Healthcare Cost Analysis

By:

Venkata Sai Mani Lakshmi Kavya Darsi

Pranjal Singh

Vipul Rajiv Sarode

Yoon Lee

#Importing the dataset and removing the rows with na BMI and Hypertension values

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(dplyr)

health_raw <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/HMO_data.csv")
```

```
## Rows: 7582 Columns: 14
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): smoker, location, location_type, education_level, yearly_physical, ...
## dbl (6): X, age, bmi, children, hypertension, cost
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
health=health_raw %>% filter(!(is.na(bmi)))
health=health %>% filter(!(is.na(hypertension)))
sapply(health,function(x) sum(is.na(x)))
```

```
##               X             age             bmi        children          smoker
##               0               0               0               0               0
##        location   location_type education_level yearly_physical        exercise
##               0               0               0               0               0
##         married    hypertension          gender            cost
##               0               0               0               0
```

```
sapply(health,function(x) sum(is.null(x)))
```

```
##                X              age              bmi         children           smoker
##                0                0                0                0                0
##         location   location_type education_level yearly_physical         exercise
##                0                0                0                0                0
##          married    hypertension           gender             cost
##                0                0                0                0
```

```
str(health)
```

```
## spc_tbl_ [7,424 x 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ X               : num [1:7424] 1 2 3 4 5 7 9 10 11 12 ...
##  $ age             : num [1:7424] 18 19 27 34 32 47 36 59 24 61 ...
##  $ bmi             : num [1:7424] 27.9 33.8 33 22.7 28.9 ...
##  $ children        : num [1:7424] 0 1 3 0 0 1 2 0 0 0 ...
##  $ smoker          : chr [1:7424] "yes" "no" "no" "no" ...
##  $ location        : chr [1:7424] "CONNECTICUT" "RHODE ISLAND" "MASSACHUSETTS" "PENNSYLVANIA" ...
##  $ location_type   : chr [1:7424] "Urban" "Urban" "Urban" "Country" ...
##  $ education_level: chr [1:7424] "Bachelor" "Bachelor" "Master" "Master" ...
##  $ yearly_physical: chr [1:7424] "No" "No" "No" "No" ...
##  $ exercise        : chr [1:7424] "Active" "Not-Active" "Active" "Not-Active" ...
##  $ married         : chr [1:7424] "Married" "Married" "Married" "Married" ...
##  $ hypertension    : num [1:7424] 0 0 0 1 0 0 0 1 0 0 ...
##  $ gender          : chr [1:7424] "female" "male" "male" "male" ...
##  $ cost            : num [1:7424] 1746 602 576 5562 836 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   X = col_double(),
##   ..   age = col_double(),
##   ..   bmi = col_double(),
##   ..   children = col_double(),
##   ..   smoker = col_character(),
##   ..   location = col_character(),
##   ..   location_type = col_character(),
##   ..   education_level = col_character(),
##   ..   yearly_physical = col_character(),
##   ..   exercise = col_character(),
##   ..   married = col_character(),
##   ..   hypertension = col_double(),
##   ..   gender = col_character(),
##   ..   cost = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
#health
```

#Creating a New column

```
threshold=quantile(health$cost,probs=(.75))
health$Expensive <- ifelse(health$cost>=threshold, 1, 0)
glimpse(health)
```

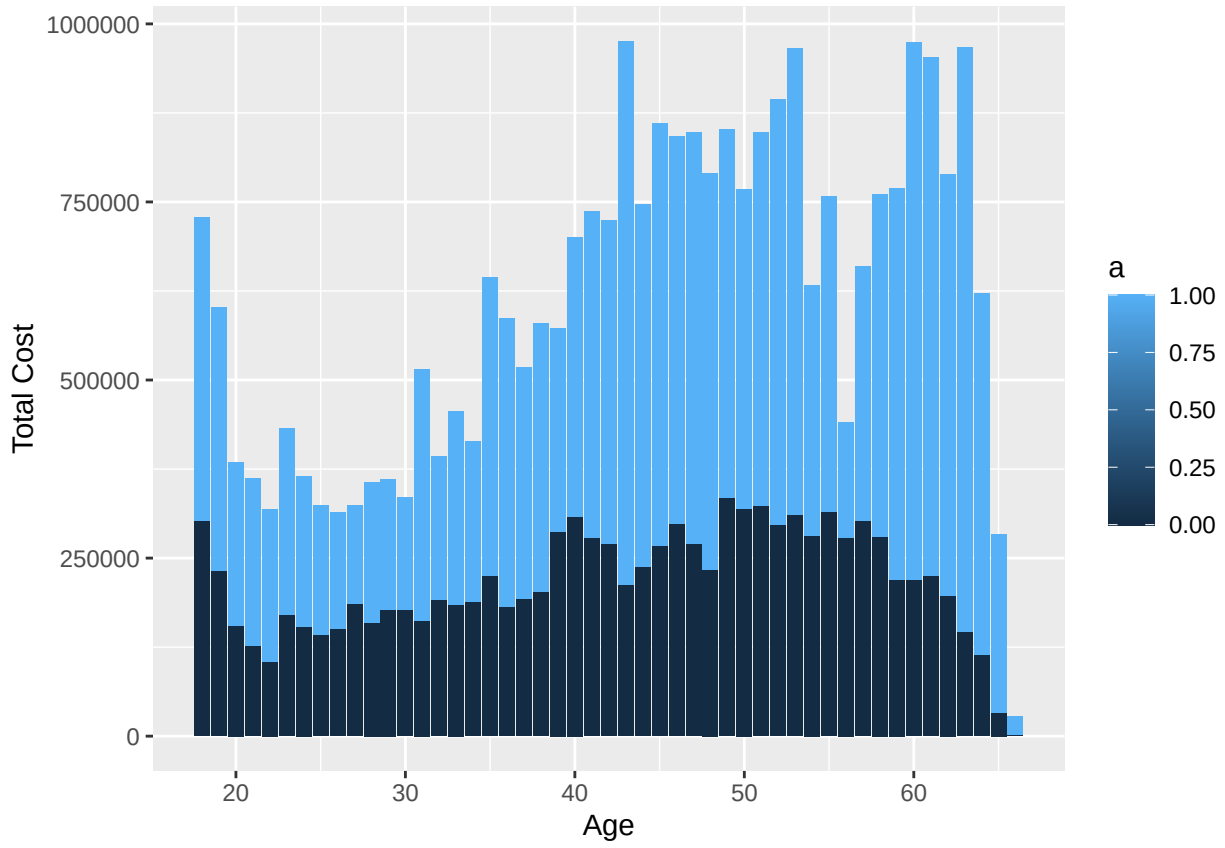
```
## Rows: 7,424
## Columns: 15
## $ X               <dbl> 1, 2, 3, 4, 5, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 1~
## $ age             <dbl> 18, 19, 27, 34, 32, 47, 36, 59, 24, 61, 22, 57, 26, 18~
## $ bmi             <dbl> 27.900, 33.770, 33.000, 22.705, 28.880, 33.440, 29.830~
## $ children        <dbl> 0, 1, 3, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ smoker          <chr> "yes", "no", "no", "no", "no", "no", "no", "no", "no",~
## $ location        <chr> "CONNECTICUT", "RHODE ISLAND", "MASSACHUSETTS", "PENNS~
## $ location_type   <chr> "Urban", "Urban", "Urban", "Country", "Country", "Urba~
## $ education_level <chr> "Bachelor", "Bachelor", "Master", "Master", "PhD", "Ba~
## $ yearly_physical <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No", ~
## $ exercise        <chr> "Active", "Not-Active", "Active", "Not-Active", "Not-A~
## $ married         <chr> "Married", "Married", "Married", "Married", "Married",~
## $ hypertension    <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ gender          <chr> "female", "male", "male", "male", "male", "female", "m~
## $ cost            <dbl> 1746, 602, 576, 5562, 836, 3842, 1304, 9724, 201, 4492~
## $ Expensive       <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, ~
```

#Creating Histograms for every numeric Variable

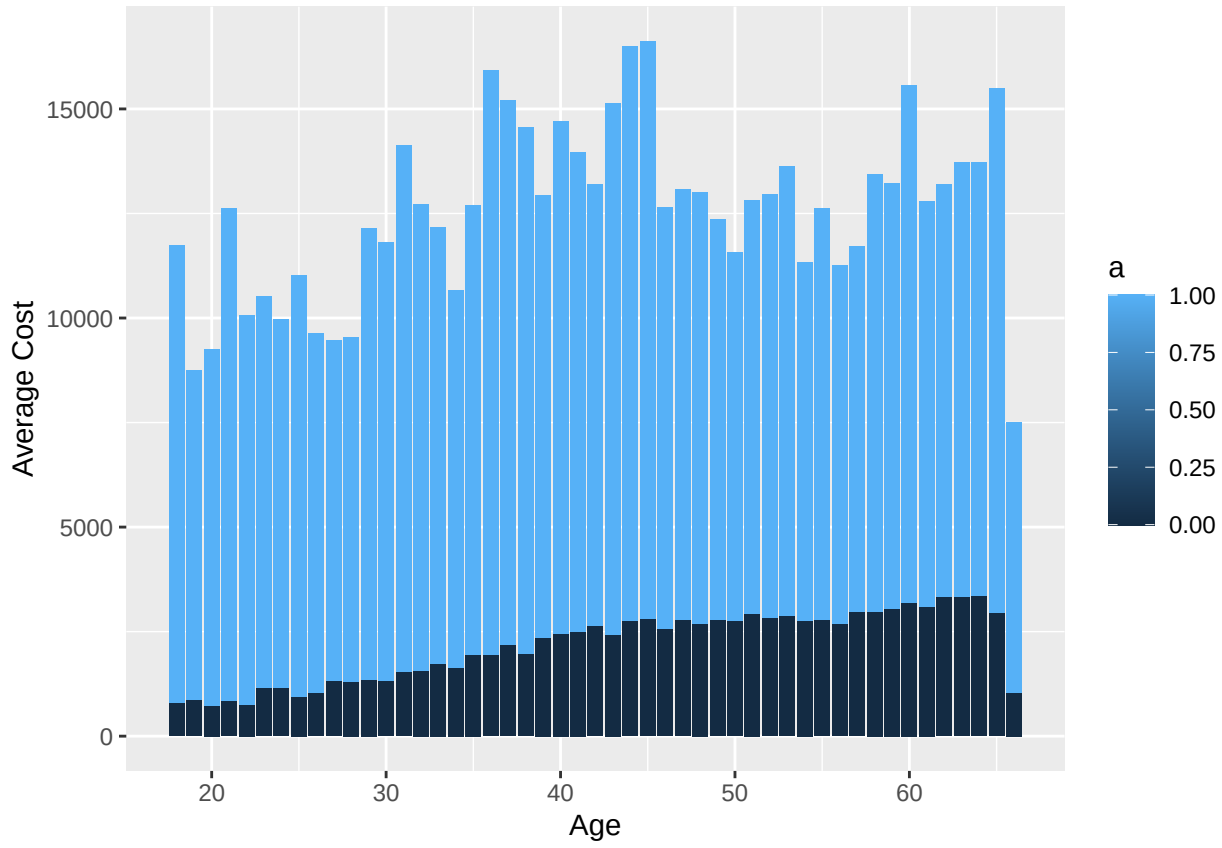
```
library(ggplot2)
#Age
s=health%>%group_by(Expensive,age) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$age
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Age")+ylab("Total Cost")
```




```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Age")+ylab("Average Cost")
```

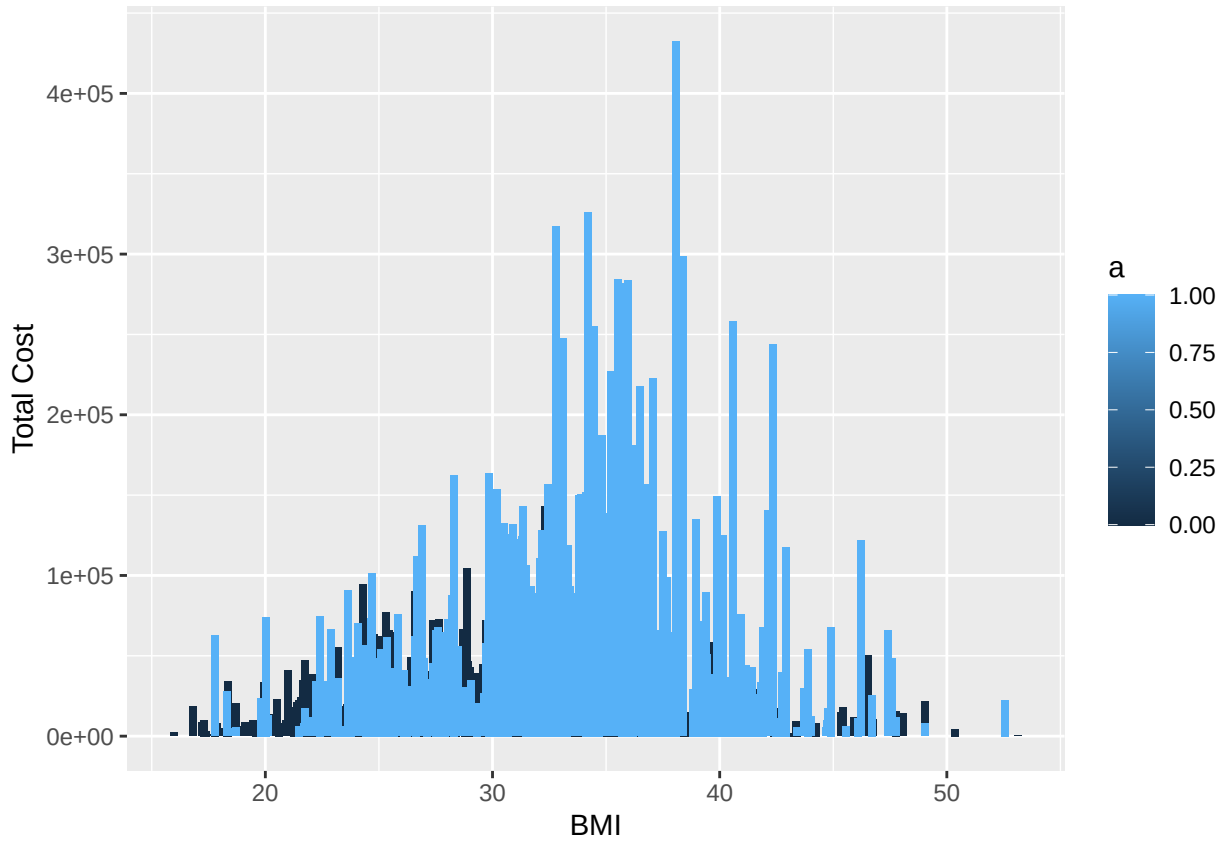



```
rm(s,a,b,c)

#BMI
s=health%>%group_by(Expensive,bmi) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

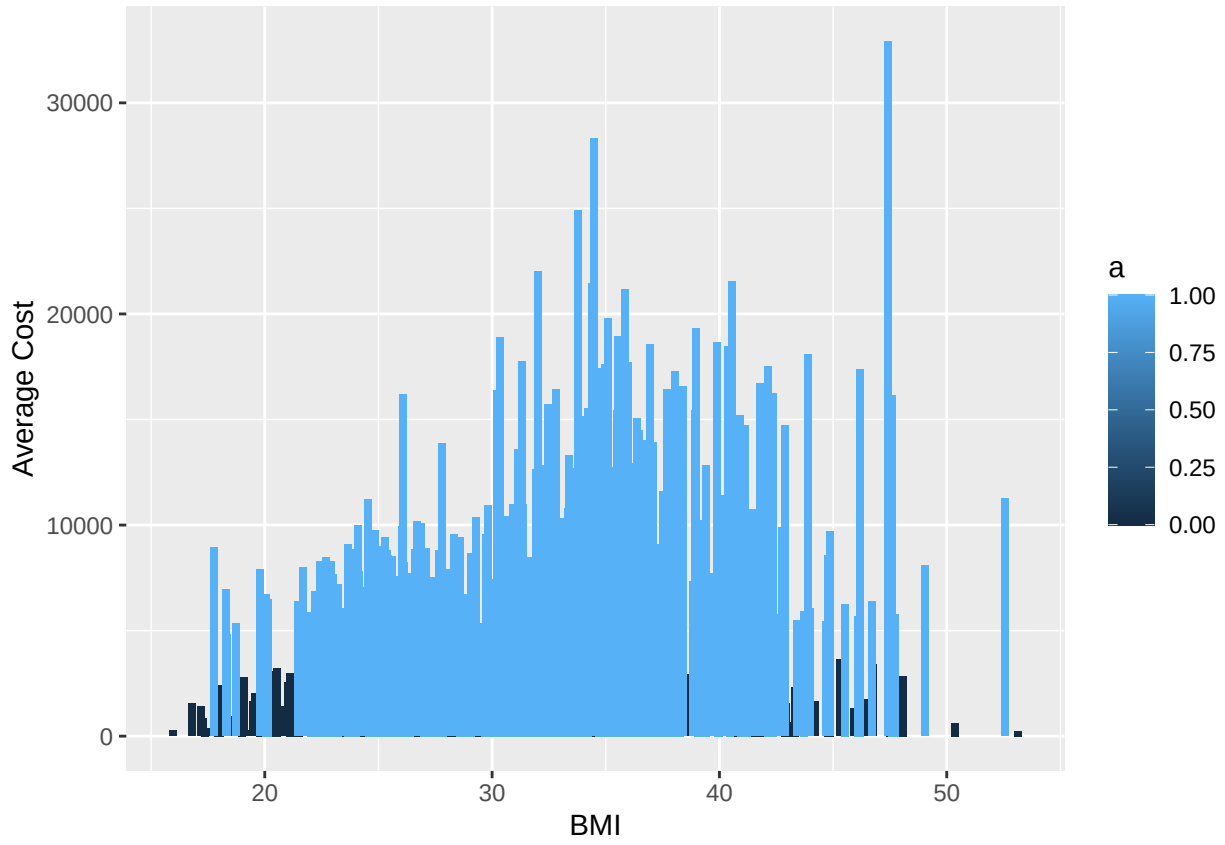
```
a=s$Expensive
b=s$Freq
c=s$bmi
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(position = "dodge",stat='identity',width = .35)+xlab("BMI")+yla
```

```
## Warning: `position_dodge()` requires non-overlapping x intervals
```




```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(position="dodge",stat='identity',width = .35)+xlab("BMI")+ylab(
```

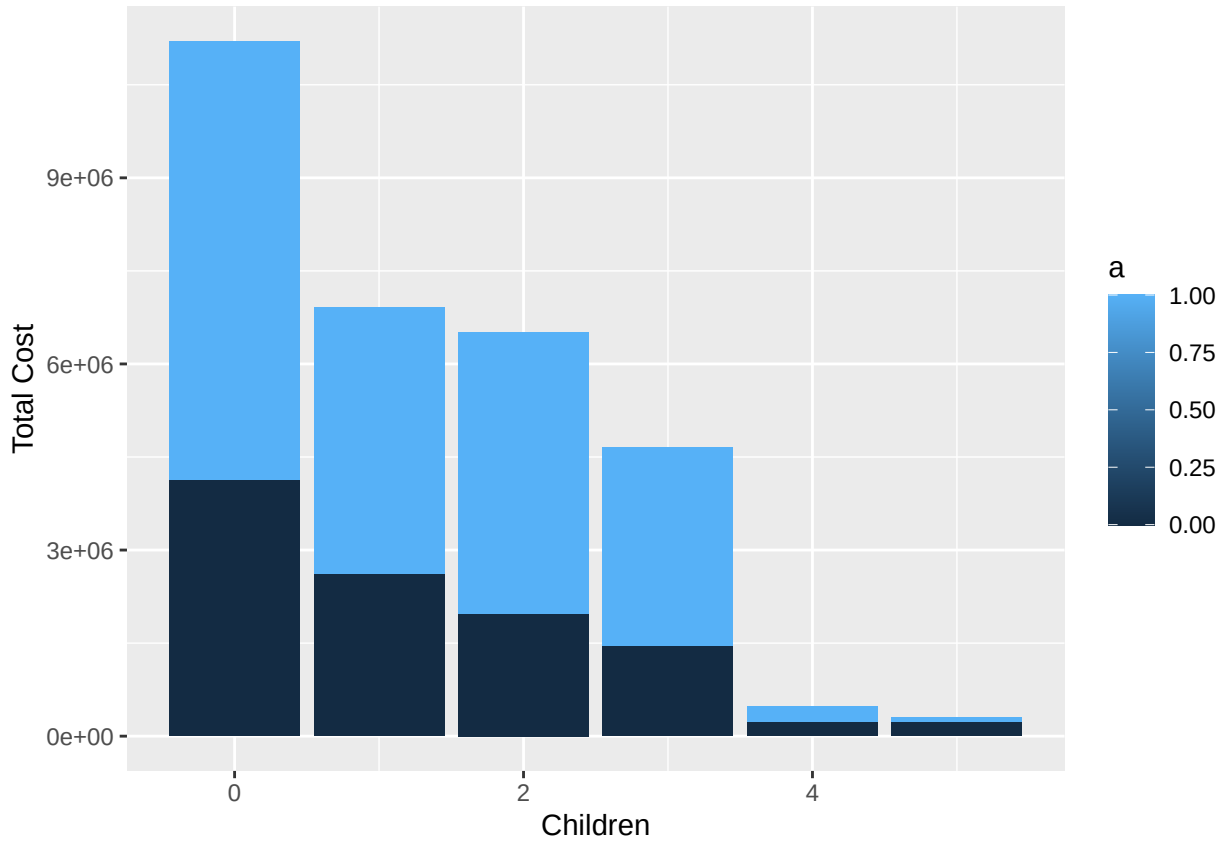
```
## Warning: `position_dodge()` requires non-overlapping x intervals
```



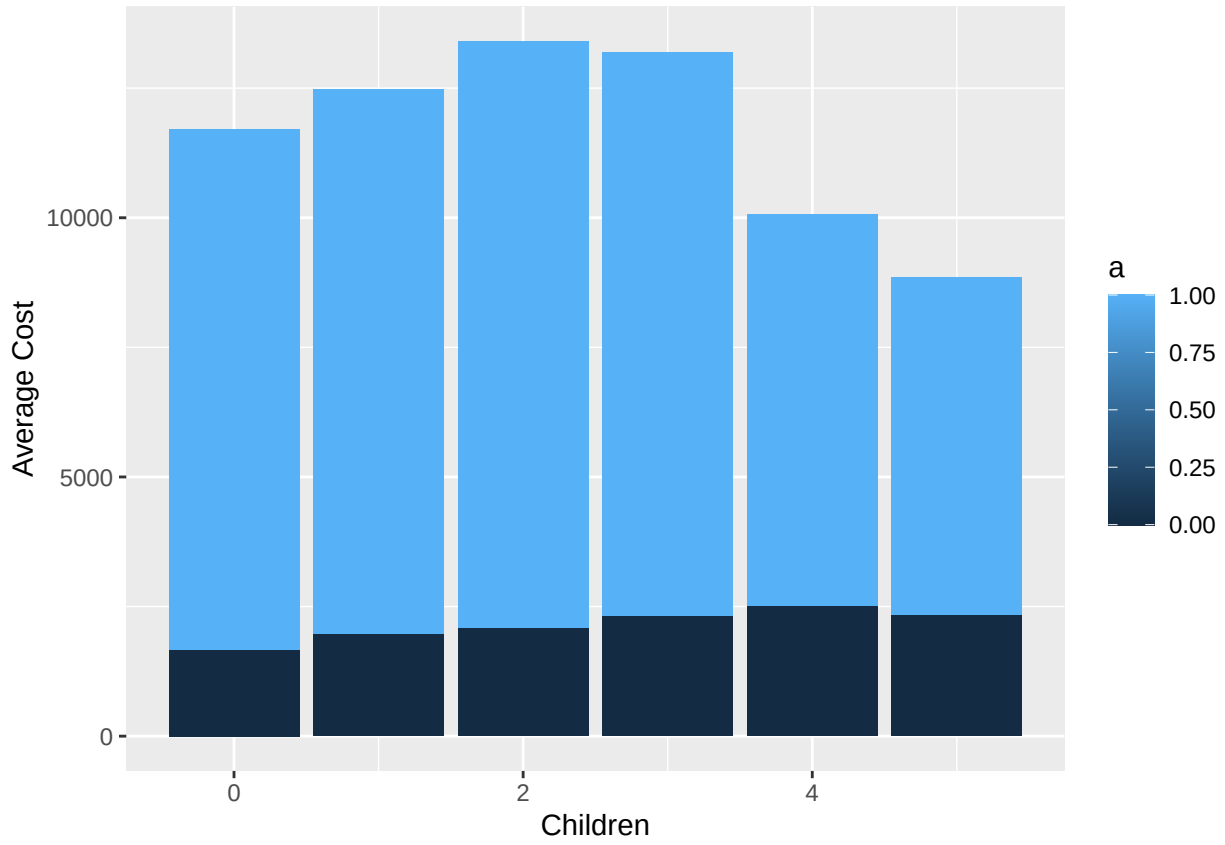

```
rm(s,a,b,c)

#Children
s=health%>%group_by(Expensive,children) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$children
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Children")+ylab("Total Cost")
```



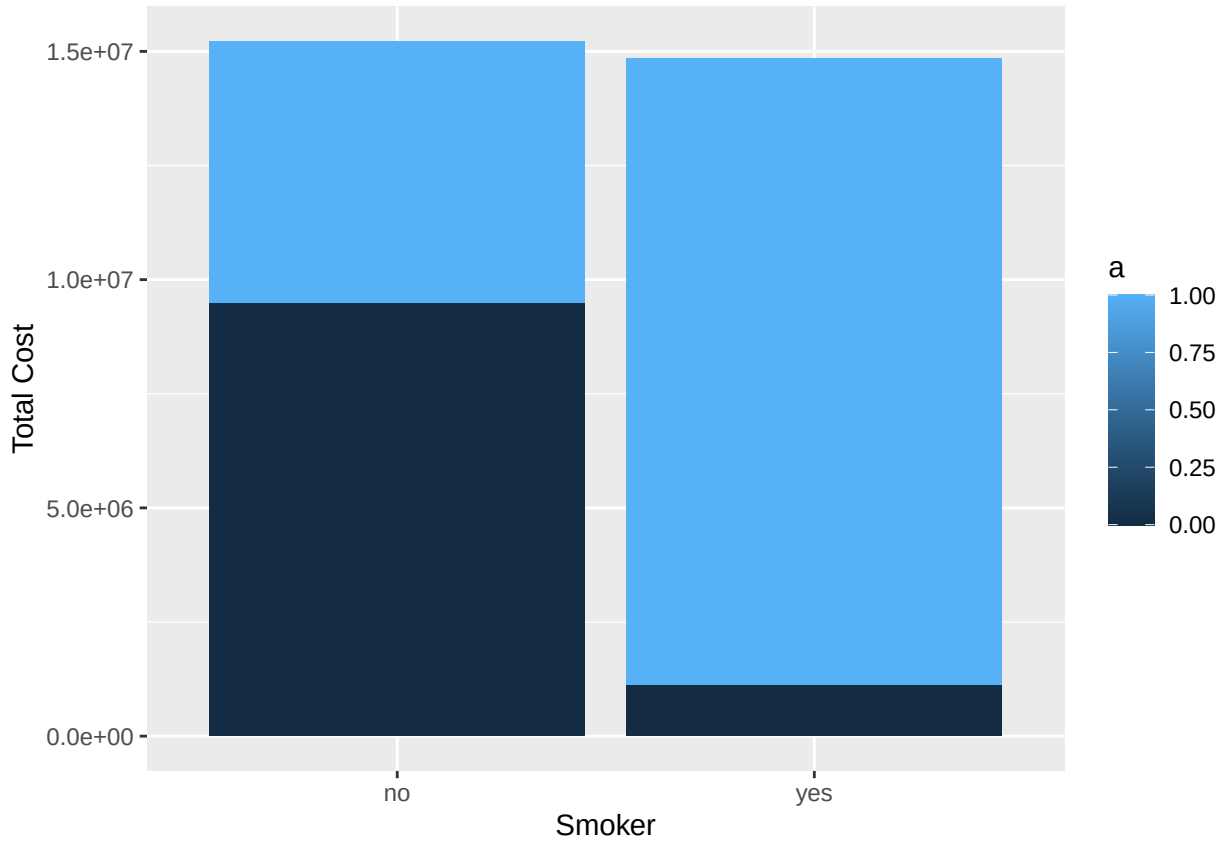

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Children")+ylab("Average Cost")
```



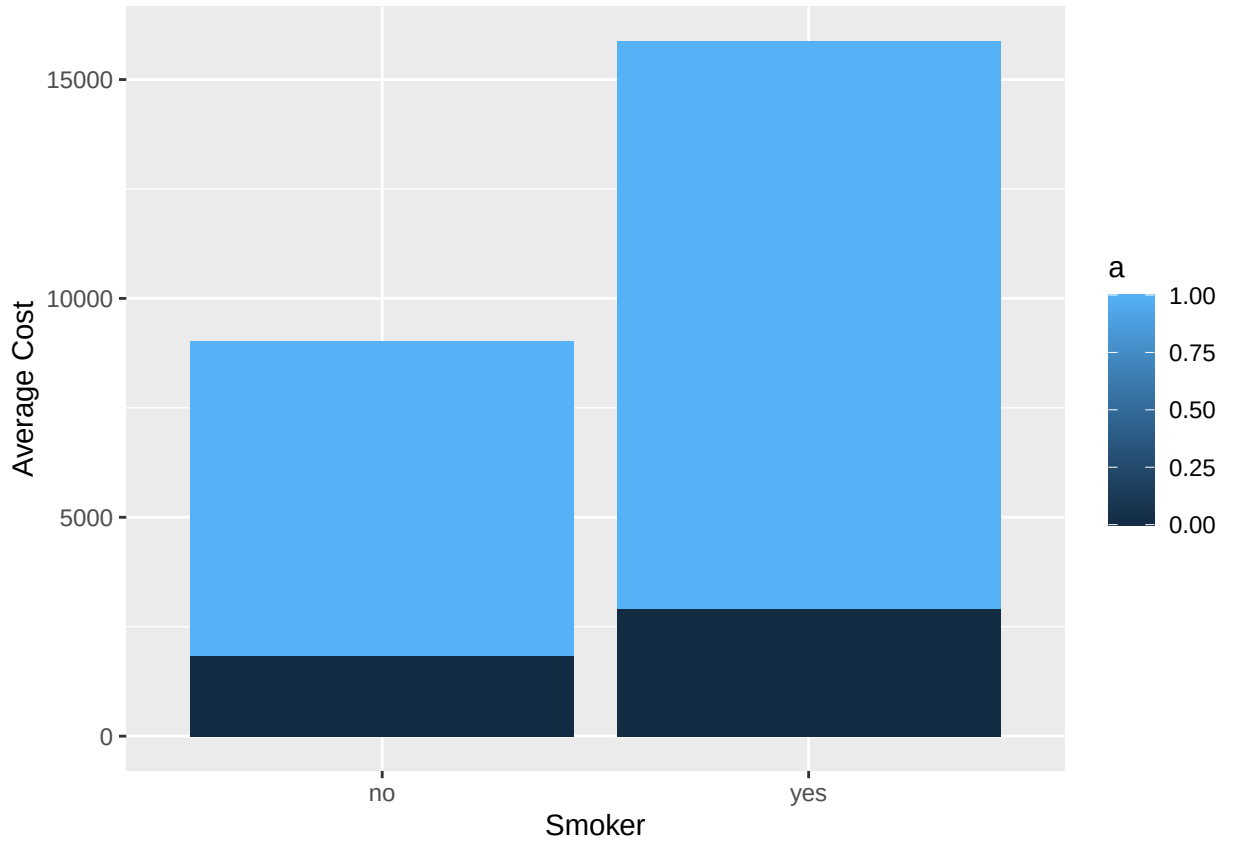

```
rm(s,a,b,c)

#hist(health$smoker)
s=health%>%group_by(Expensive,smoker) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$smoker
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Smoker")+ylab("Total Cost")
```



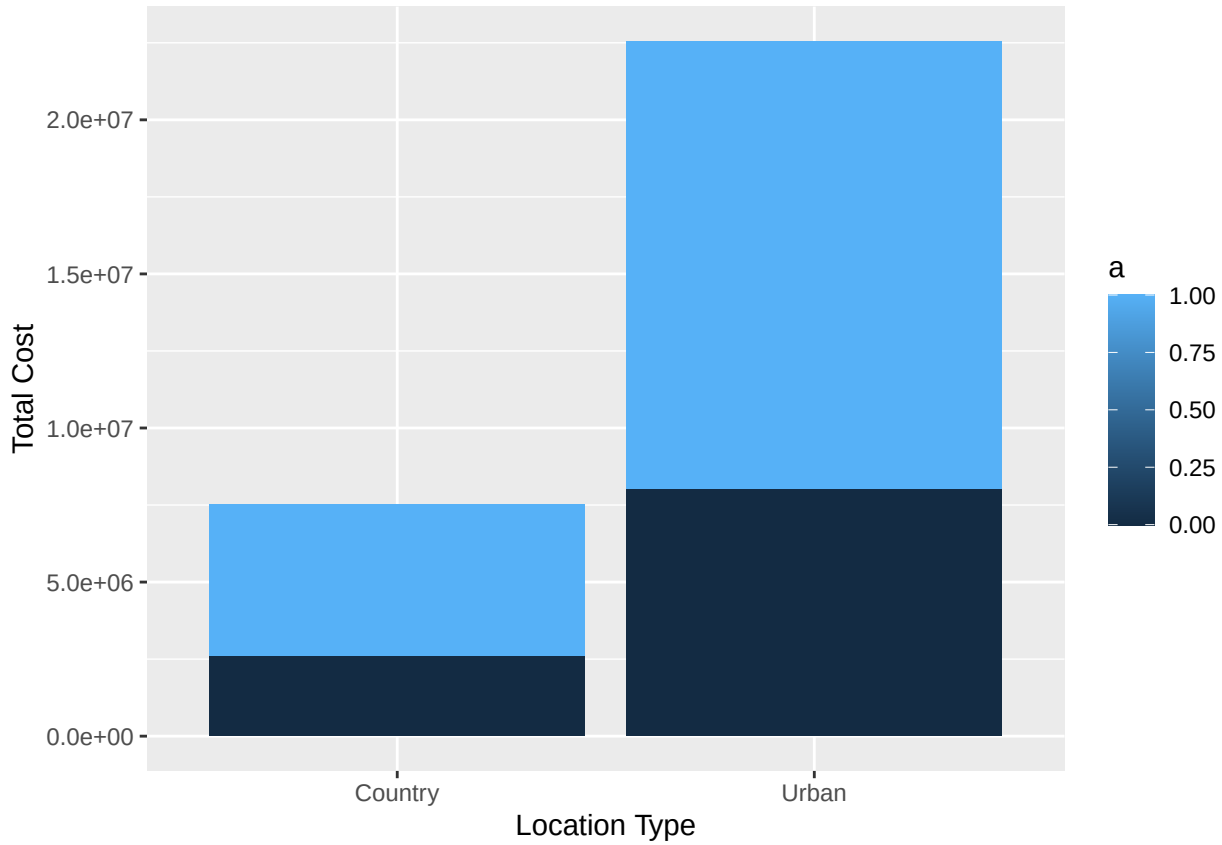

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Smoker")+ylab("Average Cost")
```



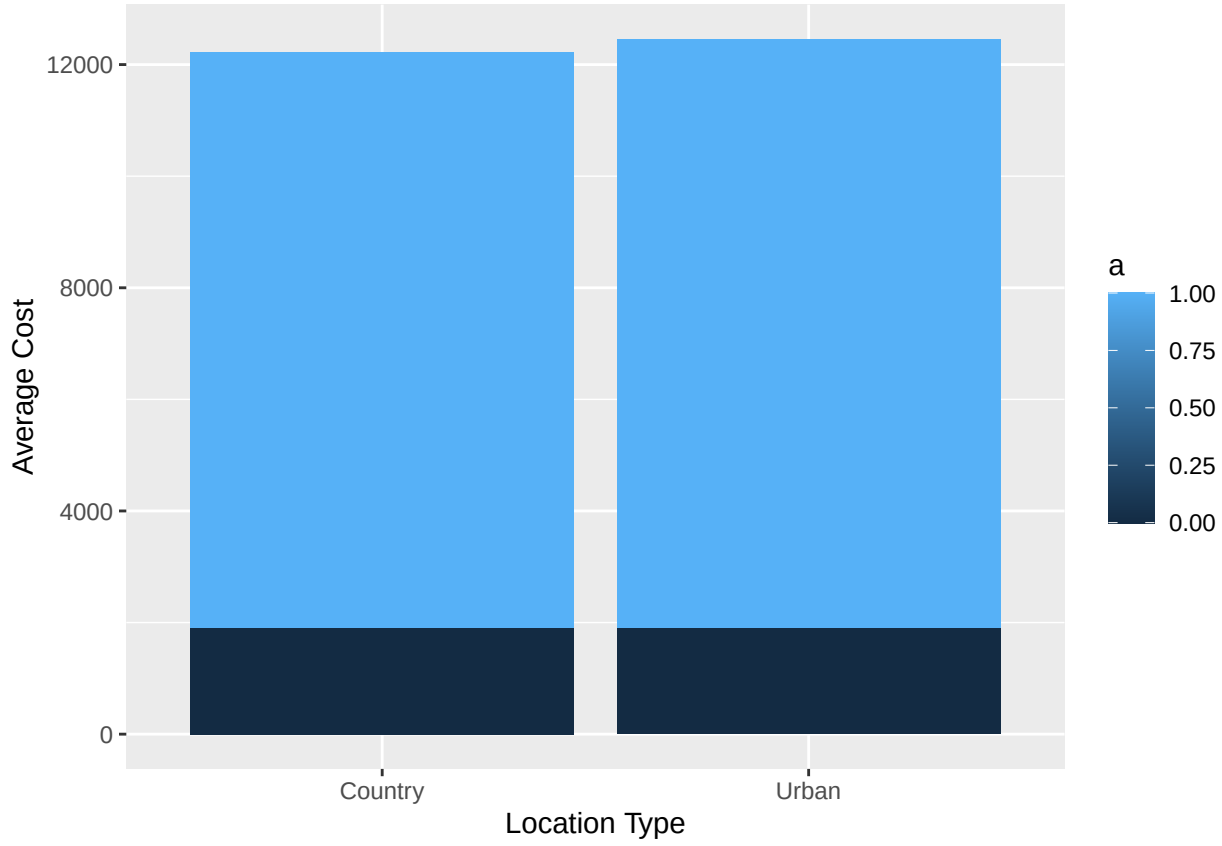

```
rm(s,a,b,c)


#Location_type
s=health%>%group_by(Expensive,location_type) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=(s$location_type)
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Location Type")+ylab("Total Cost")
```



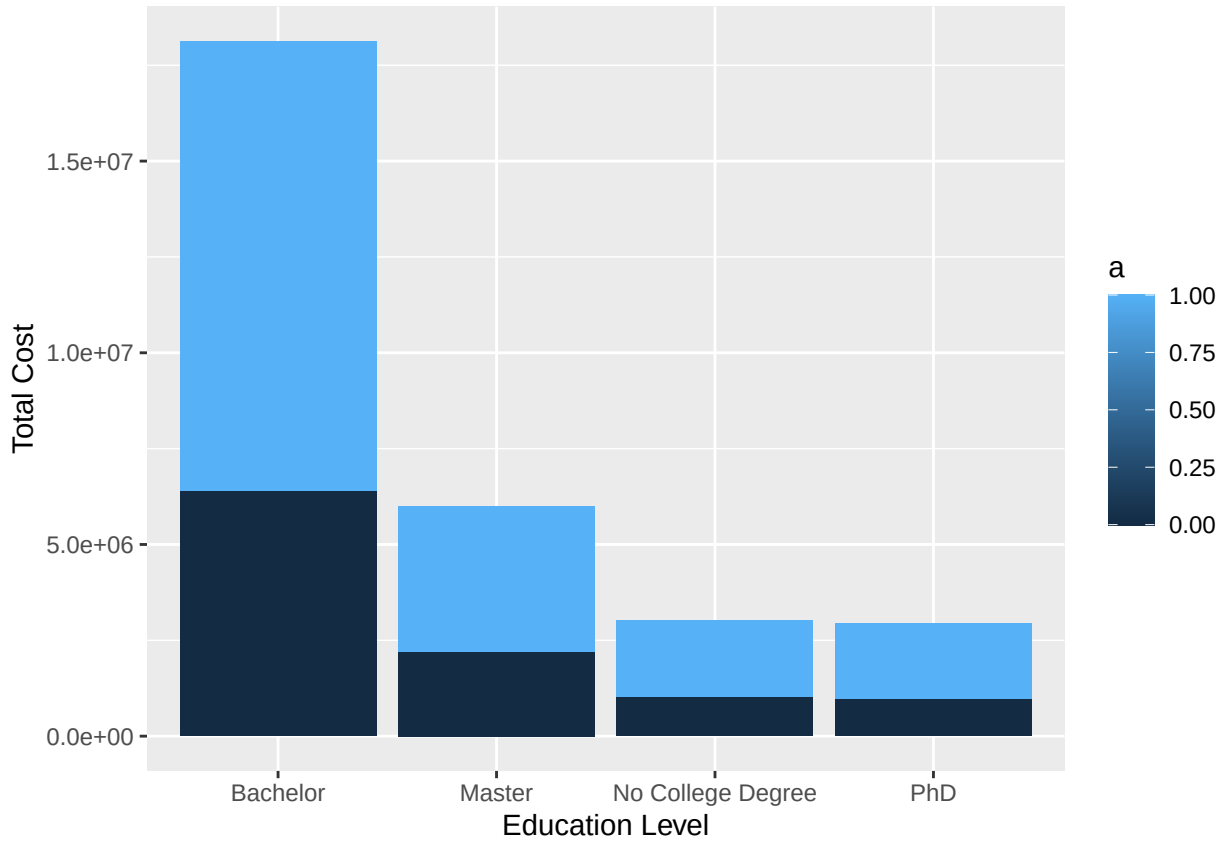

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Location Type")+ylab("Average Cost")
```



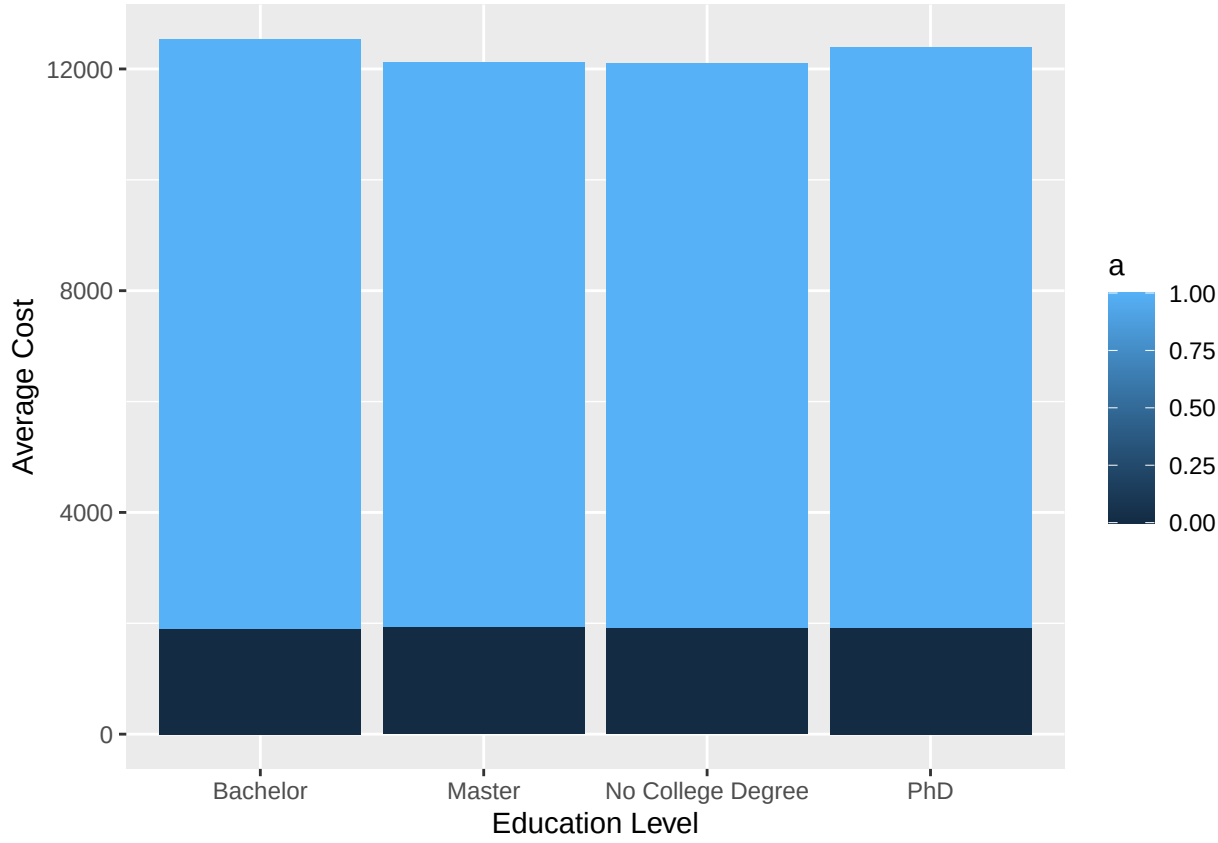

```
rm(s,a,b,c)

#hist(health$education_level)
s=health%>%group_by(Expensive,education_level) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=(s$education_level)
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Education Level")+ylab("Total Cost")
```



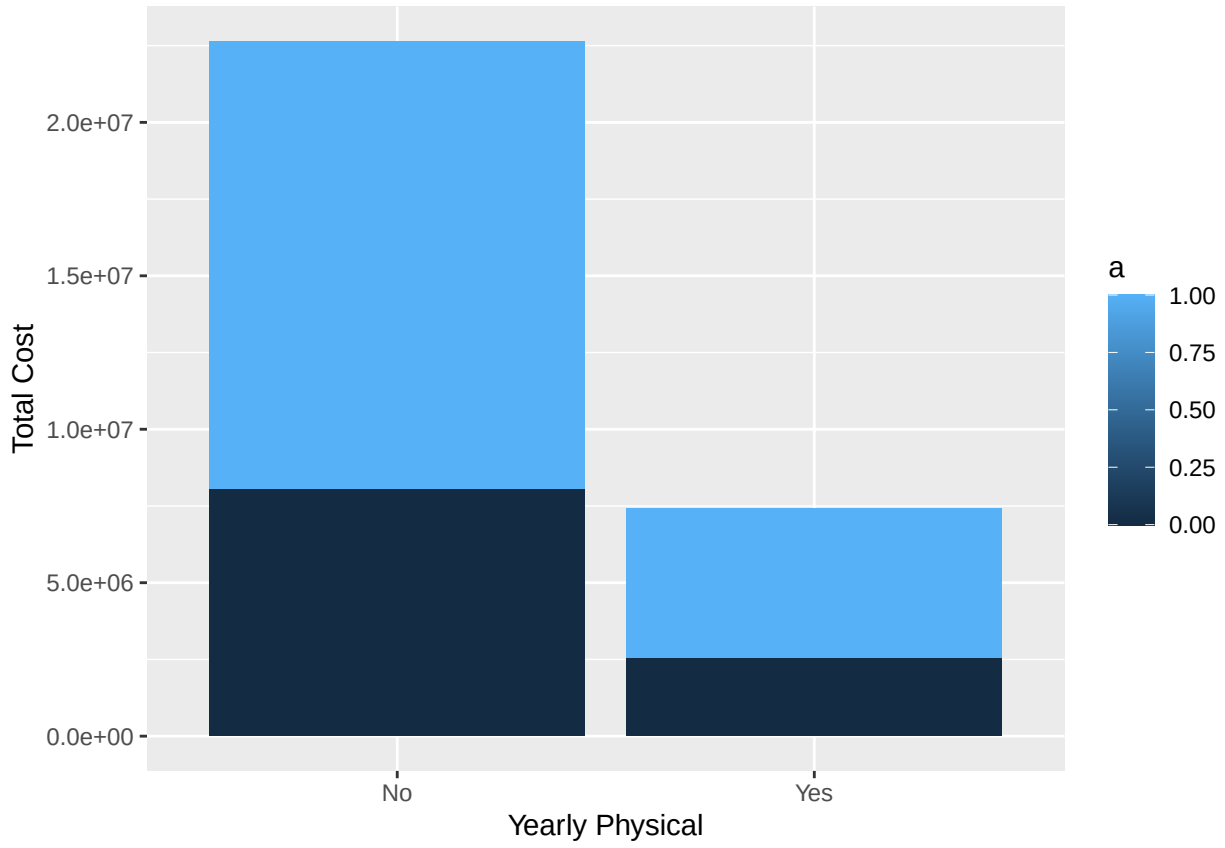

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Education Level")+ylab("Average Cost")
```



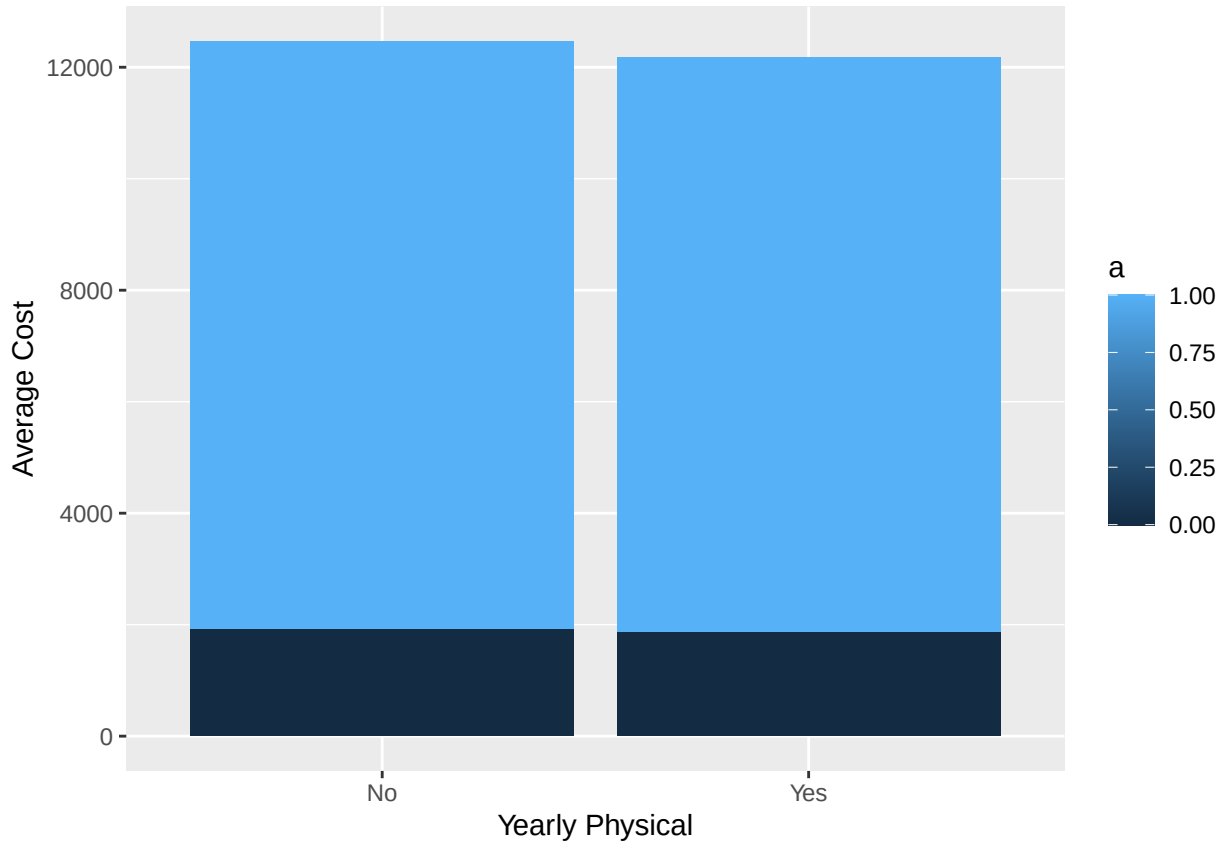

```
rm(s,a,b,c)

#hist(health$yearly_physical)
s=health%>%group_by(Expensive,yearly_physical) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$yearly_physical
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Yearly Physical")+ylab("Total Cost")
```



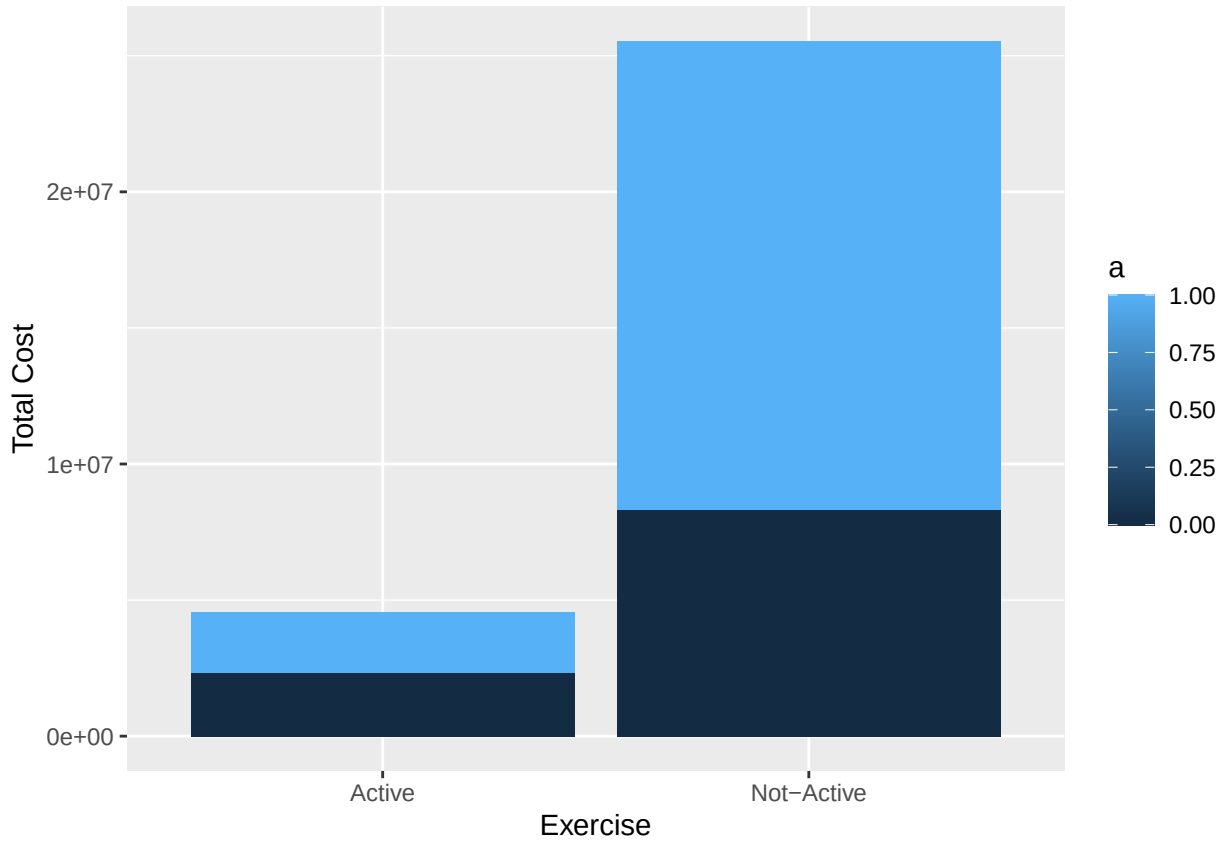

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Yearly Physical")+ylab("Average Cost")
```



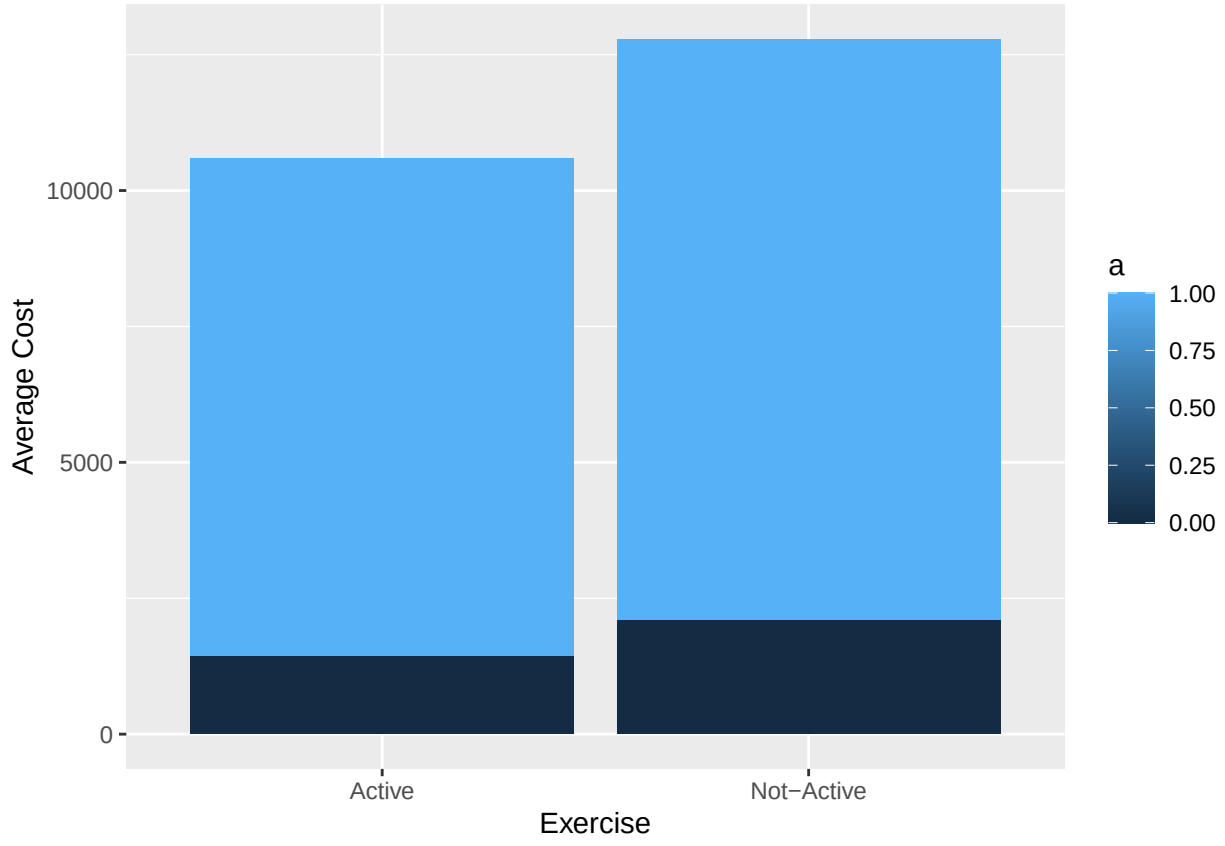

```
rm(s,a,b,c)

#hist(health$exercise)
s=health%>%group_by(Expensive,exercise) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$exercise
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Exercise")+ylab("Total Cost")
```



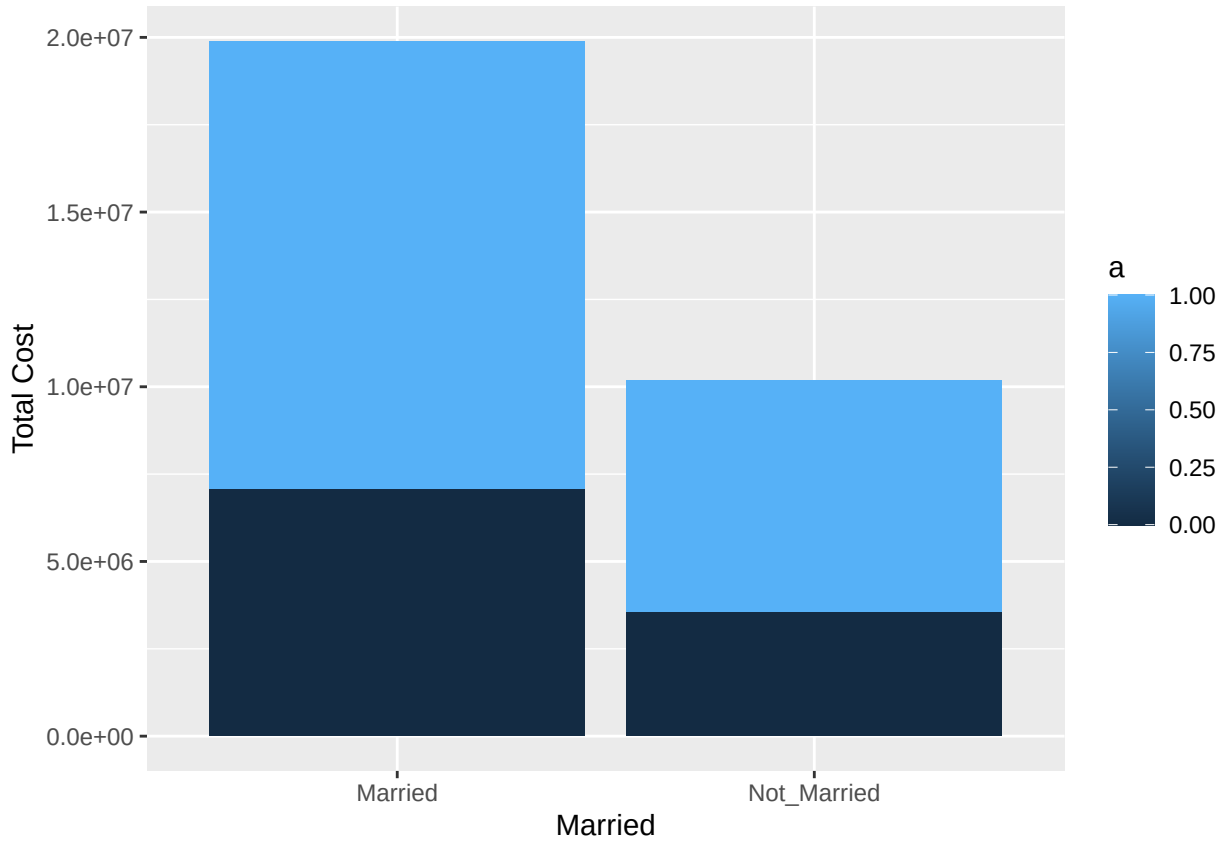

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Exercise")+ylab("Average Cost")
```



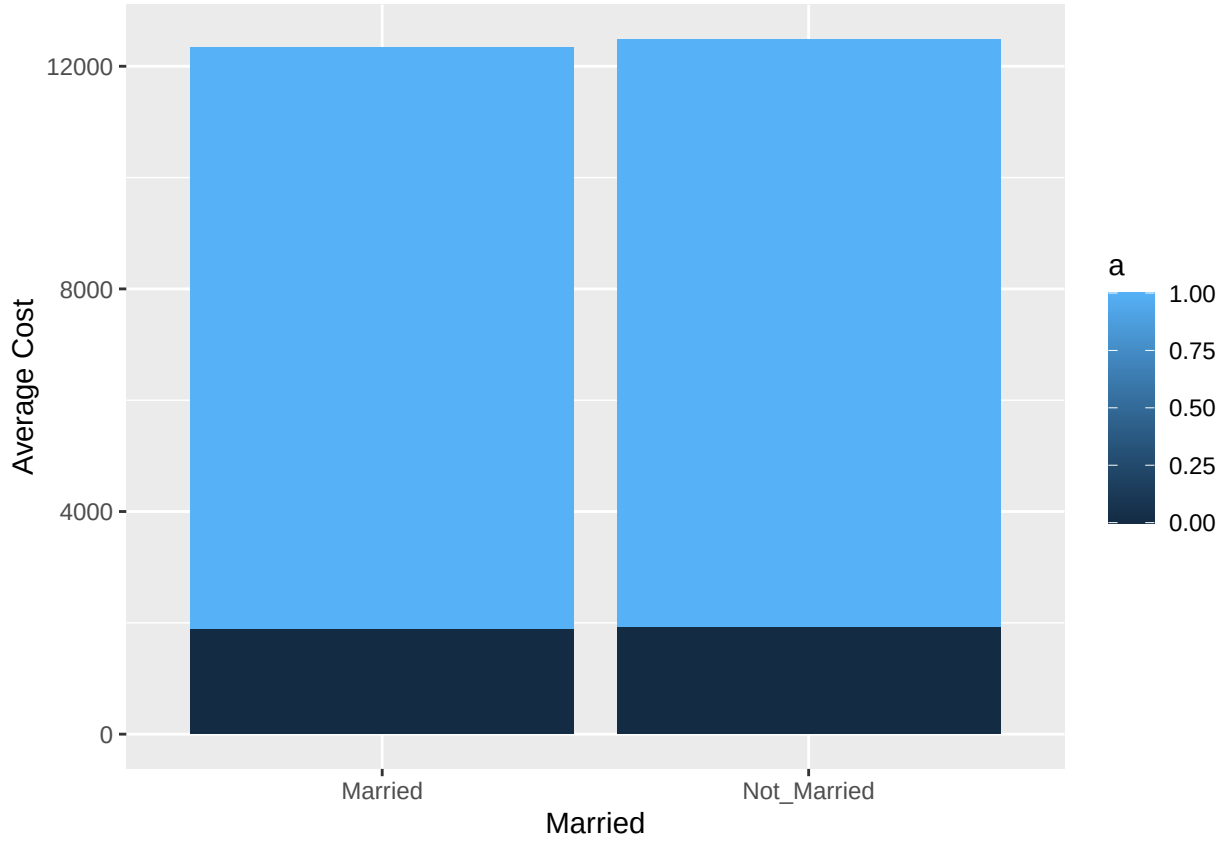

```
rm(s,a,b,c)

#hist(health$married)
s=health%>%group_by(Expensive,married) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=s$married
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Married")+ylab("Total Cost")
```



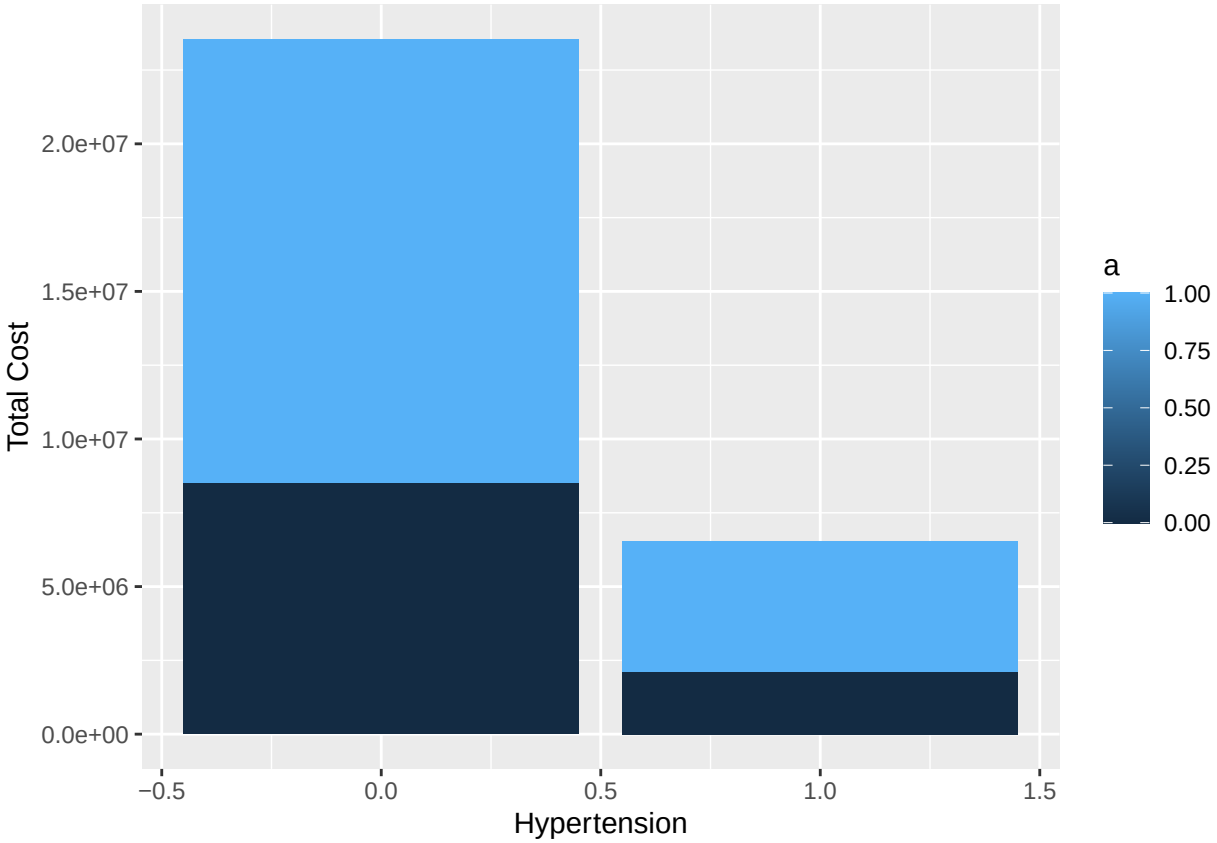

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Married")+ylab("Average Cost")
```



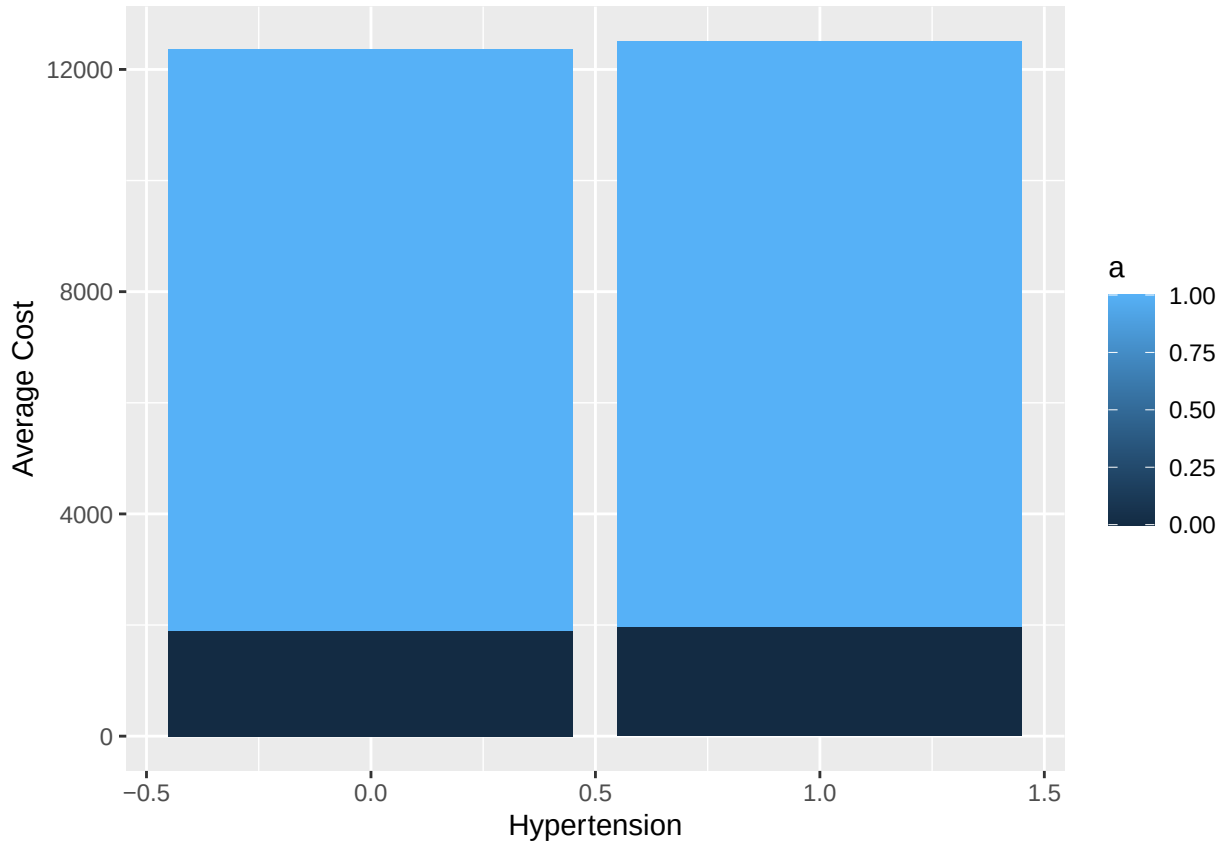

```
rm(s,a,b,c)

#hist(health$hypertension)
s=health%>%group_by(Expensive,hypertension) %>%summarise(Freq = sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

```
a=s$Expensive
b=s$Freq
c=(s$hypertension)
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Hypertension")+ylab("Total Cost")
```



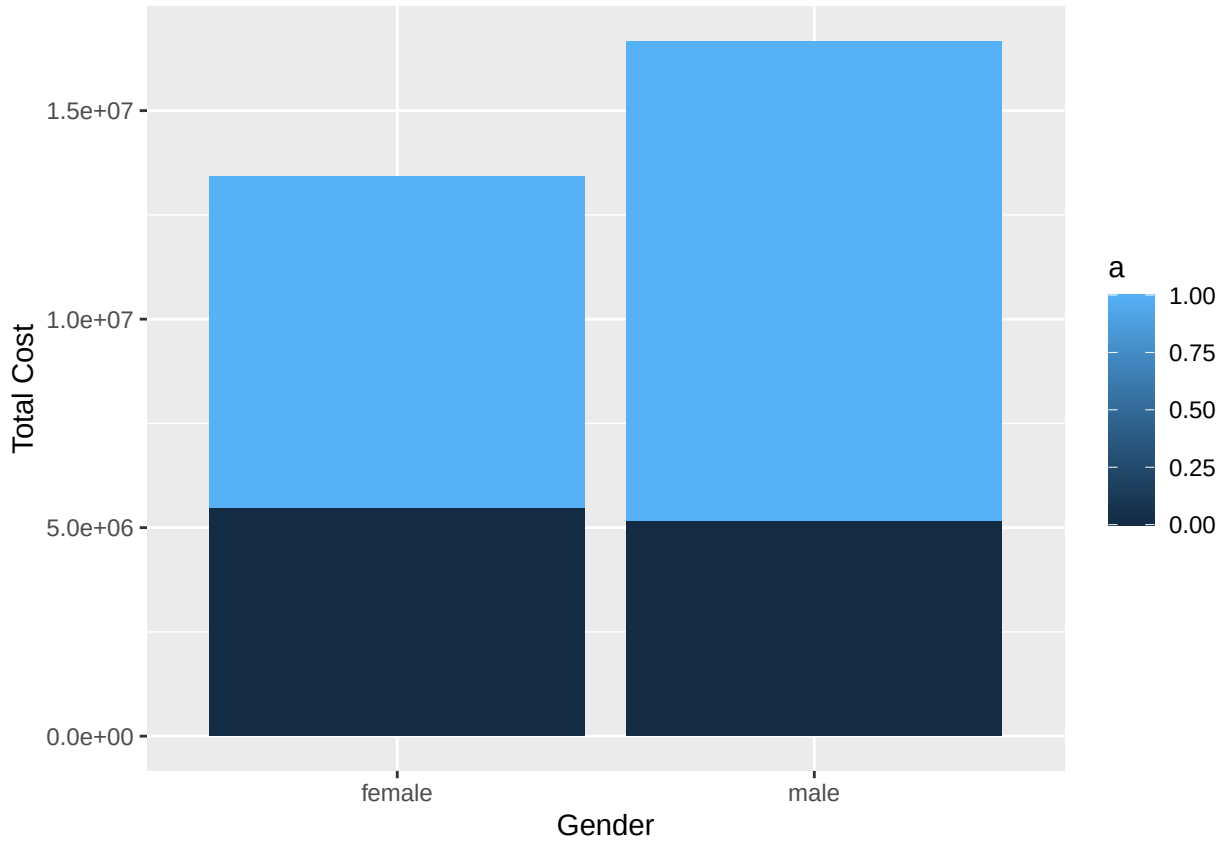

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Hypertension")+ylab("Average Cost")
```



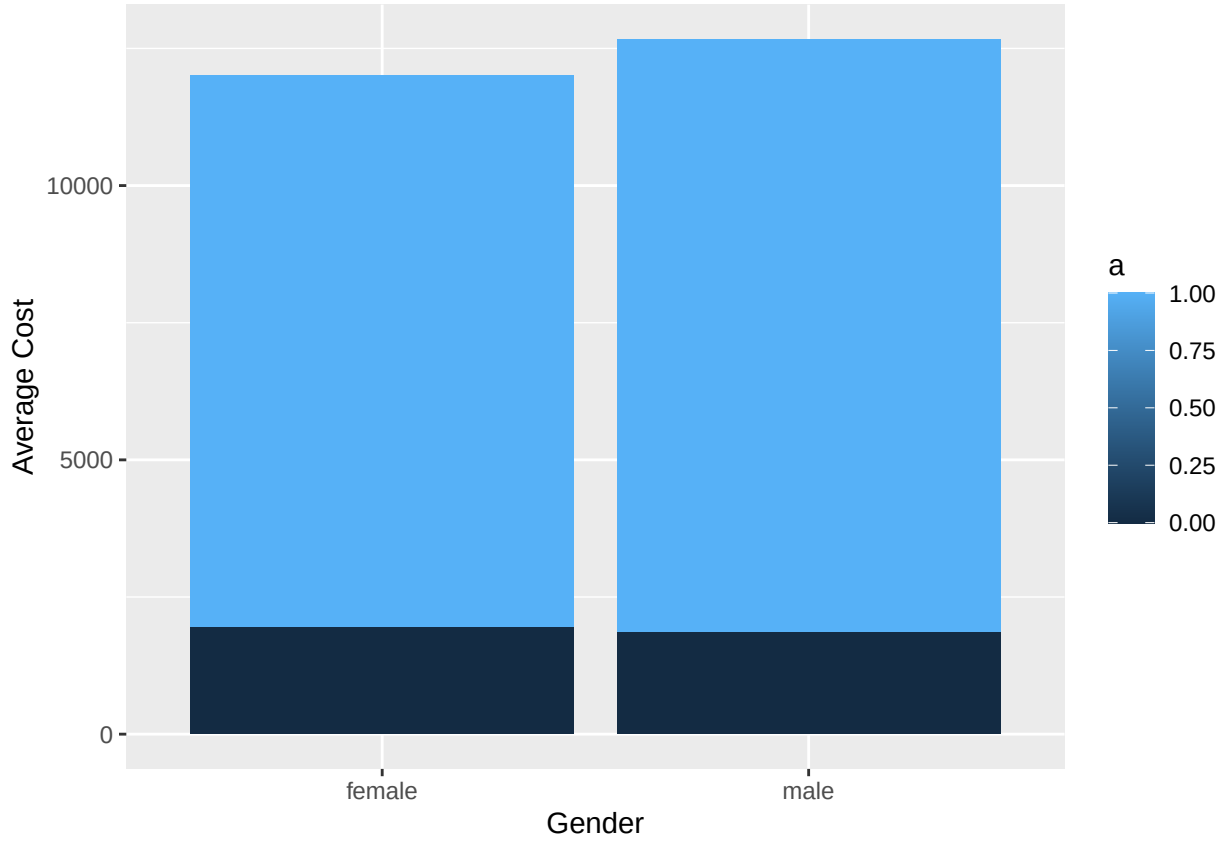

```
rm(s,a,b,c)

#hist(health$gender)
s=health%>%group_by(Expensive,gender)%>%summarise(Freq = sum(cost),avg=sum(cost),avg=mean(cost))
```

```
## `summarise()` has grouped output by 'Expensive'. You can override using the
## `.groups` argument.
```

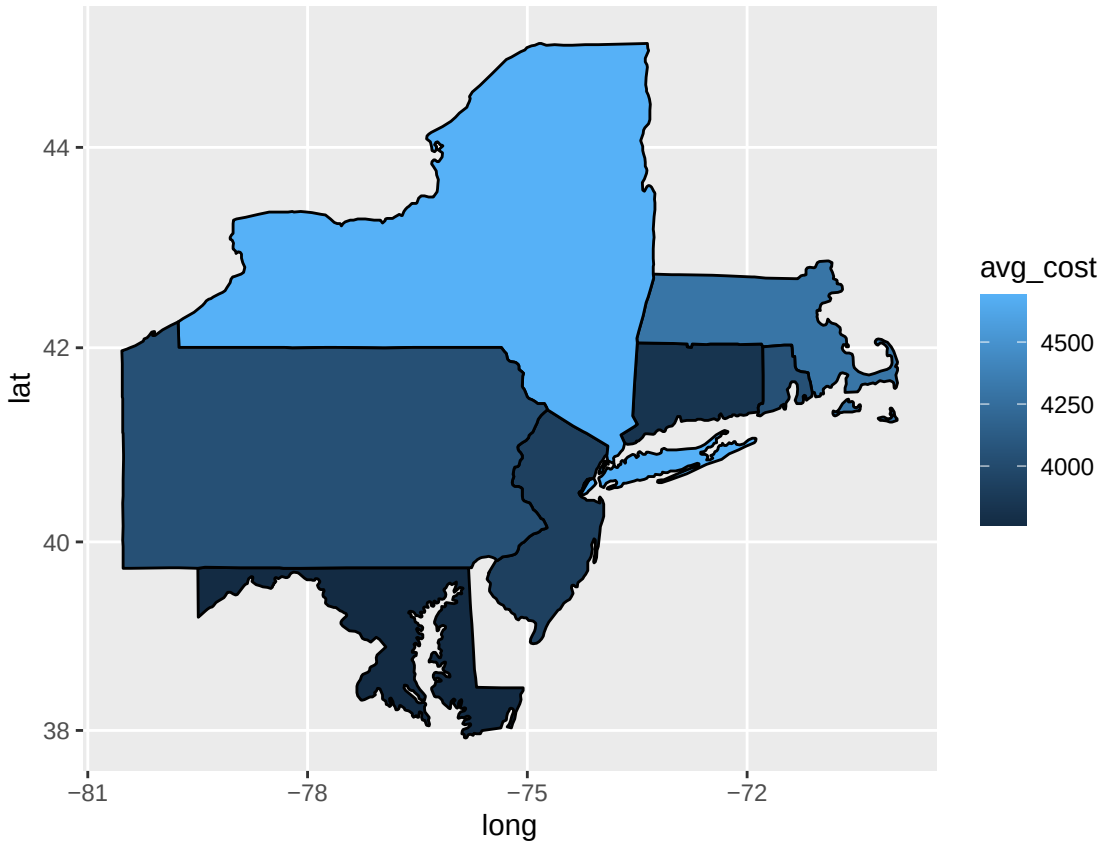
```
a=s$Expensive
b=s$Freq
c=(s$gender)
d=s$avg
ggplot(s, aes(y=b,x=c,fill=a))+geom_bar(stat='identity')+xlab("Gender")+ylab("Total Cost")
```



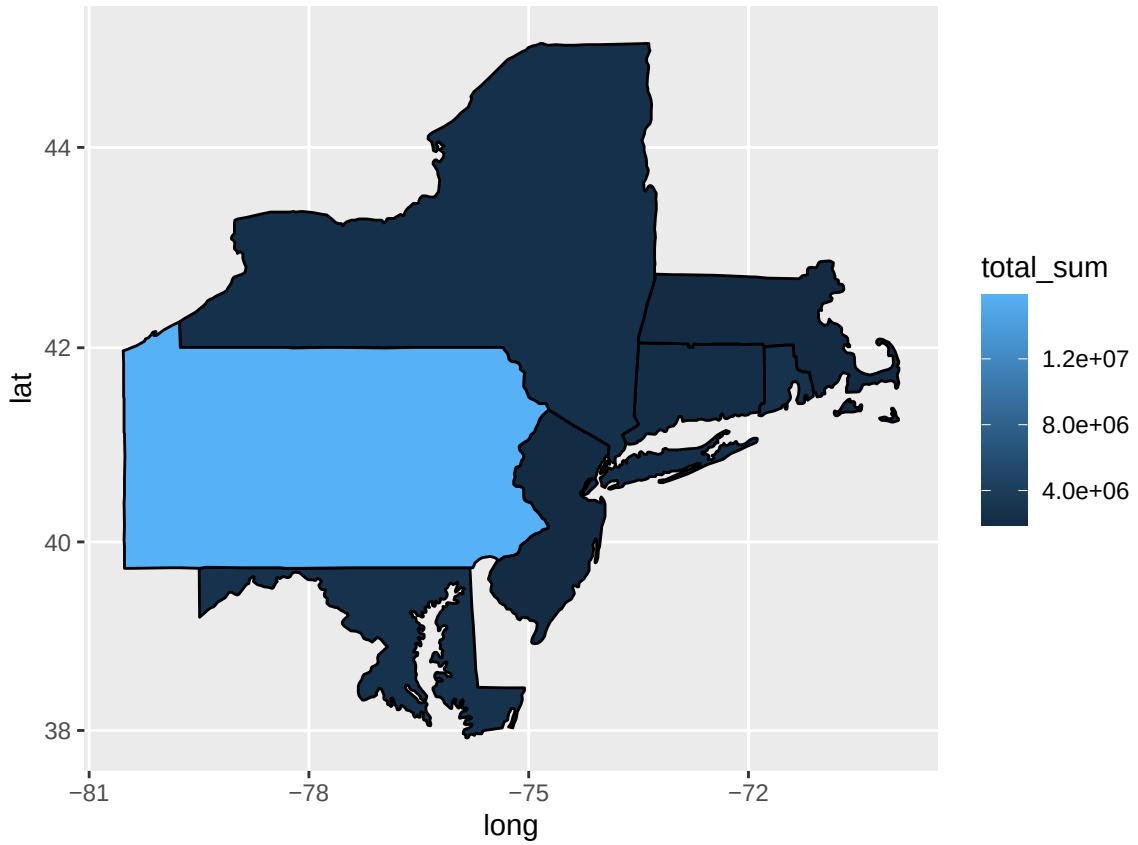

```
ggplot(s, aes(y=d,x=c,fill=a))+geom_bar(stat='identity')+xlab("Gender")+ylab("Average Cost")
```




```
rm(s,a,b,c)


library(usmap)
library(ggplot2)
us_states=map_data("state")
health$location=tolower(health$location)
data_merge=health %>% group_by(location) %>% summarize(avg_cost=mean(cost),total_sum=sum(cost))
merged=merge(data_merge,us_states,by.x='location',by.y='region')
merged=merged %>% arrange(order)
map=ggplot(merged)+geom_polygon(aes(x=long,y=lat,group=group,fill=avg_cost),color="black")
map+coord_map()
```




```
map=ggplot(merged)+geom_polygon(aes(x=long,y=lat,group=group,fill=total_sum),color="black")
map+coord_map()
```



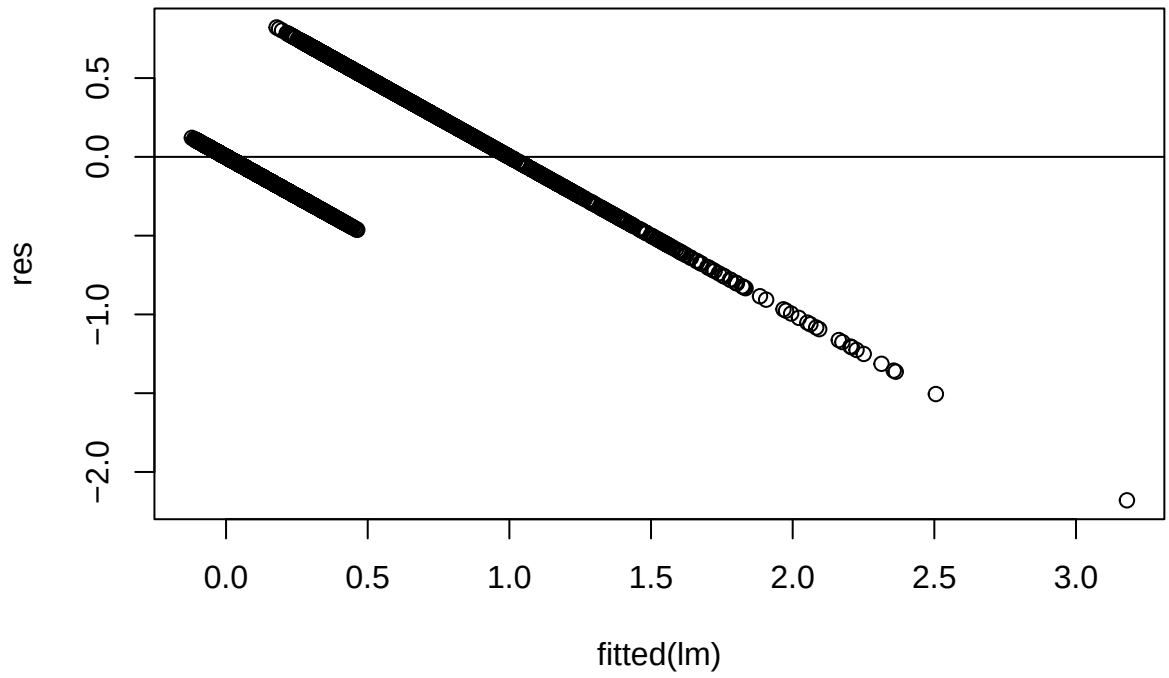
#Linear Regression

```
#First Iteration
lm<-lm(Expensive~.,data=health)
summary(lm)
```

```
##
## Call:
## lm(formula = Expensive ~ ., data = health)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.17984 -0.17003 -0.05160  0.04612  0.82159
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -2.231e-01  2.542e-02  -8.774  < 2e-16 ***
## X                             4.669e-10  6.021e-10   0.776   0.4381
## age                           1.984e-03  2.536e-04   7.824 5.83e-15 ***
## bmi                           3.081e-03  5.769e-04   5.340 9.58e-08 ***
## children                     -8.852e-04  2.698e-03  -0.328   0.7428
## smokeryes                     1.914e-01  1.133e-02  16.884  < 2e-16 ***
## locationmaryland              6.484e-03  1.550e-02   0.418   0.6757
## locationmassachusetts        -4.649e-03  1.750e-02  -0.266   0.7905
## locationnew jersey            1.910e-02  1.712e-02   1.115   0.2647
## locationnew york              1.569e-02  1.675e-02   0.936   0.3491
```

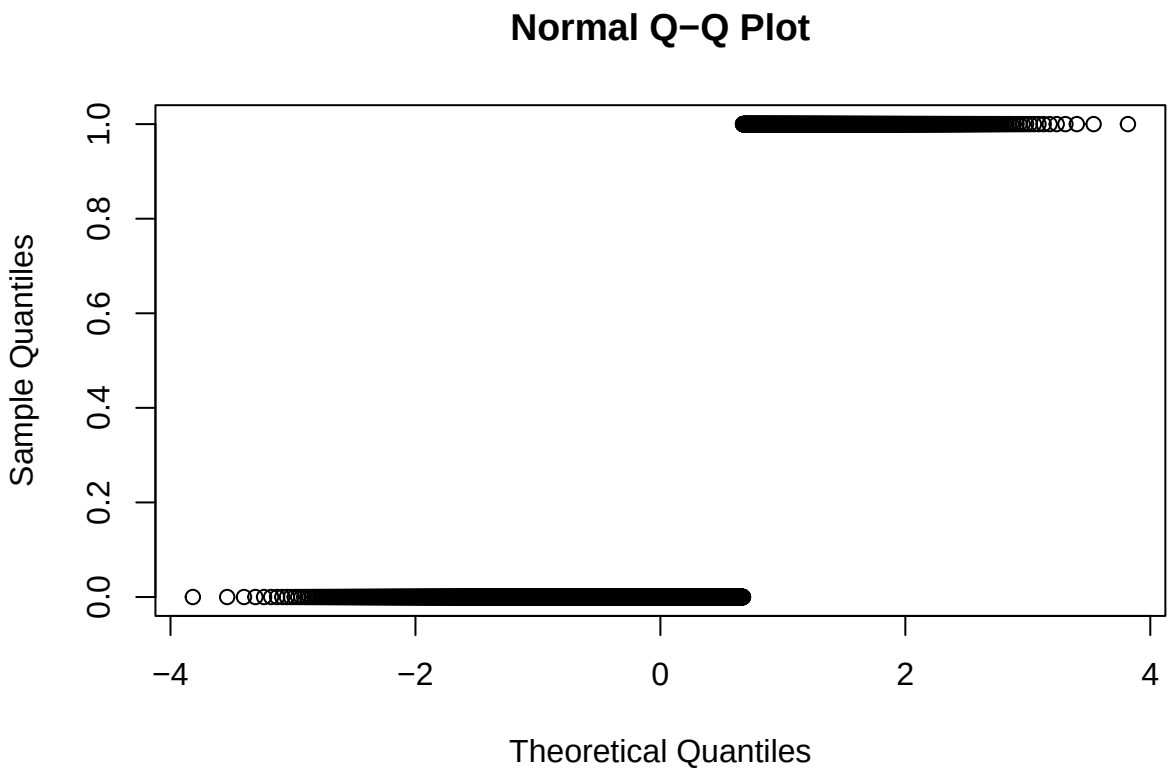
```
## locationpennsylvania              4.201e-03  1.236e-02   0.340   0.7339
## locationrhode island             -4.229e-03  1.571e-02  -0.269   0.7878
## location_typeUrban               -9.352e-03  7.532e-03  -1.242   0.2144
## education_levelMaster             4.324e-03  8.372e-03   0.516   0.6056
## education_levelNo College Degree  1.353e-02  1.112e-02   1.216   0.2239
## education_levelPhD                3.869e-03  1.144e-02   0.338   0.7351
## yearly_physicalYes                1.428e-02  7.552e-03   1.891   0.0586 .
## exerciseNot-Active                5.030e-02  7.878e-03   6.384 1.83e-10 ***
## marriedNot_Married                9.747e-04  6.918e-03   0.141   0.8880
## hypertension                      1.645e-02  8.141e-03   2.021   0.0433 *
## gendermale                        1.219e-02  6.561e-03   1.858   0.0632 .
## cost                              5.253e-05  1.010e-06  52.035  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2805 on 7402 degrees of freedom
## Multiple R-squared:  0.5817, Adjusted R-squared:  0.5805
## F-statistic: 490.2 on 21 and 7402 DF,  p-value: < 2.2e-16
```

```
res=resid(lm)
plot(fitted(lm),res)+abline(0,0)
```




```
## integer(0)
```

```
qqnorm(health$Expensive)
```




```
#Second Iteration
#lm2=lm(Expensive~age+bmi+smoker+exercise+yearly_physical+hypertension+gender, data=health)
#summary(lm2)
```

```
rm(health)
library(tidyverse)
library(ggplot2)
library(dplyr)

health_raw <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/HMO_data.csv")
```

```
## Rows: 7582 Columns: 14
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): smoker, location, location_type, education_level, yearly_physical, ...
## dbl (6): X, age, bmi, children, hypertension, cost
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
health=health_raw %>% filter(!(is.na(bmi)))
health=health %>% filter(!(is.na(hypertension)))
sapply(health,function(x) sum(is.na(x)))
```

```
##               X             age             bmi        children          smoker
##               0               0               0               0               0
##        location   location_type education_level yearly_physical        exercise
##               0               0               0               0               0
##         married    hypertension          gender            cost
##               0               0               0               0
```

```
sapply(health,function(x) sum(is.null(x)))
```

```
##               X             age             bmi        children          smoker
##               0               0               0               0               0
##        location   location_type education_level yearly_physical        exercise
##               0               0               0               0               0
##         married    hypertension          gender            cost
##               0               0               0               0
```

```
threshold=quantile(health$cost,probs=(.75))
health$cost <- ifelse(health$cost>=threshold, 1, 0)
glimpse(health)
```

```
## Rows: 7,424
## Columns: 14
## $ X               <dbl> 1, 2, 3, 4, 5, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 1~
## $ age             <dbl> 18, 19, 27, 34, 32, 47, 36, 59, 24, 61, 22, 57, 26, 18~
## $ bmi             <dbl> 27.900, 33.770, 33.000, 22.705, 28.880, 33.440, 29.830~
## $ children        <dbl> 0, 1, 3, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ smoker          <chr> "yes", "no", "no", "no", "no", "no", "no", "no", "no",~
## $ location        <chr> "CONNECTICUT", "RHODE ISLAND", "MASSACHUSETTS", "PENNS~
## $ location_type   <chr> "Urban", "Urban", "Urban", "Country", "Country", "Urba~
## $ education_level <chr> "Bachelor", "Bachelor", "Master", "Master", "PhD", "Ba~
## $ yearly_physical <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No", ~
## $ exercise        <chr> "Active", "Not-Active", "Active", "Not-Active", "Not-A~
## $ married         <chr> "Married", "Married", "Married", "Married", "Married",~
## $ hypertension    <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ gender          <chr> "female", "male", "male", "male", "male", "female", "m~
## $ cost            <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, ~
```

#Replacing values with numeric values for Models

```
#smoker
health$smoker<-str_replace_all(health$smoker,"no","0")
health$smoker<-str_replace_all(health$smoker,"yes","1")
#location_type
health$location_type<-str_replace_all(health$location_type,"Country","0")
health$location_type<-str_replace_all(health$location_type,"Urban","1")
#education_level
health$education_level<-str_replace_all(health$education_level,"No College Degree","0")
```

```
health$education_level<-str_replace_all(health$education_level,"Bachelor","1")
health$education_level<-str_replace_all(health$education_level,"Master","2")
health$education_level<-str_replace_all(health$education_level,"PhD","3")
#yearly_physical
health$yearly_physical<-str_replace_all(health$yearly_physical,"No","0")
health$yearly_physical<-str_replace_all(health$yearly_physical,"Yes","1")
#exercise
health$exercise<-str_replace_all(health$exercise,"Not-Active","0")
health$exercise<-str_replace_all(health$exercise,"Active","1")
#married
health$married<-str_replace_all(health$married,"Not_Married","0")
health$married<-str_replace_all(health$married,"Married","1")
#gender
# Make sure to re-code female first
health$gender<-str_replace_all(health$gender,"female","1")
health$gender<-str_replace_all(health$gender,"male","0")

glimpse(health)
```

```
## Rows: 7,424
## Columns: 14
## $ X               <dbl> 1, 2, 3, 4, 5, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 1~
## $ age             <dbl> 18, 19, 27, 34, 32, 47, 36, 59, 24, 61, 22, 57, 26, 18~
## $ bmi             <dbl> 27.900, 33.770, 33.000, 22.705, 28.880, 33.440, 29.830~
## $ children        <dbl> 0, 1, 3, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ smoker          <chr> "1", "0", "0", "0", "0", "0", "0", "0", "0", "1", "0",~
## $ location        <chr> "CONNECTICUT", "RHODE ISLAND", "MASSACHUSETTS", "PENNS~
## $ location_type   <chr> "1", "1", "1", "0", "0", "1", "1", "0", "1", "1", "1",~
## $ education_level <chr> "1", "1", "2", "2", "3", "1", "1", "1", "1", "0", "1",~
## $ yearly_physical <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",~
## $ exercise        <chr> "1", "0", "1", "0", "0", "0", "1", "0", "1", "1", "0",~
## $ married         <chr> "1", "1", "1", "1", "1", "1", "1", "1", "1", "1", "1",~
## $ hypertension    <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ gender          <chr> "1", "0", "0", "0", "0", "1", "0", "1", "0", "1", "0",~
## $ cost            <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, ~
```

```
#view(health)
#summary(health)
```

#Removing NA's and Changing the Column type to numerics

```
library(imputeTS)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
#install.packages("zoo")
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following object is masked from 'package:imputeTS':
##
##     na.locf

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
#delete.na <- function(DF, n) {
#  DF[rowSums(is.na(DF)) <= n,]
#}

health$smoker <- as.numeric(health$smoker)
health$location_type<- as.numeric(health$location_type)
health$education_level <- as.numeric(health$education_level)
health$yearly_physical <- as.numeric(health$yearly_physical)
health$married <- as.numeric(health$married)
health$gender <- as.numeric(health$gender)
health$exercise <- as.numeric(health$exercise)
health$bmi=as.numeric(health$bmi)
health$cost=as.factor(health$cost)


glimpse(health)
```

```
## Rows: 7,424
## Columns: 14
## $ X               <dbl> 1, 2, 3, 4, 5, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 1~
## $ age             <dbl> 18, 19, 27, 34, 32, 47, 36, 59, 24, 61, 22, 57, 26, 18~
## $ bmi             <dbl> 27.900, 33.770, 33.000, 22.705, 28.880, 33.440, 29.830~
## $ children        <dbl> 0, 1, 3, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ smoker          <dbl> 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, ~
## $ location        <chr> "CONNECTICUT", "RHODE ISLAND", "MASSACHUSETTS", "PENNS~
## $ location_type   <dbl> 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, ~
## $ education_level <dbl> 1, 1, 2, 2, 3, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 3, 3, ~
## $ yearly_physical <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, ~
## $ exercise        <dbl> 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, ~
## $ married         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, ~
## $ hypertension    <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ gender          <dbl> 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, ~
## $ cost            <fct> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, ~
```

```
#view(health)
#summary(health)
#nrow(health)
```

#Partioning the Dataset for Training and Testing

```
library(kernlab)
```

```
##
## Attaching package: 'kernlab'
```

```
## The following object is masked from 'package:purrr':
##
##     cross

## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```
health$cost=as.factor(health$cost)
trainList <-  createDataPartition(y=health$cost,p=.65,list=FALSE)
trainset <- health[trainList,]
testset <- health[-trainList,]
```

#SVM Model

```
set.seed(123)
library(cvms)
library(tibble)
svm <- train(as.factor(cost) ~ ., data=trainset, method="svmRadial",preProc=c("center","scale"))
svm
```

```
## Support Vector Machines with Radial Basis Function Kernel
##
## 4827 samples
##   13 predictor
##    2 classes: '0', '1'
##
## Pre-processing: centered (18), scaled (18)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 4827, 4827, 4827, 4827, 4827, 4827, ...
## Resampling results across tuning parameters:
##
##   C     Accuracy   Kappa
##   0.25  0.8590218  0.5721103
##   0.50  0.8624225  0.5780658
##   1.00  0.8637135  0.5834370
##
## Tuning parameter 'sigma' was held constant at a value of 0.04031823
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were sigma = 0.04031823 and C = 1.
```

```r
svmpred=predict(svm,testset)
table(svmpred, testset$cost)
```

```
##
## svmpred    0    1
##       0 1891  300
##       1   57  349
```

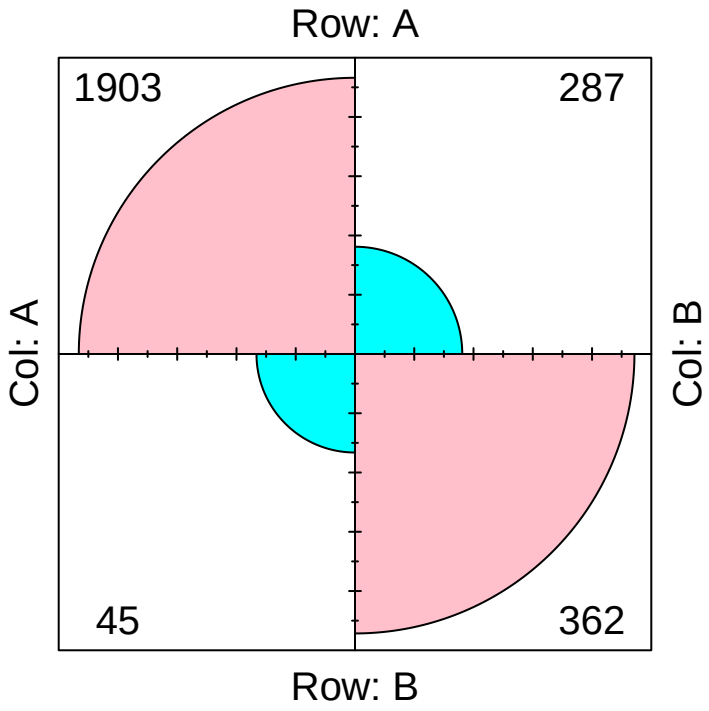
```r
sum(diag(table(svmpred,testset$cost)))/sum(table(svmpred,testset$cost))
```

```
## [1] 0.8625337
```

```r
confusionMatrix(svmpred, testset$cost)
```

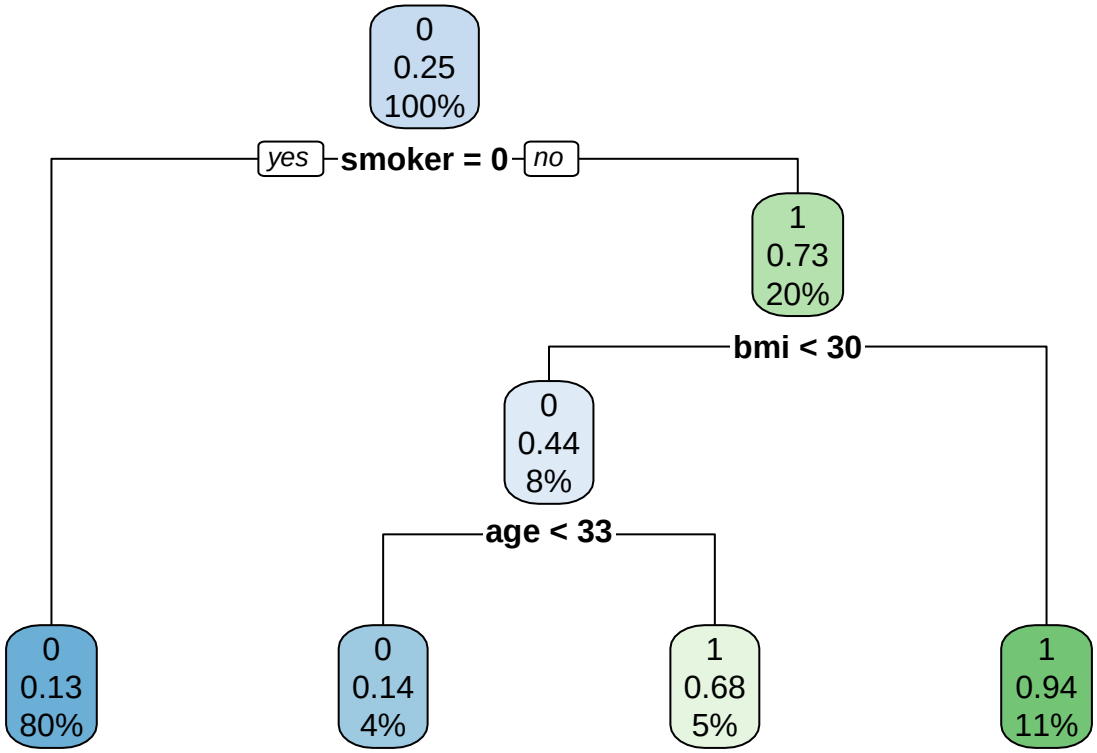
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1891  300
##          1   57  349
##
##                Accuracy : 0.8625
##                  95% CI : (0.8487, 0.8756)
##     No Information Rate : 0.7501
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.581
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9707
##             Specificity : 0.5378
##          Pos Pred Value : 0.8631
##          Neg Pred Value : 0.8596
##              Prevalence : 0.7501
##          Detection Rate : 0.7281
##    Detection Prevalence : 0.8437
##       Balanced Accuracy : 0.7542
##
##        'Positive' Class : 0
##
```

```r
ctable=as.table(matrix(c(1903,287,45,362),nrow=2,byrow=TRUE))

fourfoldplot(ctable, color = c("cyan", "pink"),
             conf.level = 0, margin = 1, main = "Confusion Matrix")
```



#Recursive Partitioning and Regression Trees

```
library(e1071)
library(caret)
library(rpart)
library(ggplot2)
library(rpart.plot)
rpartmodel<-train(cost~age+bmi+smoker+exercise+gender,data=trainset,method="rpart")
rpart.plot(rpartmodel$finalModel)
```




```
rpartPred<-predict(rpartmodel,testset)
confM2<-confusionMatrix(rpartPred,testset$cost)
confM2
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1878  294
##          1   70  355
##
##                Accuracy : 0.8598
##                  95% CI : (0.8459, 0.873)
##     No Information Rate : 0.7501
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5775
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9641
##             Specificity : 0.5470
##          Pos Pred Value : 0.8646
##          Neg Pred Value : 0.8353
##              Prevalence : 0.7501
##          Detection Rate : 0.7231
```

```
##    Detection Prevalence : 0.8363
##       Balanced Accuracy : 0.7555
##
##        'Positive' Class : 0
##
```