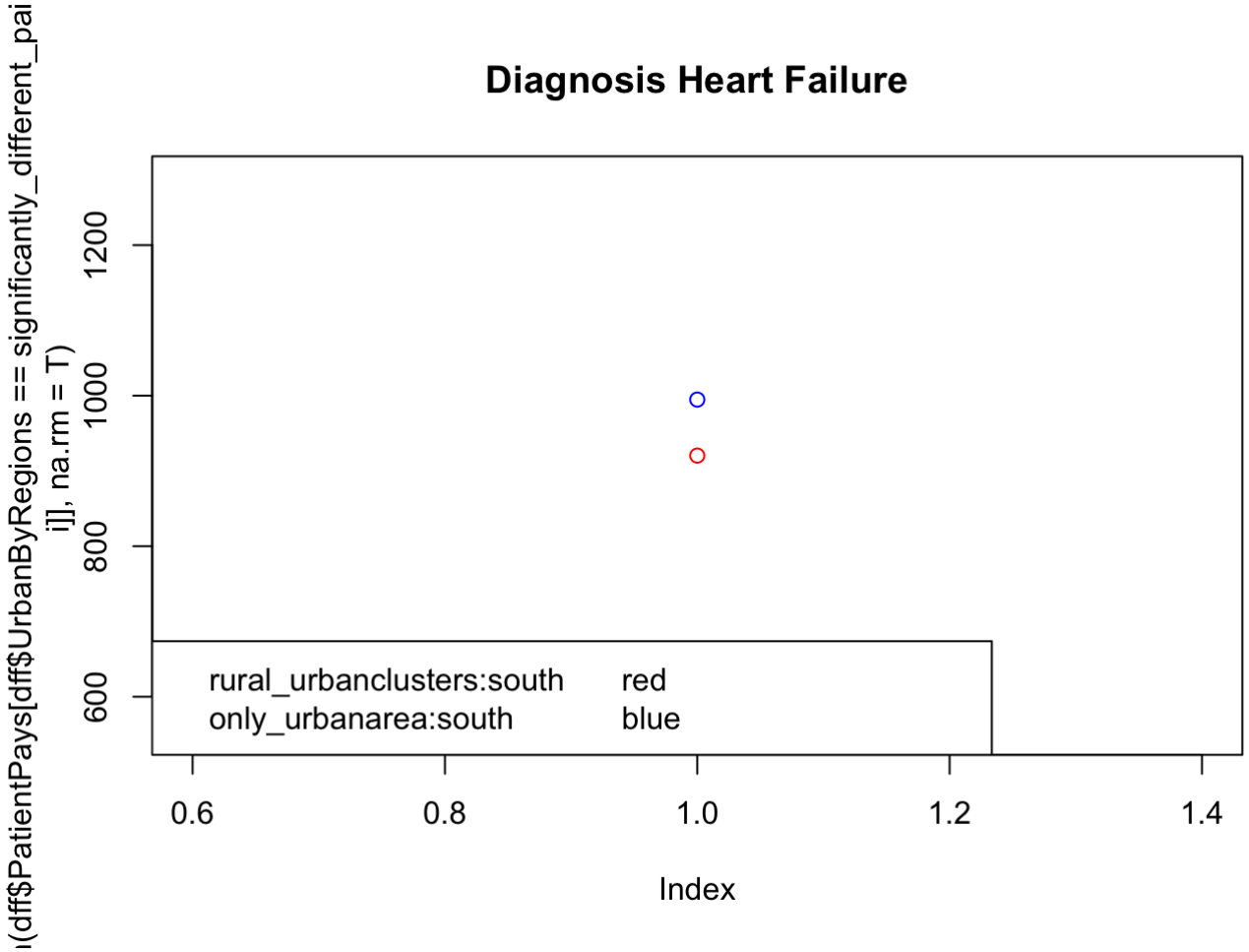

Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
0.6
0.8
1.0
1.2
1.4
Index
rural_urbanclusters:northeast red
only_urbanarea:midwest blue


Diagnosis COPD
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
rural_urbanclusters:south red
only_urbanarea:midwest blue
0.6
0.8
1.0
1.2
1.4
Index


Diagnosis Heart Failure
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
mix:west red
rural_urbanclusters:northeast blue
0.6
0.8
1.0
1.2
1.4
Index


Diagnosis Heart Failure
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
1400
0.6
0.8
1.0
1.2
1.4
Index
mix:west red
rural_urbanclusters:south blue


Diagnosis Heart Failure
(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai(dff$PatientPays[dff$UrbanByRegions == significantly_different_pai
i]], na.rm = T)
600
800
1000
1200
0.6
0.8
1.0
1.2
1.4
Index
rural_urbanclusters:northeast red
only_urbanarea:south blue



4.1.2 Permutation Testing

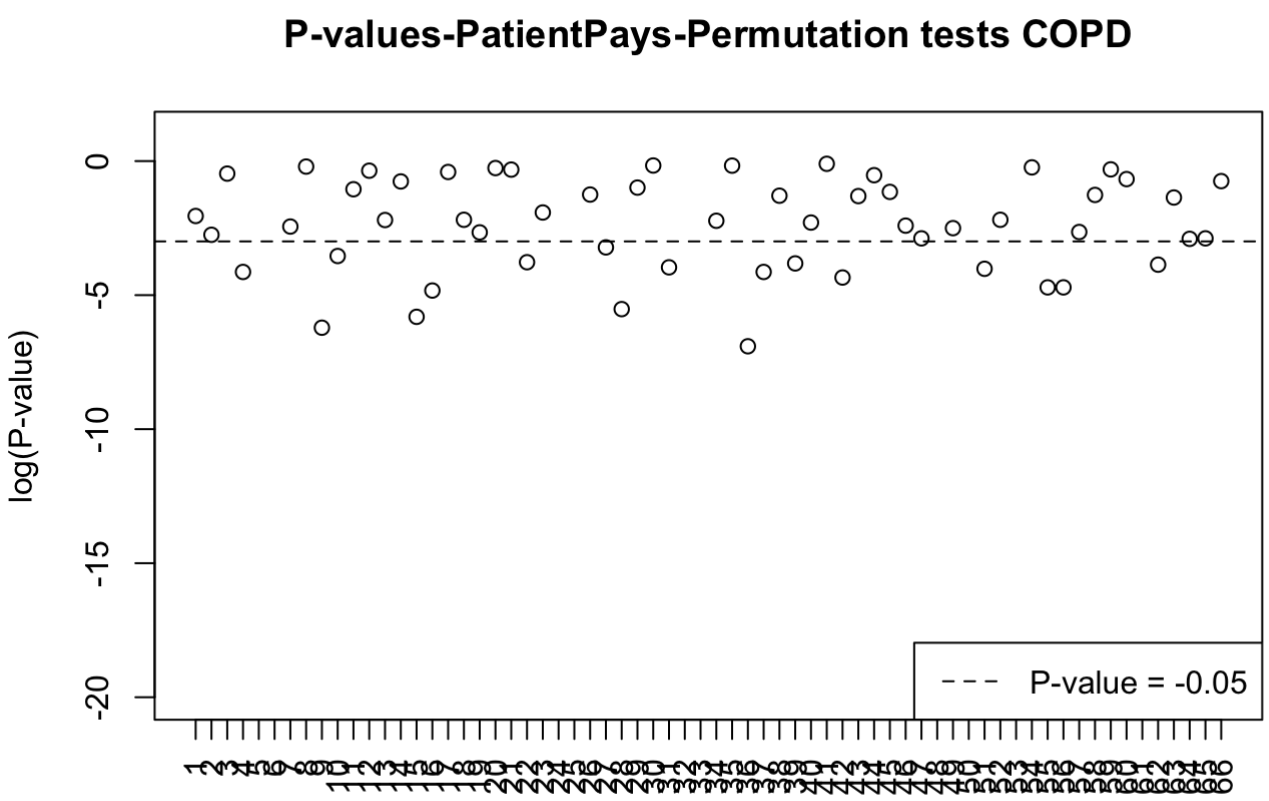
The distribution for the test statistic under the null hypothesis for a permutation tests is determined by assuming : 1. There is no difference between the distribution of PatientPays and PctPatientPays for different UrbanByRegion categories for each of the diagnosis. This is the null hypothesis to be tested. 2. The statistic observed is the result of randomly assigning the labels amongst the observed data This is the additional assumption about the random process that allows for calculating a precise null distribution of the statistic. The permutation test uses both of these assumptions to define “by chance” by assuming the data we saw we would have seen anyway even if we changed the labels (i.e.any category of UrbanByRegions). Therefore, any difference we might see between the groups is due to the luck of random permutation of the labels.

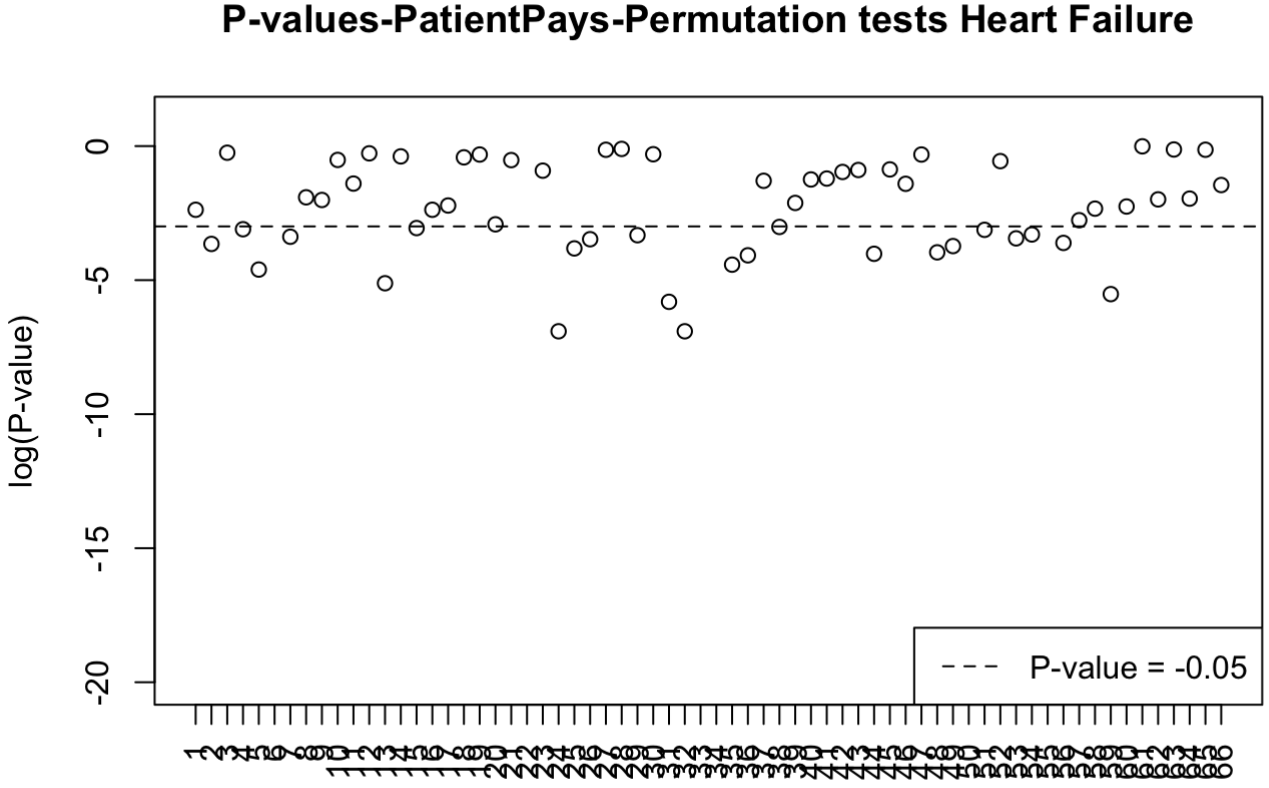
Calculating the pairwise Permutation test for all the pairs of the 4 diagnosis

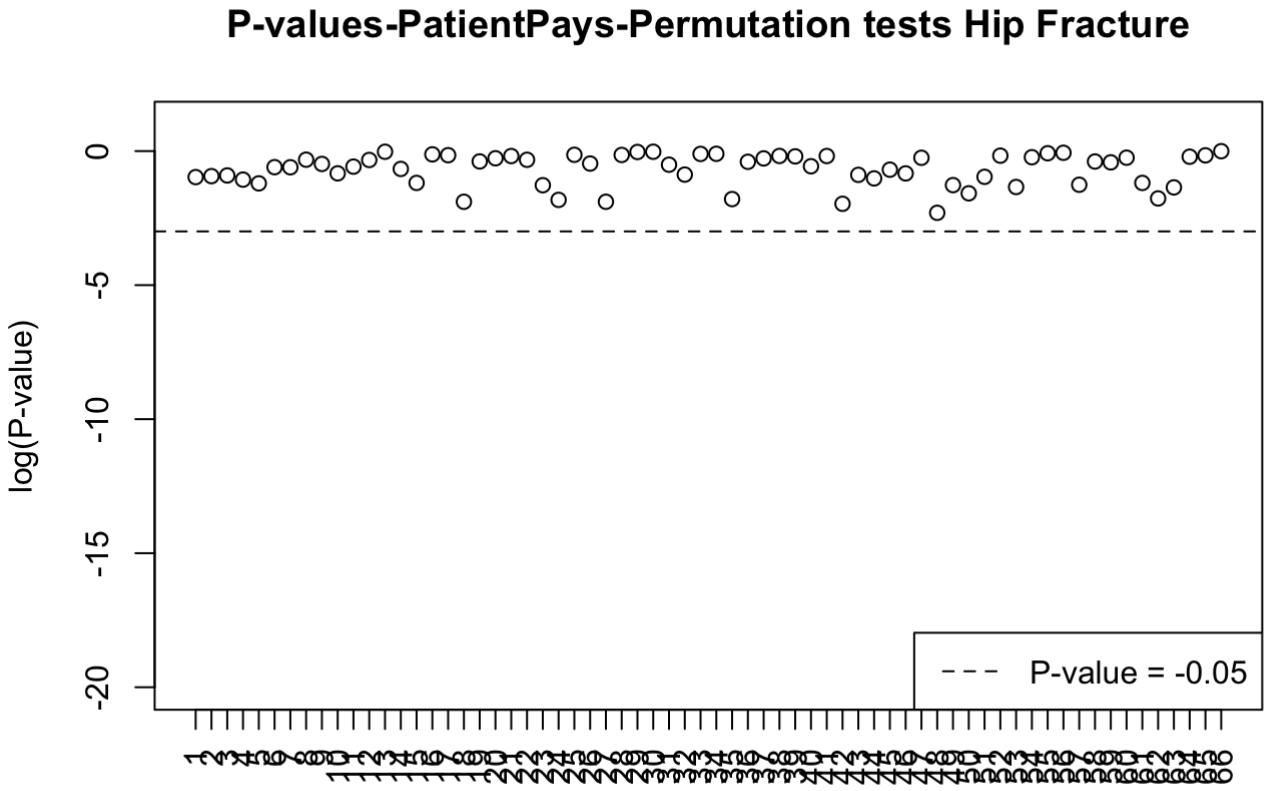
```
perm.testPairsUBR <- list()

permtestFun<-function(x = pair, df = data_frame, repetitions){
  # calculate the observed statistic
  group1 <- df[df$UrbanByRegions == x[1], ]$PatientPays
  group2 <- df[df$UrbanByRegions == x[2], ]$PatientPays
    #function to do the permutation and return statistic
  FUN <- function(x1, x2){
    x1<-na.omit(x1)
      x2<-na.omit(x2)
      return(abs(mean(x1)-mean(x2)))
  }
  stat.obs <- FUN(group1,group2)

  makePermutedStats<-function(){
      sampled <- sample(1:length(c(group1,group2)), size=length(group1),replace=FALSE) #
sample indices that will go into making new group 1
      return(FUN(c(group1, group2)[sampled], c(group1,
group2)[-sampled]))
  }
  # calculate the permuted statistic
  stat.permute <-replicate(repetitions,makePermutedStats())
  p.value <- sum(stat.permute >= stat.obs) / repetitions
  return(p.value=p.value)
}

for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  perm.testPairsUBR[[i]]<-combn(x=UBRgroups,m=2, FUN=permtestFun, repetitions = 1000, df
 = dff_temporary)
}
```

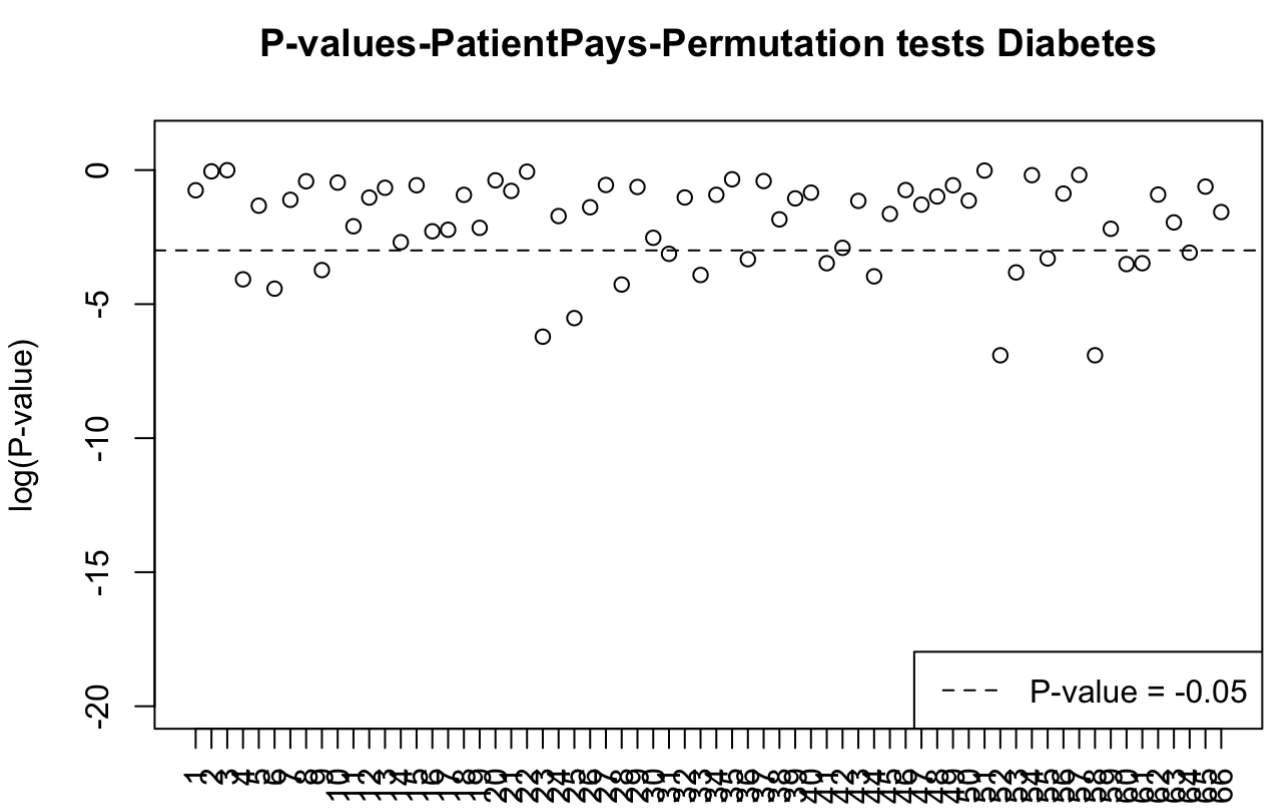
We can plot these p-values to get an idea of their value.

```
for (i in c(1:4)) {
  plot(log(perm.testPairsUBR[[i]]),ylab="log(P-value)",main= paste0("P-values-PatientPay
s-Permutation tests ", name_diag[i]),xaxt="n",xlab="", ylim = c(-20, 1))
abline(h=log(0.05),lty=2)
legend("bottomright",legend="P-value = -0.05",lty=2)
axis(1,at=1:length(perm.testPairsUBR[[i]]),labels=colnames(perm.testPairsUBR[[i]]),las=2
)
}
```


P-values-PatientPays-Permutation tests COPD
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05

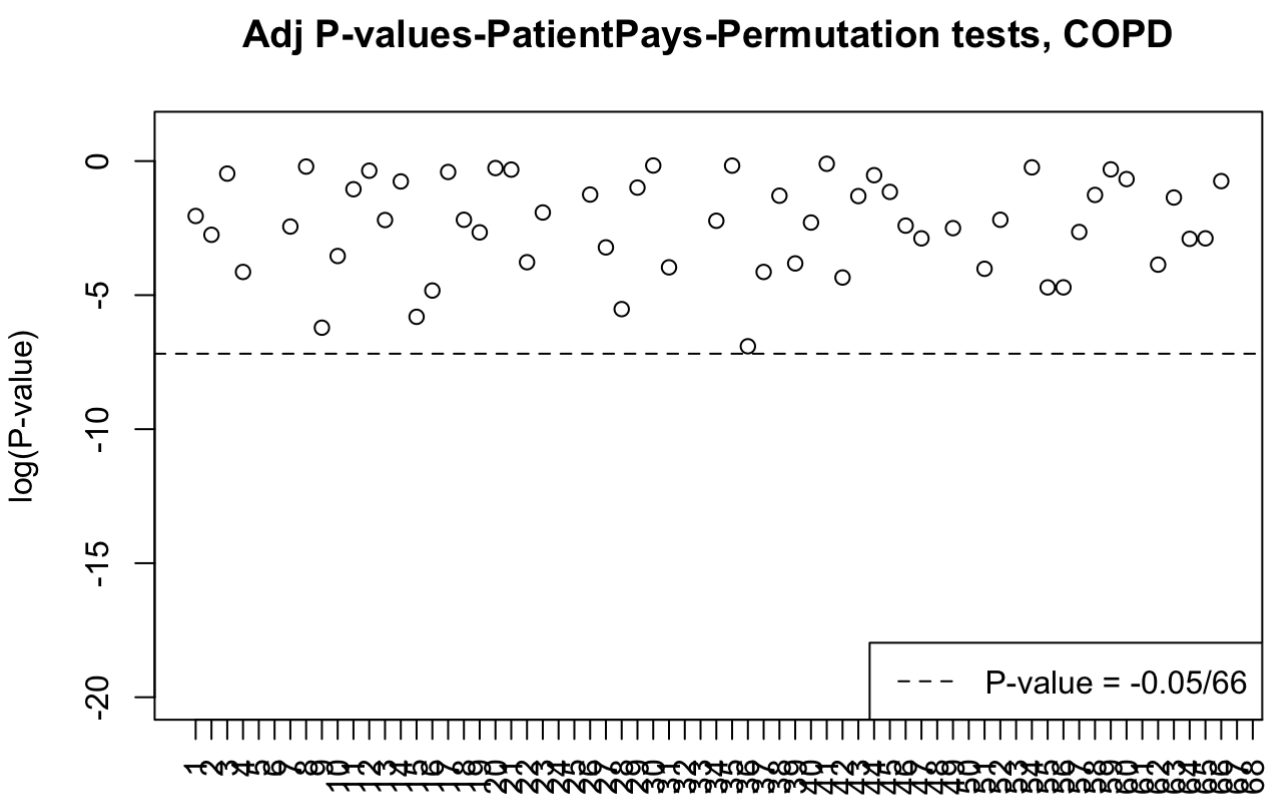

P-values-PatientPays-Permutation tests Heart Failure
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05

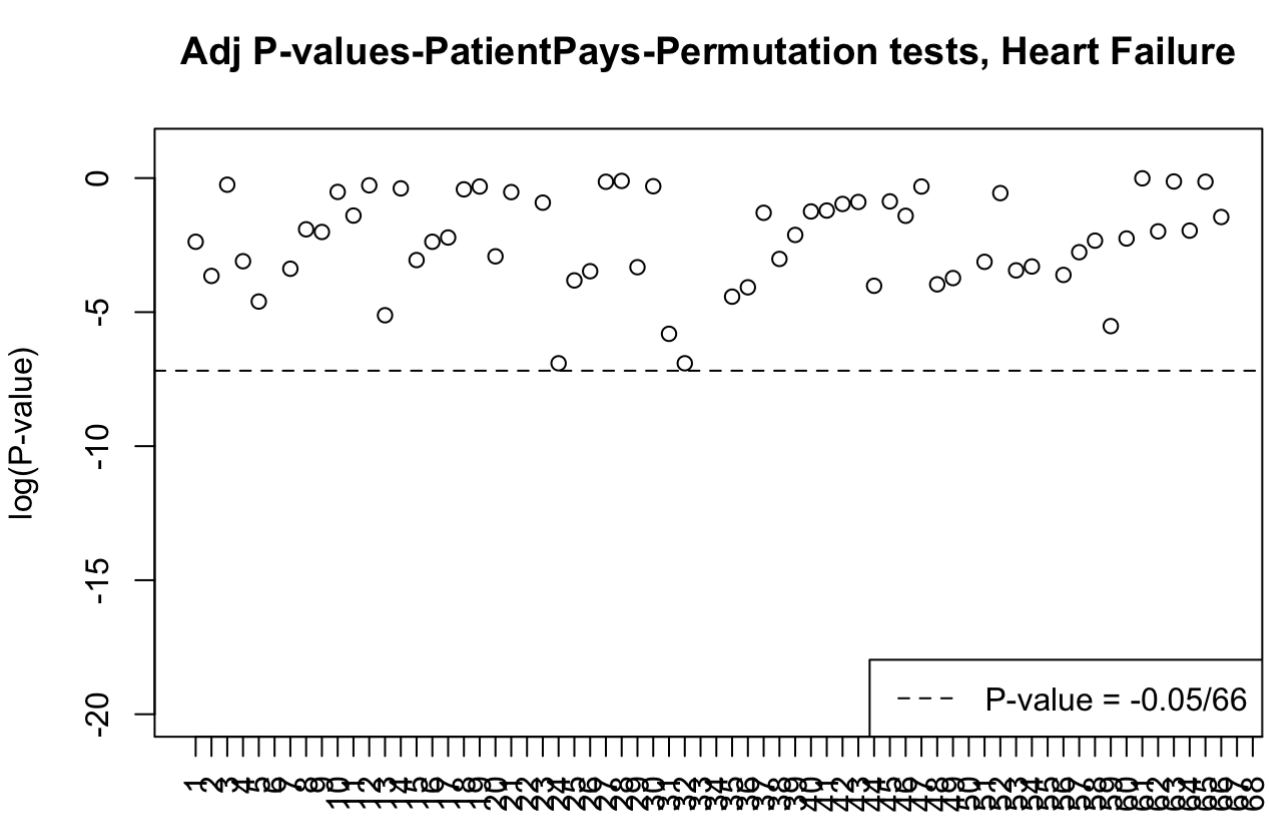


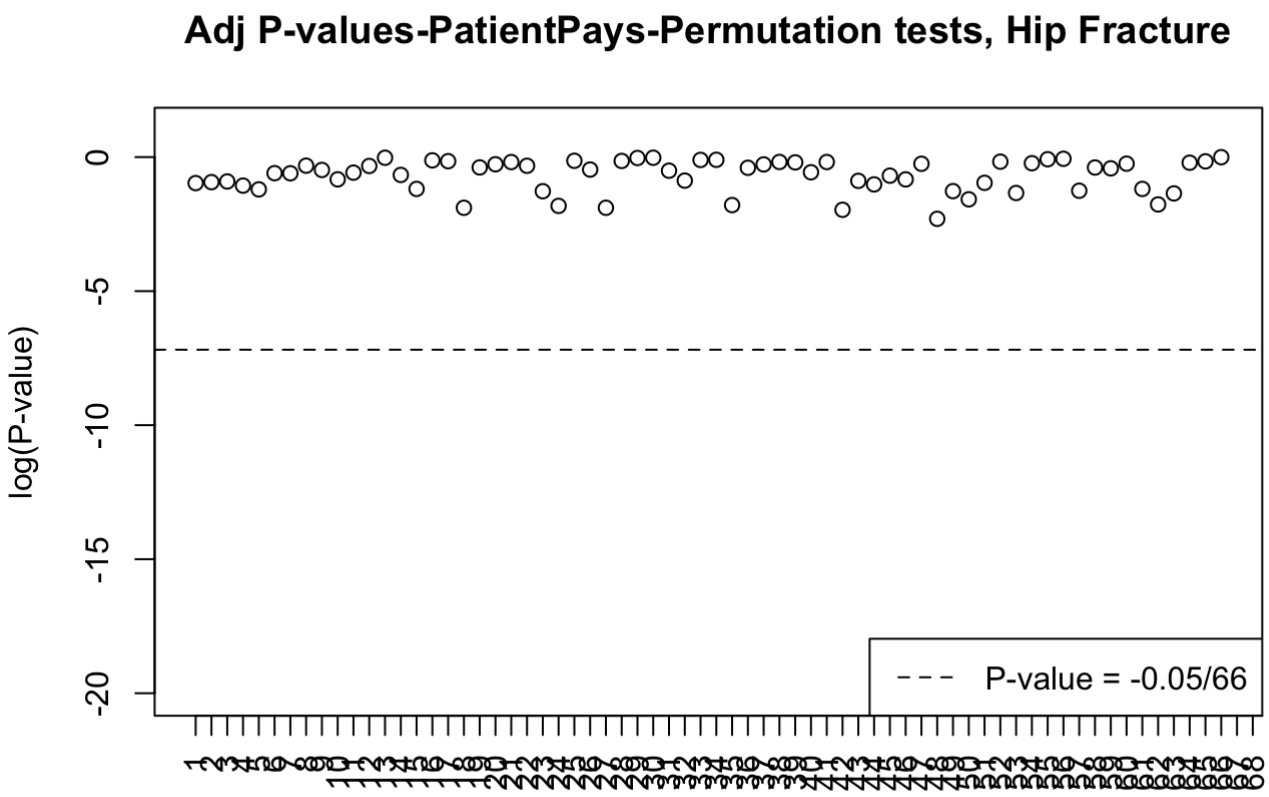


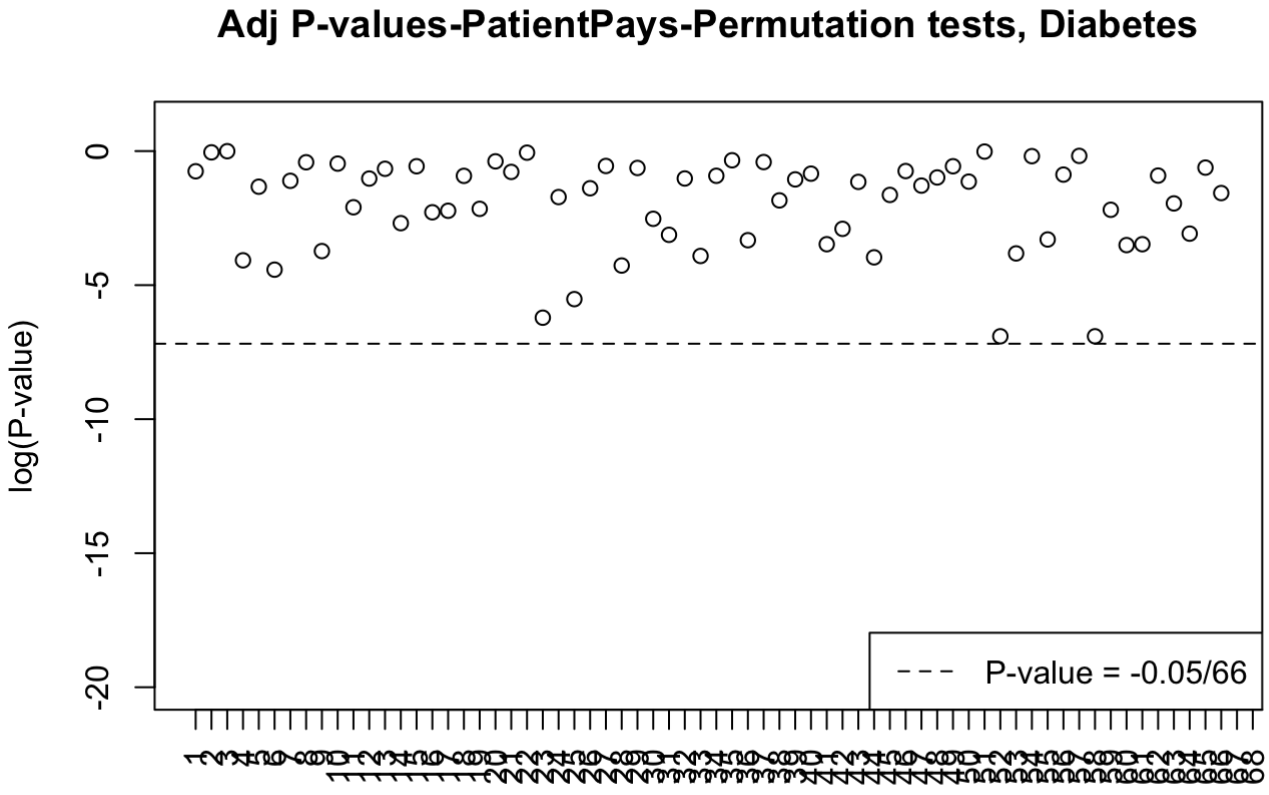
Applying Bonferroni Correction

```
for (i in c(1:4)) {
  plot(log(perm.testPairsUBR[[i]]),ylab="log(P-value)",main= paste0("Adj P-values-Patien
tPays-Permutation tests, ", name_diag[i]),xaxt="n",xlab="", ylim = c(-20, 1))
abline(h=log(0.05/npairs),lty=2)
legend("bottomright",legend="P-value = -0.05/66",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


Adj P-values-PatientPays-Permutation tests, COPD
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05/66


Adj P-values-PatientPays-Permutation tests, Heart Failure
log(P-value)
0
-5
-10
-15
-20
P-value = -0.05/66



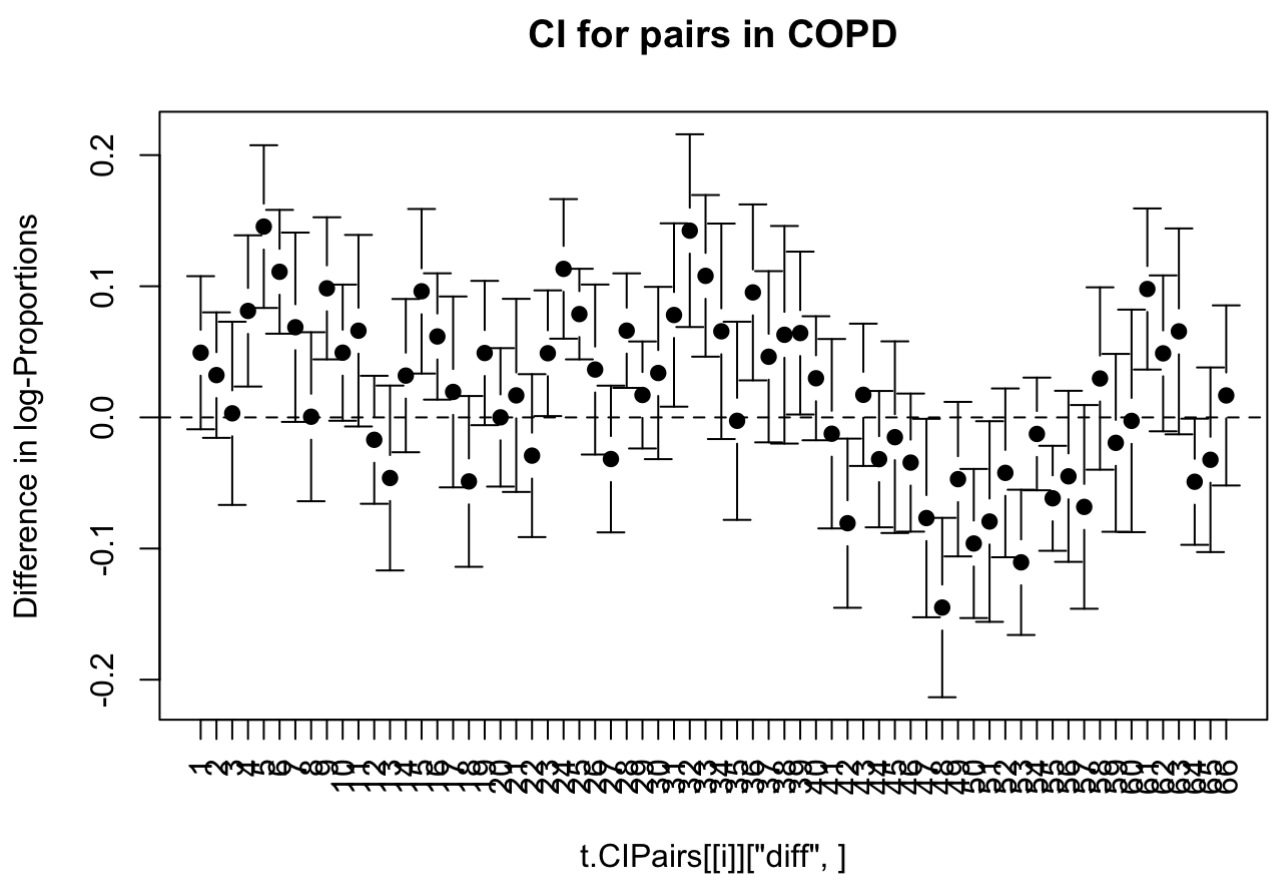


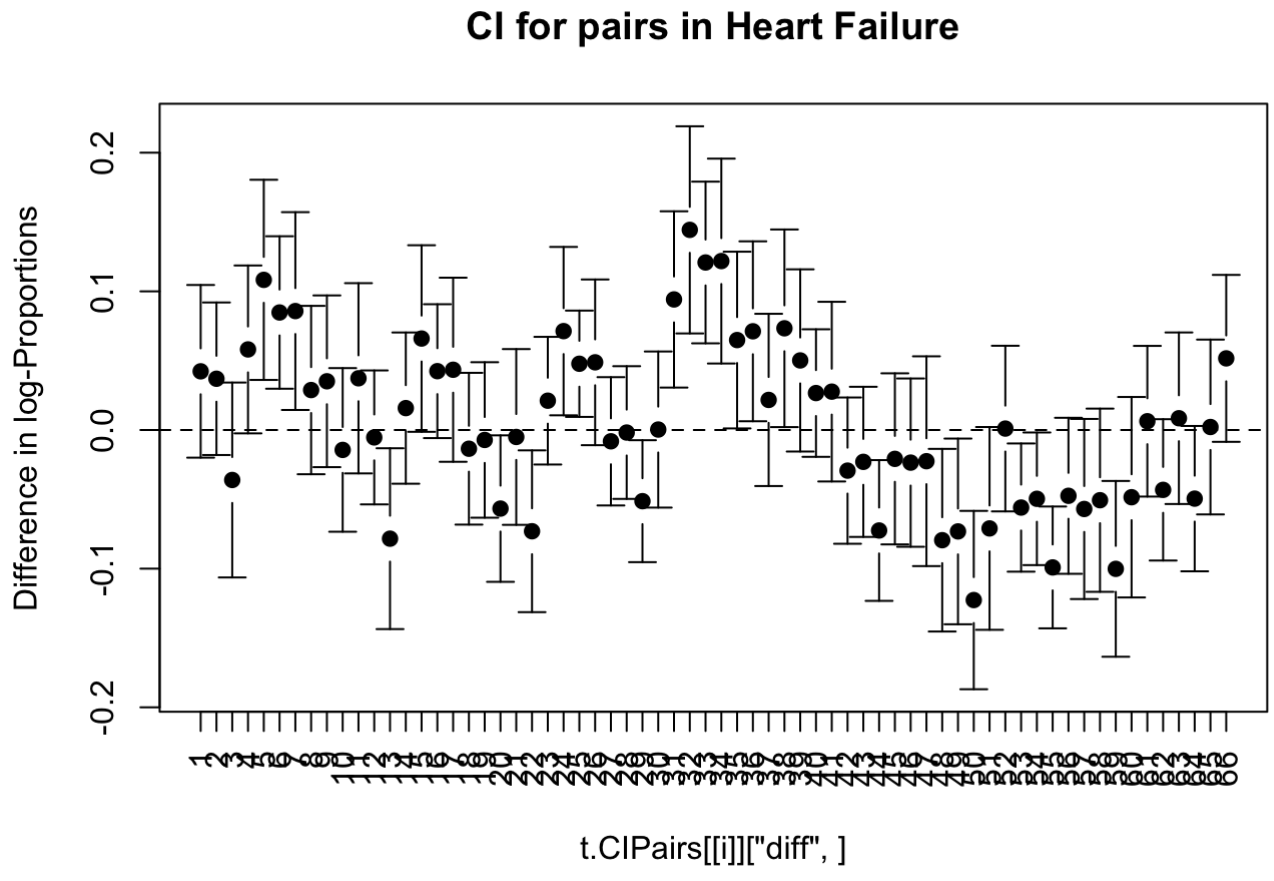
4.1.3 Creating Confidence Intervals for variabe PatientPays grouped by UrbanByRegion

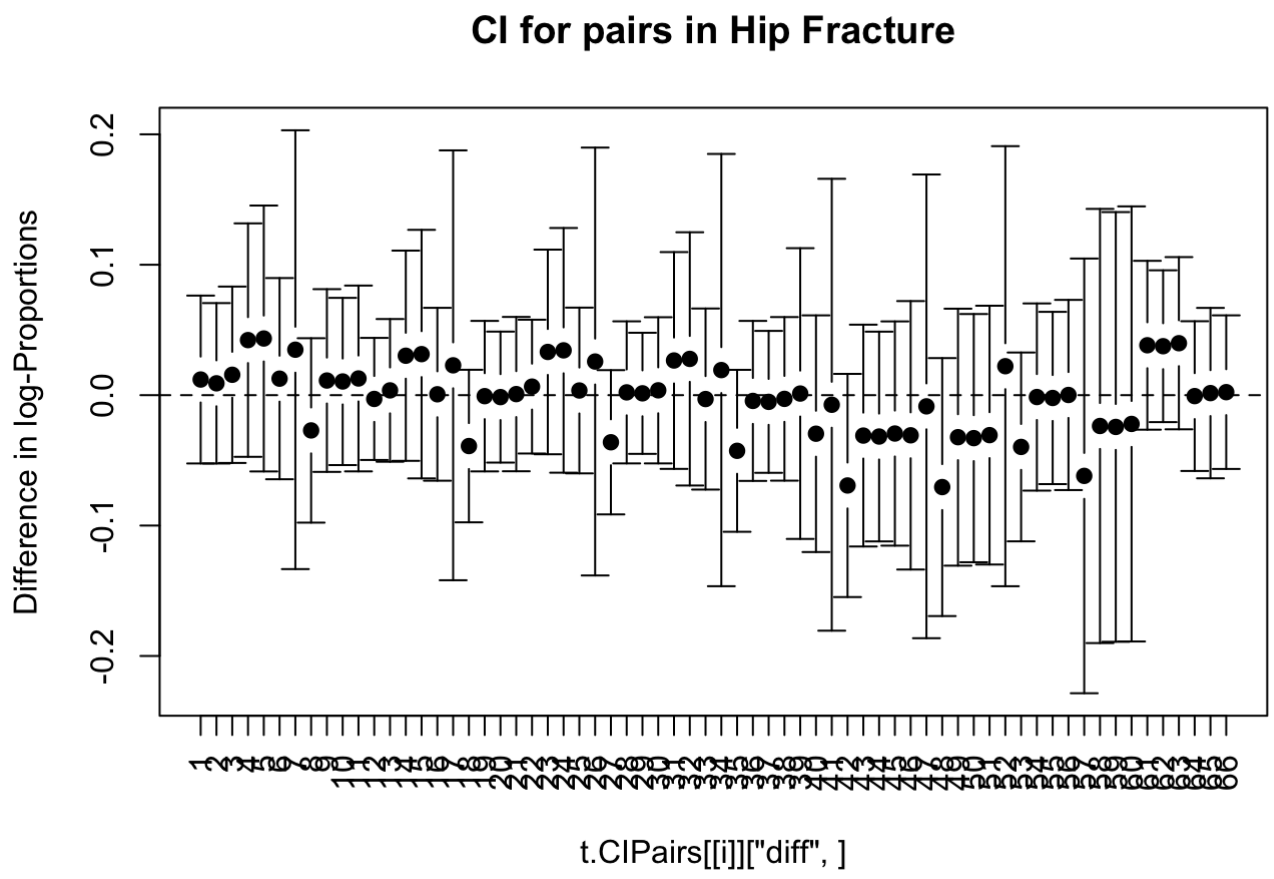
```
t.CIPairs <- list()
ttestCI <- function(x, variableName, dff_temp) {
  tout <- t.test(dff_temp$logPatientPays[dff_temp[,variableName] == x[1]],dff_temp$logPa
tientPays[dff_temp[,variableName] == x[2]])
  unlist(tout[c("estimate", "conf.int")])
}


for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.CIPairs[[i]] <- apply(X = pairsOfUBR, MARGIN = 2,
FUN = ttestCI, variableName = "UrbanByRegions", dff_temp = dff_temporary)
colnames(t.CIPairs[[i]]) <- paste(pairsOfUBR[2, ],
pairsOfUBR[1, ], sep = "--")
t.CIPairs[[i]] <- rbind(t.CIPairs[[i]], diff = t.CIPairs[[i]]["estimate.mean of x",
] - t.CIPairs[[i]]["estimate.mean of y", ])
}

require(gplots)

par(mfrow <- c(2, 2))
```

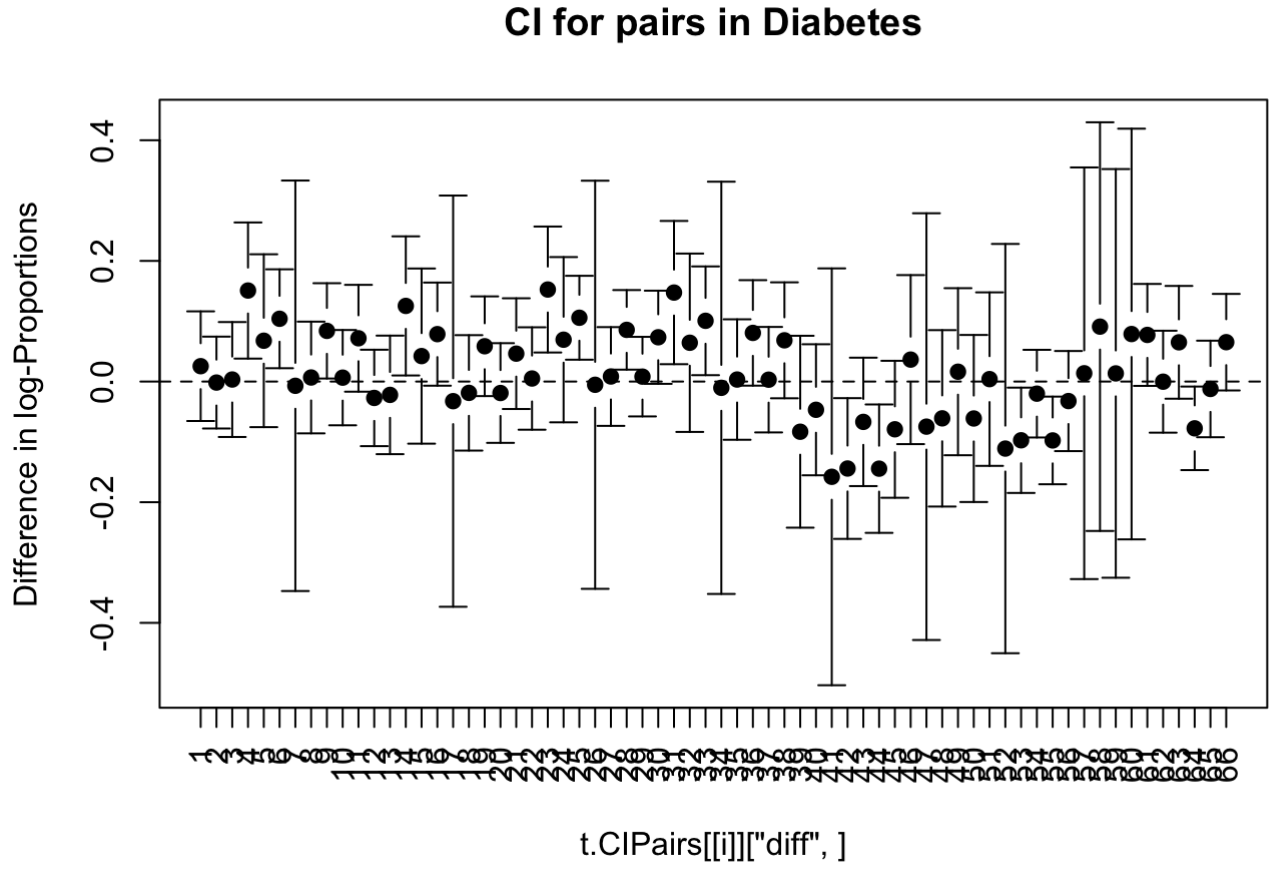
```
## NULL
```

```
for (i in c(1:4)) {
  plotCI(x = t.CIPairs[[i]]["diff", ], li = t.CIPairs[[i]]["conf.int1",
], ui = t.CIPairs[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",
pch = 19, xaxt = "n", main = paste0("CI for pairs in ", name_diag[i]))
  axis(1, at = 1:ncol(t.CIPairs[[i]]), labels = colnames(t.CIPairs), las = 2)
  abline(h = 0, lty = 2)
}
```


CI for pairs in COPD
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]

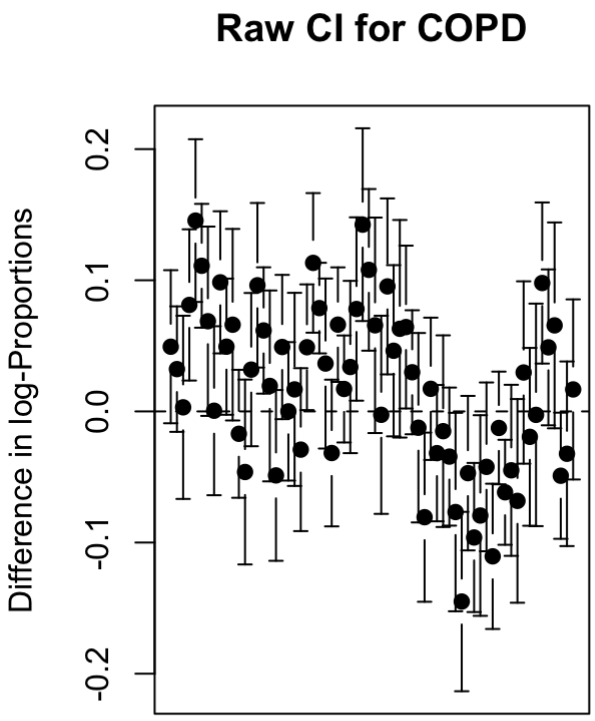

CI for pairs in Heart Failure
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]

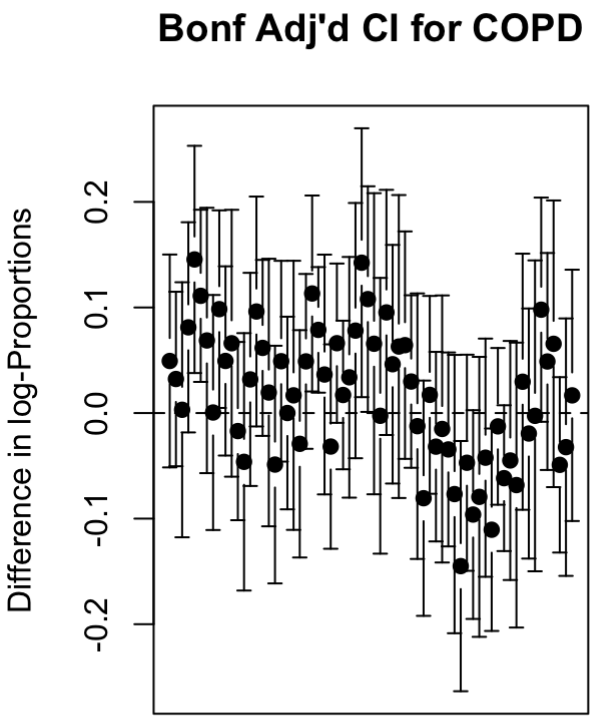

CI for pairs in Hip Fracture
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]

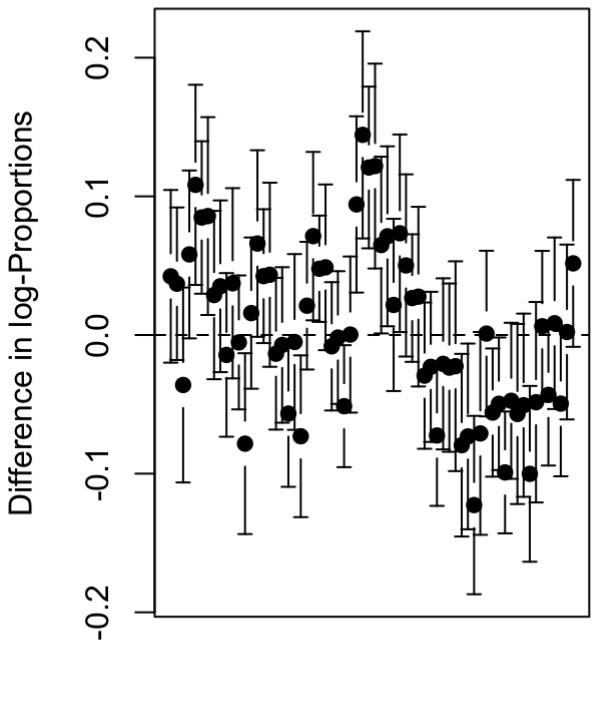

CI for pairs in Diabetes
Difference in log-Proportions
0.4
0.2
0.0
-0.2
-0.4
t.CIPairs[[i]]["diff",]

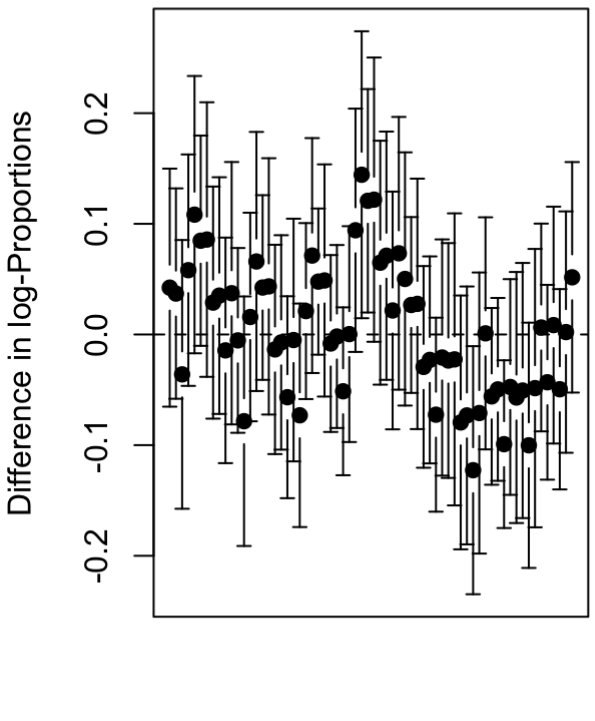
These confidence intervals suffer from the same problem as the p-values: even if the null value (0) is true in every test, roughly 5% of them will happen to not cover 0 just by chance. So we can do bonferonni corrections to the confidence intervals. Since a 95% confidence interval corresponds to a level 0.05 test, if we go to a 0.05/K level, which is the bonferonni correction, that corresponds to a 100 $*$ (1 – 0.05/K)% confidence interval.

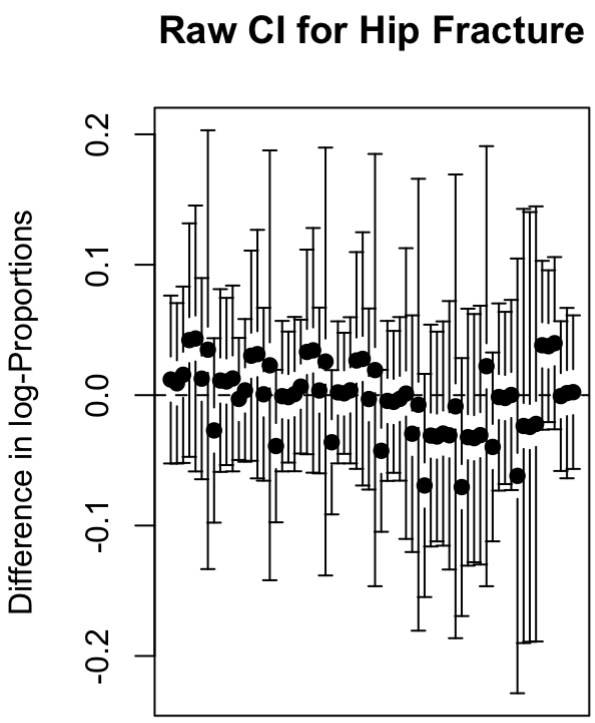
```
t.CIPairsAdj <- list()
ttestCIAdj <- function(x, variableName, dff_temp) {
  tout <- t.test(dff_temp$logPatientPays[dff_temp[,variableName] == x[1]],dff_temp$logPa
tientPays[dff_temp[,variableName] == x[2]], conf.level = 1-0.05/npairs)
  unlist(tout[c("estimate", "conf.int")])
}

for (i in c(1:4)) {
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.CIPairsAdj[[i]] <- apply(X = pairsOfUBR, MARGIN = 2,
FUN = ttestCIAdj, variableName = "UrbanByRegions", dff_temp = dff_temporary)
colnames(t.CIPairsAdj[[i]]) <- paste(pairsOfUBR[2,
], pairsOfUBR[1, ], sep = "-")
  t.CIPairsAdj[[i]] <- rbind(t.CIPairsAdj[[i]], diff = t.CIPairsAdj[[i]]["estimate.mean
 of x",
] - t.CIPairsAdj[[i]]["estimate.mean of y", ])
}
for (i in c(1:4)) {
  par(mfrow = c(1, 2))
  plotCI(x = t.CIPairs[[i]]["diff", ], li = t.CIPairs[[i]]["conf.int1",
], ui = t.CIPairs[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",
sub = "Raw CI", pch = 19, xaxt = "n", main = paste0("Raw CI for ",name_diag[i]))
  abline(h = 0, lty = 2)
  plotCI(x = t.CIPairsAdj[[i]]["diff", ], li = t.CIPairsAdj[[i]]["conf.int1",
], ui = t.CIPairsAdj[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",sub =
"Bonferonni Adjusted CI", pch = 19, xaxt = "n", , main = paste0("Bonf Adj'd CI for ",nam
e_diag[i]))
  abline(h = 0, lty = 2)
}
```

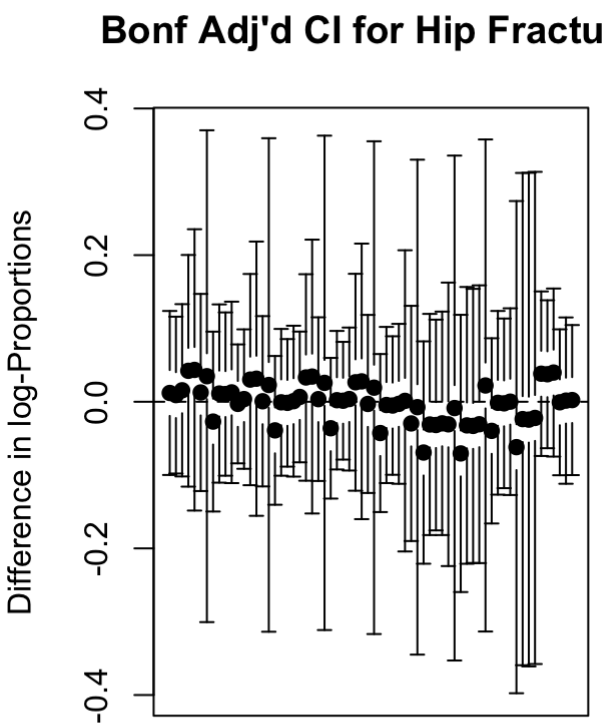

Raw CI for COPD
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]
Raw CI

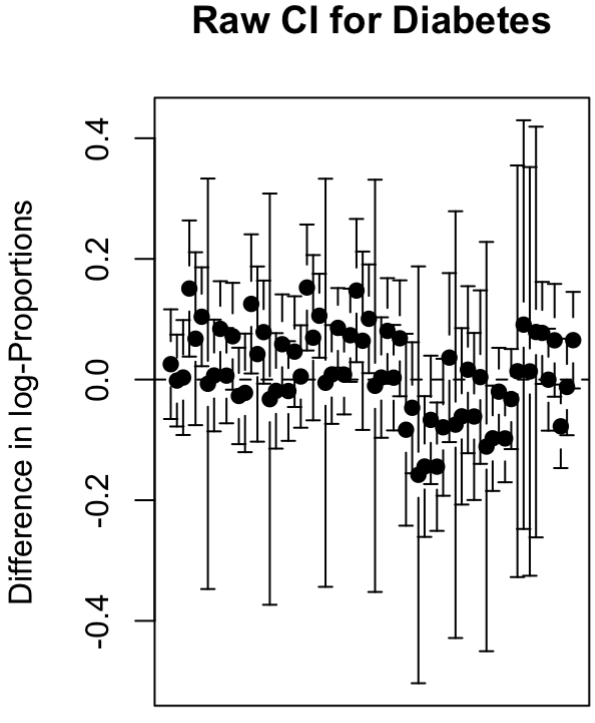

Bonf Adj'd CI for COPD
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI

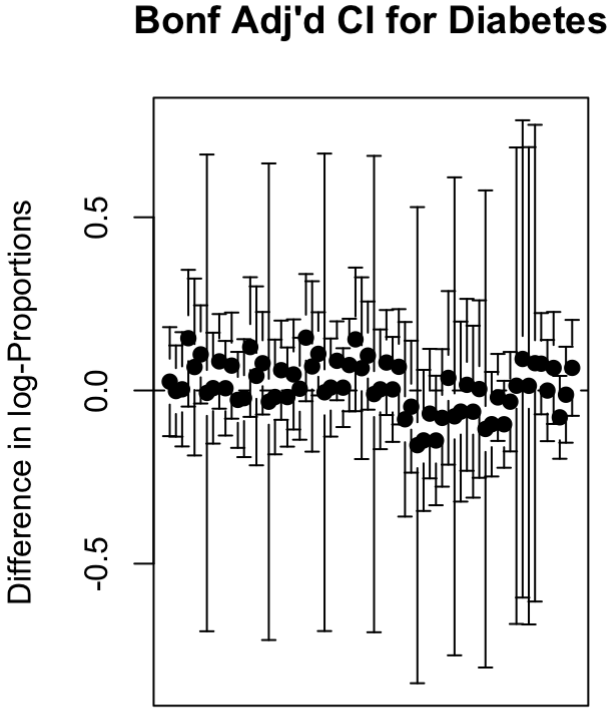

Raw CI for Heart Failure
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]
Raw CI


Bonf Adj'd CI for Heart Failure
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI


Raw CI for Hip Fracture
Difference in log-Proportions
0.2
0.1
0.0
-0.1
-0.2
t.CIPairs[[i]]["diff",]
Raw CI


Bonf Adj'd CI for Hip Fracture
Difference in log-Proportions
0.4
0.2
0.0
-0.2
-0.4
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI


Raw CI for Diabetes
Difference in log-Proportions
0.4
0.2
0.0
-0.2
-0.4
t.CIPairs[[i]]["diff",]
Raw CI


Bonf Adj'd CI for Diabetes
Difference in log-Proportions
0.5
0.0
-0.5
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI

4.2 Drawing Inference of the difference due to UrbanByRegion in PctPatientPays Variable

4.2.1 Parametric Testing of PctPatientPays

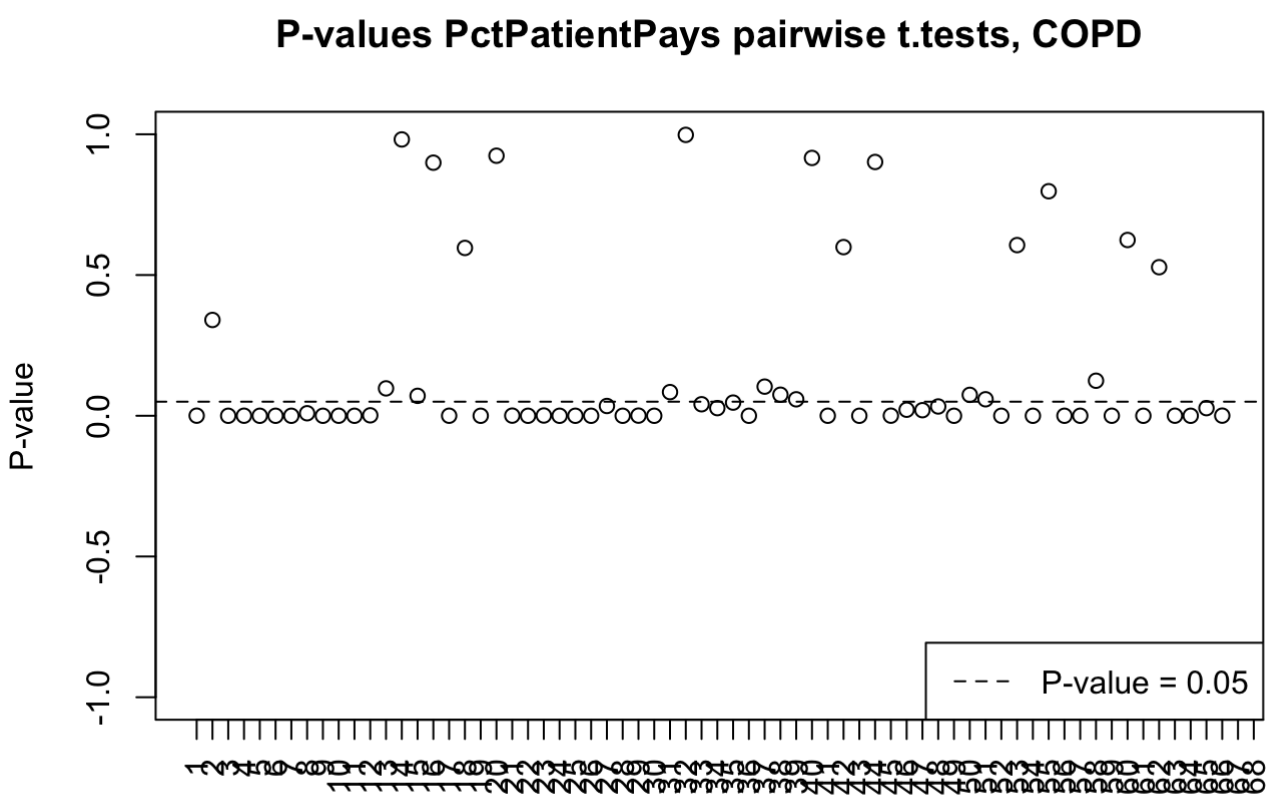
We need to compare all the categories of UrbanBy Regions and test whether any differences we see in the two response variables is due to chance. Let's start by comparing them all parametrically using t.test for all four diagnosis.

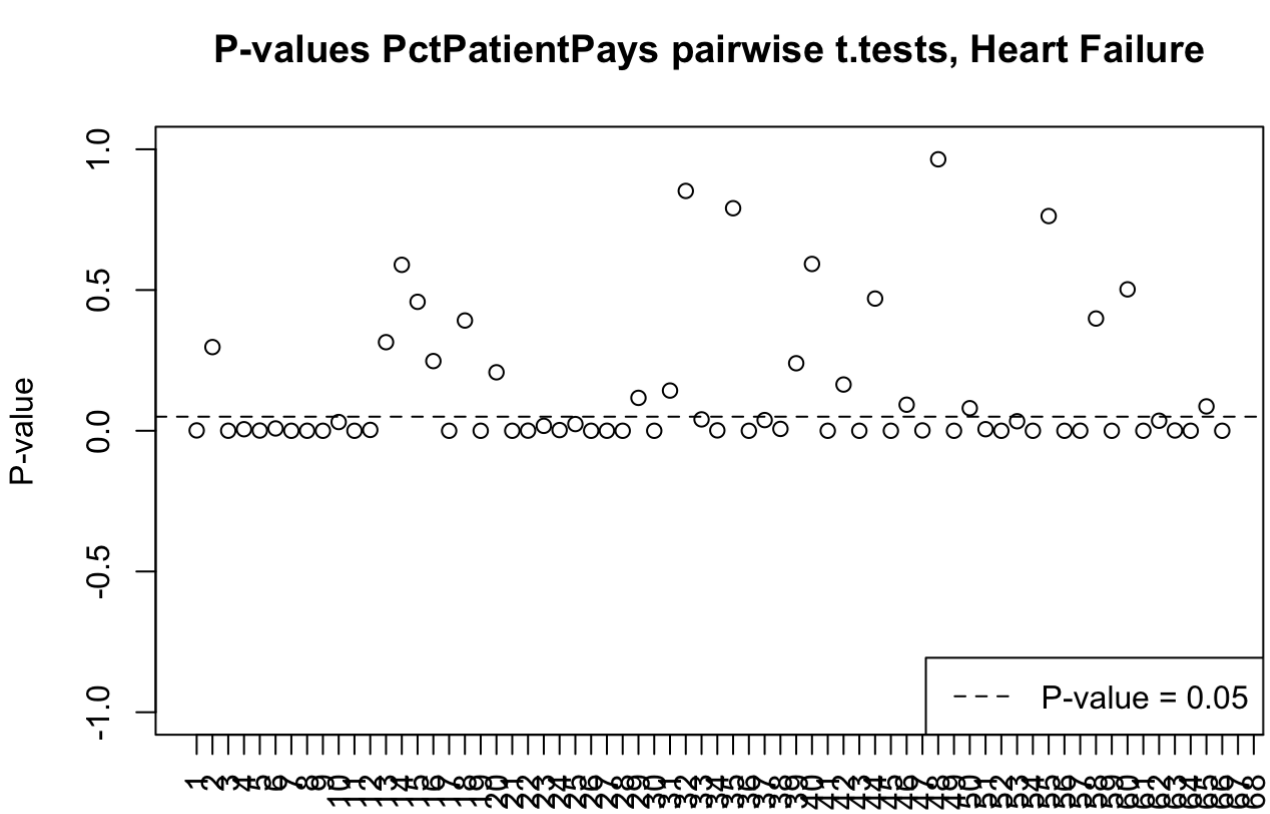
Calculating the pairwise T-Test for all the pairs of the 4 diagnosis

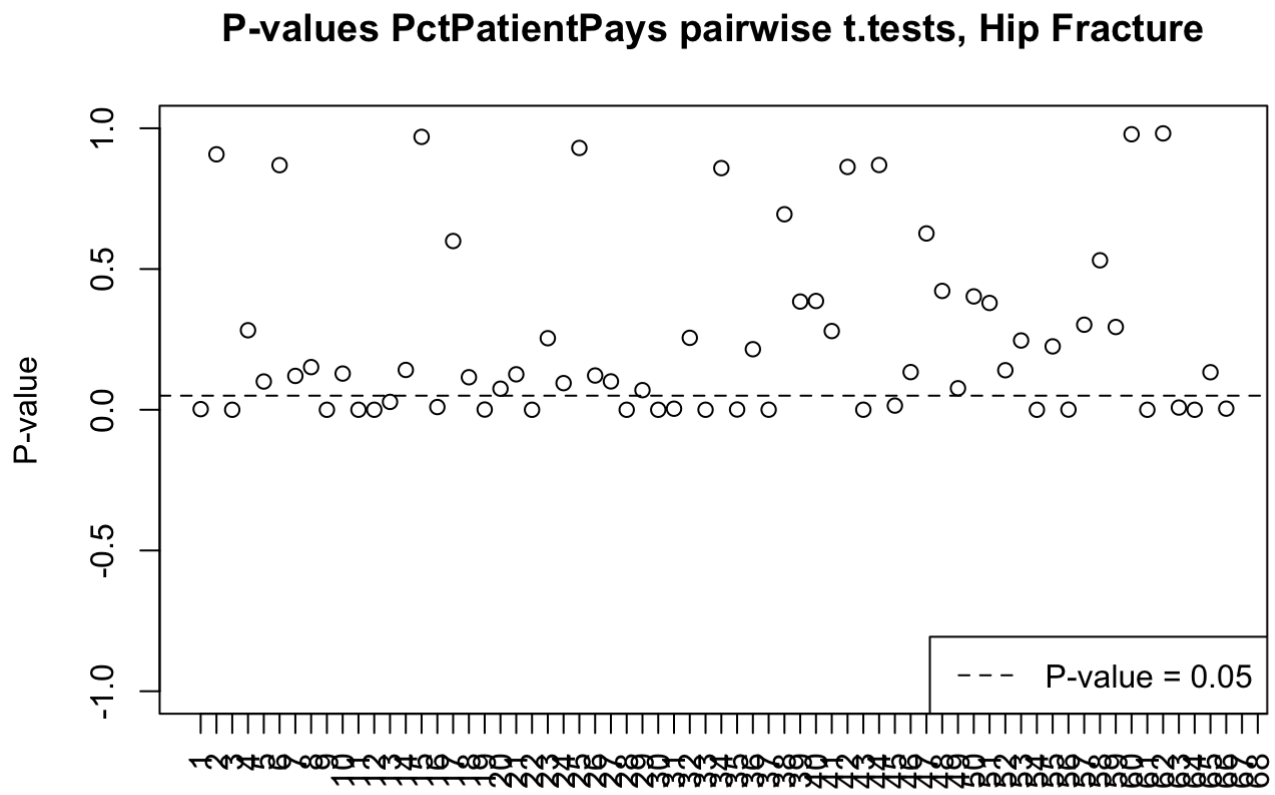
```
t.testPairsUBR <- list()

ttestFun<-function(x,variableName, dff_temp){
    tout<-t.test(dff_temp$PctPatientPays[dff_temp[,variableName] == x[1]],dff_temp$PctPa
tientPays[dff_temp[,variableName] == x[2]])
    unlist(tout[c("statistic","p.value", "conf.int", "estimate")]) #unlist makes it a ve
ctor rather than list
}

for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.testPairsUBR[[i]]<-combn(x=UBRgroups,m=2, FUN=ttestFun, variableName="UrbanByRegion
s", dff_temp = dff_temporary)
}
```

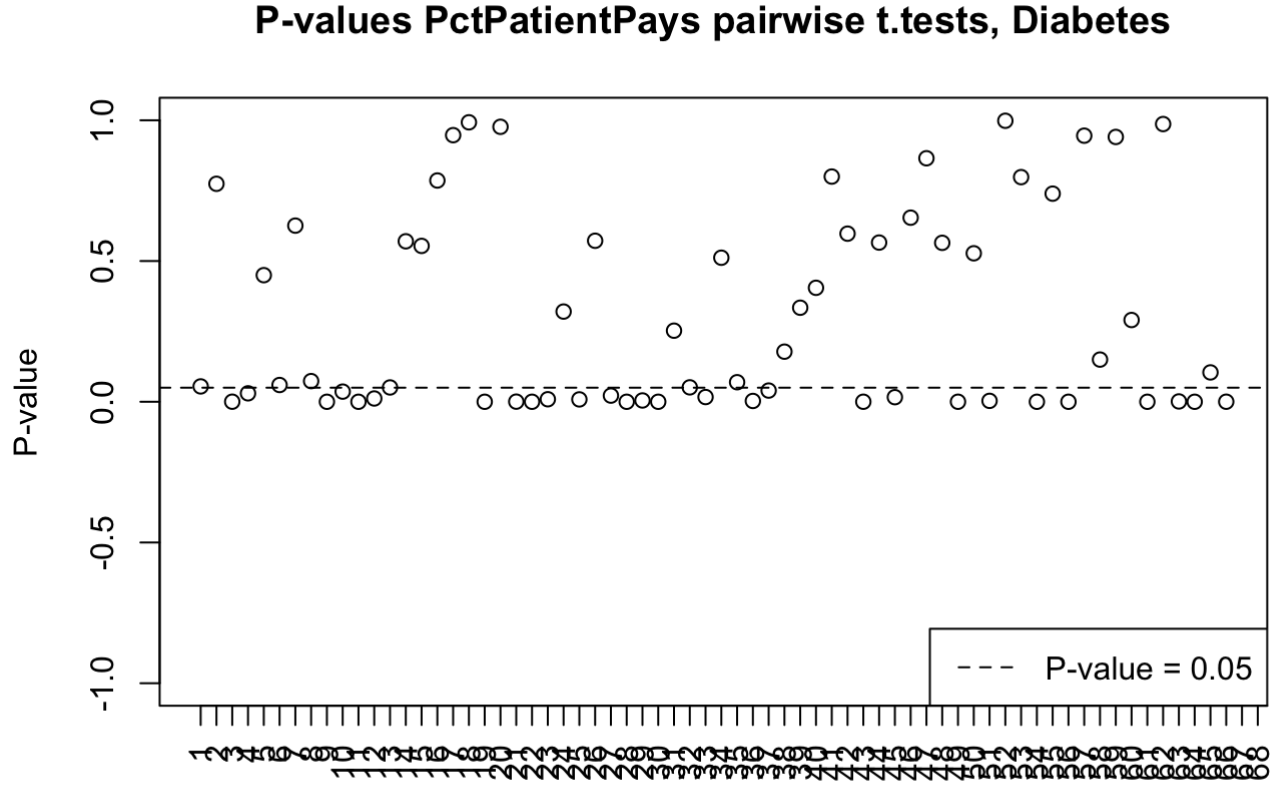
We can plot these p-values to get an idea of their value.

```
for (i in c(1:4)) {
  plot(t.testPairsUBR[[i]][2, ],ylab="P-value",main= paste0("P-values PctPatientPays pai
rwise t.tests, ", name_diag[i]),xaxt="n",xlab="", ylim = c(-1,1))
abline(h=0.05,lty=2)
legend("bottomright",legend="P-value = 0.05",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


P-values PctPatientPays pairwise t.tests, COPD
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05

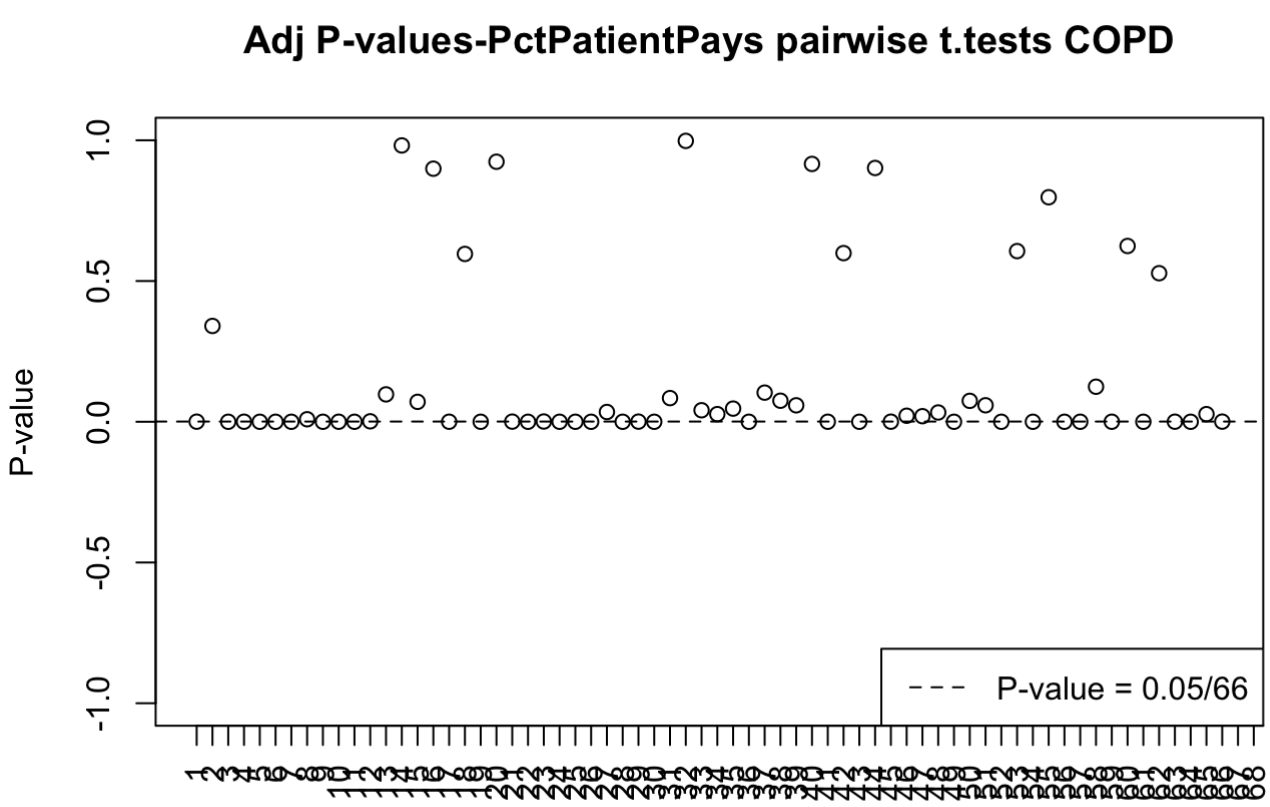

P-values PctPatientPays pairwise t.tests, Heart Failure
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05

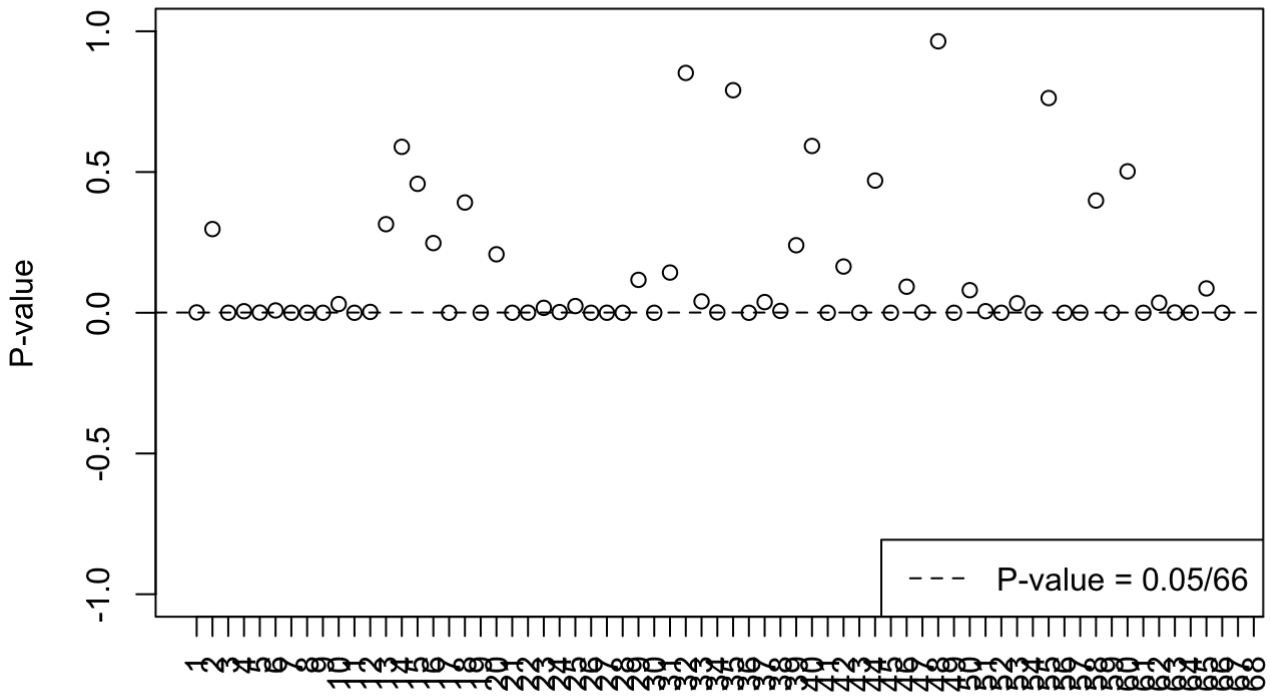


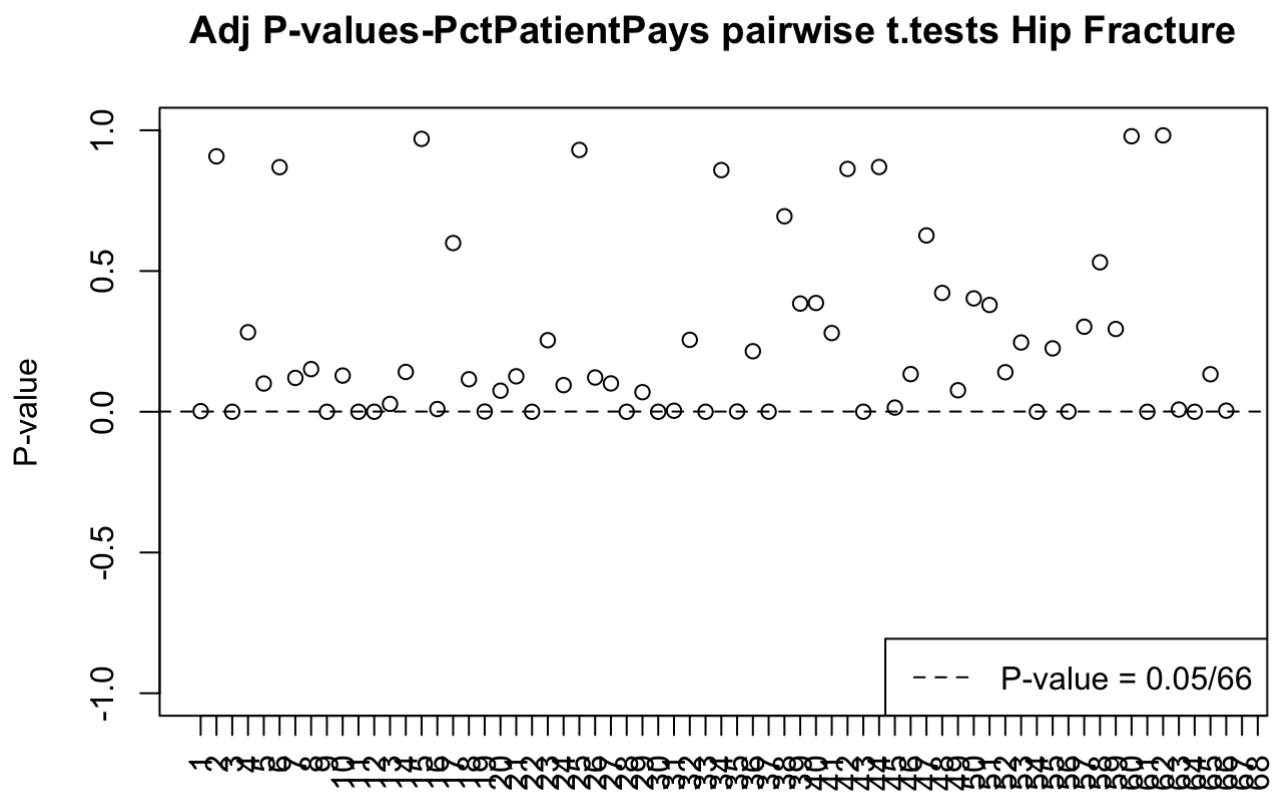


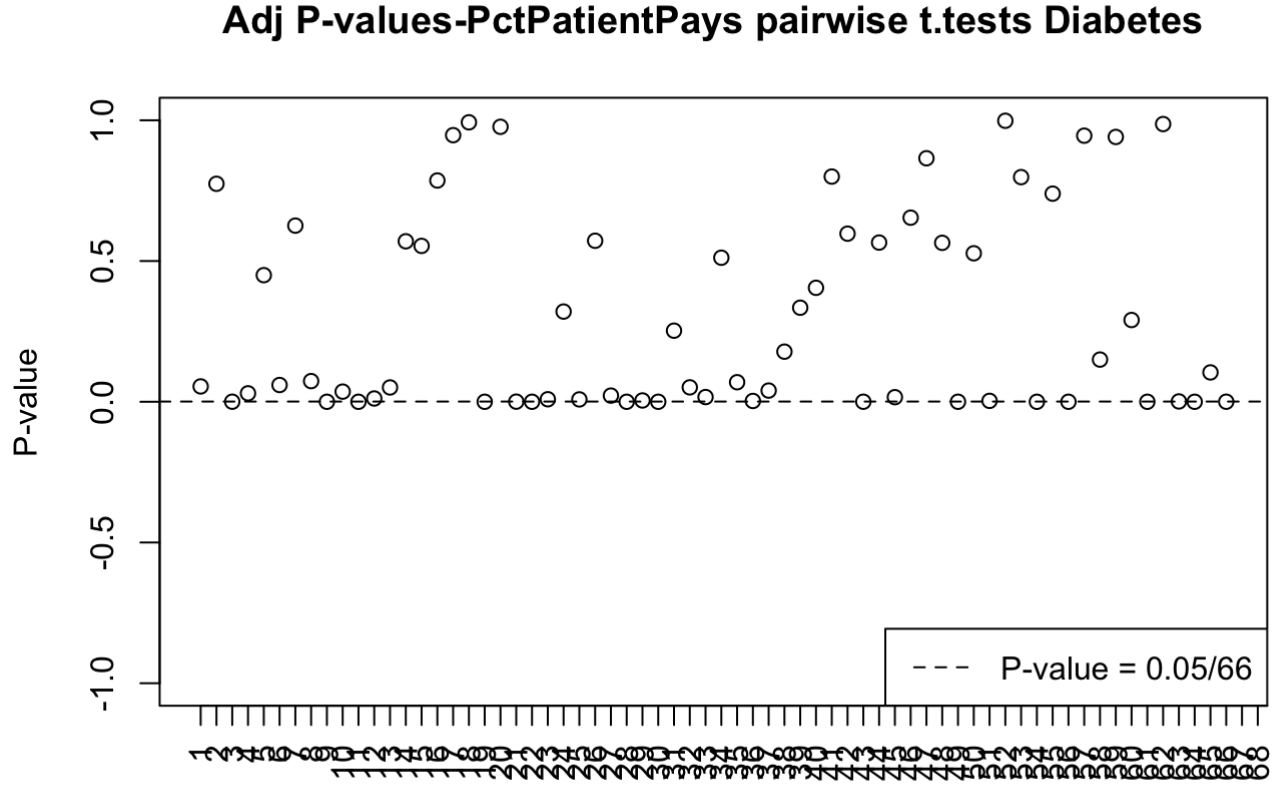
Applying Bonferroni Correction

```
for (i in c(1:4)) {
  plot(t.testPairsUBR[[i]][2, ],ylab="P-value",main= paste0("Adj P-values-PctPatientPays
 pairwise t.tests ", name_diag[i]),xaxt="n",xlab="", ylim = c(-1, 1))
abline(h=0.05/npairs,lty=2)
legend("bottomright",legend="P-value = 0.05/66",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


Adj P-values-PctPatientPays pairwise t.tests COPD
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05/66


Adj P-values-PctPatientPays pairwise t.tests Heart Failure
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05/66





Now,

let’s collect and analyse the pairs that were significantly different for the PatientPays.

```
significantly_different_pairs <- list()
for (i in c(1:4)){
  significantly_different_pairs[[i]] <- pairsOfUBR[,which(t.testPairsUBR[[i]][2, ] < 0.0
5/66)]
}

for (i in c(1:4)) {
  print(paste0("Pairs having significant differences for diagnosis ", name_diag[i], " ar
e the following :"))
 print(significantly_different_pairs[[i]] )
}
```

```
## [1] "Pairs having significant differences for diagnosis COPD are the following :"
##      [,1]            [,2]          [,3]
## [1,] "mix:midwest"   "mix:midwest" "mix:midwest"
## [2,] "mix:northeast" "mix:west"    "rural_urbanclusters:midwest"
##      [,4]                            [,5]
## [1,] "mix:midwest"                   "mix:midwest"
## [2,] "rural_urbanclusters:northeast" "rural_urbanclusters:south"
##      [,6]                       [,7]
## [1,] "mix:midwest"              "mix:midwest"
## [2,] "rural_urbanclusters:west" "only_urbanarea:northeast"
##      [,8]                   [,9]
## [1,] "mix:midwest"          "mix:midwest"
## [2,] "only_urbanarea:south" "only_urbanarea:west"
##      [,10]                      [,11]
## [1,] "mix:northeast"            "mix:northeast"
## [2,] "rural_urbanclusters:west" "only_urbanarea:northeast"
##      [,12]                 [,13]       [,14]
## [1,] "mix:northeast"       "mix:south" "mix:south"
## [2,] "only_urbanarea:west" "mix:west"  "rural_urbanclusters:northeast"
##      [,15]                       [,16]
## [1,] "mix:south"                 "mix:south"
## [2,] "rural_urbanclusters:south" "rural_urbanclusters:west"
##      [,17]                      [,18]
## [1,] "mix:south"                "mix:south"
## [2,] "only_urbanarea:northeast" "only_urbanarea:south"
##      [,19]                 [,20]
## [1,] "mix:south"           "mix:west"
## [2,] "only_urbanarea:west" "only_urbanarea:northeast"
##      [,21]                          [,22]
## [1,] "rural_urbanclusters:midwest" "rural_urbanclusters:midwest"
## [2,] "rural_urbanclusters:west"    "only_urbanarea:northeast"
##      [,23]                          [,24]
## [1,] "rural_urbanclusters:midwest" "rural_urbanclusters:northeast"
## [2,] "only_urbanarea:west"         "only_urbanarea:northeast"
##      [,25]                       [,26]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:south"
## [2,] "rural_urbanclusters:west"  "only_urbanarea:northeast"
##      [,27]                       [,28]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:west"
## [2,] "only_urbanarea:west"       "only_urbanarea:midwest"
##      [,29]                      [,30]
## [1,] "rural_urbanclusters:west" "only_urbanarea:midwest"
## [2,] "only_urbanarea:south"     "only_urbanarea:northeast"
##      [,31]                   [,32]
## [1,] "only_urbanarea:midwest" "only_urbanarea:northeast"
## [2,] "only_urbanarea:west"    "only_urbanarea:south"
##      [,33]
## [1,] "only_urbanarea:south"
## [2,] "only_urbanarea:west"
## [1] "Pairs having significant differences for diagnosis Heart Failure are the followi
ng :"
##      [,1]          [,2]                       [,3]
## [1,] "mix:midwest" "mix:midwest"              "mix:midwest"
```

```
## [2,] "mix:west"    "rural_urbanclusters:west" "only_urbanarea:midwest"
##      [,4]                        [,5]
## [1,] "mix:midwest"               "mix:midwest"
## [2,] "only_urbanarea:northeast" "only_urbanarea:west"
##      [,6]                        [,7]
## [1,] "mix:northeast"             "mix:northeast"
## [2,] "rural_urbanclusters:west" "only_urbanarea:northeast"
##      [,8]                  [,9]        [,10]
## [1,] "mix:northeast"       "mix:south" "mix:south"
## [2,] "only_urbanarea:west" "mix:west"  "rural_urbanclusters:west"
##      [,11]                    [,12]
## [1,] "mix:south"              "mix:south"
## [2,] "only_urbanarea:midwest" "only_urbanarea:northeast"
##      [,13]                 [,14]
## [1,] "mix:south"           "mix:west"
## [2,] "only_urbanarea:west" "only_urbanarea:northeast"
##      [,15]                          [,16]
## [1,] "rural_urbanclusters:midwest" "rural_urbanclusters:midwest"
## [2,] "rural_urbanclusters:west"    "only_urbanarea:northeast"
##      [,17]                          [,18]
## [1,] "rural_urbanclusters:midwest" "rural_urbanclusters:northeast"
## [2,] "only_urbanarea:west"         "only_urbanarea:northeast"
##      [,19]                        [,20]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:south"
## [2,] "rural_urbanclusters:west"  "only_urbanarea:northeast"
##      [,21]                        [,22]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:west"
## [2,] "only_urbanarea:west"       "only_urbanarea:midwest"
##      [,23]                       [,24]
## [1,] "rural_urbanclusters:west" "only_urbanarea:midwest"
## [2,] "only_urbanarea:south"     "only_urbanarea:northeast"
##      [,25]                       [,26]
## [1,] "only_urbanarea:northeast" "only_urbanarea:south"
## [2,] "only_urbanarea:south"     "only_urbanarea:west"
## [1] "Pairs having significant differences for diagnosis Hip Fracture are the followin
g :"
##      [,1]          [,2]                        [,3]
## [1,] "mix:midwest" "mix:midwest"               "mix:midwest"
## [2,] "mix:west"    "only_urbanarea:northeast" "only_urbanarea:west"
##      [,4]            [,5]        [,6]
## [1,] "mix:northeast" "mix:south" "mix:south"
## [2,] "mix:south"     "mix:west"  "only_urbanarea:northeast"
##      [,7]                  [,8]
## [1,] "mix:south"           "mix:west"
## [2,] "only_urbanarea:west" "rural_urbanclusters:south"
##      [,9]                   [,10]
## [1,] "mix:west"             "rural_urbanclusters:midwest"
## [2,] "only_urbanarea:south" "only_urbanarea:northeast"
##      [,11]                        [,12]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:south"
## [2,] "only_urbanarea:northeast"  "only_urbanarea:west"
##      [,13]                       [,14]
## [1,] "only_urbanarea:midwest"    "only_urbanarea:northeast"
## [2,] "only_urbanarea:northeast" "only_urbanarea:south"
```

```
## [1] "Pairs having significant differences for diagnosis Diabetes are the following :"
##      [,1]          [,2]                       [,3]
## [1,] "mix:midwest" "mix:midwest"              "mix:midwest"
## [2,] "mix:west"    "only_urbanarea:northeast" "only_urbanarea:west"
##      [,4]                       [,5]                  [,6]
## [1,] "mix:northeast"            "mix:northeast"       "mix:south"
## [2,] "only_urbanarea:northeast" "only_urbanarea:west" "mix:west"
##      [,7]                       [,8]
## [1,] "mix:south"                "mix:south"
## [2,] "only_urbanarea:northeast" "only_urbanarea:west"
##      [,9]                          [,10]
## [1,] "rural_urbanclusters:midwest" "rural_urbanclusters:northeast"
## [2,] "only_urbanarea:northeast"    "only_urbanarea:northeast"
##      [,11]                        [,12]
## [1,] "rural_urbanclusters:south" "rural_urbanclusters:south"
## [2,] "only_urbanarea:northeast"  "only_urbanarea:west"
##      [,13]                      [,14]
## [1,] "only_urbanarea:midwest"   "only_urbanarea:northeast"
## [2,] "only_urbanarea:northeast" "only_urbanarea:south"
##      [,15]
## [1,] "only_urbanarea:south"
## [2,] "only_urbanarea:west"
```

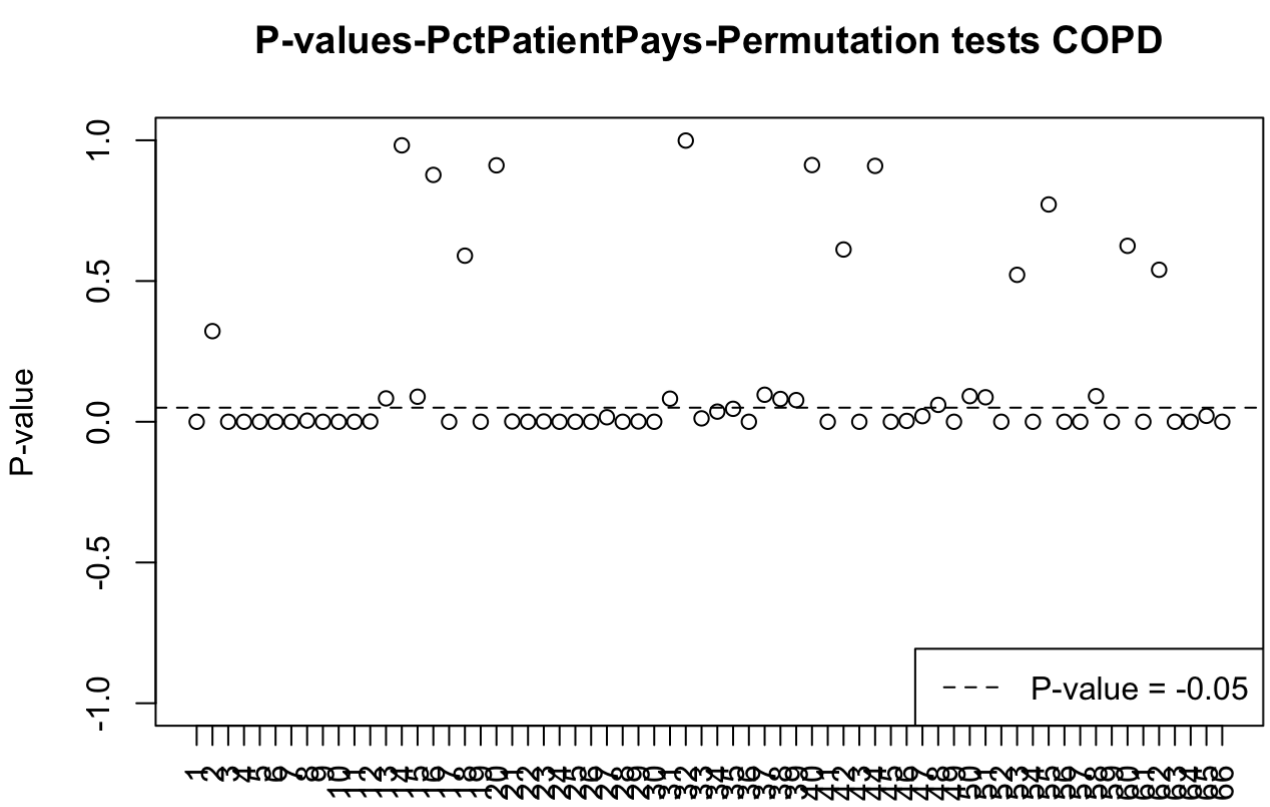
4.2.2 Permutation Testing for PctPatientPays

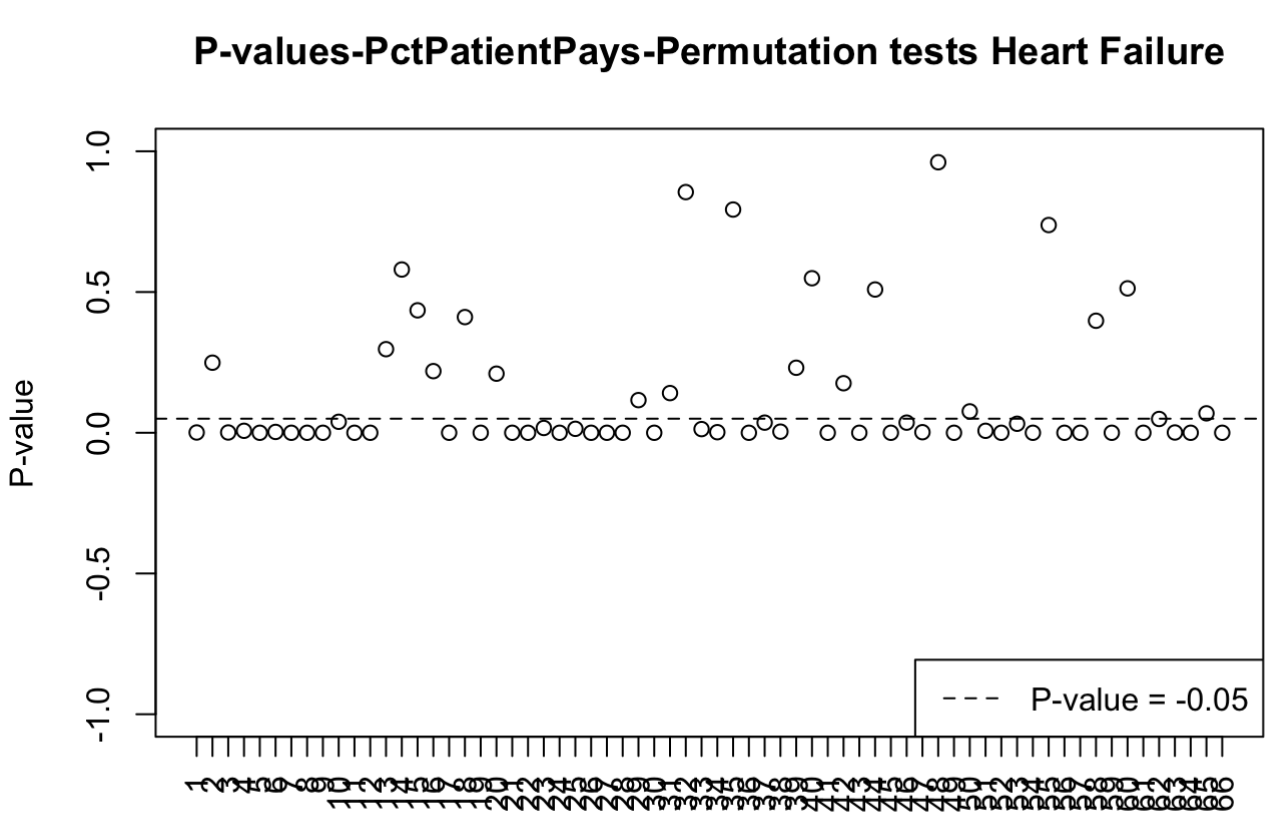
Calculating the pairwise Permutation test for all the pairs of the 4 diagnosis for the variable PctPatientPays

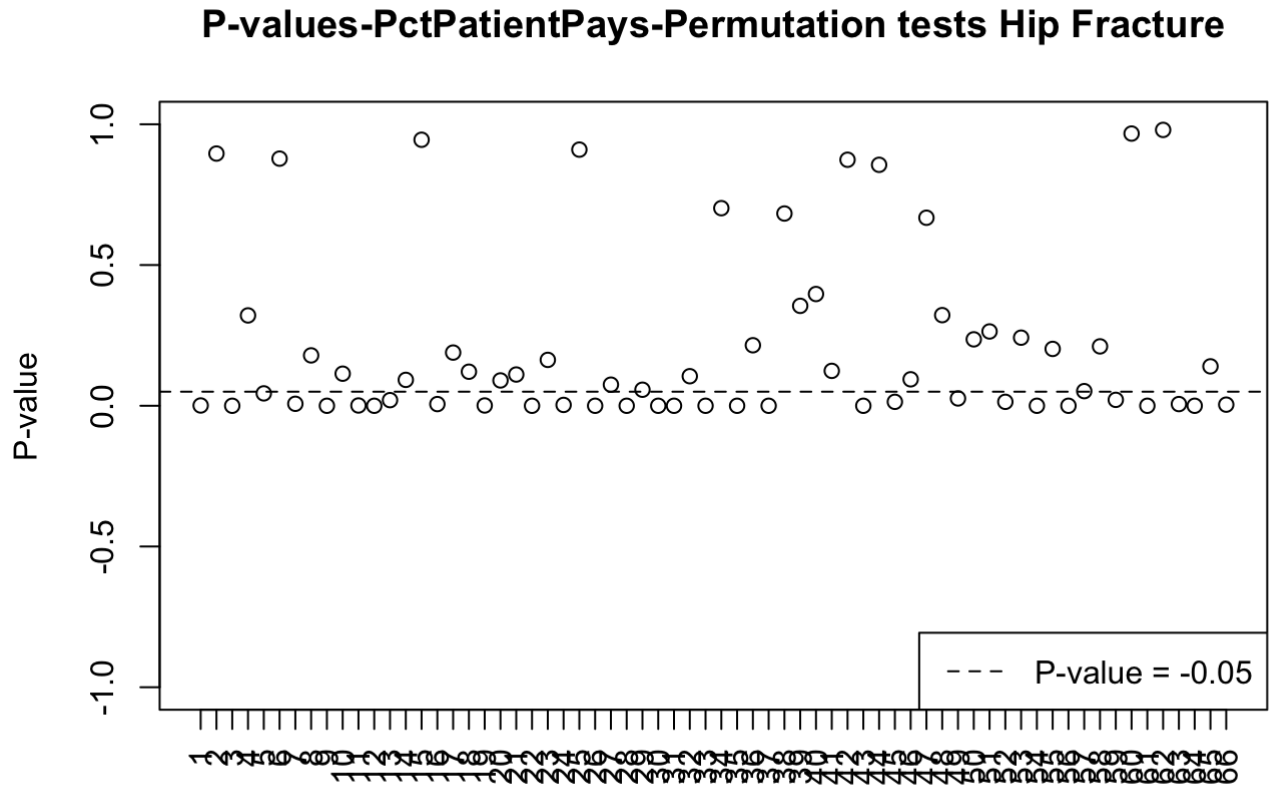
```
perm.testPairsUBR <- list()

permtestFun<-function(x = pair, df = data_frame, repetitions){
  # calculate the observed statistic
  group1 <- df[df$UrbanByRegions == x[1], ]$PctPatientPays
  group2 <- df[df$UrbanByRegions == x[2], ]$PctPatientPays
    #function to do the permutation and return statistic
  FUN <- function(x1, x2){
    x1<-na.omit(x1)
      x2<-na.omit(x2)
      return(abs(mean(x1)-mean(x2)))
  }
  stat.obs <- FUN(group1,group2)

  makePermutedStats<-function(){
      sampled <- sample(1:length(c(group1,group2)), size=length(group1),replace=FALSE) #
sample indices that will go into making new group 1
      return(FUN(c(group1, group2)[sampled], c(group1,
group2)[-sampled]))
  }
  # calculate the permuted statistic
  stat.permute <-replicate(repetitions,makePermutedStats())
  p.value <- sum(stat.permute >= stat.obs) / repetitions
  return(p.value=p.value)
}

for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  perm.testPairsUBR[[i]]<-combn(x=UBRgroups,m=2, FUN=permtestFun, repetitions = 1000, df
 = dff_temporary)
}
```

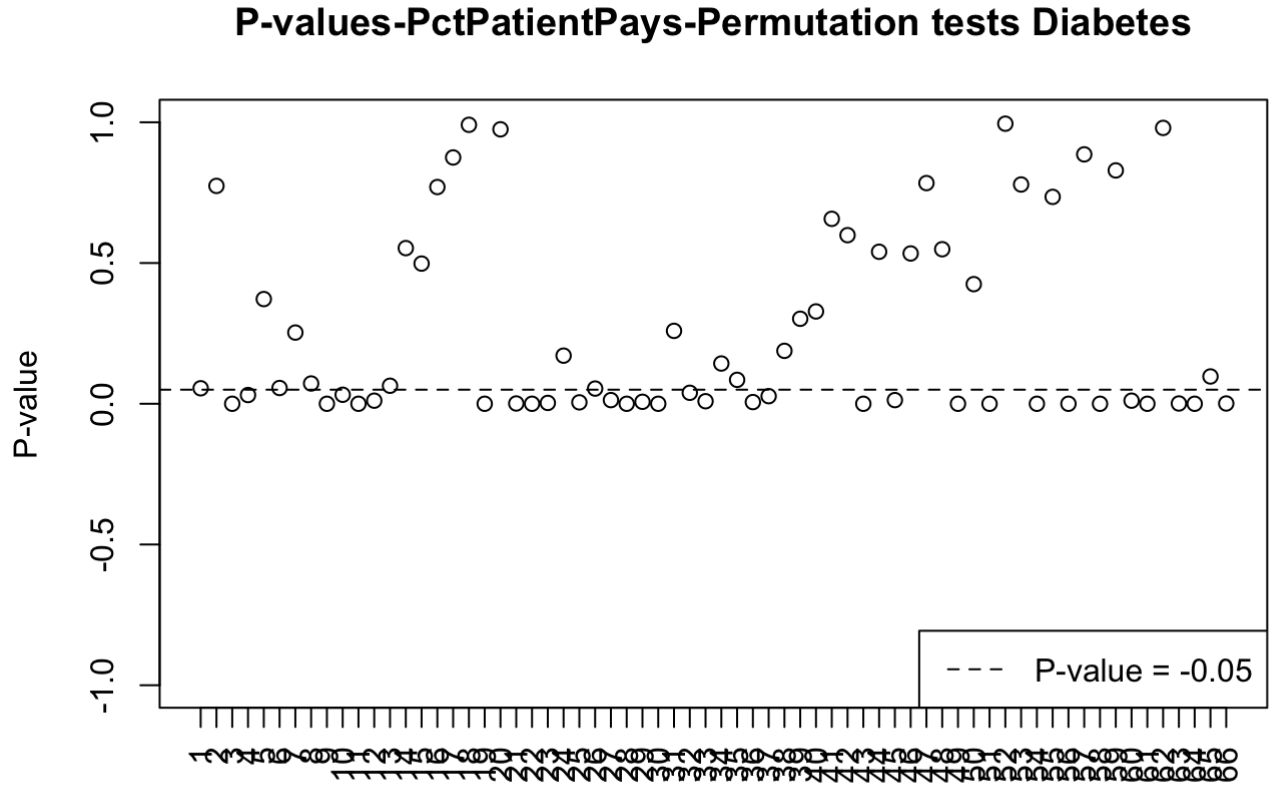
We can plot these p-values to get an idea of their value.

```
for (i in c(1:4)) {
  plot(perm.testPairsUBR[[i]],ylab="P-value",main= paste0("P-values-PctPatientPays-Permu
tation tests ", name_diag[i]),xaxt="n",xlab="", ylim = c(-1, 1))
abline(a = 0.05, b = 0, lty=2)
legend("bottomright",legend="P-value = -0.05",lty=2)
axis(1,at=1:length(perm.testPairsUBR[[i]]),labels=colnames(perm.testPairsUBR[[i]]),las=2
)
}
```


P-values-PctPatientPays-Permutation tests COPD
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = -0.05

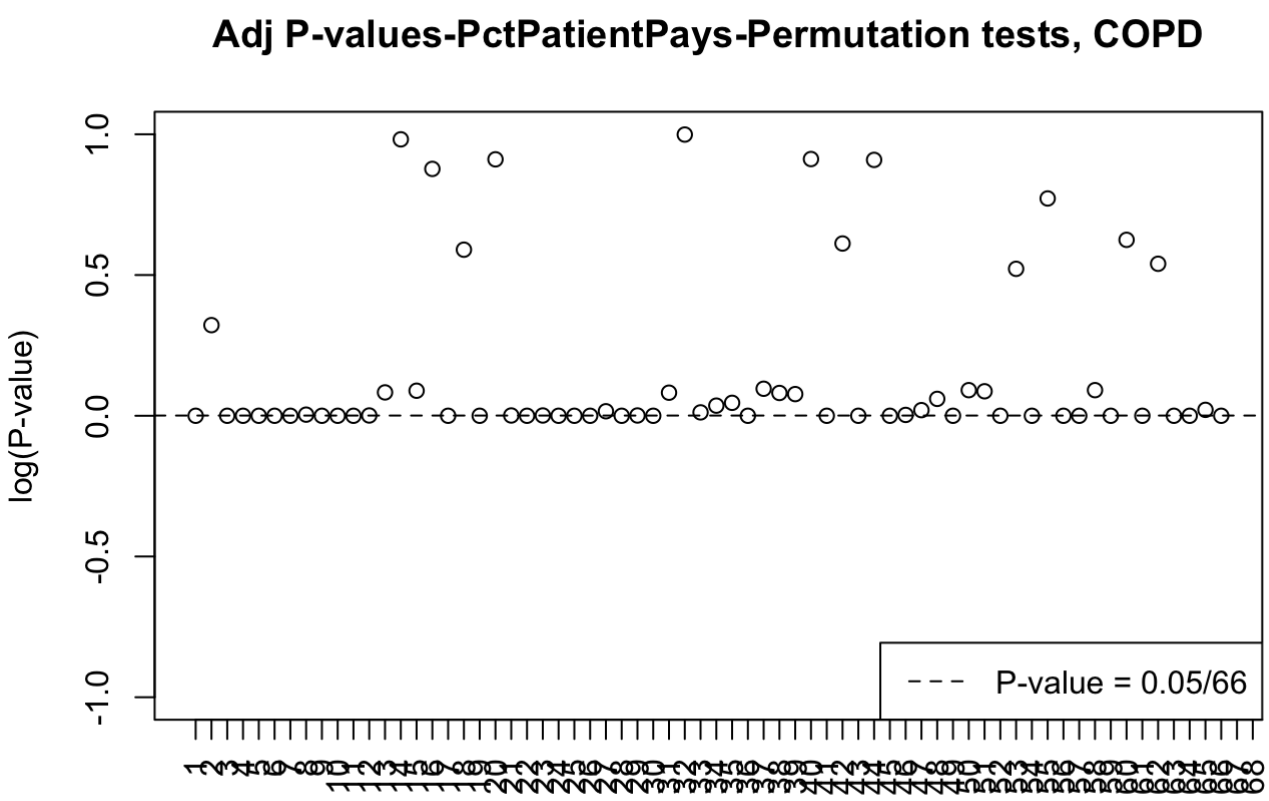

P-values-PctPatientPays-Permutation tests Heart Failure
P-value
1.0
0.5
0.0
-0.5
-1.0
P-value = -0.05

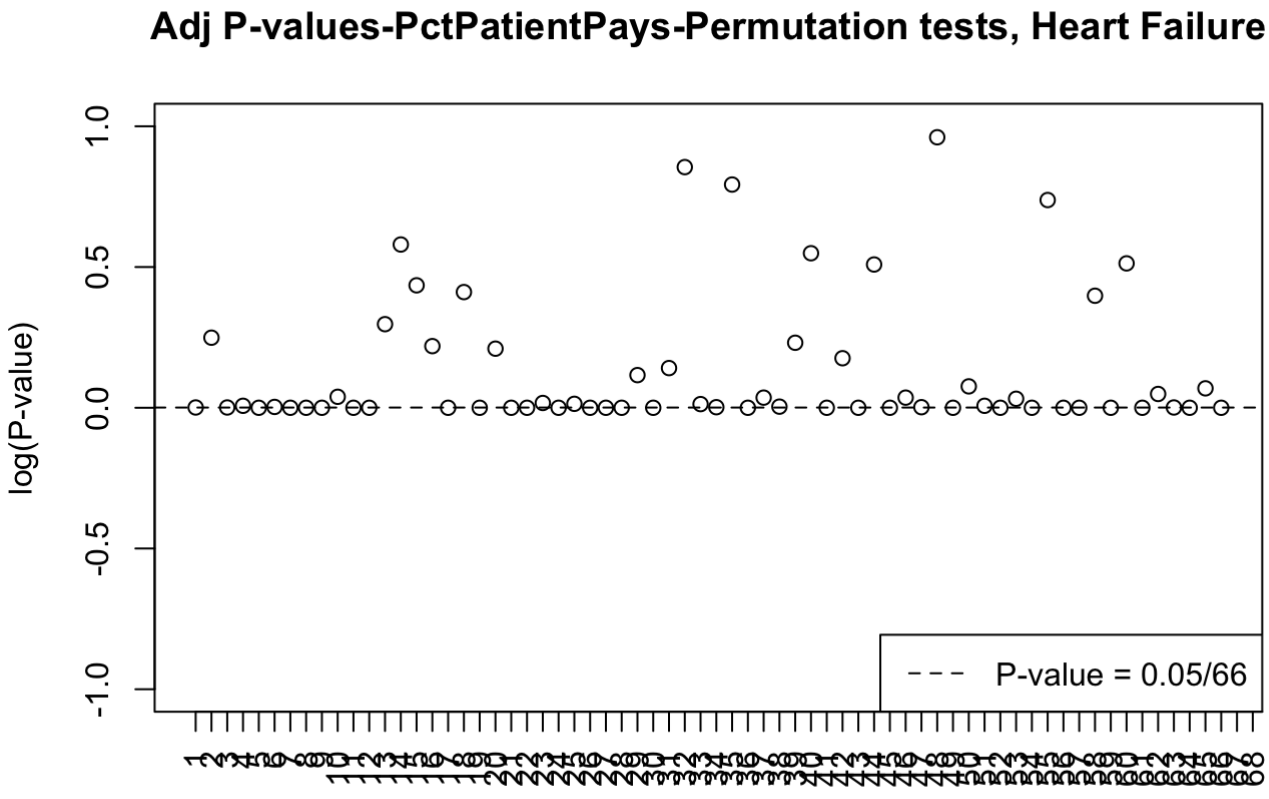


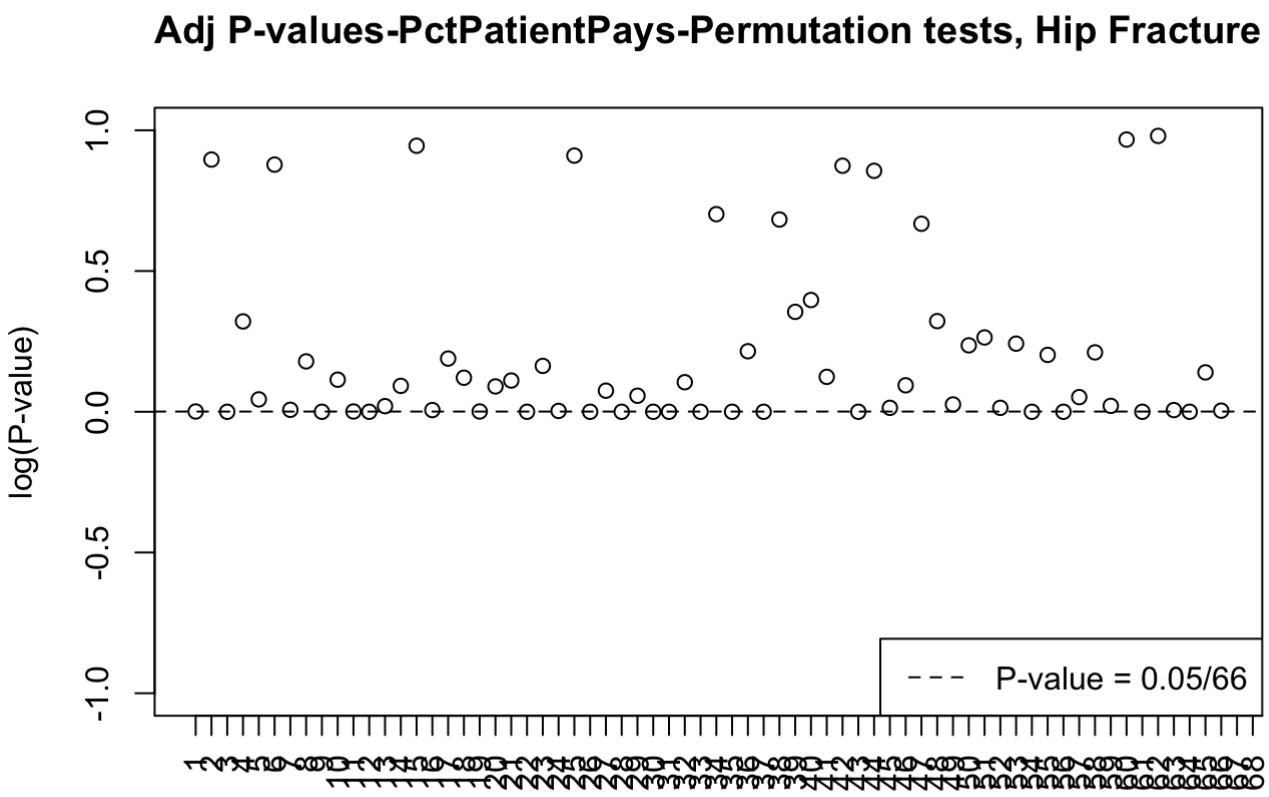


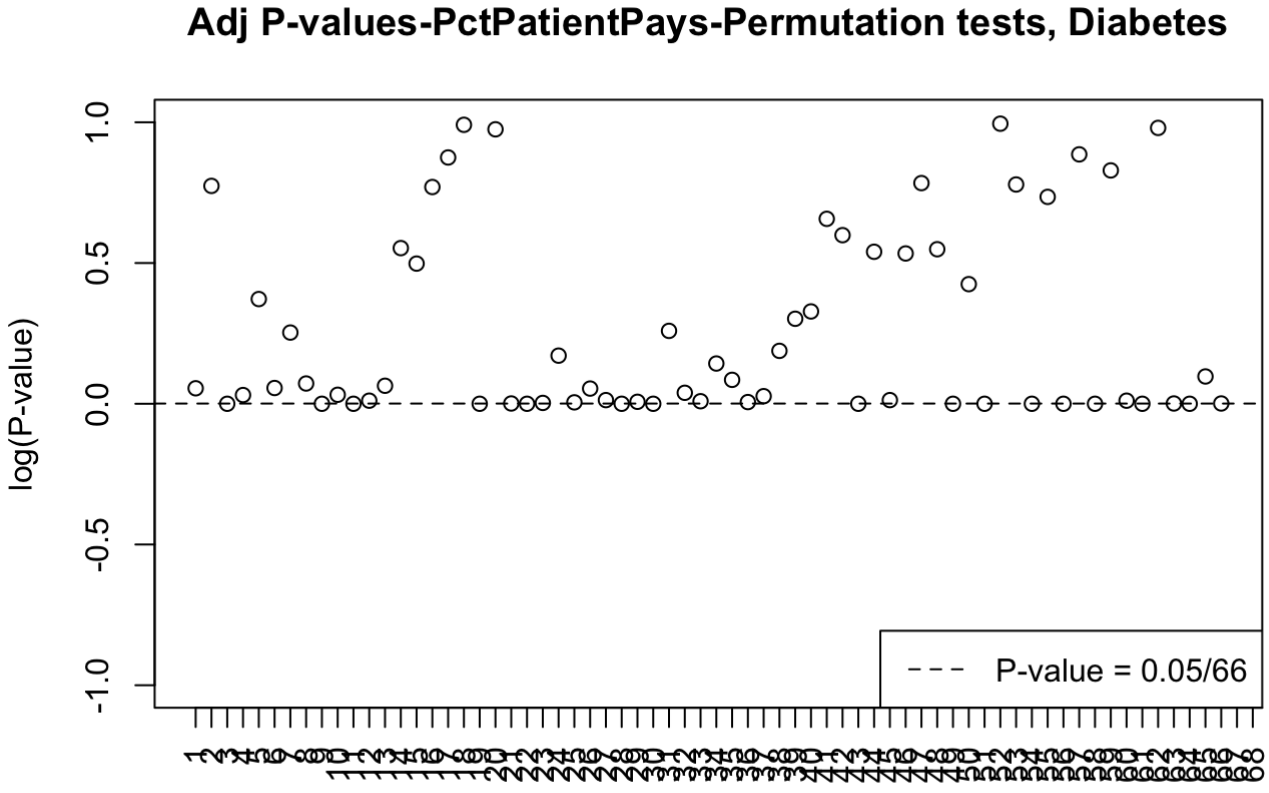
Applying Bonferroni Correction

```
for (i in c(1:4)) {
  plot(perm.testPairsUBR[[i]],ylab="log(P-value)",main= paste0("Adj P-values-PctPatientP
ays-Permutation tests, ", name_diag[i]),xaxt="n",xlab="", ylim = c(-1, 1))
abline(h=0.05/npairs,lty=2)
legend("bottomright",legend="P-value = 0.05/66",lty=2)
axis(1,at=1:length(t.testPairsUBR[[i]]),labels=colnames(t.testPairsUBR[[i]]),las=2)
}
```


Adj P-values-PctPatientPays-Permutation tests, COPD
log(P-value)
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05/66


Adj P-values-PctPatientPays-Permutation tests, Heart Failure
log(P-value)
1.0
0.5
0.0
-0.5
-1.0
P-value = 0.05/66



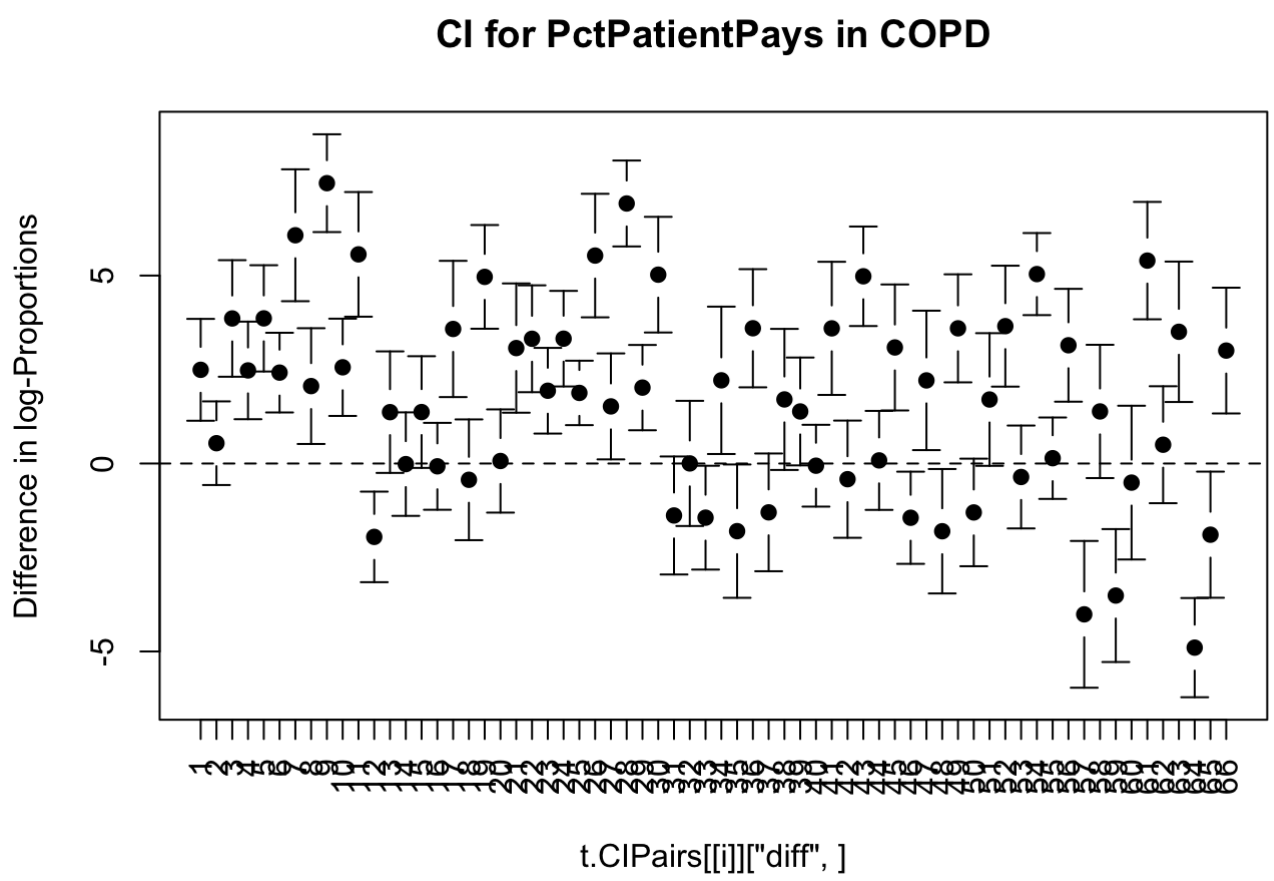


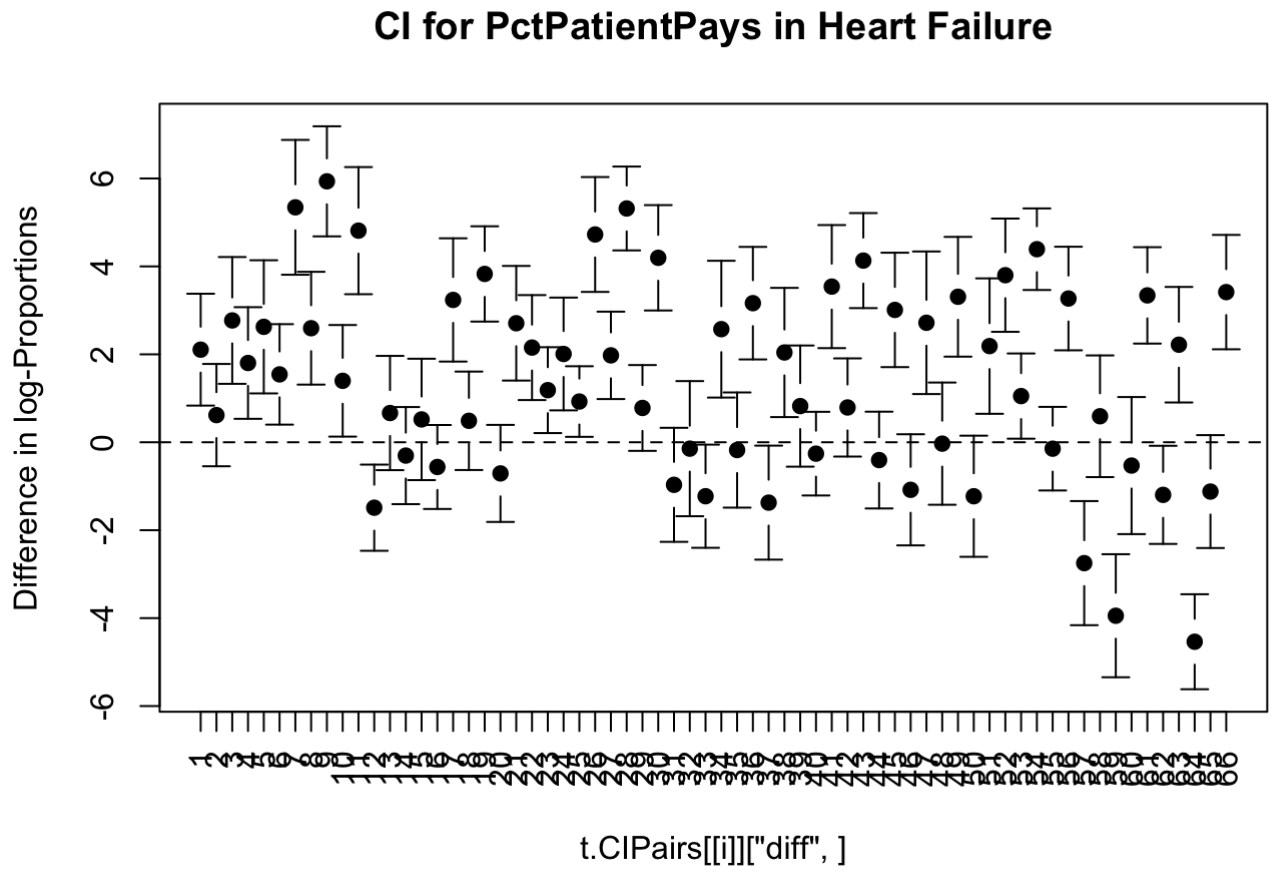
4.2.3 Creating Confidence Intervals for variabe PctPatientPays grouped by UrbanByRegion

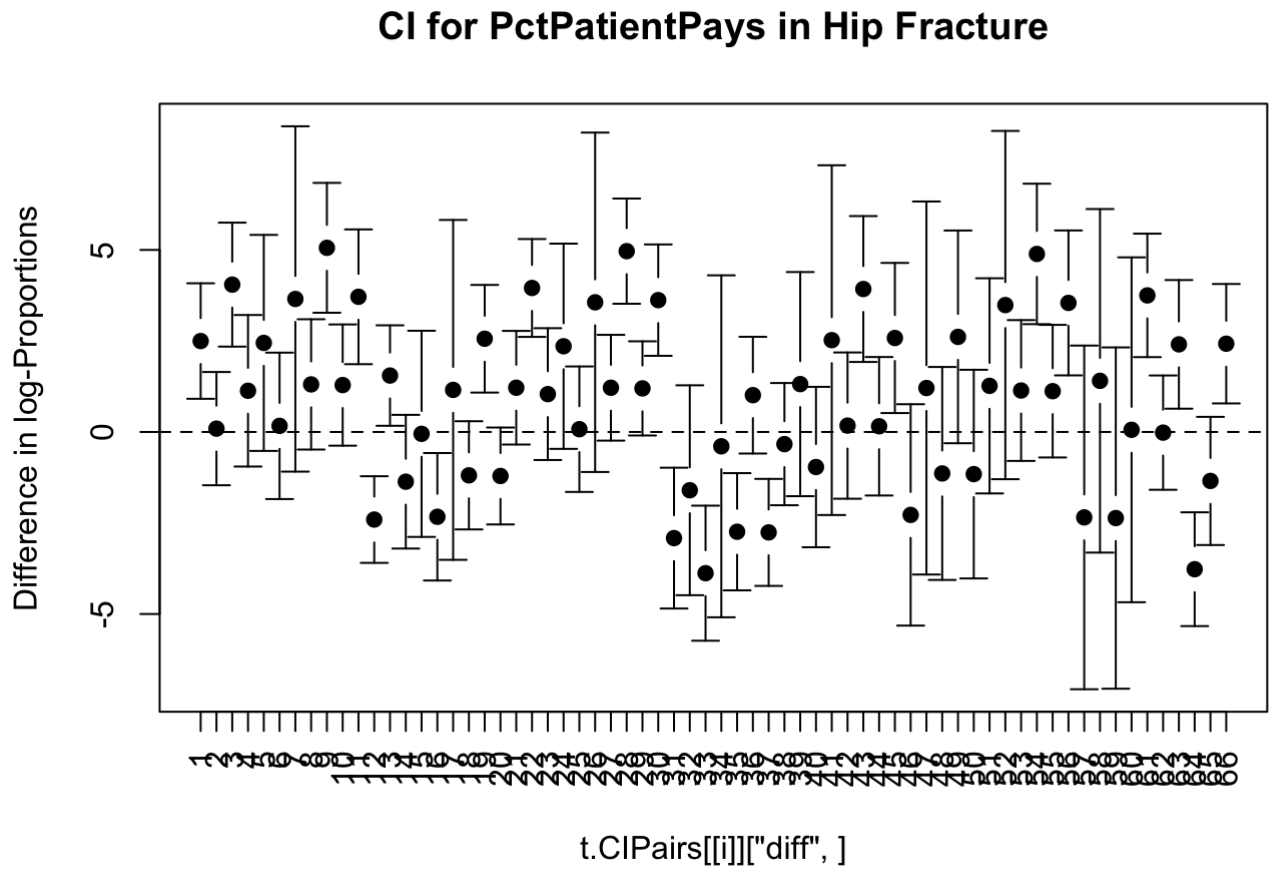
```
t.CIPairs <- list()
ttestCI <- function(x, variableName, dff_temp) {
  tout <- t.test(dff_temp$PctPatientPays[dff_temp[,variableName] == x[1]],dff_temp$PctPa
tientPays[dff_temp[,variableName] == x[2]])
  unlist(tout[c("estimate", "conf.int")])
}


for(i in c(1:4)){
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.CIPairs[[i]] <- apply(X = pairsOfUBR, MARGIN = 2,
FUN = ttestCI, variableName = "UrbanByRegions", dff_temp = dff_temporary)
colnames(t.CIPairs[[i]]) <- paste(pairsOfUBR[2, ],
pairsOfUBR[1, ], sep = "--")
t.CIPairs[[i]] <- rbind(t.CIPairs[[i]], diff = t.CIPairs[[i]]["estimate.mean of x",
] - t.CIPairs[[i]]["estimate.mean of y", ])
}

require(gplots)

par(mfrow <- c(2, 2))
```

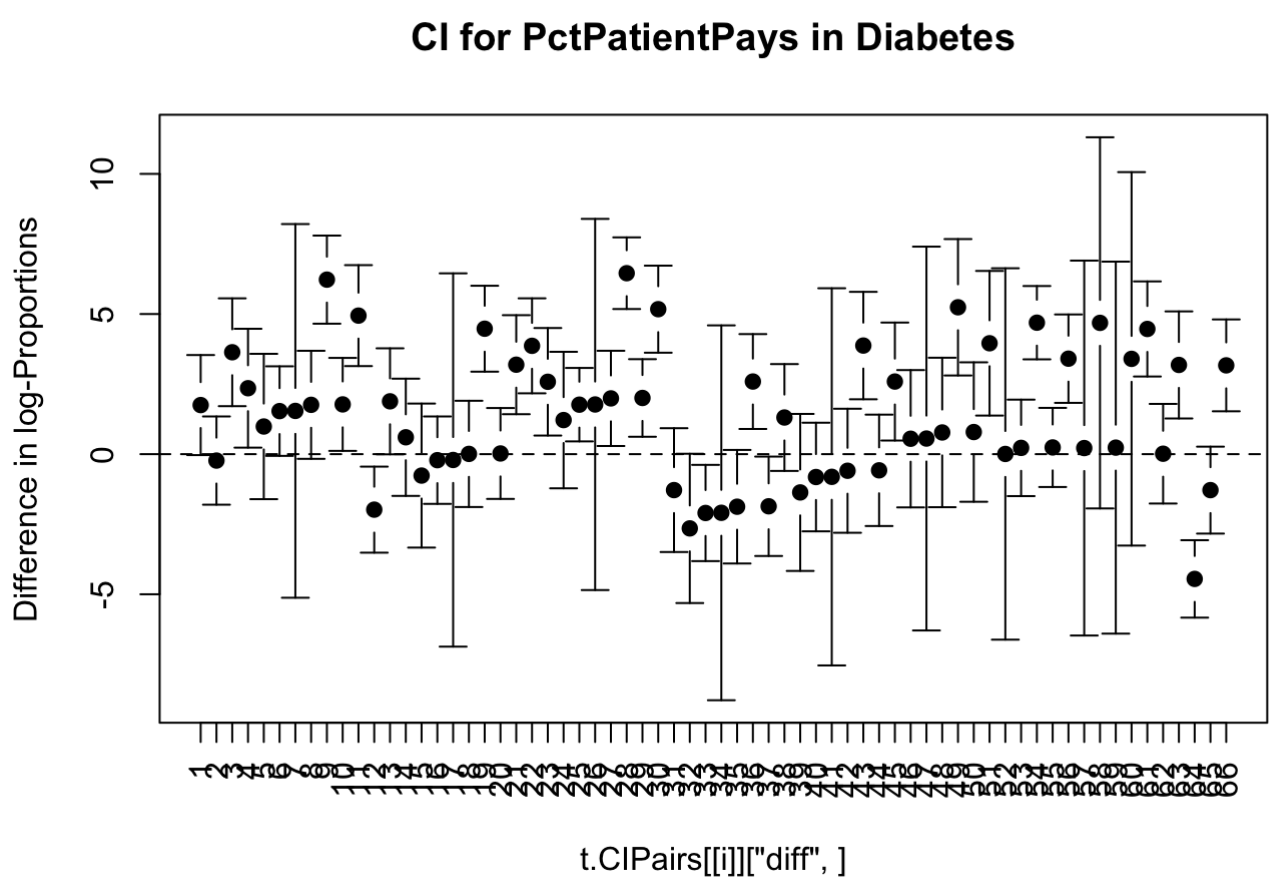
```
## NULL
```

```
for (i in c(1:4)) {
  plotCI(x = t.CIPairs[[i]]["diff", ], li = t.CIPairs[[i]]["conf.int1",
], ui = t.CIPairs[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",
pch = 19, xaxt = "n", main = paste0("CI for PctPatientPays in ", name_diag[i]))
  axis(1, at = 1:ncol(t.CIPairs[[i]]), labels = colnames(t.CIPairs), las = 2)
  abline(h = 0, lty = 2)
}
```


CI for PctPatientPays in COPD
Difference in log-Proportions
5
0
-5
t.CIPairs[[i]]["diff",]

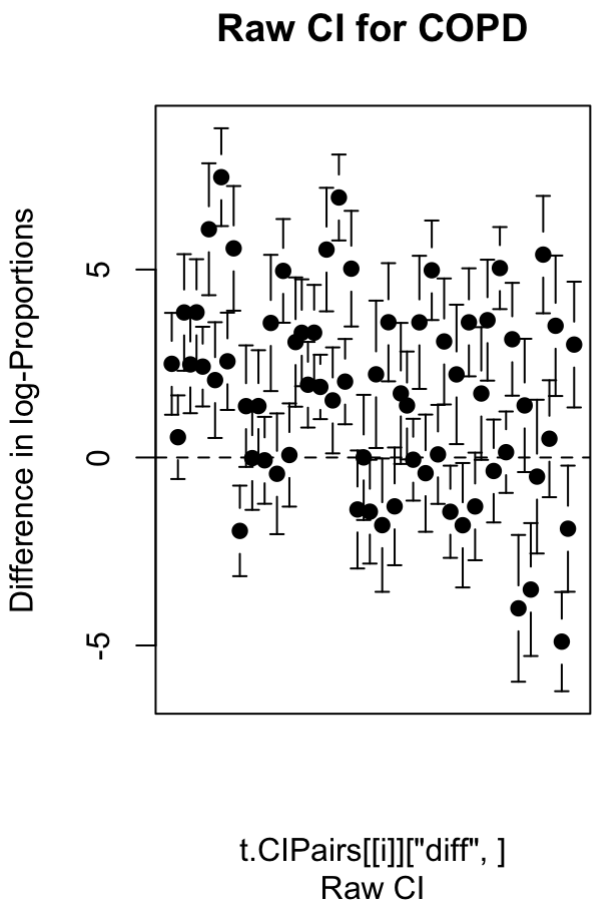

CI for PctPatientPays in Heart Failure
Difference in log-Proportions
6
4
2
0
-2
-4
-6
t.CIPairs[[i]]["diff",]

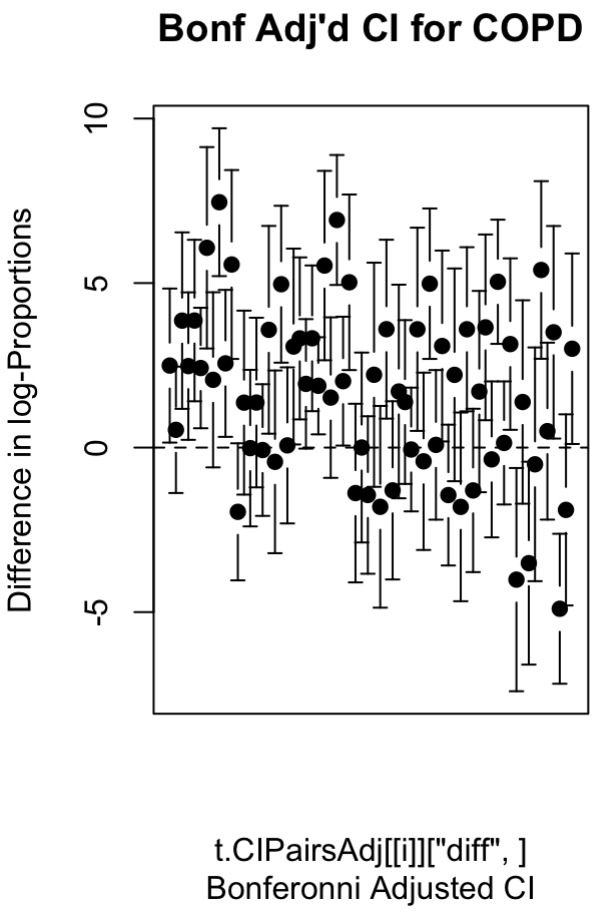

CI for PctPatientPays in Hip Fracture
Difference in log-Proportions
5
0
-5
t.CIPairs[[i]]["diff",]

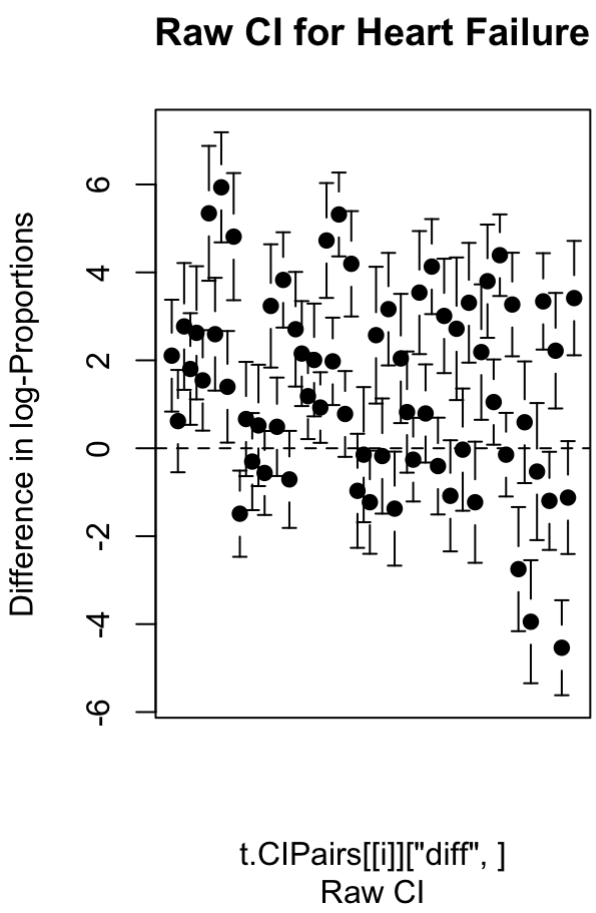

CI for PctPatientPays in Diabetes
Difference in log-Proportions
10
5
0
-5
t.CIPairs[[i]]["diff",]

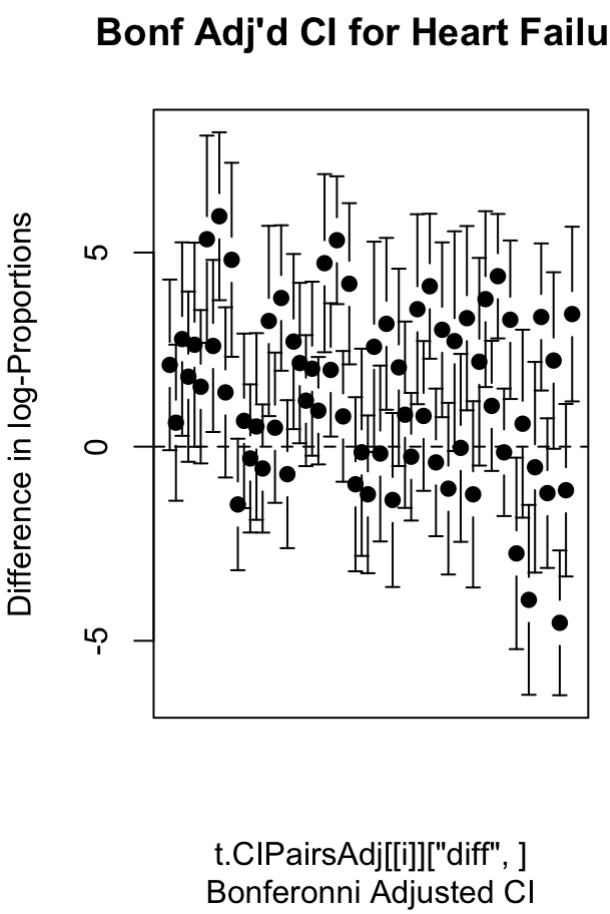
These confidence intervals suffer from the same problem as the p-values: even if the null value (0) is true in every test, roughly 5% of them will happen to not cover 0 just by chance. So we can do bonferonni corrections to the confidence intervals. Since a 95% confidence interval corresponds to a level 0.05 test, if we go to a 0.05/K level, which is the bonferonni correction, that corresponds to a 100 $*$ (1 – 0.05/K)% confidence interval.

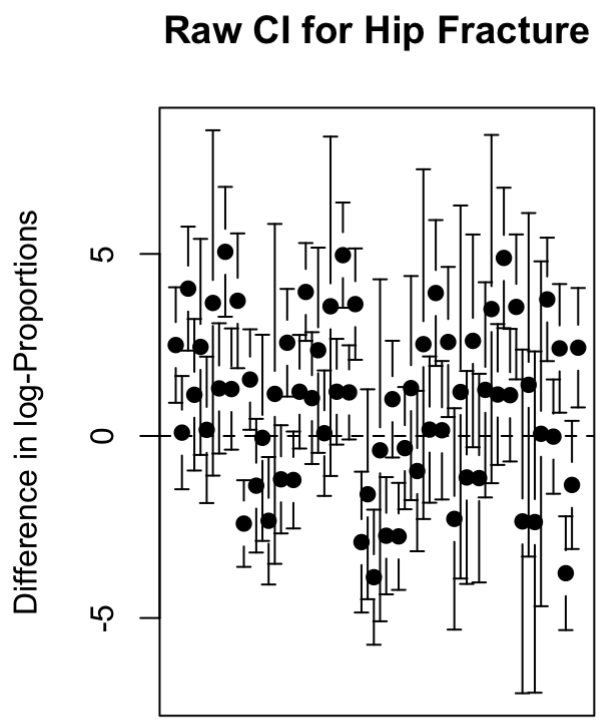
```
t.CIPairsAdj <- list()
ttestCIAdj <- function(x, variableName, dff_temp) {
  tout <- t.test(dff_temp$PctPatientPays[dff_temp[,variableName] == x[1]],dff_temp$PctPa
tientPays[dff_temp[,variableName] == x[2]], conf.level = 1-0.05/npairs)
  unlist(tout[c("estimate", "conf.int")])
}

for (i in c(1:4)) {
  dff_temporary <- dff[dff$DRG.Definition==levels(dff$DRG.Definition)[i], ]
  t.CIPairsAdj[[i]] <- apply(X = pairsOfUBR, MARGIN = 2,
FUN = ttestCIAdj, variableName = "UrbanByRegions", dff_temp = dff_temporary)
colnames(t.CIPairsAdj[[i]]) <- paste(pairsOfUBR[2,
], pairsOfUBR[1, ], sep = "-")
  t.CIPairsAdj[[i]] <- rbind(t.CIPairsAdj[[i]], diff = t.CIPairsAdj[[i]]["estimate.mean
 of x",
] - t.CIPairsAdj[[i]]["estimate.mean of y", ])
}
for (i in c(1:4)) {
  par(mfrow = c(1, 2))
  plotCI(x = t.CIPairs[[i]]["diff", ], li = t.CIPairs[[i]]["conf.int1",
], ui = t.CIPairs[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",
sub = "Raw CI", pch = 19, xaxt = "n", main = paste0("Raw CI for ",name_diag[i]))
  abline(h = 0, lty = 2)
  plotCI(x = t.CIPairsAdj[[i]]["diff", ], li = t.CIPairsAdj[[i]]["conf.int1",
], ui = t.CIPairsAdj[[i]]["conf.int2", ], ylab = "Difference in log-Proportions",sub =
"Bonferonni Adjusted CI", pch = 19, xaxt = "n", , main = paste0("Bonf Adj'd CI for ",nam
e_diag[i]))
  abline(h = 0, lty = 2)
}
```

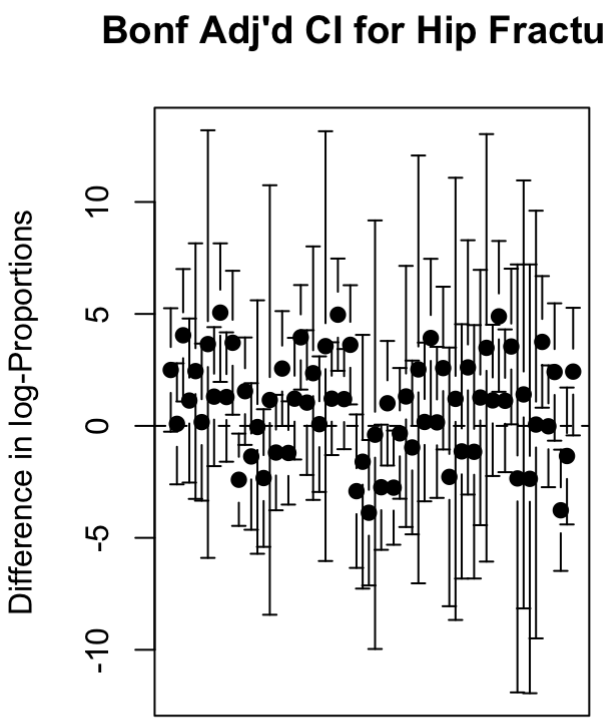

Raw CI for COPD
Difference in log-Proportions
-5
0
5
t.CIPairs[[i]]["diff",]
Raw CI

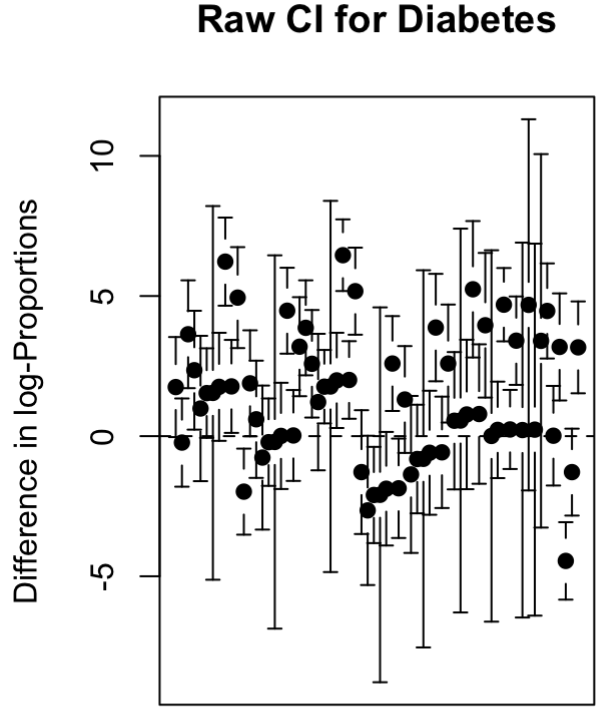

Bonf Adj'd CI for COPD
Difference in log-Proportions
-5
0
5
10
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI

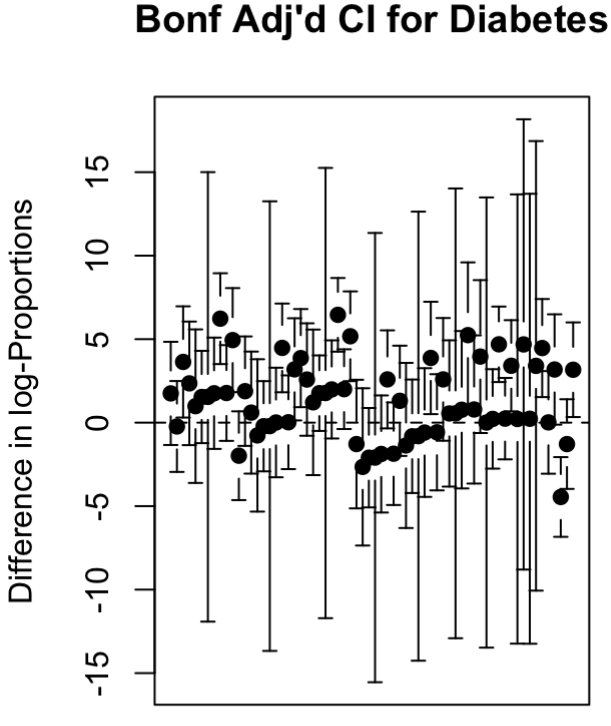

Raw CI for Heart Failure
Difference in log-Proportions
-6
-4
-2
0
2
4
6
t.CIPairs[[i]]["diff",]
Raw CI


Bonf Adj'd CI for Heart Failure
Difference in log-Proportions
-5
0
5
t.CIPairsAdj[[i]]["diff",]
Bonferonni Adjusted CI









####

4.3 Interpreting the results of Inference

If we analyse the pairs that we have rejected the null hypothesis for in each of the four diagnosis, we find that the overall percentage cost paid by the patient for all the four diagnosis, we would find that the regions of *west* is the *cheapest* followed by the *northeast*. Also, the *rural urban clusters* are more gentle on the pockets of the patients than the *only-urban* areas.

5. Reflections

In retrospect about the permutation tests I did, I think that in order for my permutation tests to be valid, I must make atleast 1320 iterations (66(pairs) * 20(1/20=0.05)) to ensure any false positive would show up with probability 0.05 and in order to be more confident, the iterations should be around 10000(1320 times 66) which was not feasible for my computer. Therefore, I didnot have confidence in the results of permutation tests and hence, I didnot go ahead with calculating the confidence intervals for the same. (Bootstrapping would need millions of iterations).

Going with the parametric test with the normal assumption was justified by Q-Q plots to which the data pretty much adhered. Hence, I am positive about the validity of the the t.test results and confidence intervals.

Talking about the limitations of the analysis, one thing that I find could be improved is incorporating the weights of the number of abservations made to give the average value for the response variables. We must account for the fact that we give more weightage to the values that were derived from a large data points than those with only few as they might mess up the analysis incase they are not representative of the population.

Another question that I would ask from the data is if instead of regions, the percentage patient cost and the absolute patient cost are a function of state and state laws for the medical insurance. The government party ruling the state might have different stances for the medical support to the citizens according to if their principals are capitalist, socialist, or marxist which might be the reason behind the different cost observed in the different states. Therefore, in future, we must analyse the data with respect to states as well and do Principal component analysis so that we might capture the true factor that causes the variation in the cost.