Econ 104 Project 3 Marcus Young Geoffrey Penarubia Kyle Almon

2023-03-08

```
library(AER)
```

```
## Loading required package: car

## Loading required package: carData

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

## Loading required package: sandwich

## Loading required package: survival
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --

## v ggplot2 3.4.0      v purrr   1.0.1
## 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()
## x dplyr::recode() masks car::recode()
## x purrr::some()   masks car::some()
```

```
library(readr)
library(knitr)
library(xtable)
library(effects)
```

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
library(broom)
library(jtools)
library(leaps)
library(car)
library(Boruta)
library(lmtest)
library(AICcmodavg)
library(flexmix)
```

```
## Loading required package: lattice
```

```
library(caret)
```

```
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
##
## The following object is masked from 'package:survival':
##
##     cluster
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(RColorBrewer)
library(ggplot2)
library(rlang)
```

```
##
## Attaching package: 'rlang'
##
## The following objects are masked from 'package:purrr':
##
##     %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
##     flatten_raw, invoke, splice
```

```
library(base)
library(xfun)
```

```
##
## Attaching package: 'xfun'
##
## The following objects are masked from 'package:base':
##
##     attr, isFALSE
```

```
library(tinytex)
```

```
##
## Attaching package: 'tinytex'
##
## The following object is masked from 'package:rlang':
##
##     check_installed
```

```
library(stats)
library(TSA)
```

```
##
## Attaching package: 'TSA'
##
## The following object is masked from 'package:readr':
##
```

```
##     spec
##
## The following objects are masked from 'package:stats':
##
##     acf, arima
##
## The following object is masked from 'package:utils':
##
##     tar
```

```r
library(timeSeries)
```

```
## Loading required package: timeDate
##
## Attaching package: 'timeDate'
##
## The following objects are masked from 'package:TSA':
##
##     kurtosis, skewness
##
## The following object is masked from 'package:xtable':
##
##     align
##
##
## Attaching package: 'timeSeries'
##
## The following object is masked from 'package:zoo':
##
##     time<-
```

```r
library(fUnitRoots)
library(fBasics)
```

```
##
## Attaching package: 'fBasics'
##
## The following objects are masked from 'package:TSA':
##
##     kurtosis, skewness
##
## The following object is masked from 'package:flexmix':
##
##     getModel
##
## The following object is masked from 'package:car':
##
##     densityPlot
```

```r
library(tseries)
```

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

```r
library(timsac)
```

```r
library(TTR)
```

```
##
## Attaching package: 'TTR'
##
## The following object is masked from 'package:fBasics':
##
##     volatility
```

```r
library(fpp)
```

```
## Loading required package: forecast
## Registered S3 methods overwritten by 'forecast':
##   method       from
##   fitted.Arima TSA
##   plot.Arima   TSA
## Loading required package: fma
## Loading required package: expsmooth
```

```r
library(strucchange)
```

```
##
## Attaching package: 'strucchange'
##
## The following object is masked from 'package:stringr':
##
##     boundary
```

```r
library(lattice)
library(foreign)
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following objects are masked from 'package:fma':
##
##     cement, housing, petrol
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(car)
require(stats)
require(stats4)
```

```
## Loading required package: stats4
```

```r
library(KernSmooth)
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```r
library(fastICA)
library(cluster)
library(leaps)
library(mgcv)
```

```
## Loading required package: nlme
##
## Attaching package: 'nlme'
##
## The following object is masked from 'package:forecast':
##
##     getResponse
##
## The following object is masked from 'package:dplyr':
##
##     collapse
##
## This is mgcv 1.8-41. For overview type 'help("mgcv-package")'.
```

```
library(rpart)
library(pan)
library(mgcv)
library(DAAG)
```

```
##
## Attaching package: 'DAAG'
##
## The following object is masked from 'package:MASS':
##
##     hills
##
## The following objects are masked from 'package:fma':
##
##     milk, ozone
##
## The following object is masked from 'package:survival':
##
##     lung
##
## The following object is masked from 'package:car':
##
##     vif
```

```
library(TTR)
library(tis)
```

```
##
## Attaching package: 'tis'
##
## The following object is masked from 'package:mgcv':
##
##     ti
##
## The following object is masked from 'package:forecast':
##
##     easter
##
## The following object is masked from 'package:TTR':
##
##     lags
```

```
##
## The following objects are masked from 'package:timeSeries':
##
##     description, interpNA
##
## The following objects are masked from 'package:timeDate':
##
##     dayOfWeek, dayOfYear, isHoliday
##
## The following object is masked from 'package:dplyr':
##
##     between
```

```
require(graphics)
library(forecast)
library(xtable)
library(dynlm)
library(vars)
```

```
## Loading required package: urca
##
## Attaching package: 'urca'
##
## The following objects are masked from 'package:fUnitRoots':
##
##     punitroot, qunitroot, unitrootTable
```

```
library(plm)
```

```
##
## Attaching package: 'plm'
##
## The following object is masked from 'package:tis':
##
##     between
##
## The following object is masked from 'package:stats4':
##
##     nobs
##
## The following object is masked from 'package:timeSeries':
##
##     lag
##
## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```
library(coefplot)
```

```
## Registered S3 methods overwritten by 'useful':
##   method       from
##   autoplot.acf forecast
##   fortify.ts   forecast
```

```r
library(gplots)
```

```
##
## Attaching package: 'gplots'
##
## The following object is masked from 'package:tis':
##
##     barplot2
##
## The following object is masked from 'package:stats':
##
##     lowess
```

```r
library(graphics)
library(ggeffects)


CountryGDP <- read_csv("~/Desktop/School/Econ 104/CountryGDPGrowthRate.csv",
                       col_types = cols(Country = col_factor(levels =
                                                              c("1","2", "3", "4", "5")),
                      Year = col_factor(levels = c("2002", "2003", "2004", "2005",
                                                  "2006", "2007","2008", "2009",
                                                  "2010", "2011", "2012","2013",
                                                  "2014", "2015", "2016", "2017",
                                                  "2018", "2019", "2020", "2021",
                                                  "2022"))))

Unemployment <- CountryGDP$`Unemployment Rate`
Inflation <- CountryGDP$`Inflation Rate`
GDPGR <- CountryGDP$`GDP Growth Rate`
Year <- CountryGDP$Year
Country <- CountryGDP$Country

#Country 1 is United States of America
#Country 2 is China
#Country 3 is Japan
#Country 4 is India
#Country 5 is Germany
```

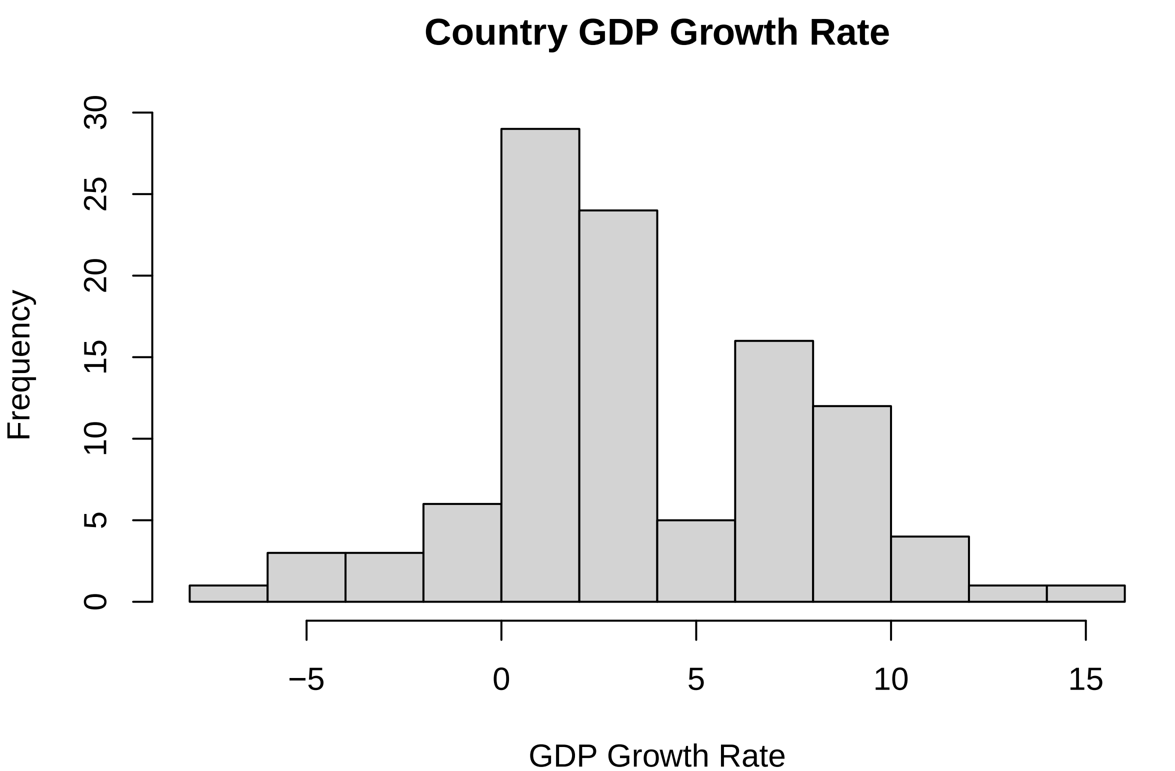
```r
#Question 1 Part 1
#For this part of the project, we are taking the Top 5 ranking countries with
#the highest GDPs within the last 20 years(2002-2022) and looking at their growth
#GDP rates. In this project, we plan to compute the fixed and random effects, or
#models to see what predictors have the greatest effect on the GDP growth rate.
#Using these techniques, we will be able to address if heterogeneity is present
#across these countries. This will also help us predict how these variables interact
#with each other.
```

```r
#Question 1 Part 2
#Summary Comments - Before starting, We specifically identified each of our country
#based on the country's highest GDP total. So Country 1: United States of America,
#Country 2: China 3: Japan, Country 4: India, and Country 5: Germany. In terms of
#GDP growth, there was a big gap between the lowest GDP Growth in an annual year,
#which was a negative one, @-6.6%, which came from India. On the other hand, the
#max growth rate % in GDP came from Country 2 (China). When looking at all 5
```

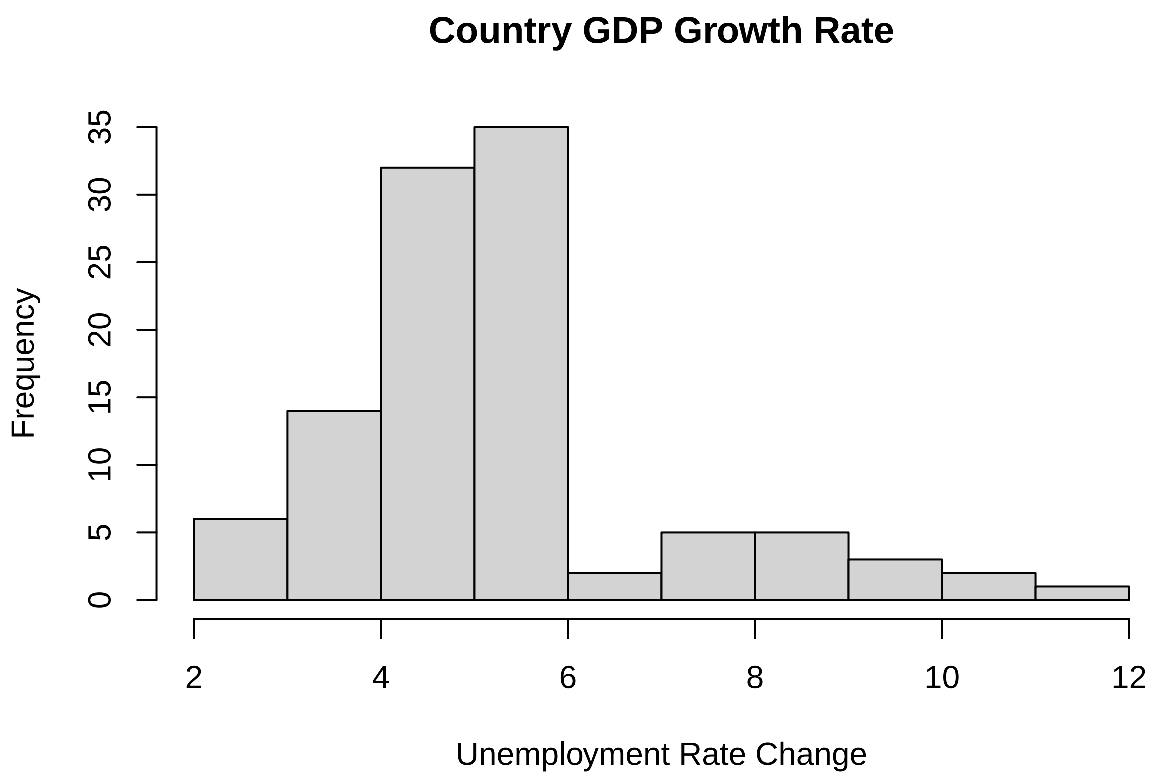
```r
#countries, the average growth rate % in GDP was around 3.73%, which was about a
#percent higher than the median. Looking at Unemployment Rate %s, the lowest was
#from Country 3(Japan) @ 2.4%, and highest came from Country 5(Germany) @ 11.17%.
#The inflation rate minimum came from Country 3(Japan) and highest inflation rate
#came from country 4(India).
summary(CountryGDP)
```

```
##  Country      Year      GDP Growth Rate Unemployment Rate Inflation Rate
##  1:21     2002   : 5   Min.   :-6.60   Min.   : 2.400    Min.   :-1.35
##  2:21     2003   : 5   1st Qu.: 1.50   1st Qu.: 4.360    1st Qu.: 0.98
##  3:21     2004   : 5   Median : 2.70   Median : 5.070    Median : 2.00
##  4:21     2005   : 5   Mean   : 3.73   Mean   : 5.312    Mean   : 2.61
##  5:21     2006   : 5   3rd Qu.: 6.90   3rd Qu.: 5.600    3rd Qu.: 3.77
##           2007   : 5   Max.   :14.20   Max.   :11.170    Max.   :11.99
##           (Other):75
```

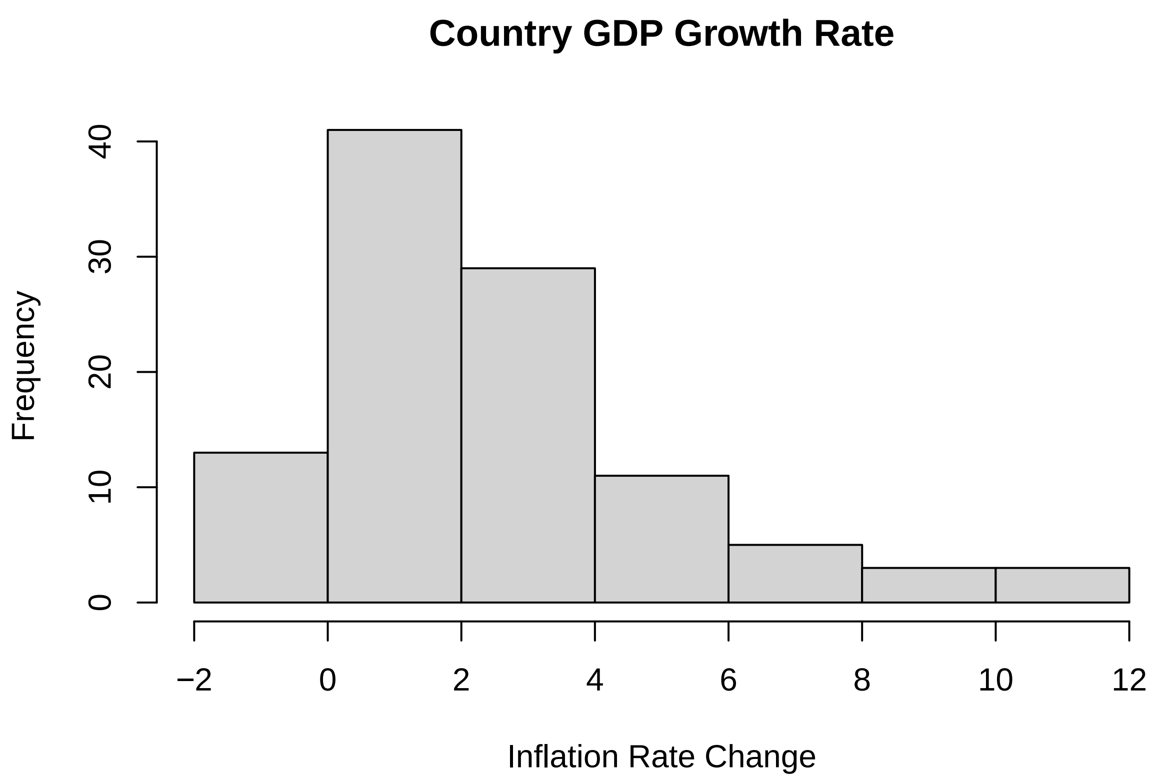
```r
#Histogram Comments
#When looking at the histogram of the GDP Growth Rates, the mean of 3.73 seems
#to be greater than the median value of 2.7. This is a distrbution that is skewed
#to the right,since there are a few values that bring the mean up, but do not
#really affect the median. As far as looking at the unemployment,the same
#can be said, as the histogram is still skewed a bit to the right, but the
#mean(5.312) is much closer to the median of 5.070%. Finally, when looking at
#the inflation rates for each of the 5 countries, it also shows a right skewing
#distribution, as the mean is still greater than the median.
hist(GDPGR, main= "Country GDP Growth Rate", xlab="GDP Growth Rate")
```



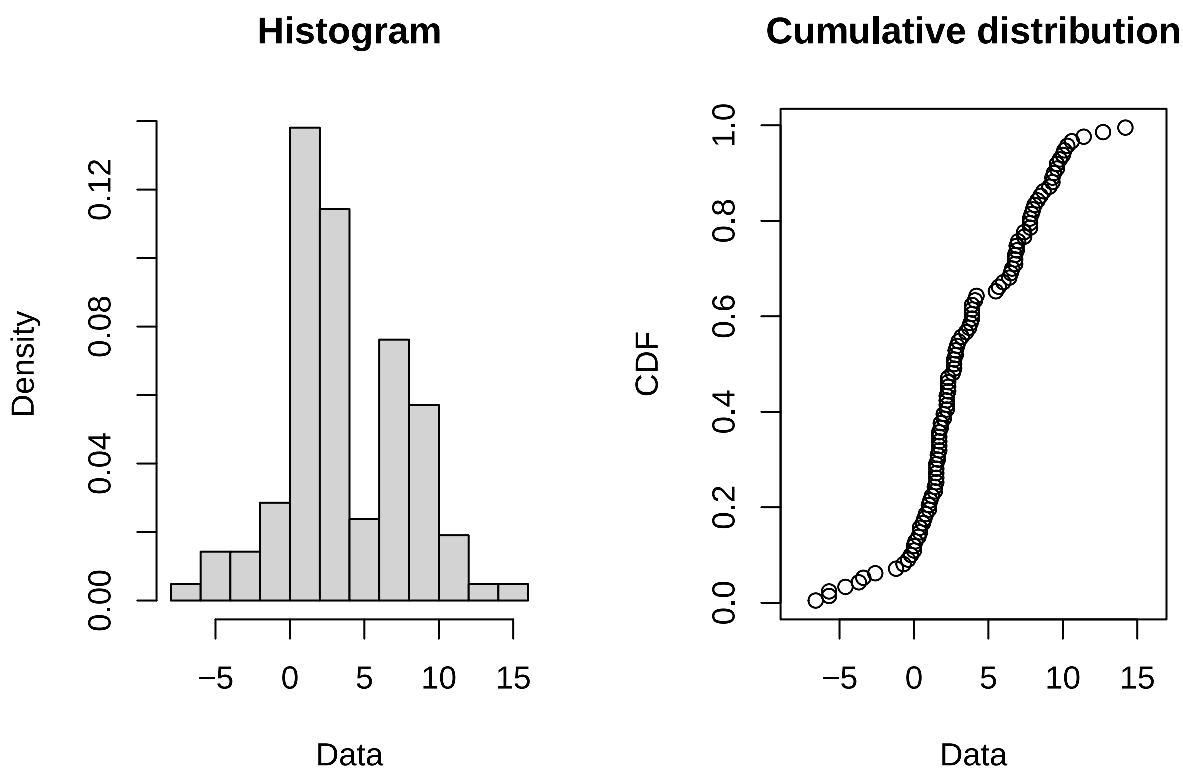

```r
hist(Unemployment, main= "Country GDP Growth Rate",xlab= "Unemployment Rate Change")
```



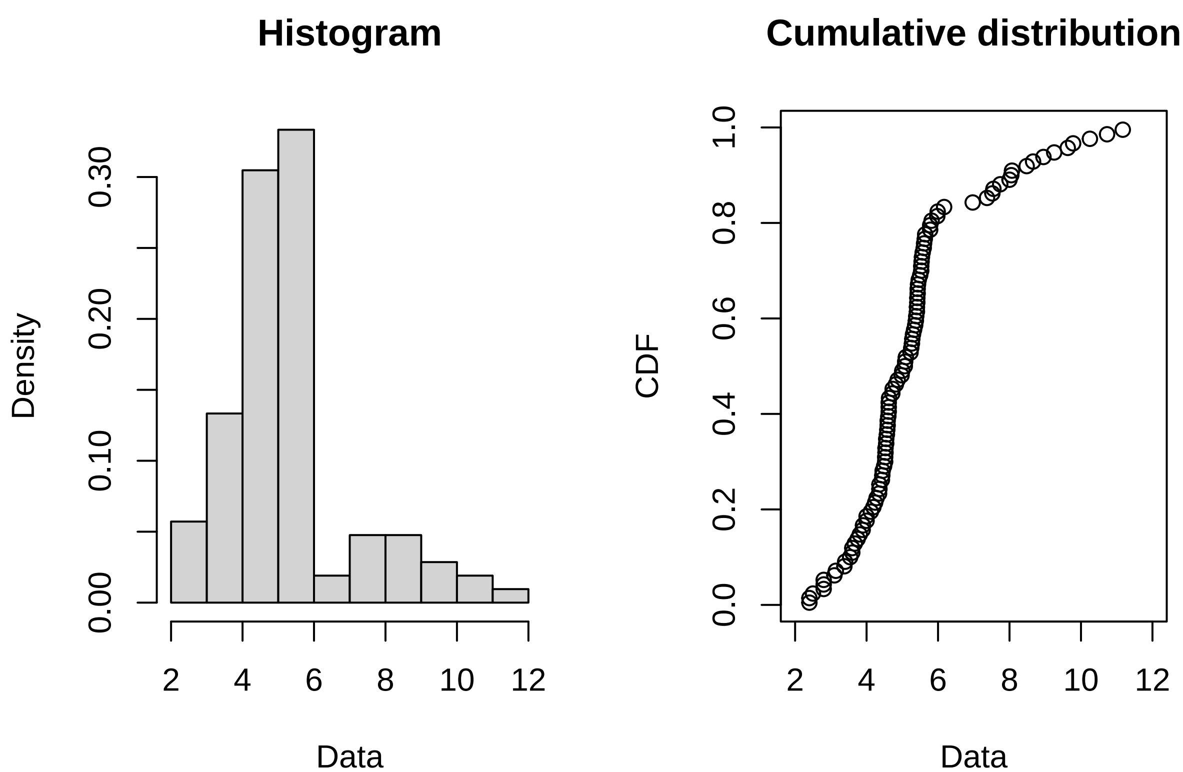

```
hist(Inflation, main= "Country GDP Growth Rate", xlab= "Inflation Rate Change")
```



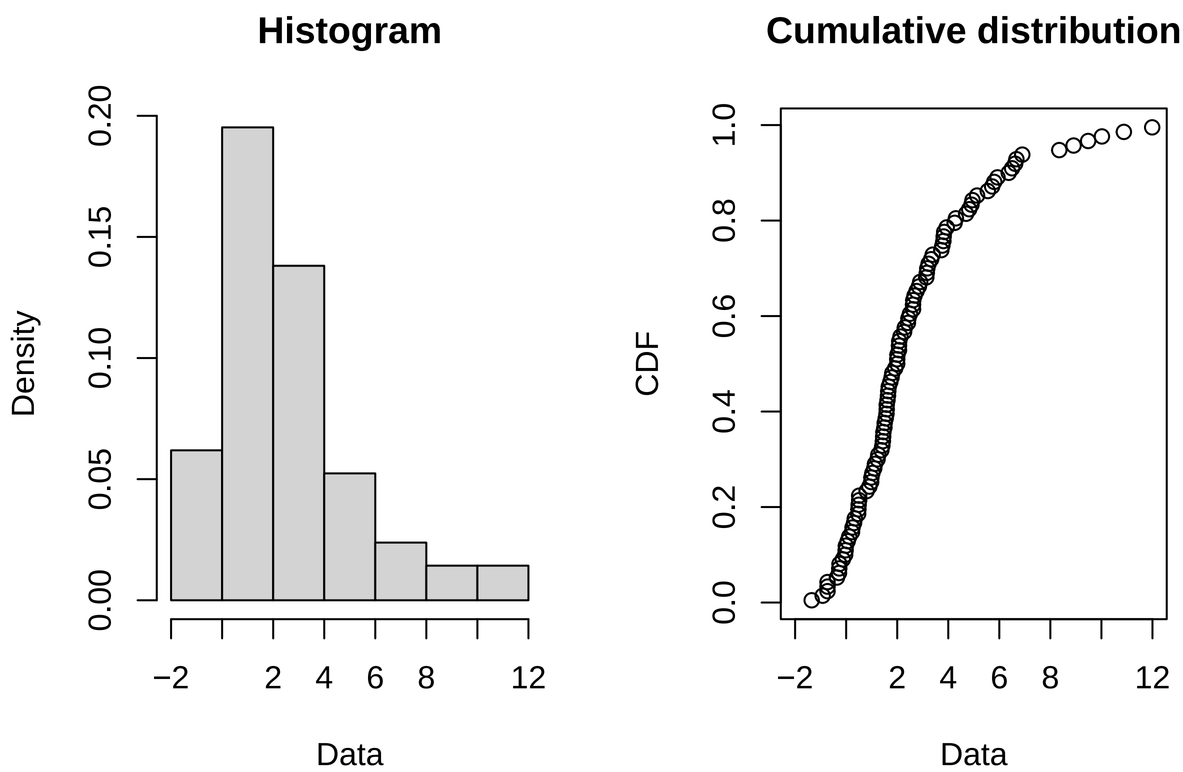

```
library(fitdistrplus)
#Fitted Distributions
plotdist(GDPGR, histo=TRUE)
```



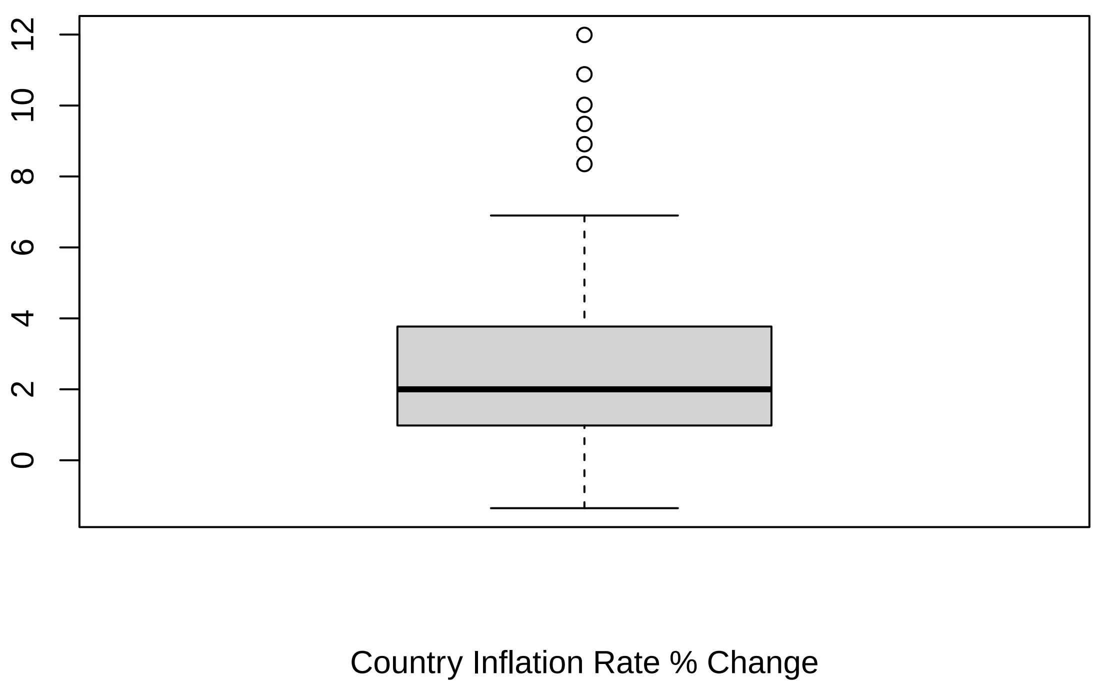

```
plotdist(Unemployment, histo=TRUE)
```




```
plotdist(Inflation, histo=TRUE)
```

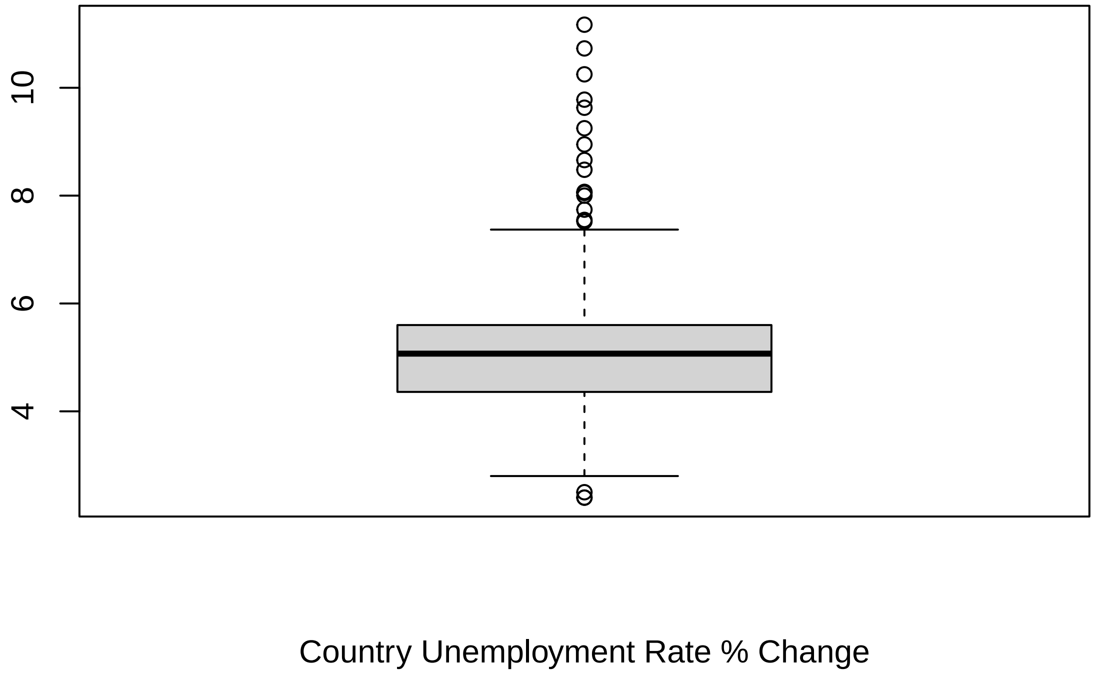



```
#Boxplots
boxplot(Inflation, xlab="Country Inflation Rate % Change")
```

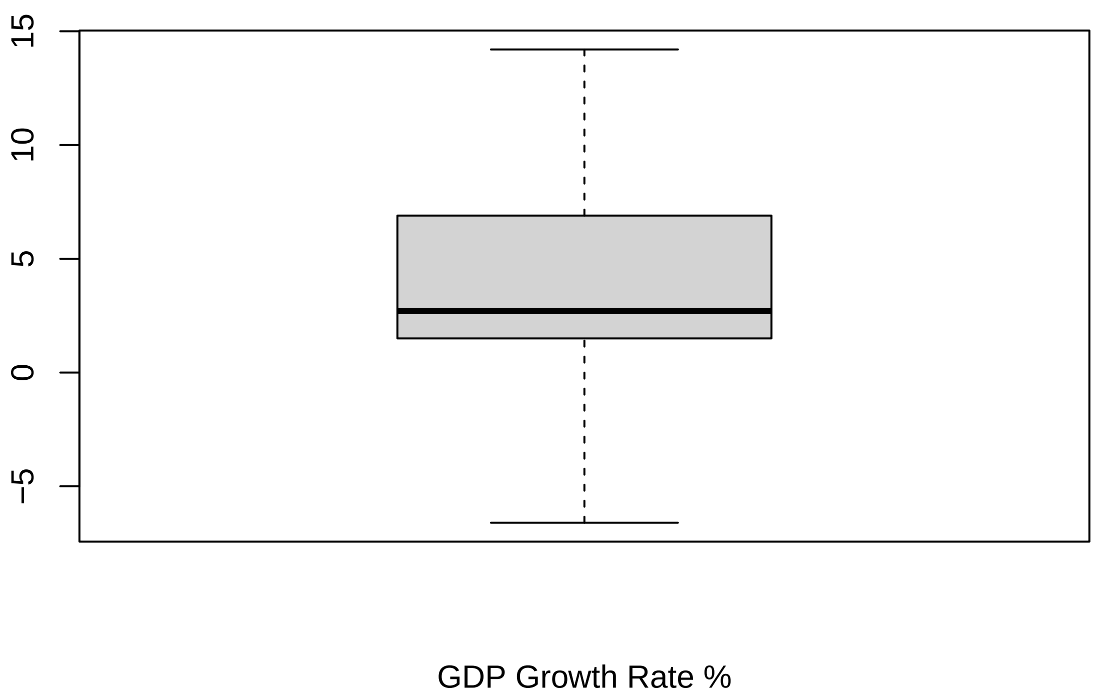



```
#When looking at the boxplot of the different inlation rates for the different
#countries, we see that there are values that are definitely outliers, with the
#max value of 11.99% coming from India, and many of these outliers coming from this country.
```

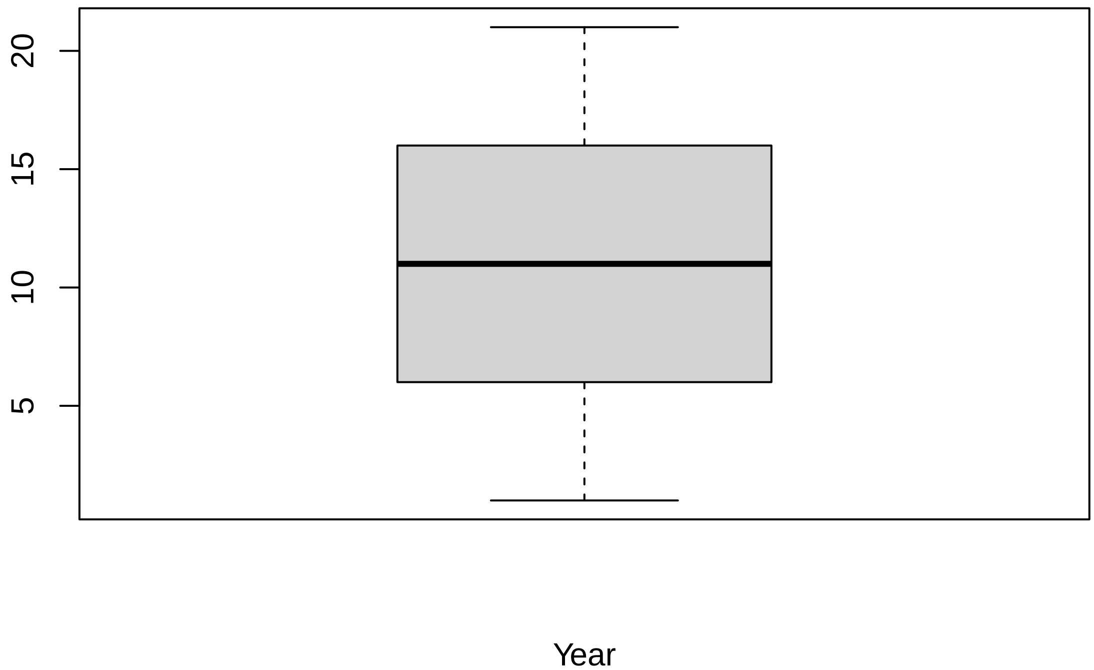
```
boxplot(Unemployment, xlab="Country Unemployment Rate % Change")
```



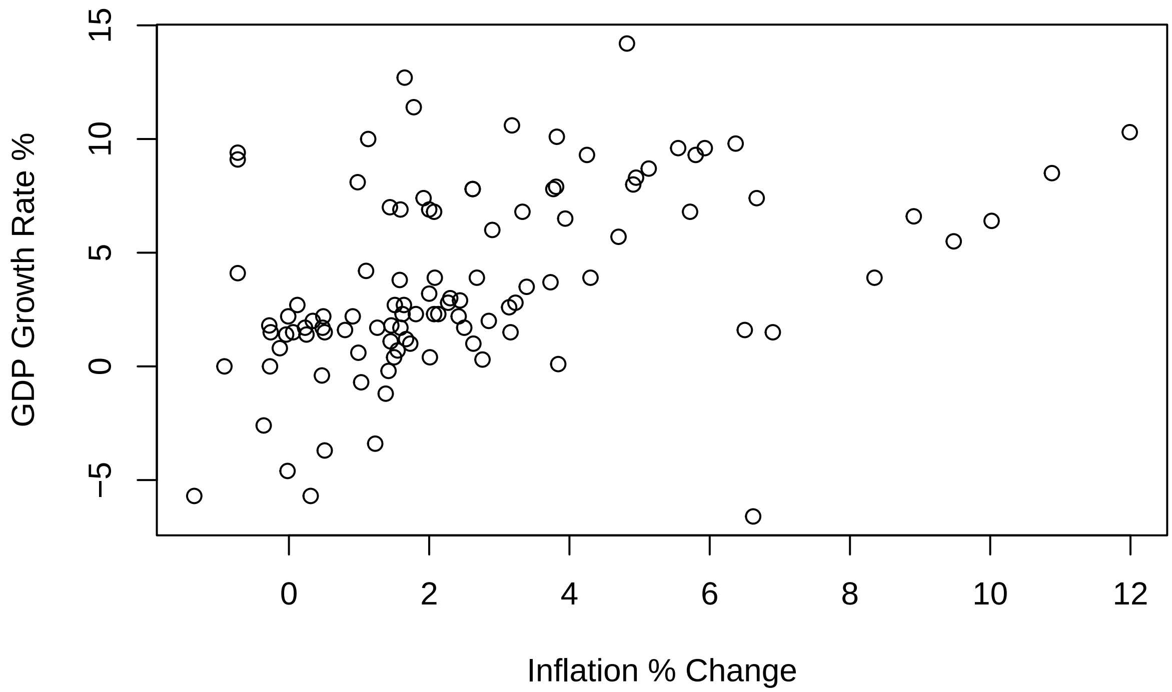

```
#When looking at the boxplot of different unemployment rates for the 5 countries,
#we see that there are many outliers that are present in the data. When looking at
#the data, we can see that there was a period from 2004-2006, where country 5(Germany)
#has these outliers compared to the mean and median values, with unemployment rates of 10+ %

boxplot(GDPGR, xlab= "GDP Growth Rate %")
```



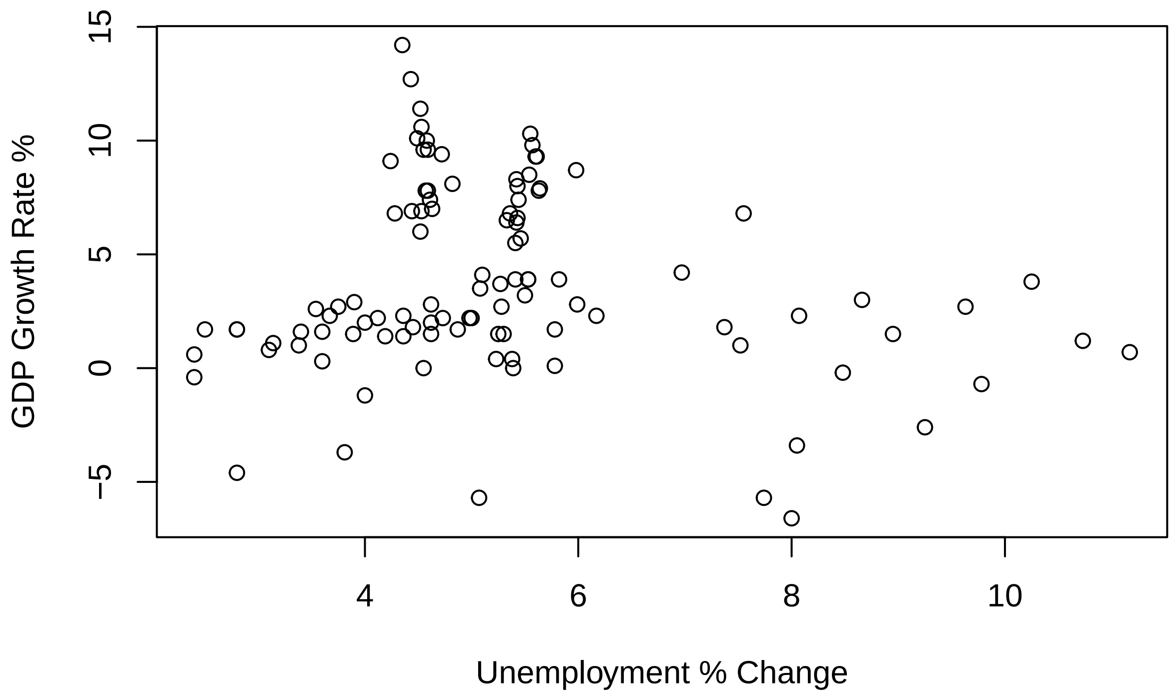

```
boxplot(Year, xlab= "Year")
```




```
#Scatterplots
plot(Inflation, GDPGR,xlab="Inflation % Change", ylab="GDP Growth Rate %")
```




```
#When looking at our scatterplot of the relationship between inflation and the GDP
#growth rate of each country, we found that there seems to be a positive/ increase
#in GDP growth as inflation rate % changes start to increase.


plot(Unemployment, GDPGR, xlab="Unemployment % Change", ylab= "GDP Growth Rate %")
```




```r
#Correlations
cor(Inflation, GDPGR)
```

```
## [1] 0.431203
```

```r
#There seems to be a low, positive correlation between Inflation and the Country's
#Annual GDP Growth Rate. With an increase in inflation %s, there seems to be slight
#increases in country GDP Growth Rate %.
cor(Unemployment, GDPGR)
```

```
## [1] -0.1420829
```

```r
# For correlation between unemployment rate % change and GDP Growth Rate, there
#is an indication of a negative correlation of -.14, with a rise in unemployment
#rates leading to a decrease in total GDP Growth Rate % change.
```

```r
#Question 1 Part 3

#Pooled
FE.Pool <- lm(GDPGR~Unemployment+Inflation)

#Full Effect
FE.Full <- lm(GDPGR~Unemployment+Inflation+Year+Country)

#Fixed Effect(Time)
FE.Time <- lm(GDPGR~Unemployment+Inflation+Year)

#Fixed Effect(Country)
FE.Country <- lm(GDPGR~Unemployment+Inflation+Country)

#Full vs Pooled
#At least country or year has a significant effect on the model
anova(FE.Full,FE.Pool)
```

```
## Analysis of Variance Table
##
## Model 1: GDPGR ~ Unemployment + Inflation + Year + Country
## Model 2: GDPGR ~ Unemployment + Inflation
##   Res.Df     RSS  Df Sum of Sq      F    Pr(>F)
## 1     78  219.93
## 2    102 1370.08 -24   -1150.2 16.996 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

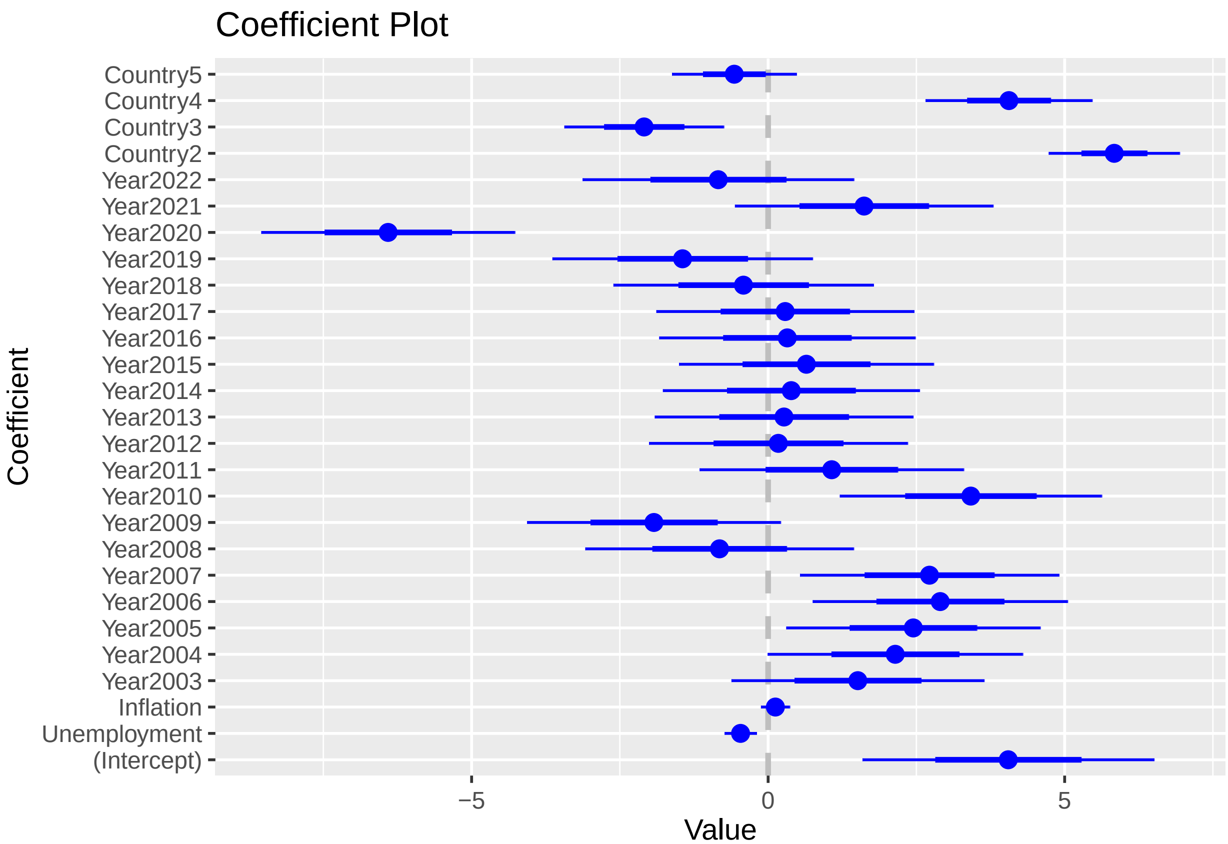
```r
#Controlled for time effects
#
anova(FE.Full,FE.Country)
```

```
## Analysis of Variance Table
##
## Model 1: GDPGR ~ Unemployment + Inflation + Year + Country
## Model 2: GDPGR ~ Unemployment + Inflation + Country
##   Res.Df    RSS  Df Sum of Sq      F    Pr(>F)
## 1     78 219.93
## 2     98 661.33 -20    -441.4 7.8273 1.116e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

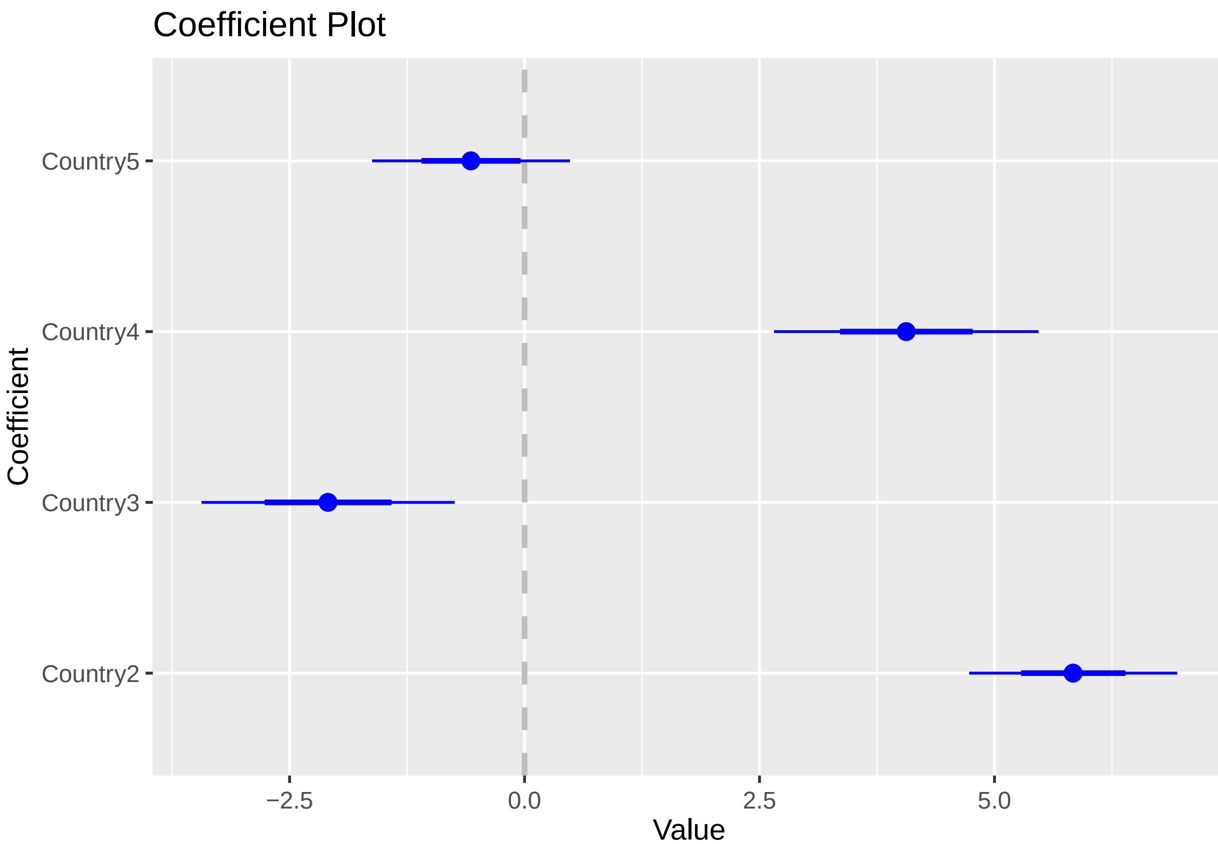
```r
#Controlled for country effects
#
anova(FE.Full, FE.Time)
```

```
## Analysis of Variance Table
##
## Model 1: GDPGR ~ Unemployment + Inflation + Year + Country
## Model 2: GDPGR ~ Unemployment + Inflation + Year
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     78 219.93
## 2     82 908.22 -4   -688.29 61.028 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

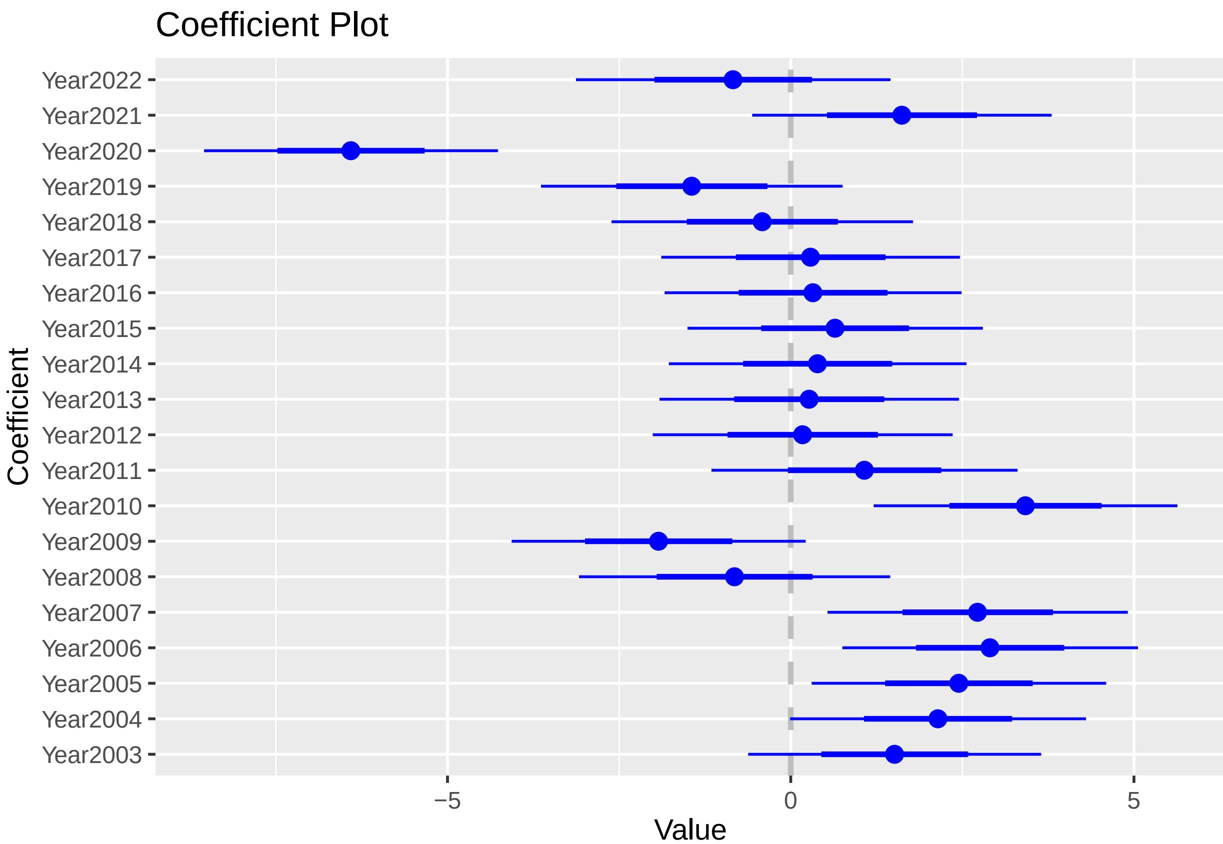
```r
#Coefficient Plot
coefplot(FE.Full)
```



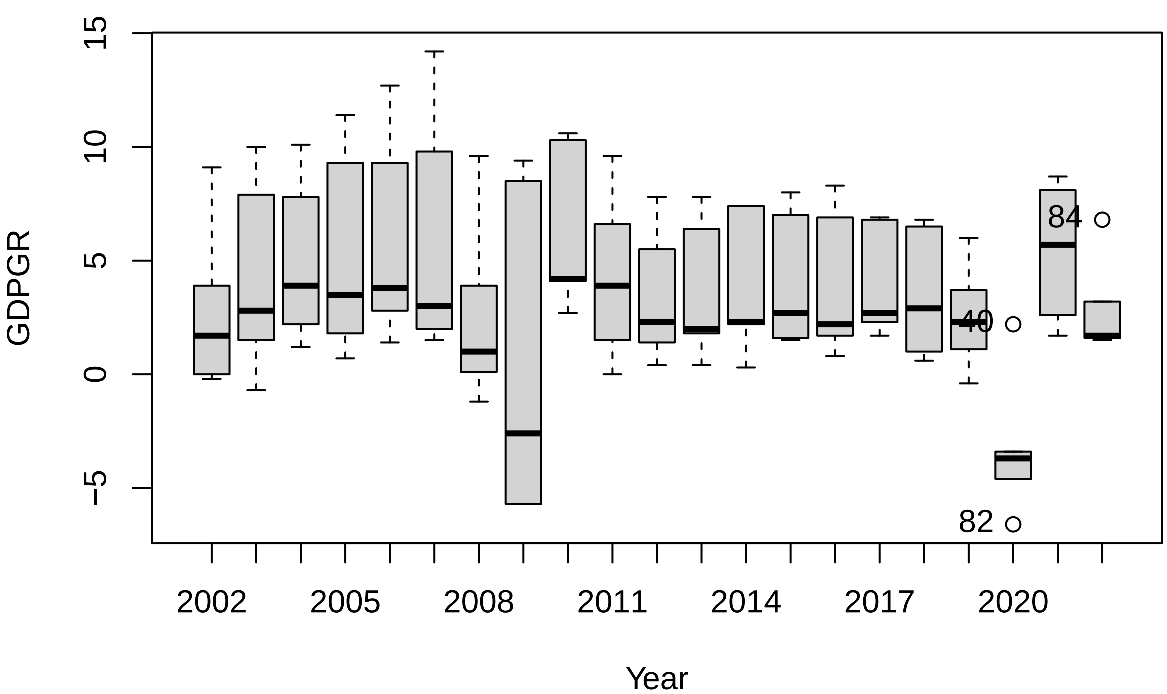

```
coefplot(FE.Full, predictors="Country")
```




```
coefpplot(FE.Full, predictors="Year")
```

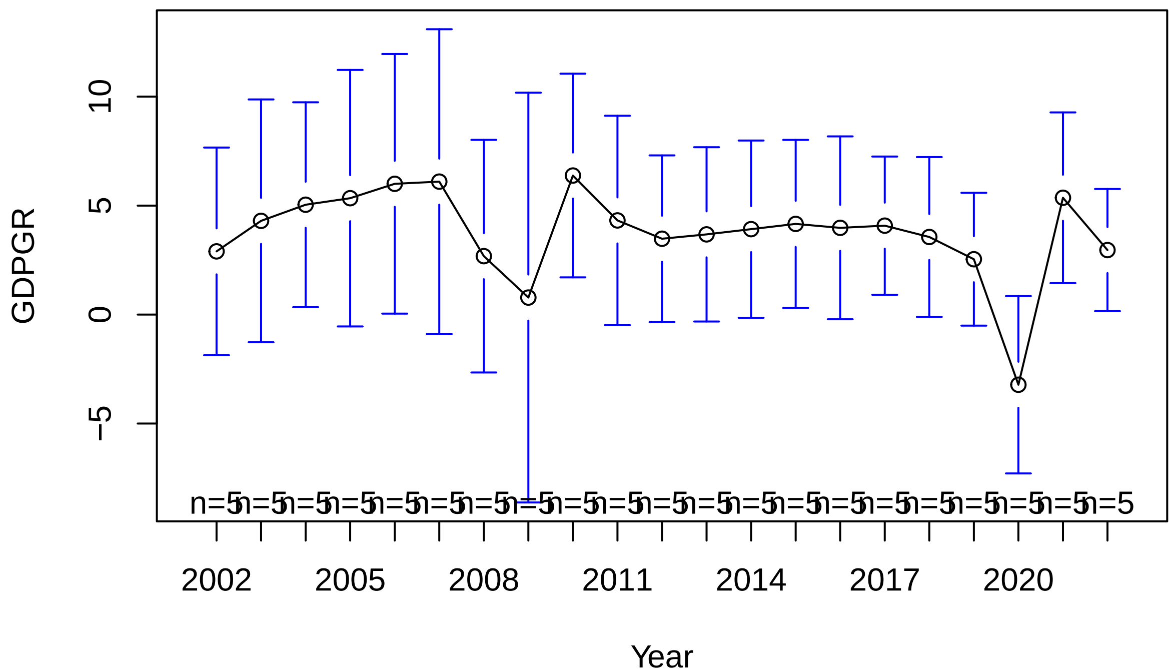



```
#Heterogeneity across time
scatterplot(GDPGR~Year|Country)
```

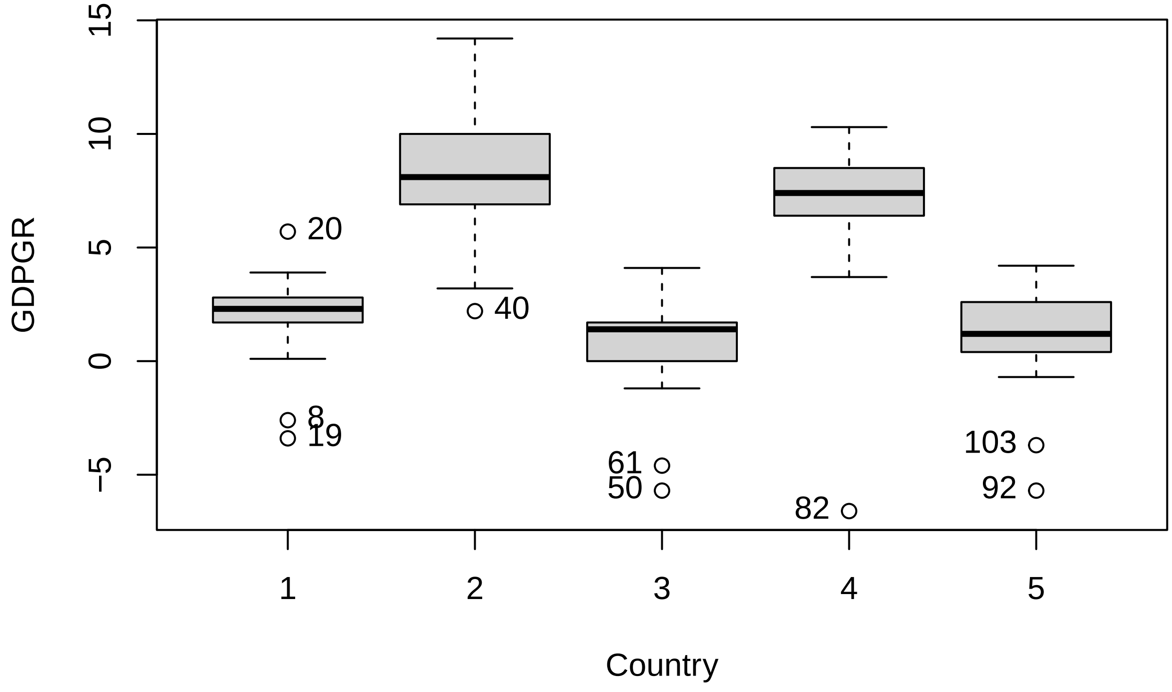



```
## [1] "82" "40" "84"
```

```
plotmeans(GDPGR~Year)
```

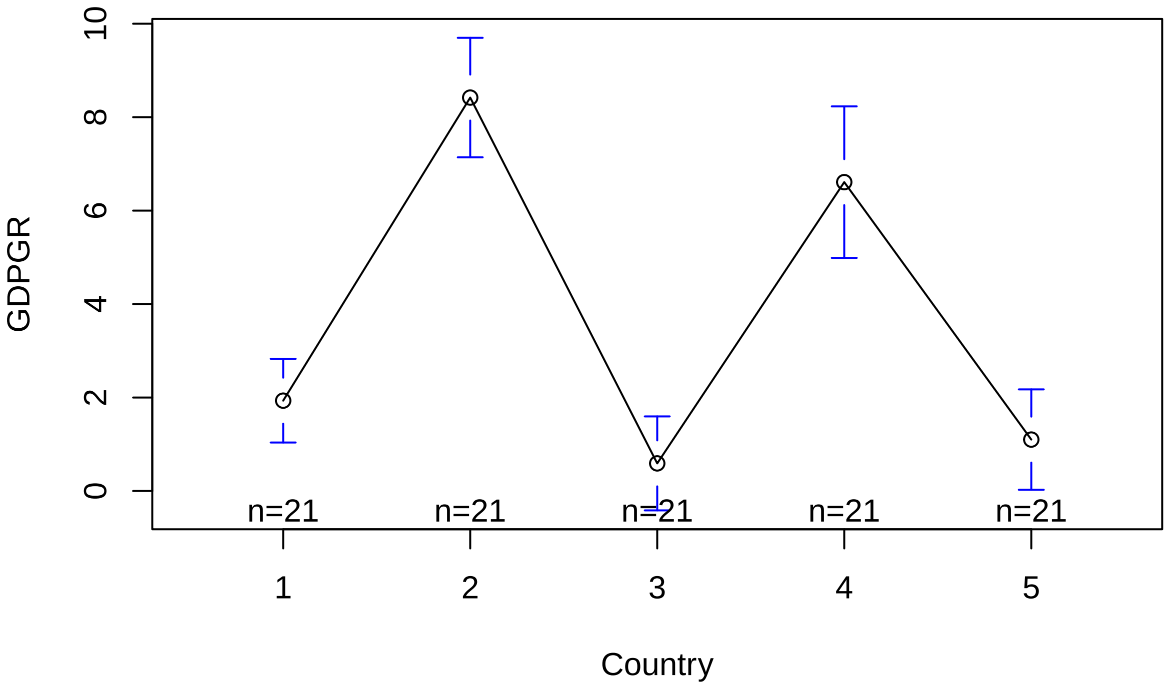



```
#Heterogeneity across country
scatterplot(GDPGR~Country|Year)
```

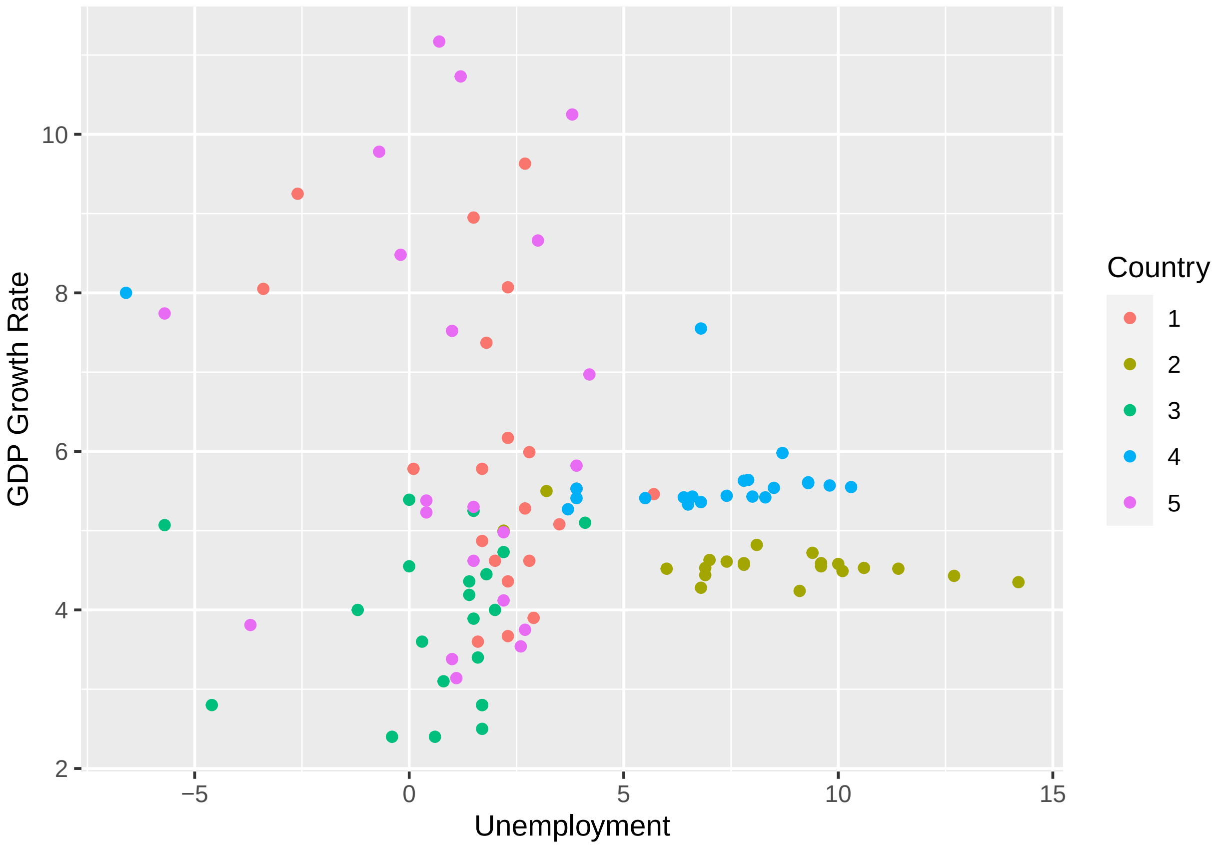



```
## [1] "8"   "19"  "20"  "40"  "50"  "61"  "82"  "92"  "103"
```

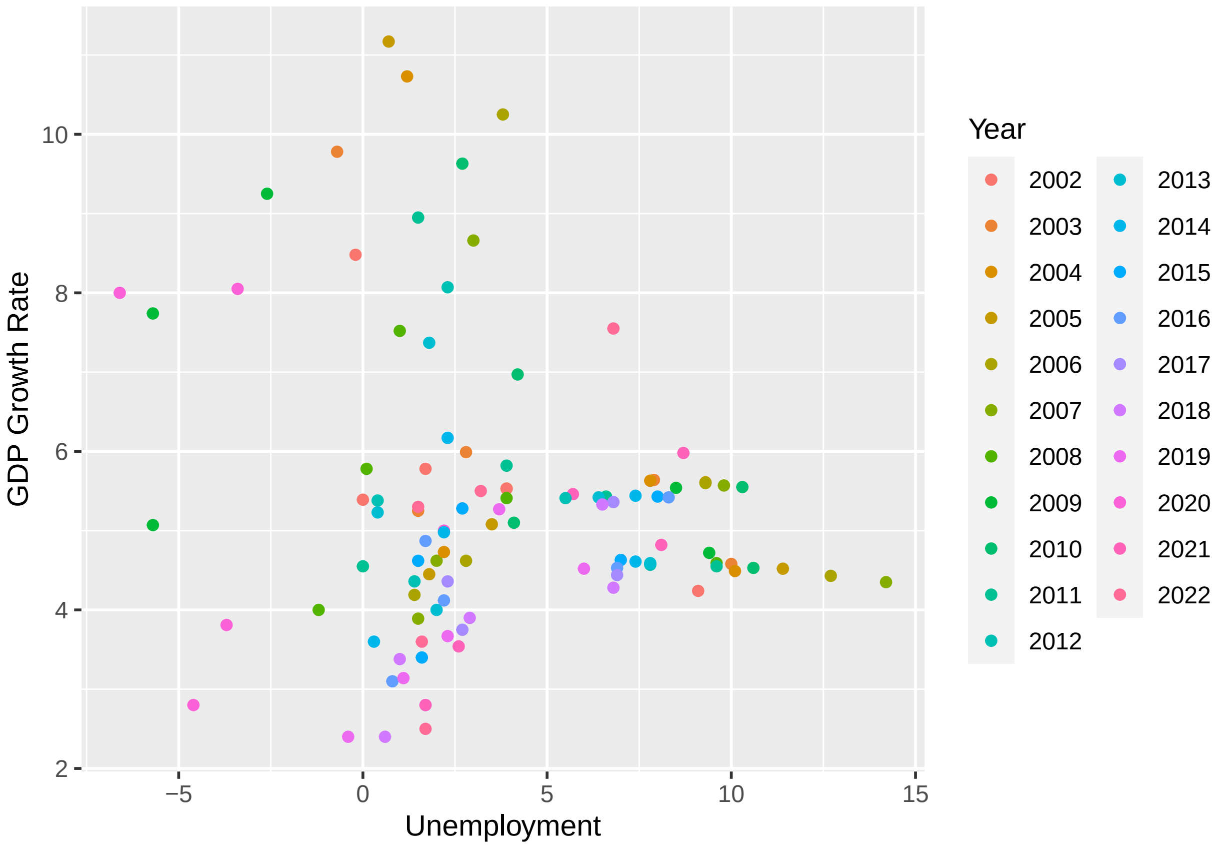
```
plotmeans(GDPGR~Country)
```




```
#GDP growth rate vs Unemployment by country
ggplot(CountryGDP, aes(x=GDPGR, y=Unemployment, colour=factor(Country))) + geom_point() + xlab("Unemploy
```




```
#GDP growth rate vs Unemployment by year
ggplot(CountryGDP, aes(x=GDPGR, y=Unemployment, colour=factor(Year))) + geom_point() + xlab("Unemploymen
```




```r
#FE model and Fixed effects by country
mreg.within <- plm(GDPGR~Inflation+Unemployment+Country+Year,data = CountryGDP,
                   model="within")
summary(mreg.within)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = GDPGR ~ Inflation + Unemployment + Country + Year,
##     data = CountryGDP, model = "within")
##
## Balanced Panel: n = 5, T = 21, N = 105
##
## Residuals:
##     Min.  1st Qu.   Median  3rd Qu.     Max.
## -5.39315 -0.88238  0.14238  0.78103  3.72348
##
## Coefficients:
##                Estimate Std. Error t-value  Pr(>|t|)
## Inflation       0.12224    0.12161  1.0051 0.3179372
## Unemployment -0.46469    0.13435 -3.4587 0.0008826 ***
## Year2003        1.51243    1.06527  1.4198 0.1596611
## Year2004        2.14378    1.07546  1.9934 0.0497195 *
## Year2005        2.44807    1.07078  2.2863 0.0249531 *
## Year2006        2.90186    1.07468  2.7002 0.0084951 **
## Year2007        2.72089    1.09170  2.4923 0.0148110 *
## Year2008       -0.81968    1.13144 -0.7245 0.4709491
```

```
## Year2009     -1.92602    1.06887 -1.8019 0.0754215 .
## Year2010      3.41769    1.10420  3.0952 0.0027314 **
## Year2011      1.07220    1.11332  0.9631 0.3384908
## Year2012      0.17230    1.09048  0.1580 0.8748632
## Year2013      0.26752    1.08932  0.2456 0.8066477
## Year2014      0.38967    1.08204  0.3601 0.7197287
## Year2015      0.64497    1.07369  0.6007 0.5497803
## Year2016      0.32298    1.07942  0.2992 0.7655737
## Year2017      0.28788    1.08616  0.2650 0.7916761
## Year2018     -0.41617    1.09620 -0.3796 0.7052403
## Year2019     -1.44332    1.09695 -1.3158 0.1921102
## Year2020     -6.40825    1.06889 -5.9952 5.954e-08 ***
## Year2021      1.61787    1.08875  1.4860 0.1413124
## Year2022     -0.84098    1.14338 -0.7355 0.4642272
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    697.86
## Residual Sum of Squares: 219.93
## R-Squared:      0.68485
## Adj. R-Squared: 0.5798
## F-statistic: 7.70463 on 22 and 78 DF, p-value: 5.822e-12
```

```r
fixef(mreg.within)
```

```
##      1      2      3      4      5
## 4.0492 9.8851 1.9575 8.1097 3.4778
```

```r
#Random Effects
GDPGrowth <- pdata.frame(CountryGDP, c("Country","Year"))
fm.time <- plm(GDPGR~Unemployment+Inflation,data=GDPGrowth, model= "within",
               effect = "time")
fm.rtime <- plm(GDPGR~Unemployment+Inflation,data=GDPGrowth, model= "random")


phtest(fm.time, fm.rtime)
```

```
##
##  Hausman Test
##
## data:  GDPGR ~ Unemployment + Inflation
## chisq = 14.757, df = 2, p-value = 0.0006246
## alternative hypothesis: one model is inconsistent
```

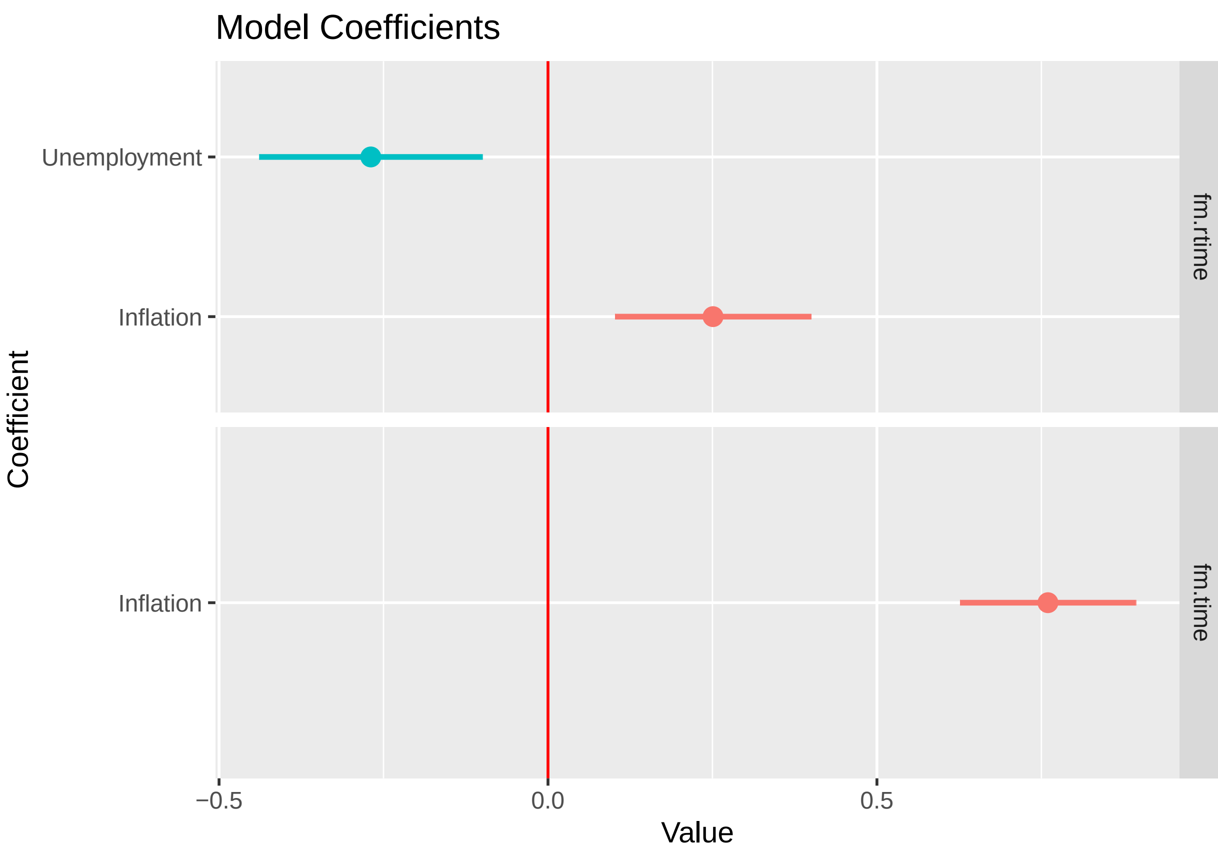
```r
ce <- function(model.obj) {
summ.model <- summary(get(model.obj))$coefficients
extract <- summ.model[2:nrow(summ.model),drop=FALSE, 1:2]
return(data.frame(extract, vars = row.names(extract), model = model.obj))
}
coefs <- do.call(rbind, sapply(paste0(list(
"fm.time", "fm.rtime"
)), ce, simplify = FALSE))
names(coefs)[2] <- "se"
gg_coef <- ggplot(coefs, aes(vars, Estimate)) +
geom_hline(yintercept = 0, lty = 1, lwd = 0.5, colour = "red") +
geom_errorbar(aes(ymin = Estimate - se, ymax = Estimate + se, colour = vars),
```

```r
lwd = 1, width = 0
) +
geom_point(size = 3, aes(colour = vars)) +
facet_grid(model ~ ., scales="free") +
coord_flip() +
guides(colour = FALSE) +
labs(x = "Coefficient", y = "Value") +
ggtitle("Model Coefficients")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```

```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
## of ggplot2 3.3.4.
```

```r
gg_coef
```




```r
#After testing the fixed effects, pooled, and random effects model, we came to the
#conclusion that the fixed effects model with time was the preferred model. When we
#tested the full model versus pooled model there was significance when using the anova
#test, which meant that either country or time has a significant effect on the model.
#When we tested the fixed effects models for time and country,  we found that that time
#had a more significant effect on the model. We then tested the fixed effects model to
#the random effects model and found that the fixed effects model was preferred over
#the random effects model using the Hausmann test.
```

```r
#Qualitative Dependent Variable Models
heart <- read_csv("~/Desktop/School/Econ 104/heart_data.csv")
```

```
## Rows: 70000 Columns: 16
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (16): index, id, age, age years, gender, gender dummy, height, weight, a...
##
## 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.
```

```
cardio <- heart$cardio
gender <- heart$`gender dummy`
age <- heart$`age years`
height <- heart$height
weight <- heart$weight
systolic <- heart$ap_hi
diastolic <- heart$ap_lo
cholesterol <- heart$cholesterol
glucose <- heart$gluc
smoke <- heart$smoke
alcohol <- heart$alco
active <- heart$active
```

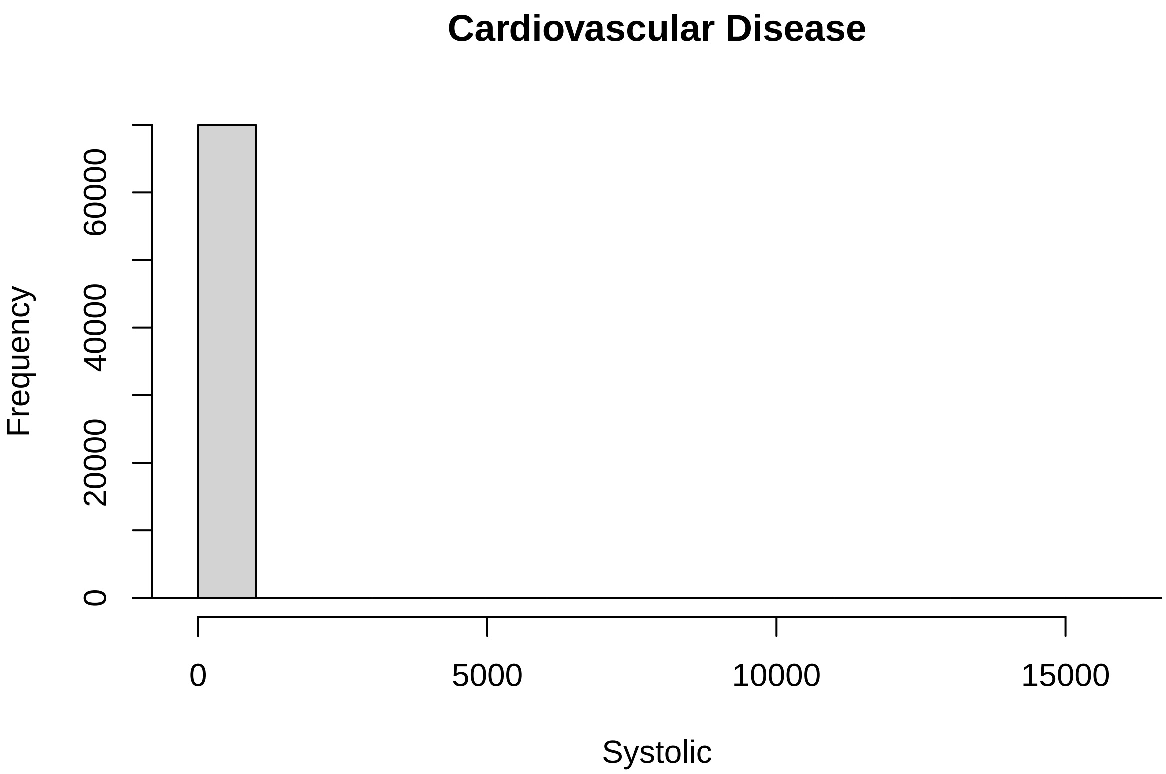
```
#Question 2 Part 1
#The question that our group was trying to answer was that we were trying to predict
#whether a patient would have some sort of cardiovascular disease, by analyzing the
#effect of lifestyle and environmental factors, such as their gender, weight, height,
#if they were active, if they smoked or consumed alcohol, etc... We gathered our data
#from Kaggle, and was updated two months ago. We will try and detect any patterns
#and predict different outcomes by using our best model and changing values within
#our model to see how it will affect the probability that the patient will encounter
#a cardiovascular disease or not.
```

```
summary(heart)
```

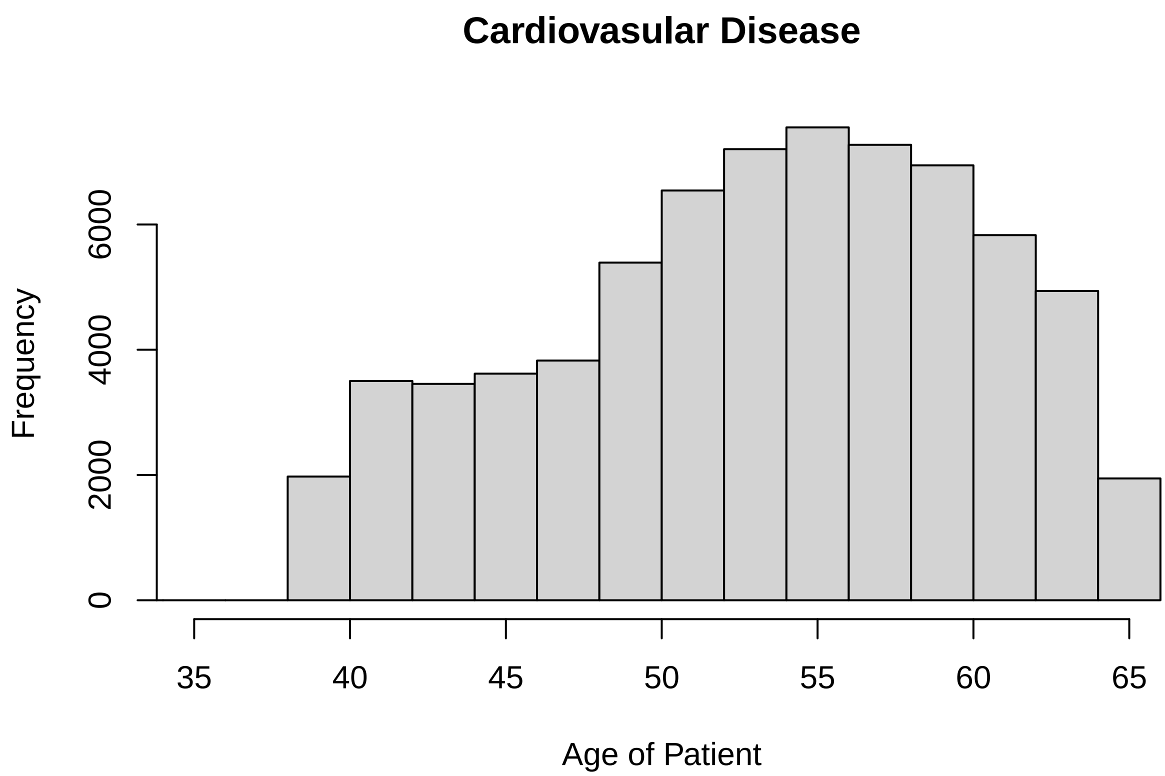
```
##      index             id              age          age years         gender
##  Min.   :    0   Min.   :    0   Min.   :10798   Min.   :29.60   Min.   :1.00
##  1st Qu.:17500   1st Qu.:25007   1st Qu.:17664   1st Qu.:48.40   1st Qu.:1.00
##  Median :35000   Median :50002   Median :19703   Median :54.00   Median :1.00
##  Mean   :35000   Mean   :49972   Mean   :19469   Mean   :53.34   Mean   :1.35
##  3rd Qu.:52499   3rd Qu.:74889   3rd Qu.:21327   3rd Qu.:58.40   3rd Qu.:2.00
##  Max.   :69999   Max.   :99999   Max.   :23713   Max.   :65.00   Max.   :2.00
##   gender dummy        height          weight             ap_hi
##  Min.   :0.0000   Min.   : 55.0   Min.   : 10.00   Min.   : -150.0
##  1st Qu.:0.0000   1st Qu.:159.0   1st Qu.: 65.00   1st Qu.:  120.0
##  Median :1.0000   Median :165.0   Median : 72.00   Median :  120.0
##  Mean   :0.6504   Mean   :164.4   Mean   : 74.21   Mean   :  128.8
##  3rd Qu.:1.0000   3rd Qu.:170.0   3rd Qu.: 82.00   3rd Qu.:  140.0
##  Max.   :1.0000   Max.   :250.0   Max.   :200.00   Max.   :16020.0
##      ap_lo            cholesterol         gluc            smoke
##  Min.   :  -70.00   Min.   :1.000   Min.   :1.000   Min.   :0.00000
##  1st Qu.:   80.00   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:0.00000
##  Median :   80.00   Median :1.000   Median :1.000   Median :0.00000
##  Mean   :   96.63   Mean   :1.367   Mean   :1.226   Mean   :0.08813
##  3rd Qu.:   90.00   3rd Qu.:2.000   3rd Qu.:1.000   3rd Qu.:0.00000
##  Max.   :11000.00   Max.   :3.000   Max.   :3.000   Max.   :1.00000
##       alco             active           cardio
```

```
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.00000   1st Qu.:1.0000   1st Qu.:0.0000
##  Median :0.00000   Median :1.0000   Median :0.0000
##  Mean   :0.05377   Mean   :0.8037   Mean   :0.4997
##  3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.00000   Max.   :1.0000   Max.   :1.0000
```

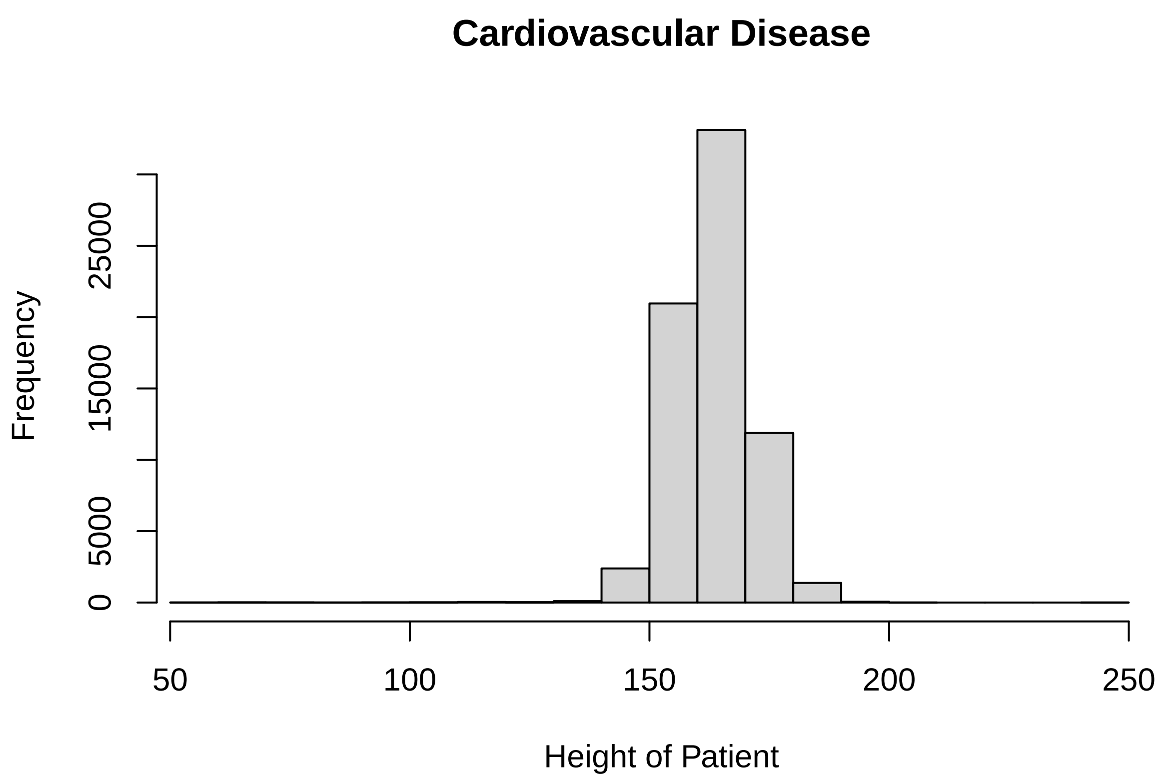
```
#When looking at the data, there were a couple of major outliers, specifically when
#looking at the systolic blood pressure reading,diastolic blood pressure reading. With
#a mean of 128.8 for systolic reading, there was one value that was at 16020, but we
#weren't sure if this was an actual reading since it was so high. This was also the
#case for the diastolic reading(11000)

#Histograms
hist(systolic,main="Cardiovascular Disease", xlab= "Systolic", xlim=c(-150,16020))
```



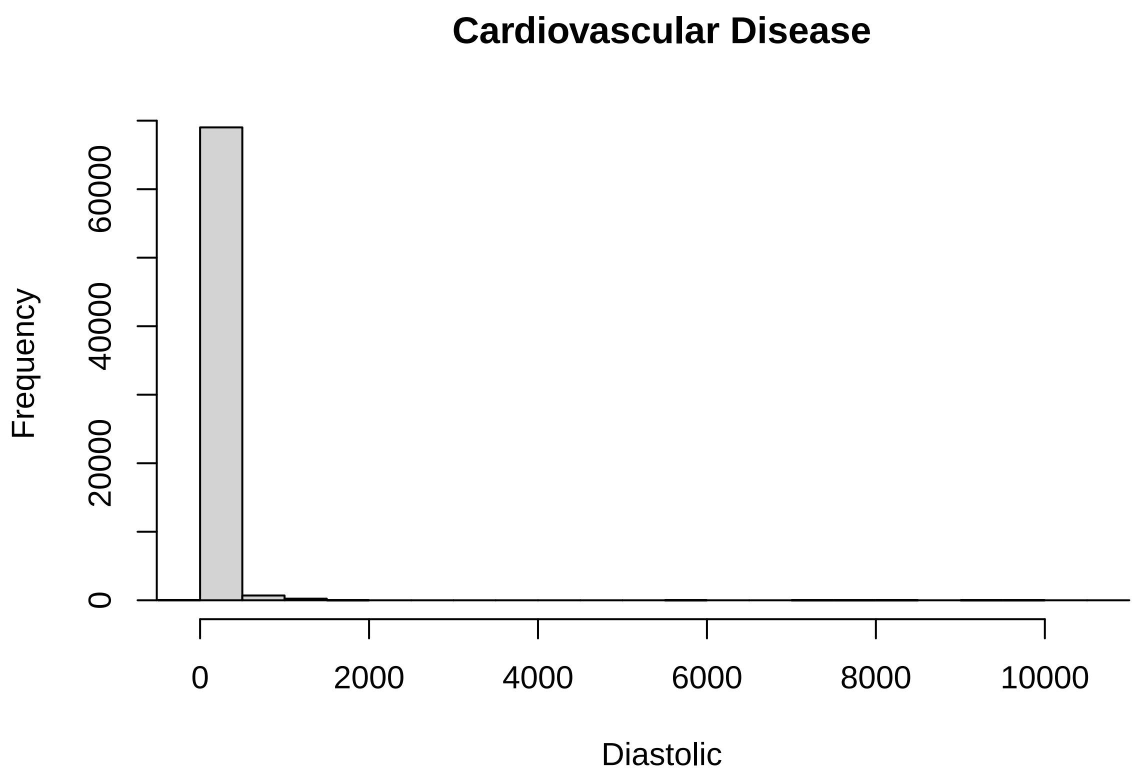

```
hist(age, main="Cardiovasular Disease", xlab= "Age of Patient", xlim=c(35,65))
```



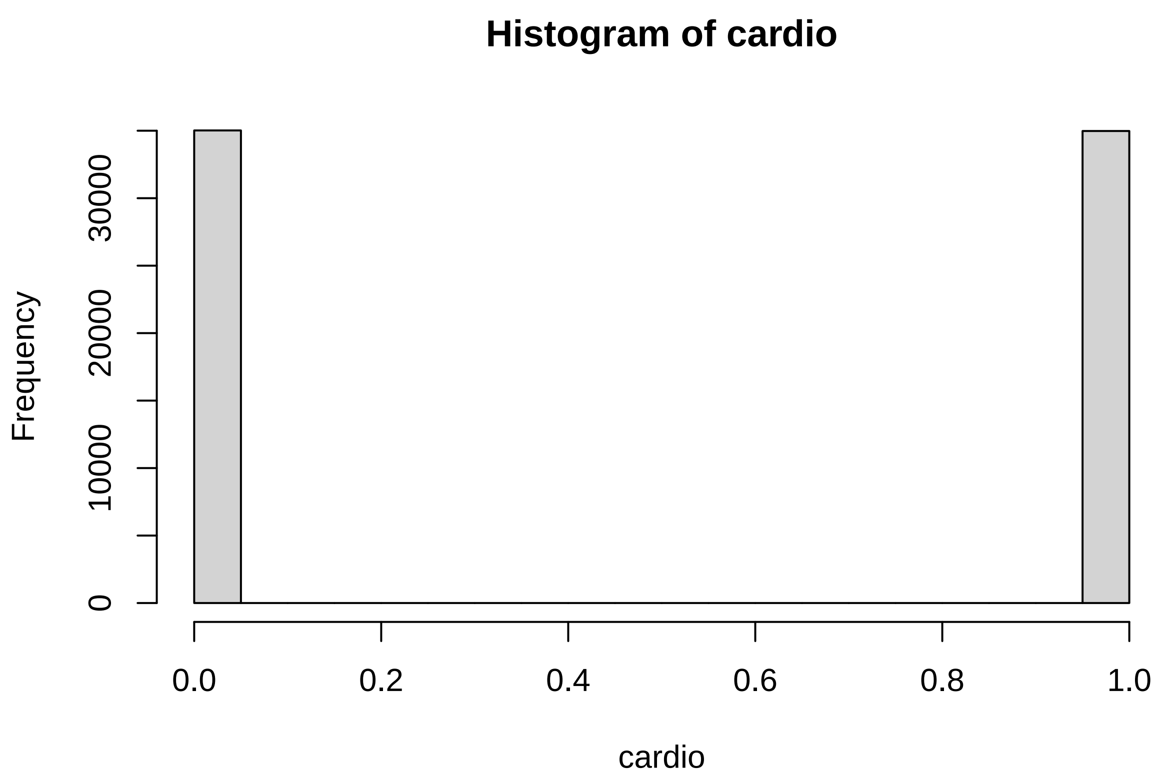

```r
hist(height, main="Cardiovascular Disease", xlab= "Height of Patient", xlim=c(55,250))
```




```r
hist(diastolic, main="Cardiovascular Disease", xlab="Diastolic",xlim=c(-70, 11000))
```

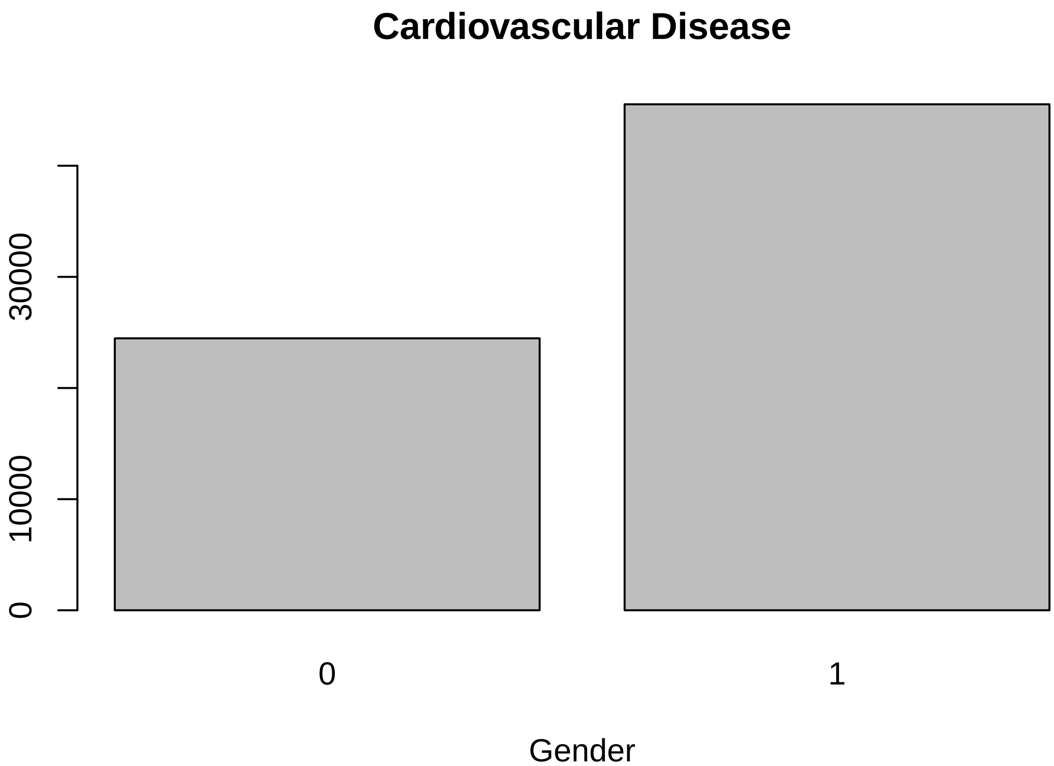



```
hist(cardio)
```

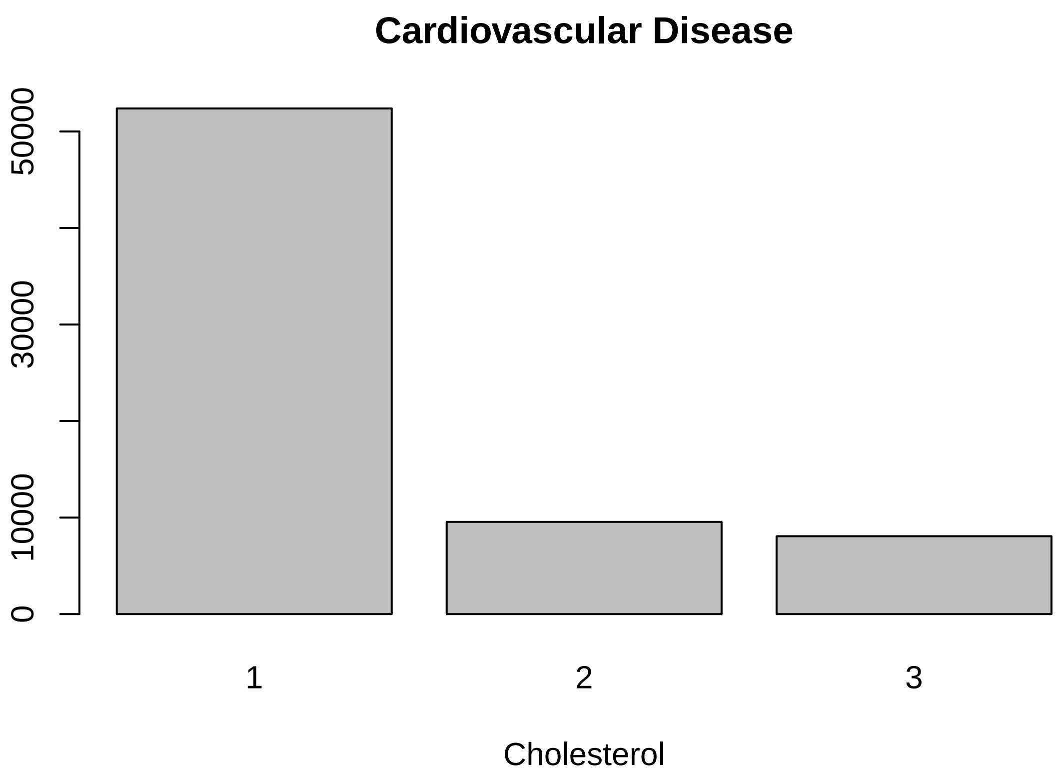



```
#When looking at the histograms of age in years, the mean of 53.34 is a bit lower
#than the median of 54, and is a left skewing distribution. This can also be seen
#with weight of the patient. Also the weight of the patient's histogram is right skewing .
```

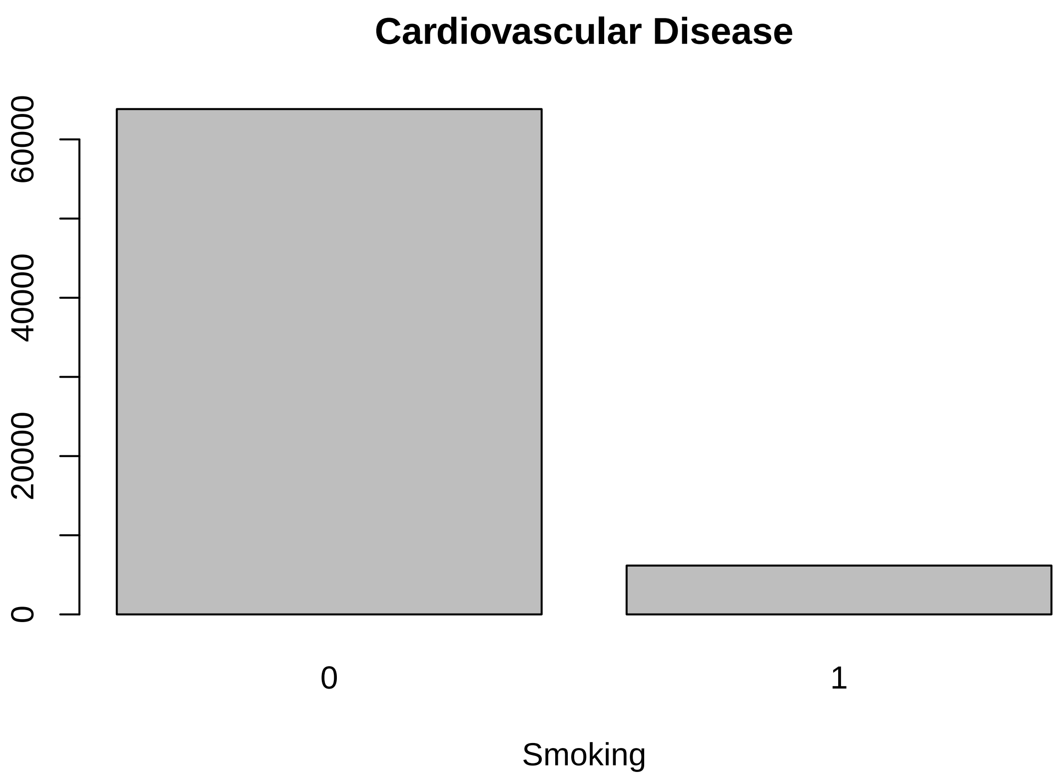
```r
#Bar Charts
barchart1 <- table(gender)
barplot(barchart1, main= "Cardiovascular Disease",
        xlab="Gender")
```



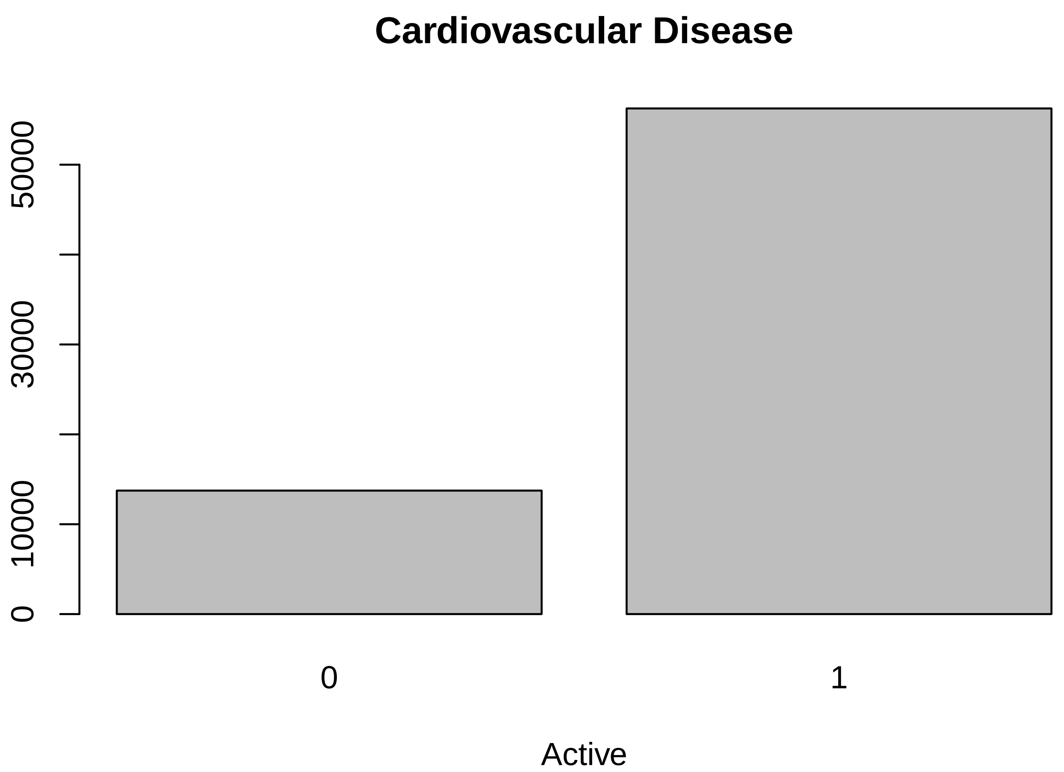

```r
barchart2 <- table(cholesterol)
barplot(barchart2,main="Cardiovascular Disease", xlab="Cholesterol")
```




```
barchart3 <- table(smoke)
barplot(barchart3, main="Cardiovascular Disease", xlab="Smoking")
```



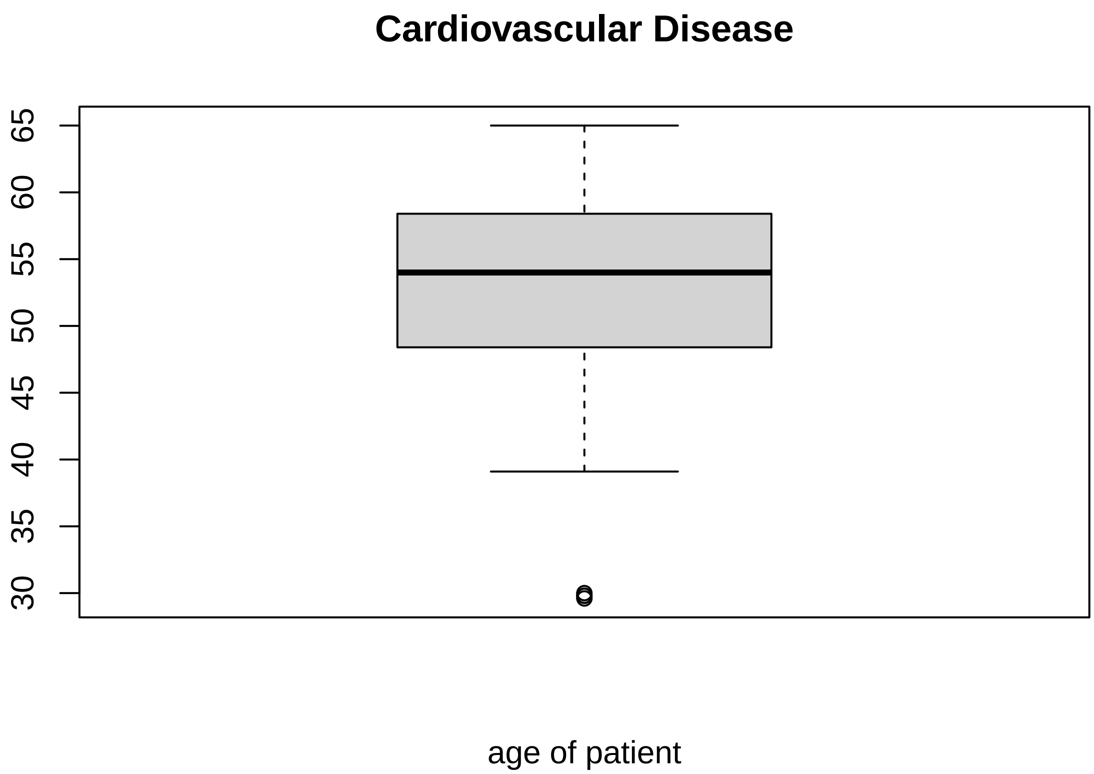

```
barchart4<- table(active)
barplot(barchart4, main="Cardiovascular Disease", xlab="Active")
```



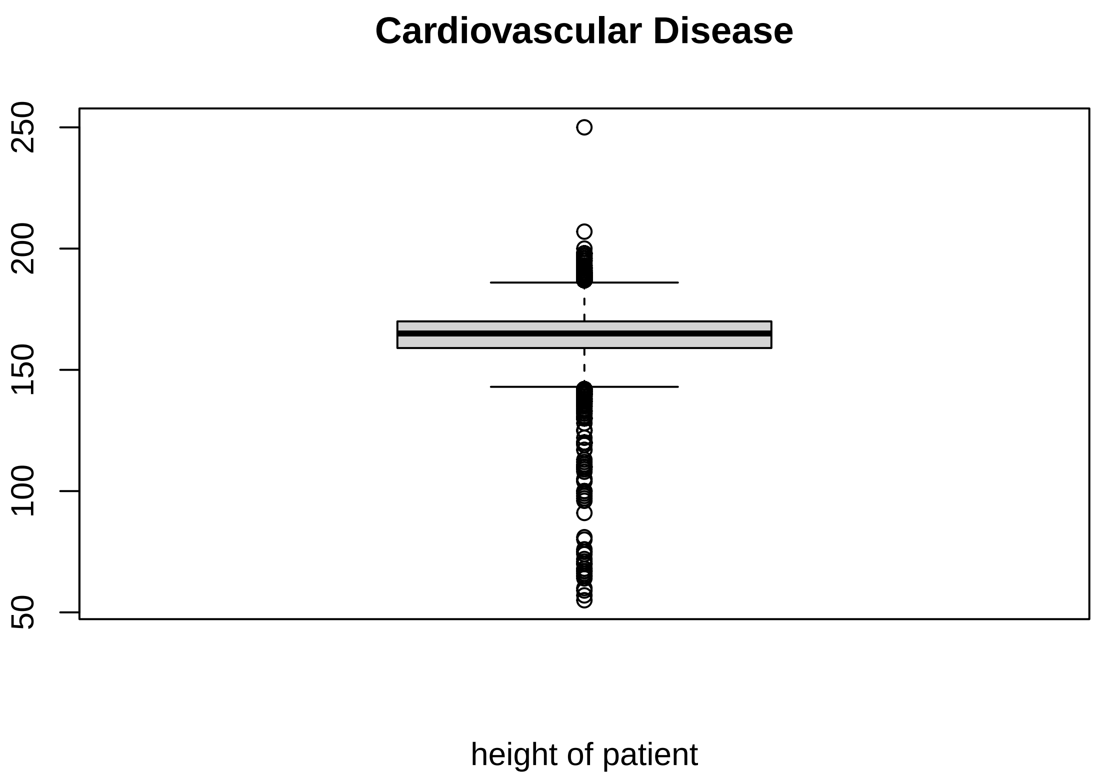

```
barchart5<-table(alcohol)
barplot(barchart5, main="Cardiovascular Disease", xlab="Alcohol")
```


```
#When looking at the bar charts for the binary variables in the data, we see that
#a majority of participants were males(1). Most of the particpants did not smoke(0),
#and did not consume alcohol(0), and a majority were active(1).
```

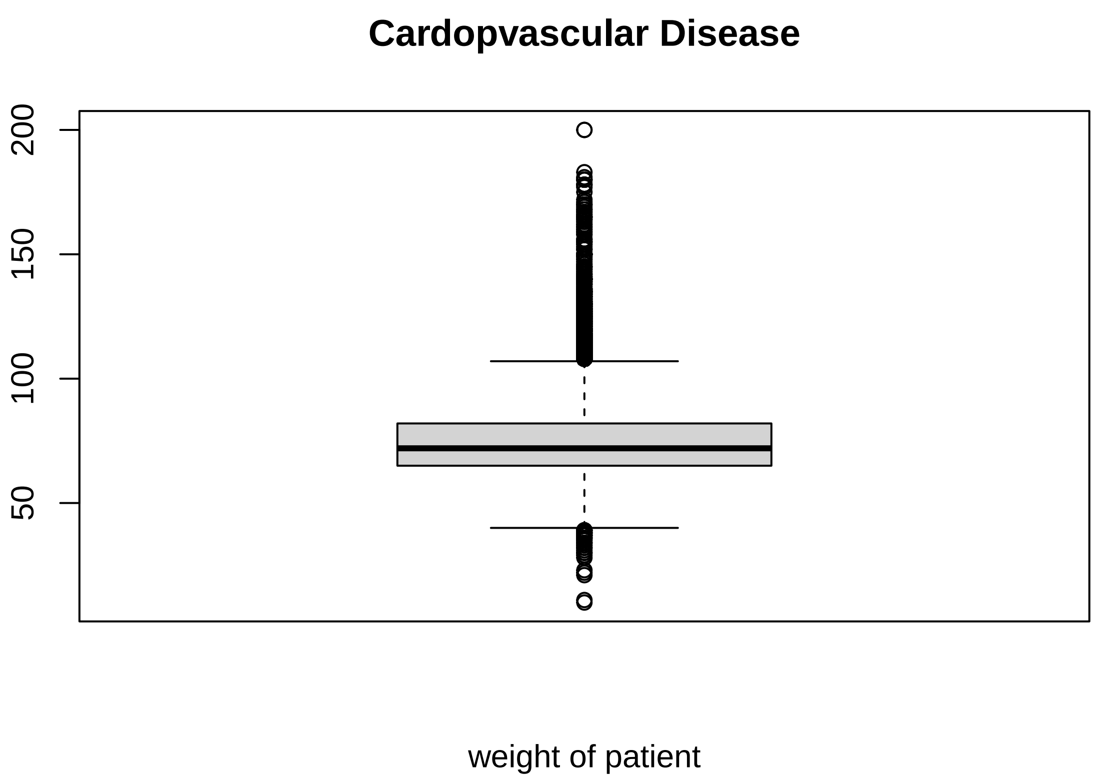
```r
#Boxplots
boxplot(age,xlab= "age of patient", main="Cardiovascular Disease")
```



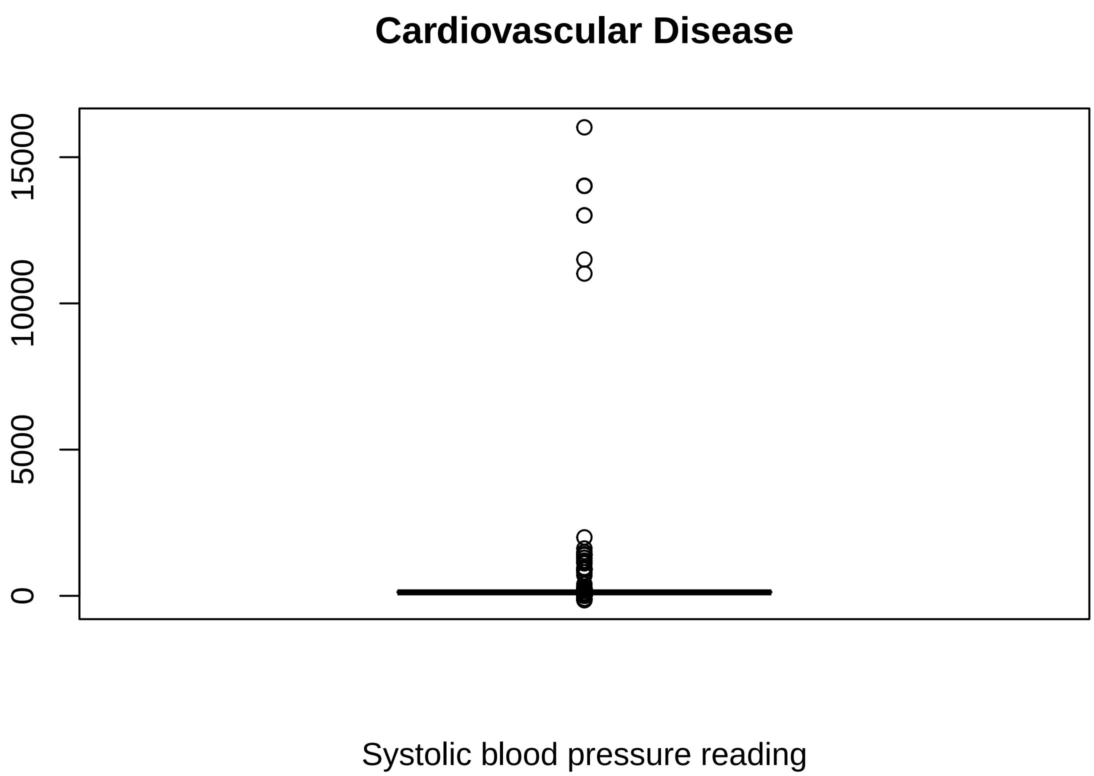

```r
boxplot(height, xlab="height of patient", main="Cardiovascular Disease")
```



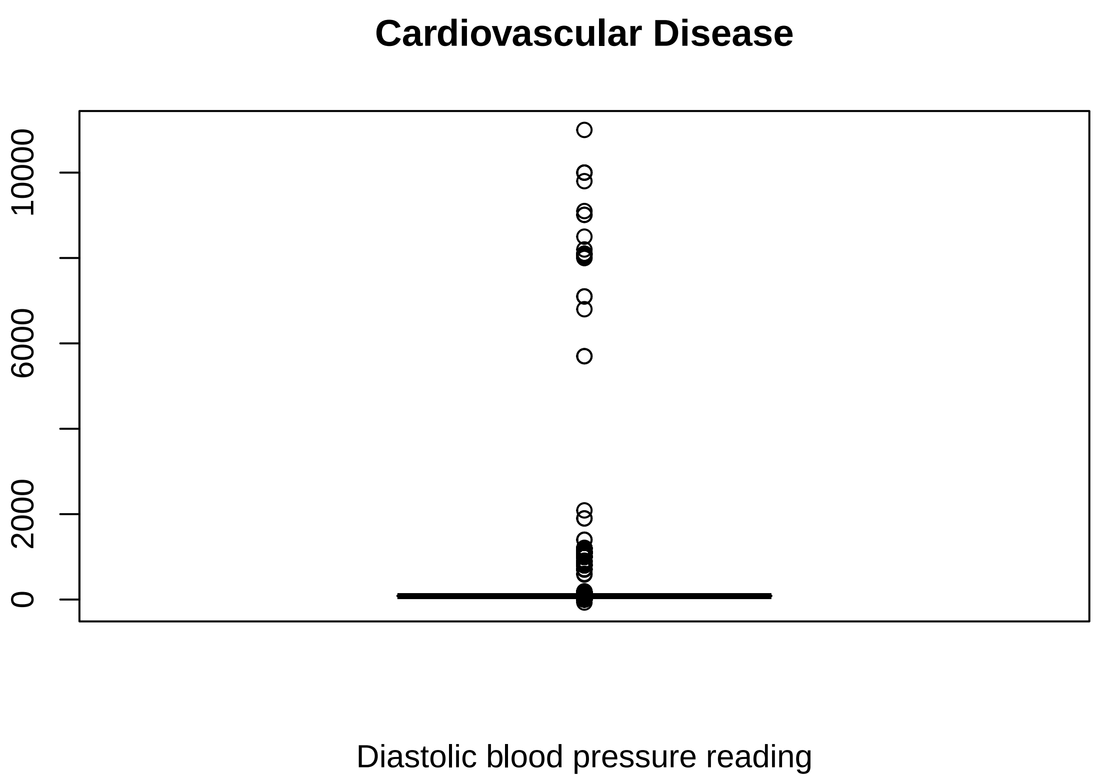

```r
boxplot(weight, xlab="weight of patient", main="Cardopvascular Disease")
```




```r
boxplot(systolic, xlab= "Systolic blood pressure reading", main="Cardiovascular Disease")
```




```r
boxplot(diastolic, xlab= "Diastolic blood pressure reading", main="Cardiovascular Disease")
```




```r
#Correlations
cor(age, cardio)
```

```
## [1] 0.2381366
```

```
cor(height, cardio)
```

```
## [1] -0.01082106
```

```
cor(weight, cardio)
```

```
## [1] 0.1816596
```

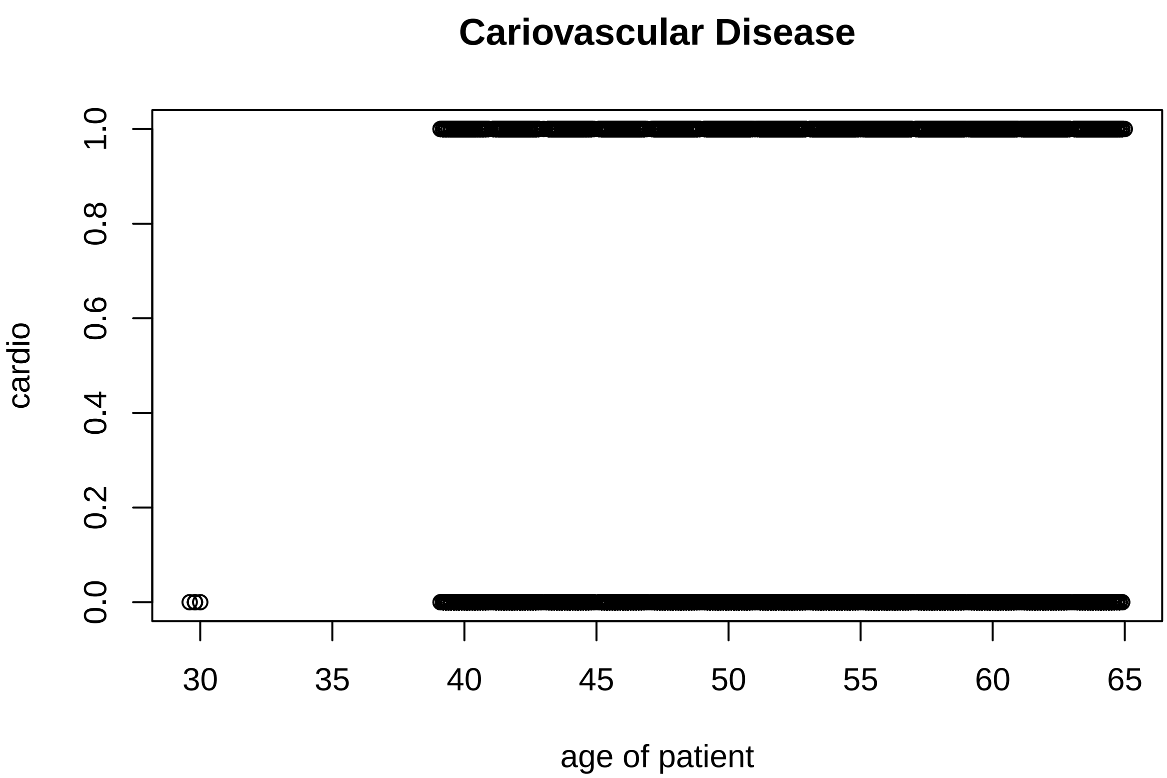
```
cor(cholesterol, cardio)
```

```
## [1] 0.2211473
```

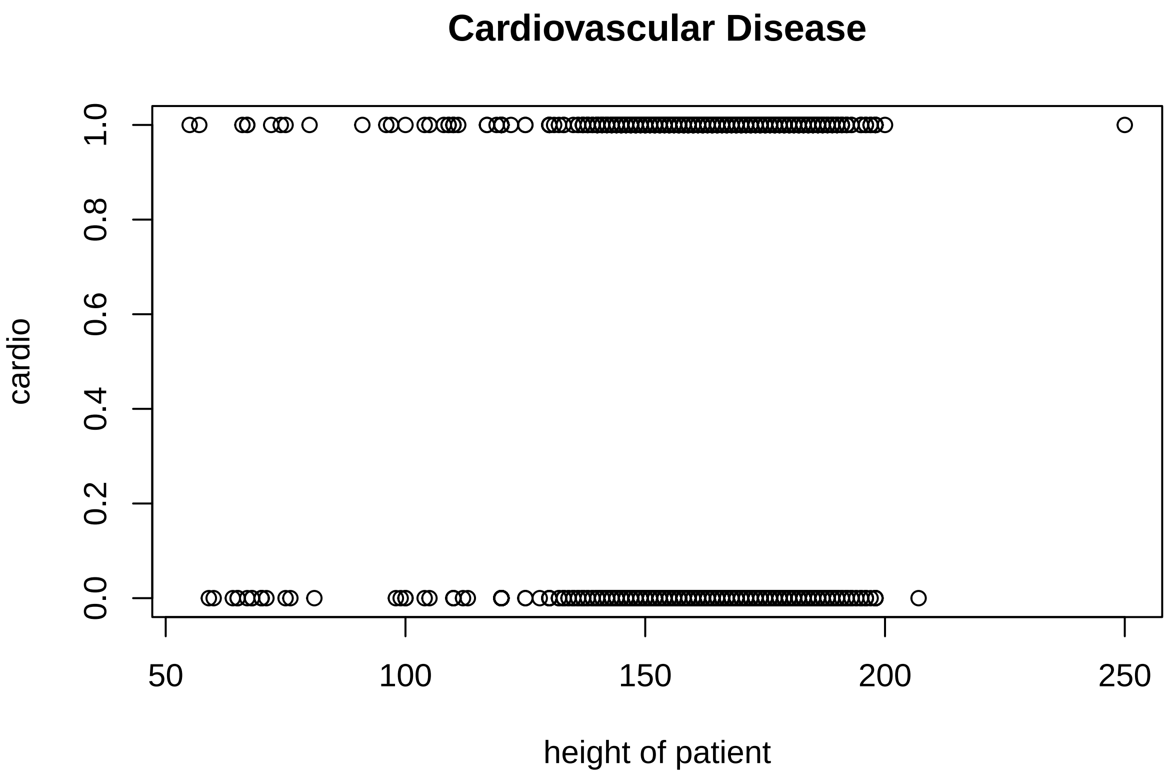
```
cor(active, cardio)
```

```
## [1] -0.03565325
```

```
#Before identifying the three models: Linear Probability Model, Probit and Logit Model,
#we were able to calculate different correlations between the predictors and cardiovascular
#disease being present.The three highest positive correlations that we found with cardiovascular
#disease is with age (0.24),cholesterol(0.22), and weight(0.18). With an increase in
#these values, there was also increases in risk of cardiovascular disease being present.
#Some negative correlations that are important to point out are whether the patient is
#active(-0.04),and height(-0.01). Although very small values, it shows that as values were
#higher for these predictors, the rates of having cardiovascular disease decreased.

#Scatterplots
plot(age, cardio, main="Cariovascular Disease", xlab="age of patient")
```



```r
plot(height, cardio, main="Cardiovascular Disease", xlab= "height of patient")
```




```r
#Question 2 Part 3
#Linear Probability Model
lp.model <- lm(cardio~gender+age+height+weight+systolic+diastolic+cholesterol+
                 glucose+smoke+alcohol+active, data=heart)

summary(lp.model)
```

```
##
## Call:
## lm(formula = cardio ~ gender + age + height + weight + systolic +
##     diastolic + cholesterol + glucose + smoke + alcohol + active,
##     data = heart)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5205 -0.4315 -0.1108  0.4563  0.9579
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.770e-01  4.258e-02 -11.202  < 2e-16 ***
## gender       1.988e-02  4.490e-03   4.428 9.55e-06 ***
## age          4.041e-05  7.321e-07  55.188  < 2e-16 ***
## height      -2.291e-03  2.592e-04  -8.841  < 2e-16 ***
## weight       5.362e-03  1.311e-04  40.896  < 2e-16 ***
## systolic     1.330e-04  1.152e-05  11.541  < 2e-16 ***
## diastolic    1.356e-04  9.420e-06  14.394  < 2e-16 ***
## cholesterol  1.317e-01  2.970e-03  44.346  < 2e-16 ***
## glucose     -2.558e-02  3.478e-03  -7.355 1.94e-13 ***
```

```
## smoke       -2.249e-02  6.985e-03  -3.220 0.001280 **
## alcohol     -2.871e-02  8.388e-03  -3.423 0.000619 ***
## active      -4.153e-02  4.468e-03  -9.295  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.469 on 69988 degrees of freedom
## Multiple R-squared:  0.1202, Adjusted R-squared:   0.12
## F-statistic:   869 on 11 and 69988 DF,  p-value: < 2.2e-16
```

```
confint(lp.model)
```

```
##                     2.5 %        97.5 %
## (Intercept) -5.605119e-01 -3.935804e-01
## gender       1.107874e-02  2.867882e-02
## age          3.897092e-05  4.184092e-05
## height      -2.799258e-03 -1.783342e-03
## weight       5.104587e-03  5.618512e-03
## systolic     1.103867e-04  1.555505e-04
## diastolic    1.171266e-04  1.540519e-04
## cholesterol  1.258999e-01  1.375434e-01
## glucose     -3.240018e-02 -1.876462e-02
## smoke       -3.618301e-02 -8.803782e-03
## alcohol     -4.515529e-02 -1.227323e-02
## active      -5.028218e-02 -3.276959e-02
```

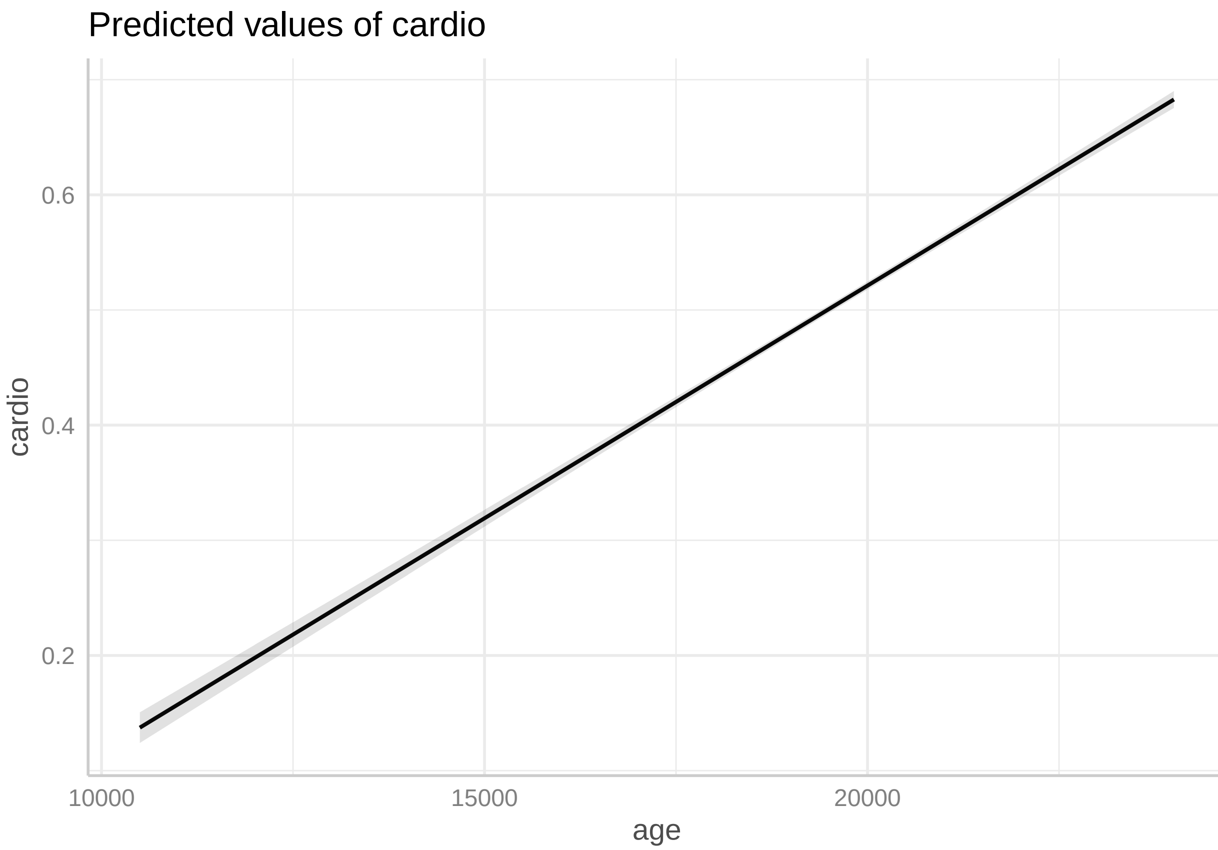
```
lp.pred.model <- ifelse(fitted(lp.model) > 0.5, 1, 0)
table(lp.pred.model,cardio)
```

```
##              cardio
## lp.pred.model     0     1
##             0 23904 13558
##             1 11117 21421
```

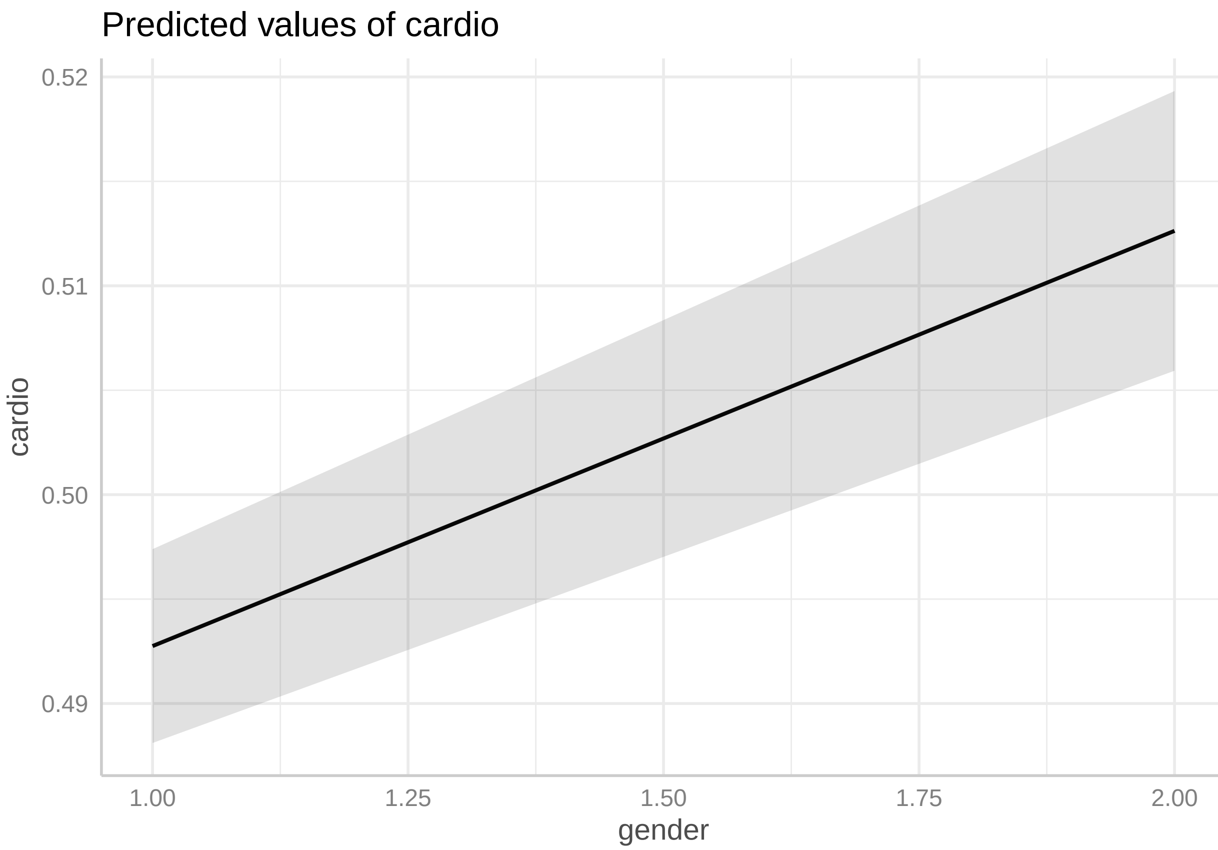
```
mean(lp.pred.model == cardio)
```

```
## [1] 0.6475
```

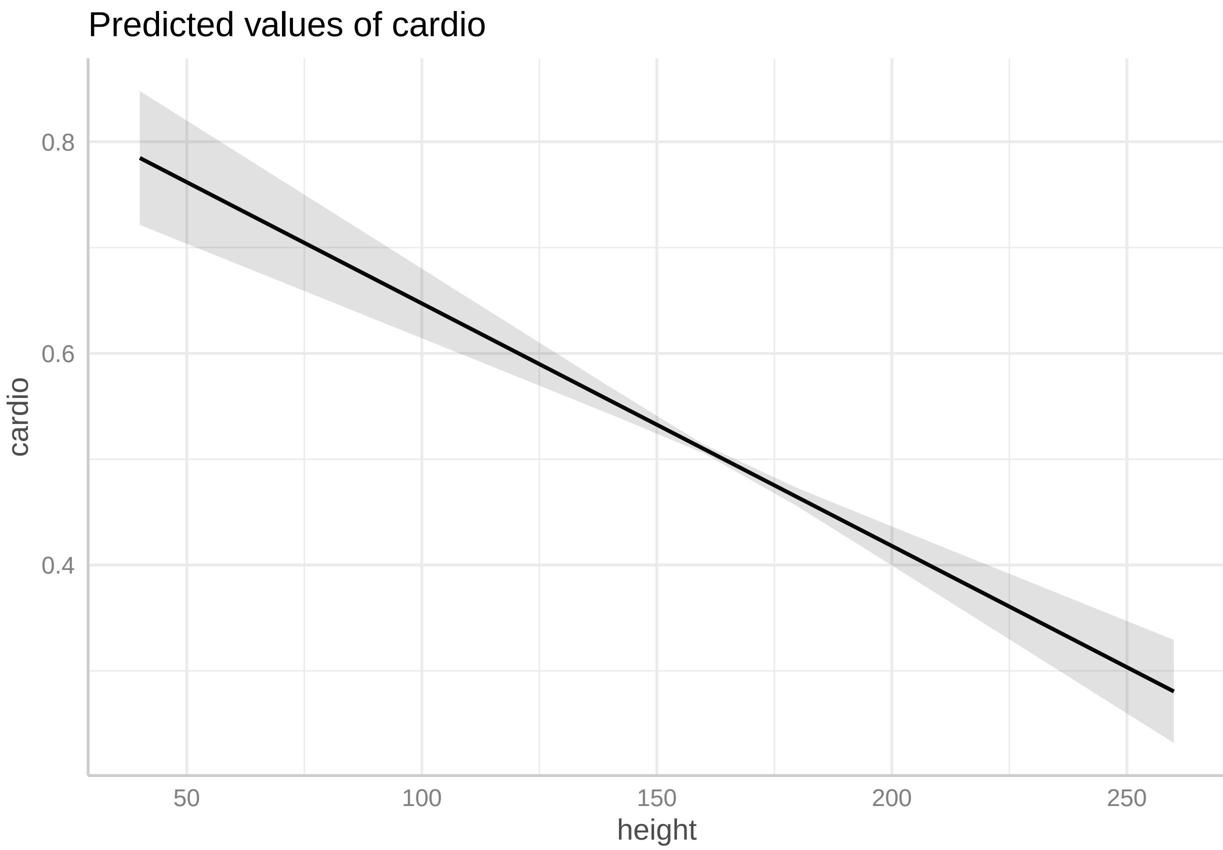
```
plot(ggpredict(lp.model,"age"))
```



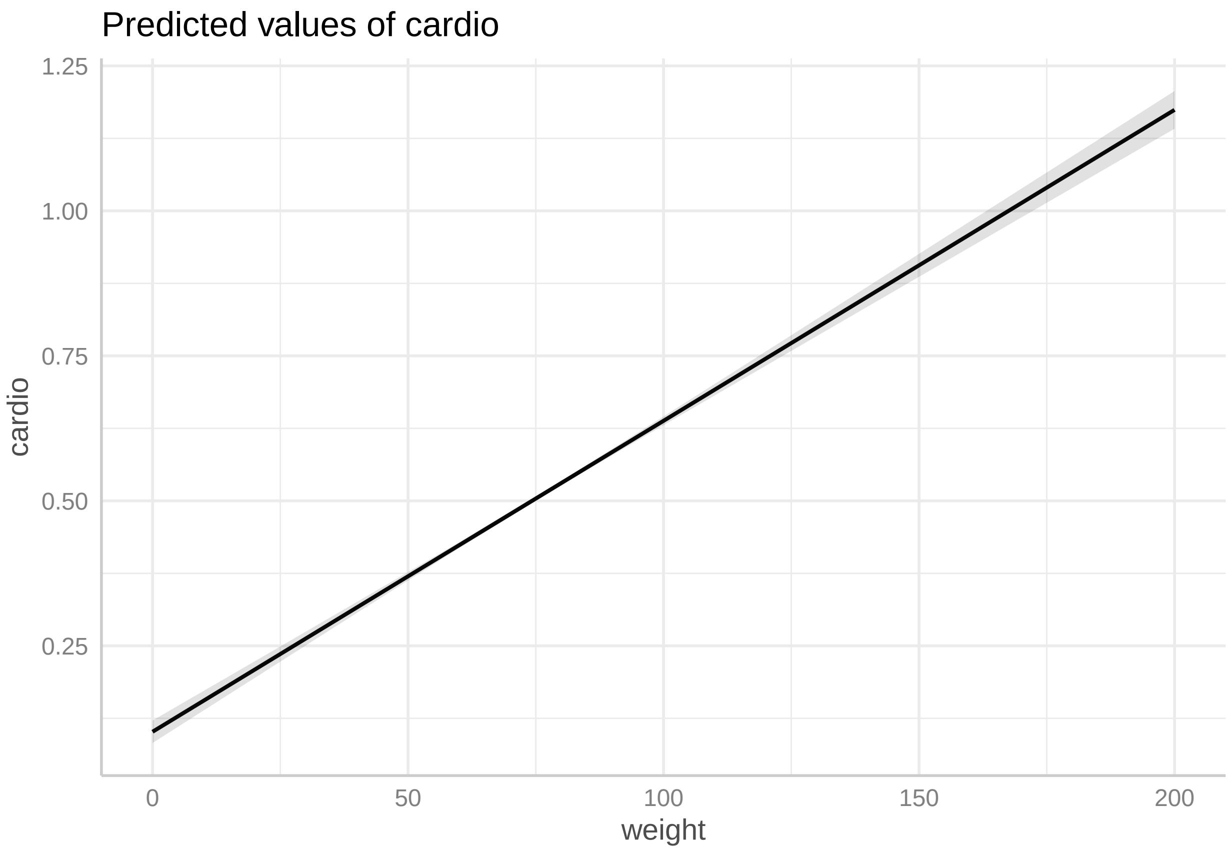

```
plot(ggpredict(lp.model,"gender"))
```



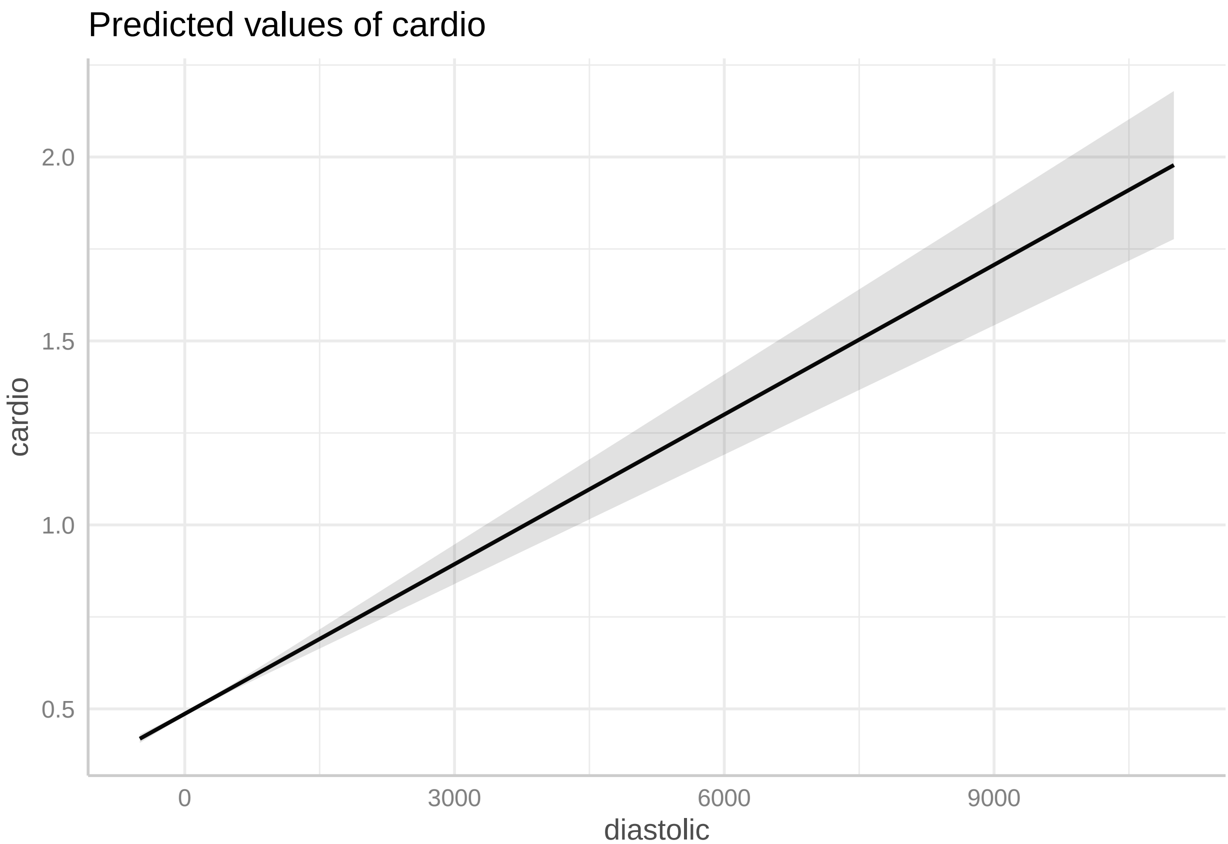

```
plot(ggpredict(lp.model,"height"))
```



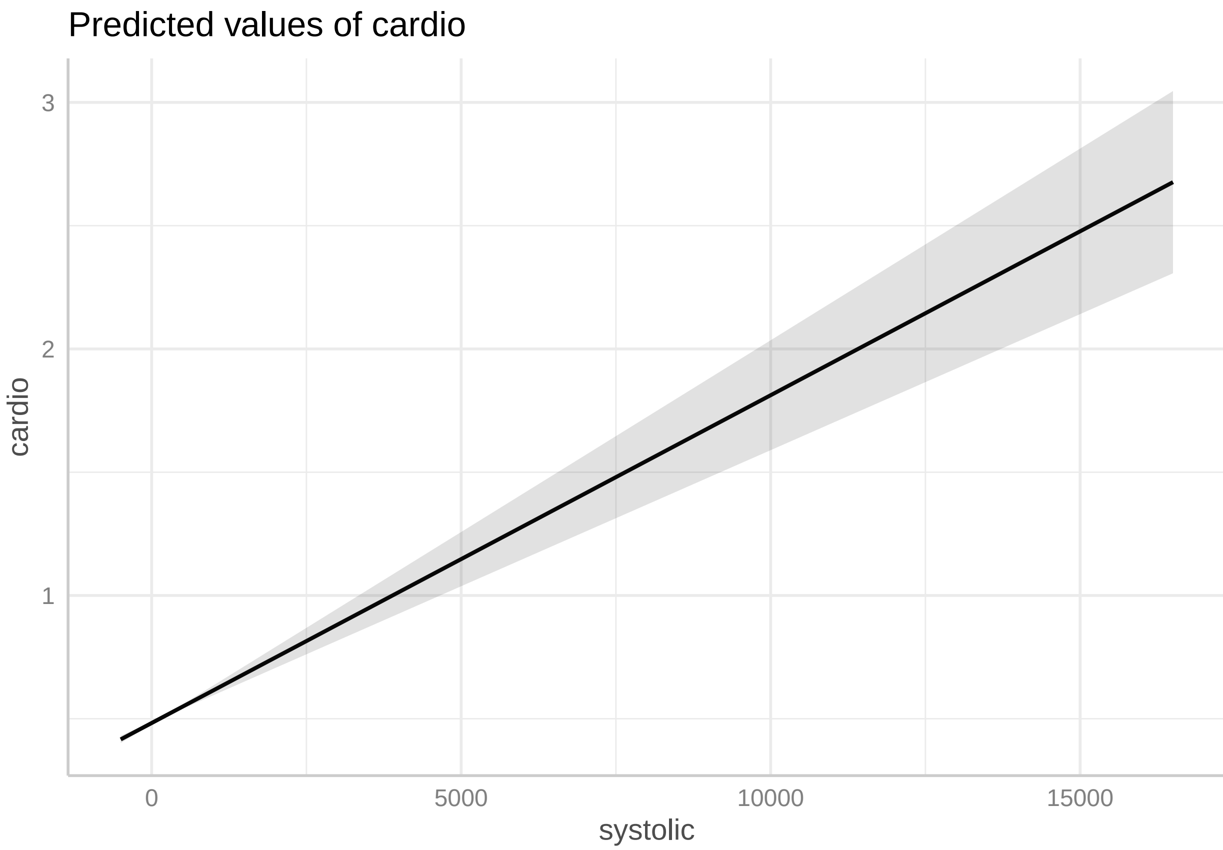

```
plot(ggpredict(lp.model,"weight"))
```



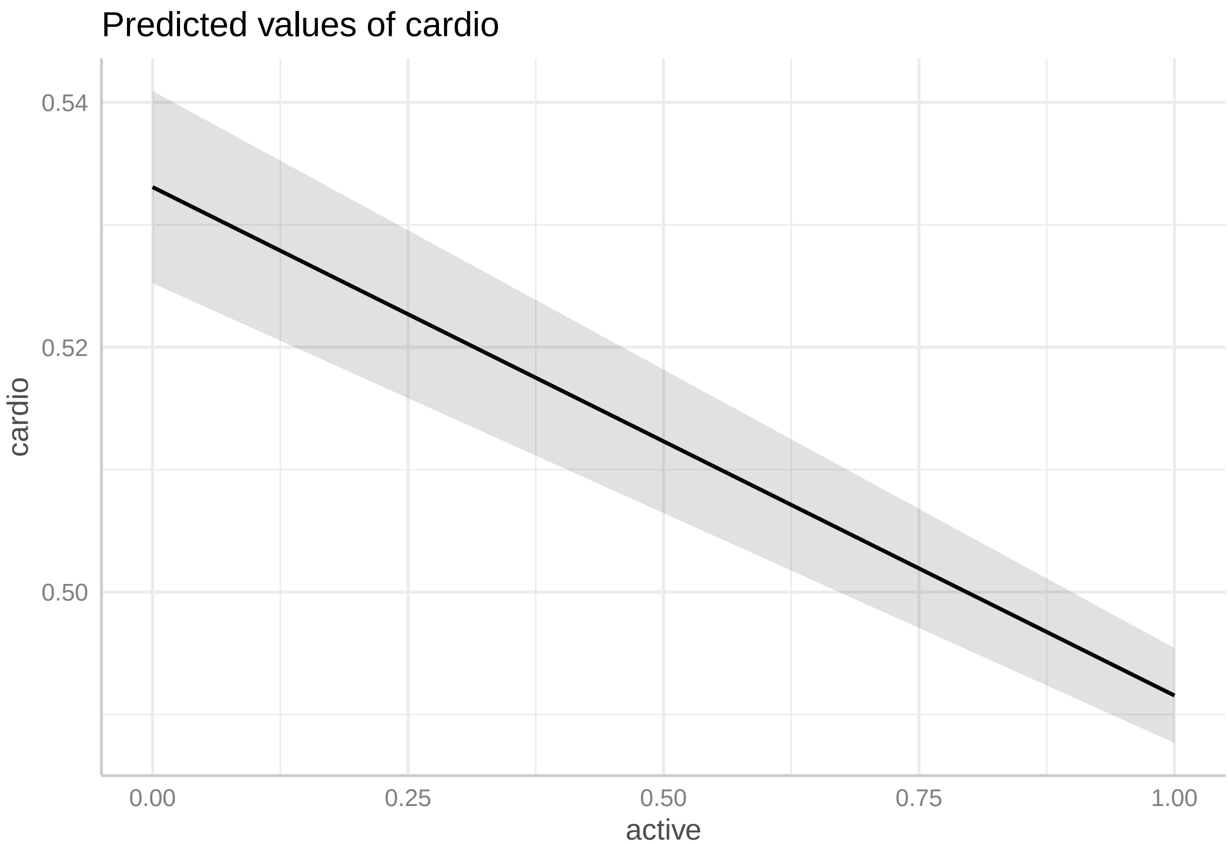

```
plot(ggpredict(lp.model,"diastolic"))
```



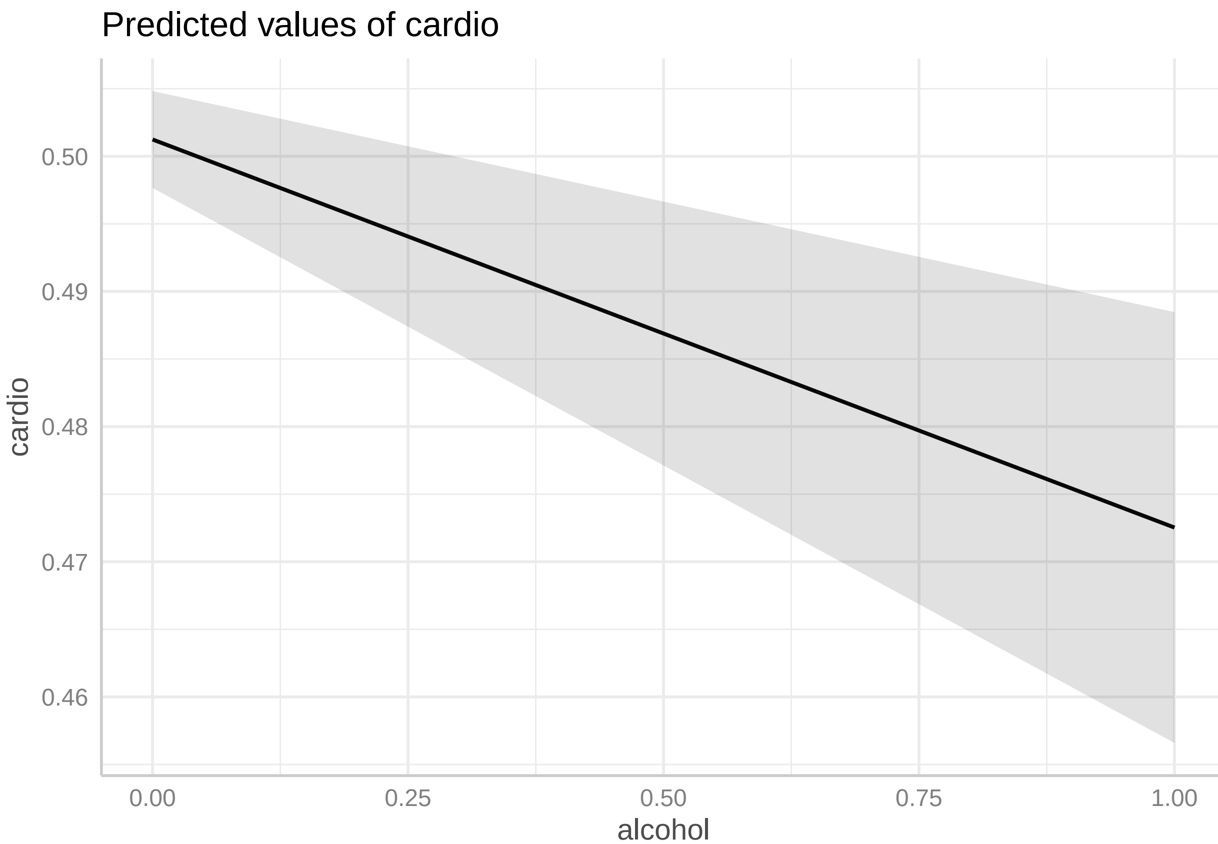

```
plot(ggpredict(lp.model,"systolic"))
```



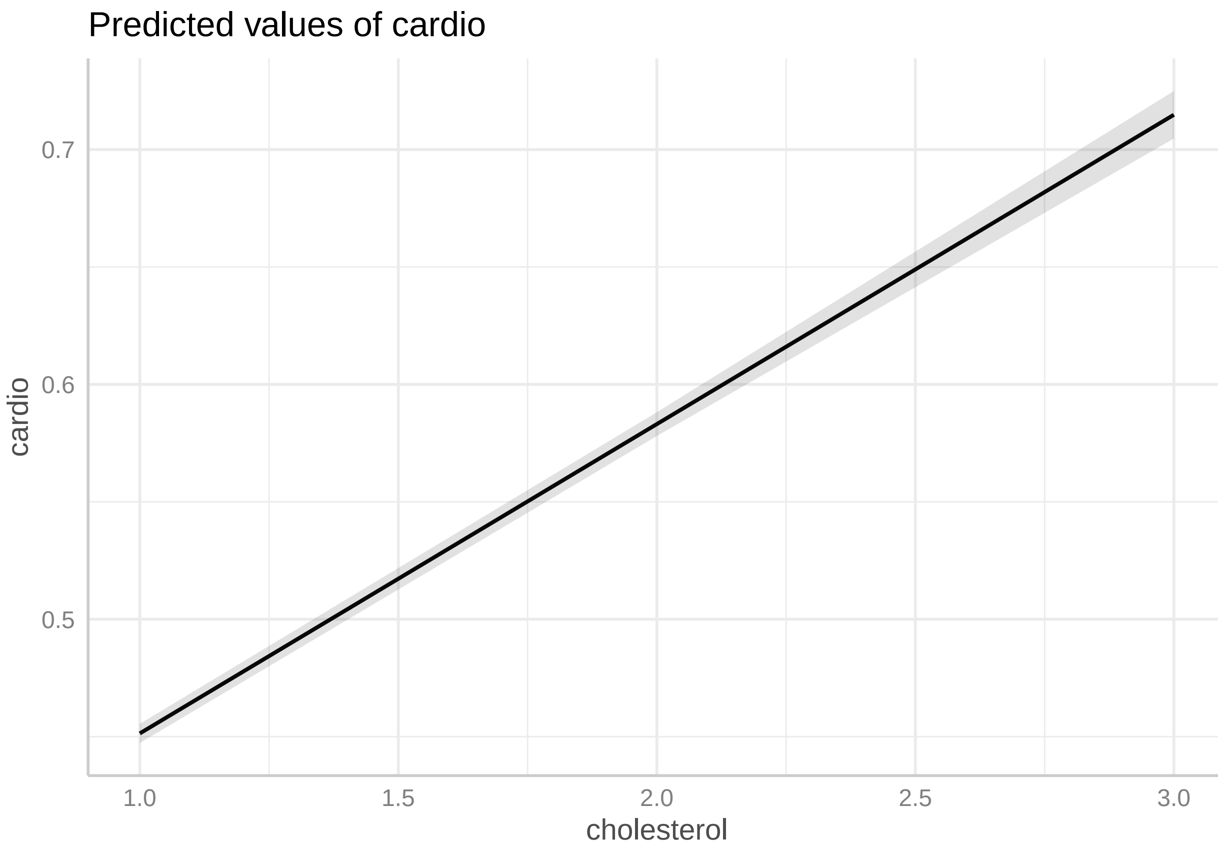

```
plot(ggpredict(lp.model,"active"))
```



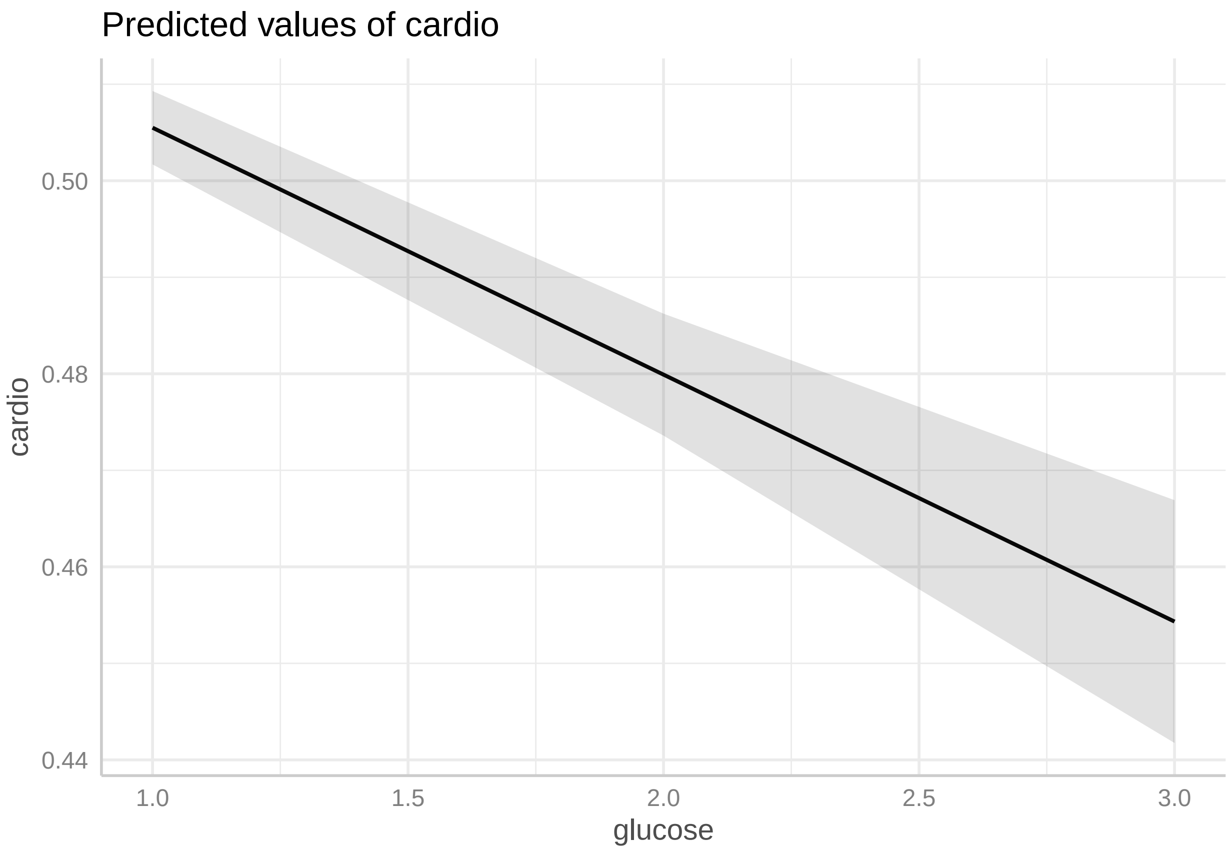

```
plot(ggpredict(lp.model,"alcohol"))
```



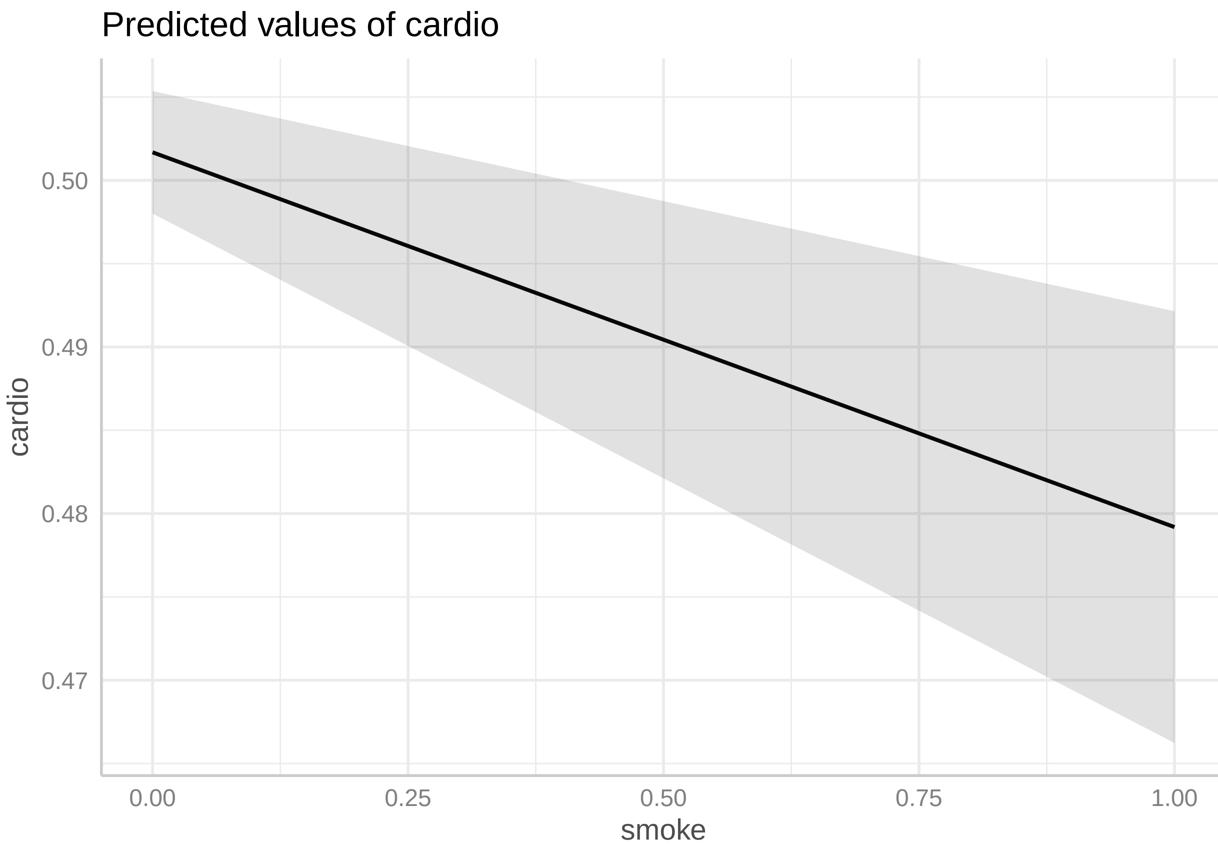

```
plot(ggpredict(lp.model,"cholesterol"))
```




```
plot(ggpredict(lp.model,"glucose"))
```




```
plot(ggpredict(lp.model,"smoke"))
```




```r
#After using the ggpredict command, we were able to support the correlation that we
#found earlier on between certain predictors. When looking at height, as the height
#of the patient increased, the predicted value of having a cardiovascular disease went
#from about 0.50-0.55 range when there height was at the 150 cm threshold, and decreased
#to around 0.35 when their height was greater than 250cm. There was a small error band
#around the lower and higher threshold values but was still pretty close to what we
#expected. The accuracy on the looking at weight predictions were very accurate.
#As the weight increased from 50kg which had a predicted value of 0.38, to about a
#predicted value of 0.63 when the weight was greater than 100kg which made sense.


#Probit Model
probit.model = glm(cardio~gender+age+height+weight+systolic+diastolic+cholesterol+
                 glucose+smoke+alcohol+active, family=binomial(link="probit"),data = heart)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(probit.model)
```

```
##
## Call:
## glm(formula = cardio ~ gender + age + height + weight + systolic +
##     diastolic + cholesterol + glucose + smoke + alcohol + active,
##     family = binomial(link = "probit"), data = heart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4904  -0.9691  -0.0346   0.9981   7.0151
```

```
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.186e+00  1.282e-01 -40.460  < 2e-16 ***
## gender       9.976e-03  1.276e-02   0.782    0.434
## age          9.106e-05  2.134e-06  42.675  < 2e-16 ***
## height      -3.260e-03  7.436e-04  -4.385 1.16e-05 ***
## weight       9.068e-03  3.948e-04  22.972  < 2e-16 ***
## systolic     2.398e-02  3.526e-04  68.012  < 2e-16 ***
## diastolic    1.483e-04  3.493e-05   4.247 2.17e-05 ***
## cholesterol  3.099e-01  8.873e-03  34.921  < 2e-16 ***
## glucose     -7.198e-02  1.016e-02  -7.083 1.41e-12 ***
## smoke       -8.084e-02  1.999e-02  -4.044 5.26e-05 ***
## alcohol     -1.038e-01  2.417e-02  -4.293 1.76e-05 ***
## active      -1.272e-01  1.273e-02  -9.990  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 97041  on 69999  degrees of freedom
## Residual deviance: 81454  on 69988  degrees of freedom
## AIC: 81478
##
## Number of Fisher Scoring iterations: 13
```

```
probit.pred.model <- ifelse(fitted(probit.model) > 0.5,1,0)
table(probit.pred.model, cardio)
```

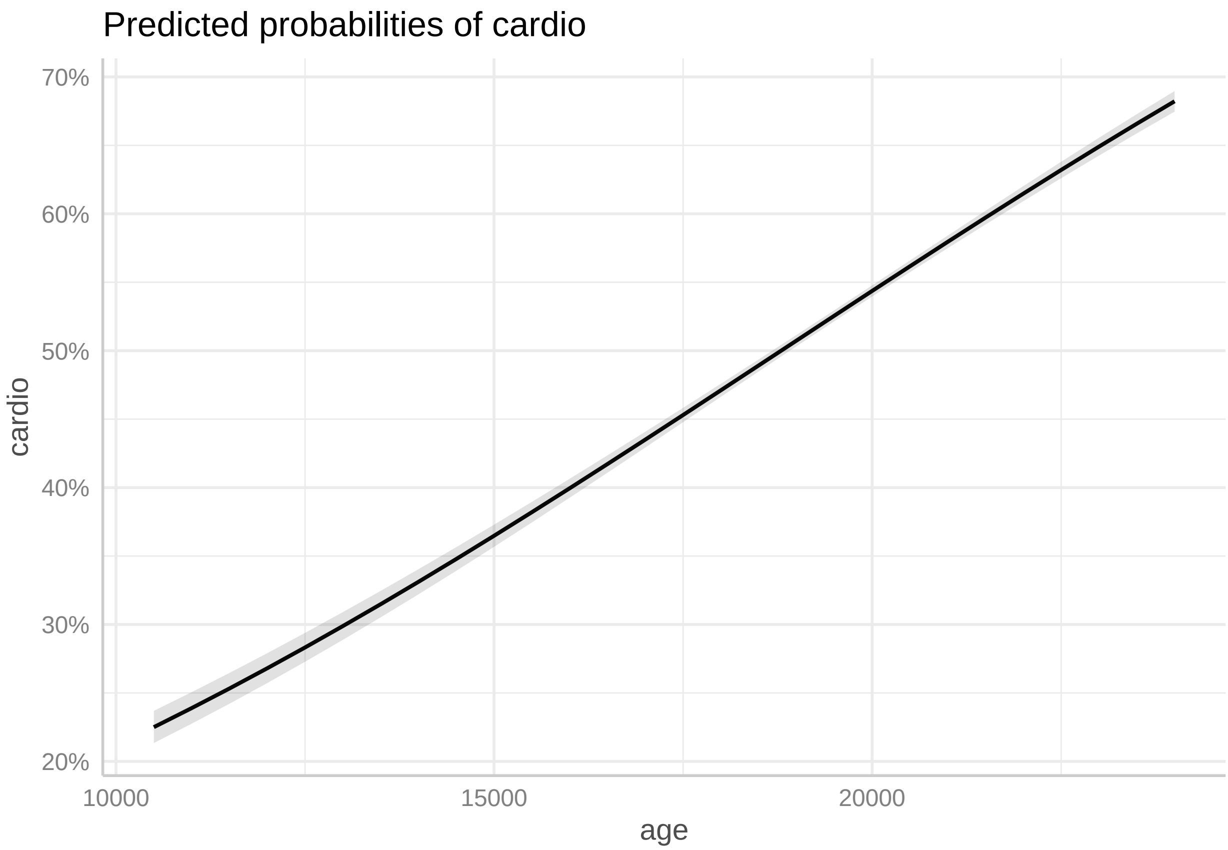
```
##                  cardio
## probit.pred.model     0     1
##                 0 26816 11281
##                 1  8205 23698
```

```
mean(probit.pred.model == cardio)
```

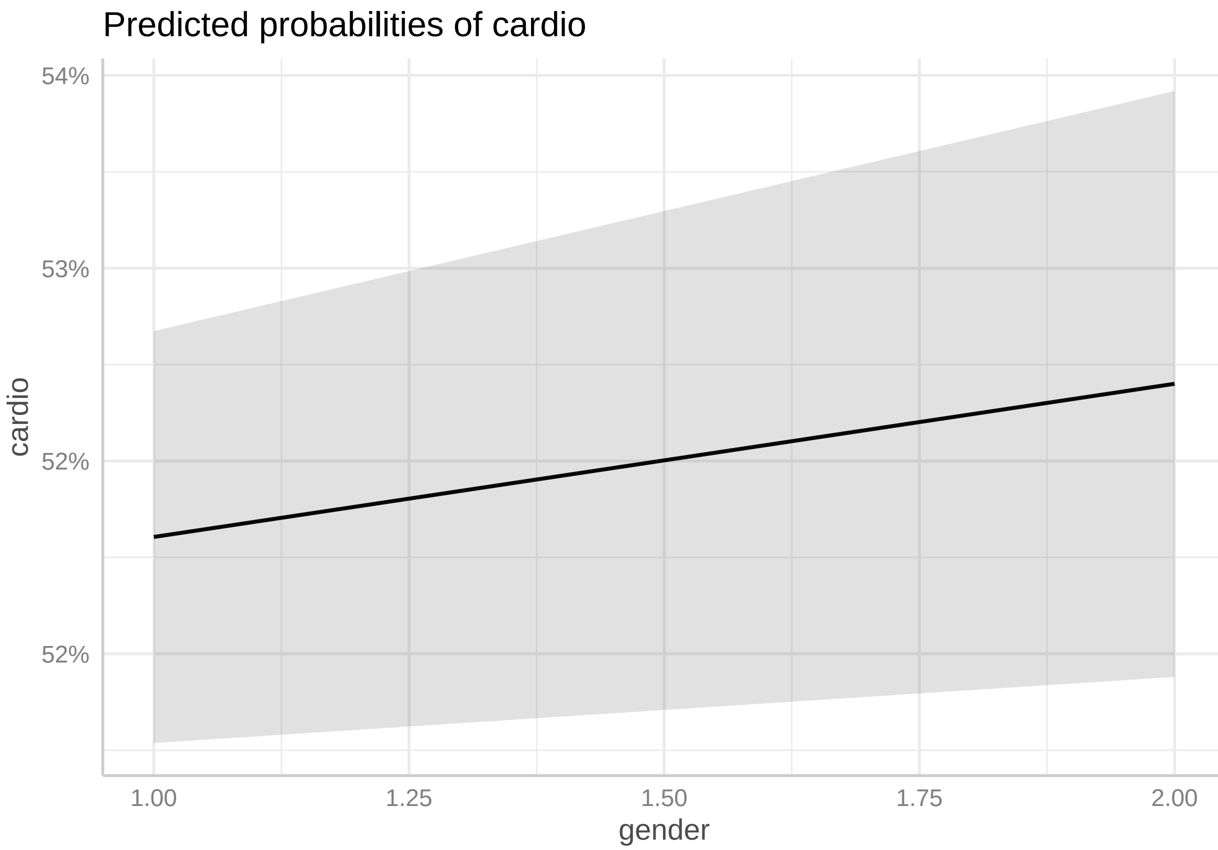
```
## [1] 0.7216286
```

```
plot(ggpredict(probit.model,"age"))
```

```
## Data were 'prettified'. Consider using `terms="age [all]"` to get smooth
##   plots.
```

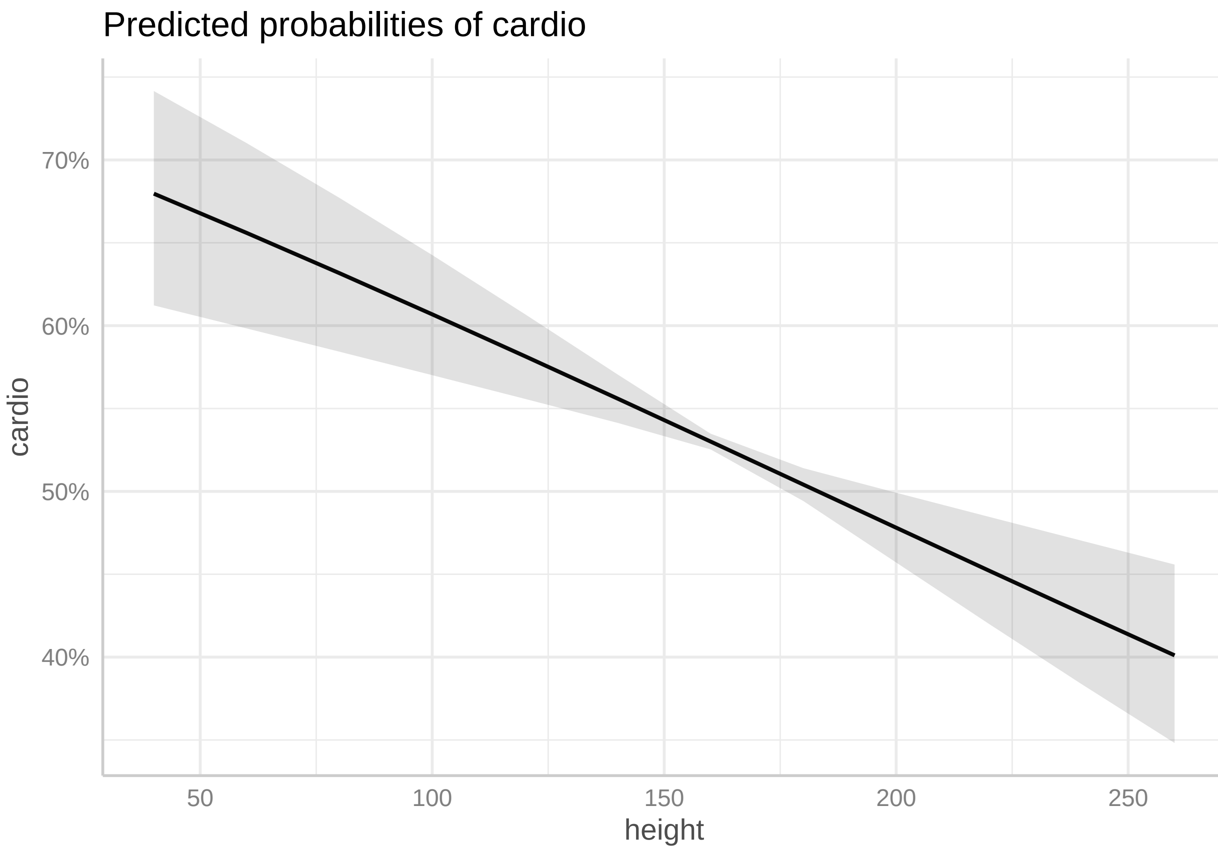



```
plot(ggpredict(probit.model,"gender"))
```



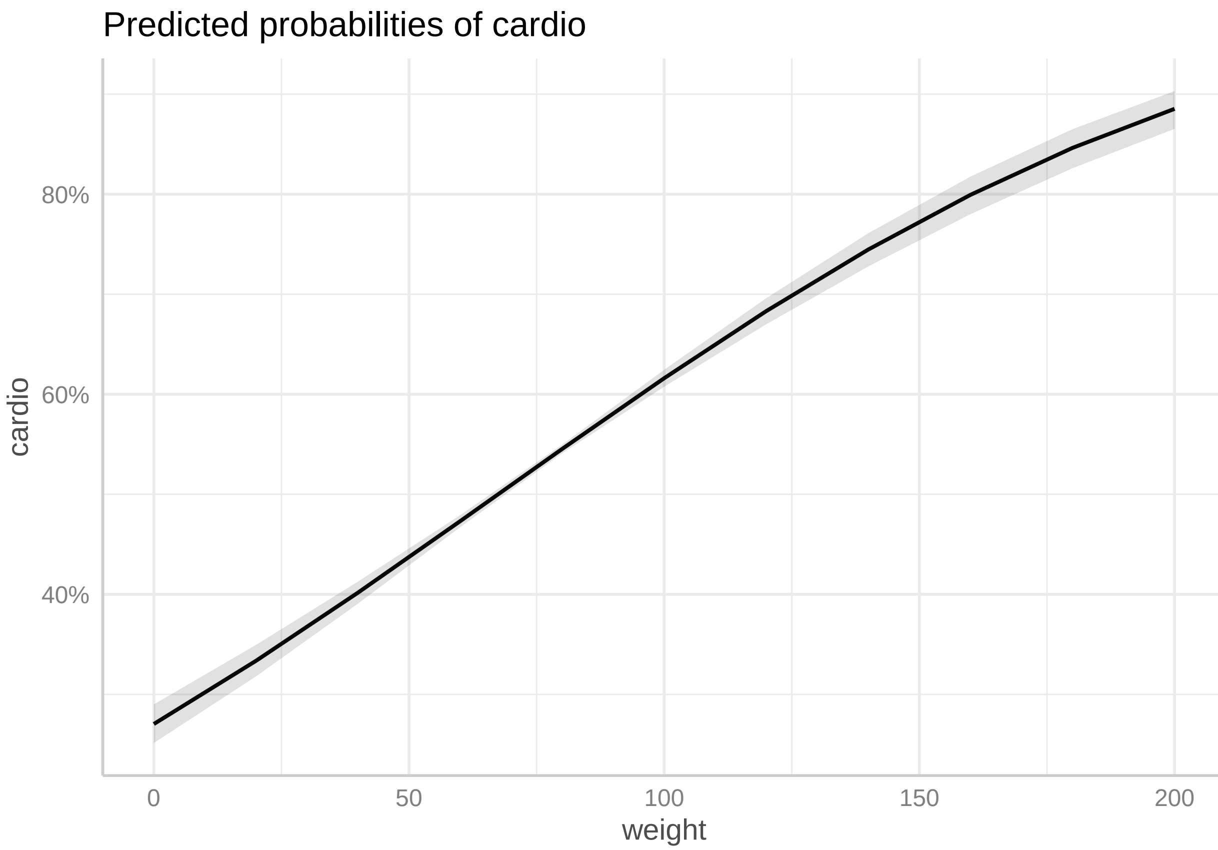

```
plot(ggpredict(probit.model,"height"))
```

```
## Data were 'prettified'. Consider using `terms="height [all]"` to get
##   smooth plots.
```



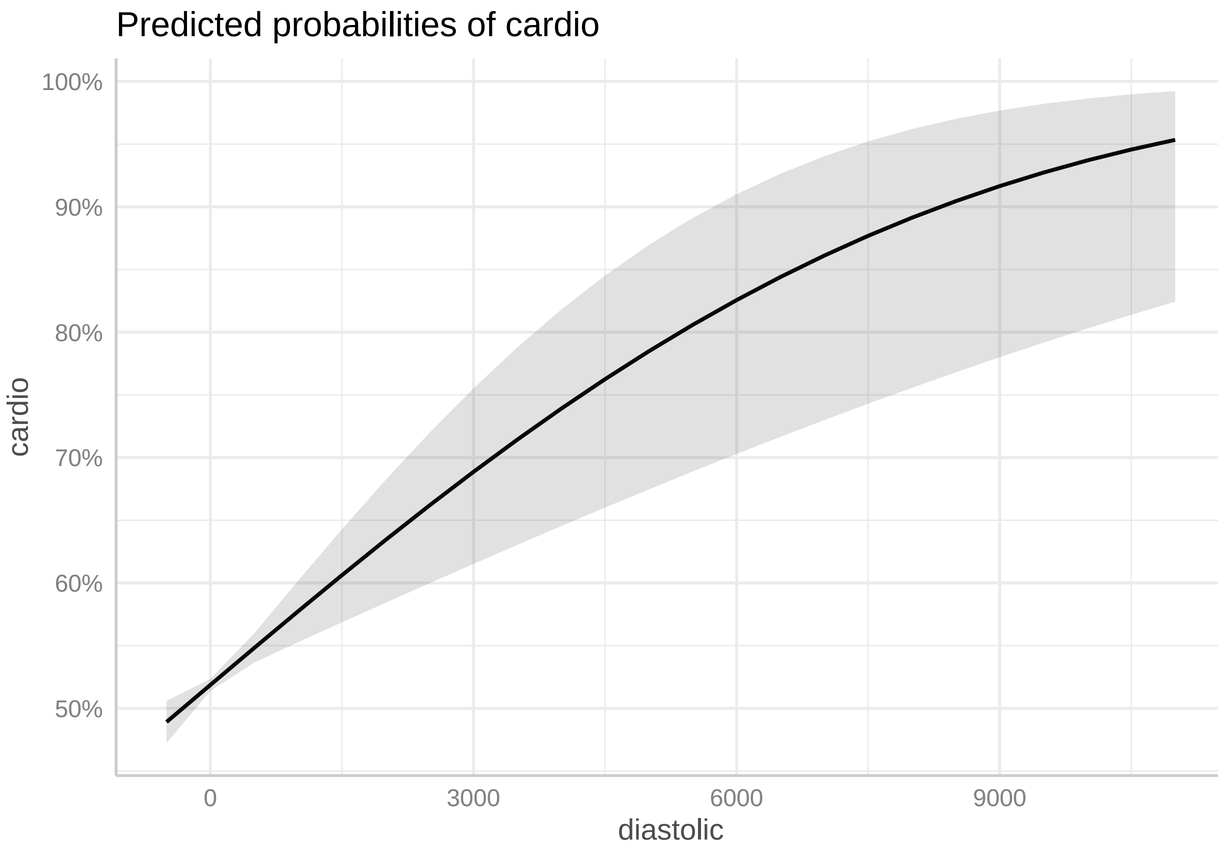

```
plot(ggpredict(probit.model,"weight"))
```

```
## Data were 'prettified'. Consider using `terms="weight [all]"` to get
##   smooth plots.
```



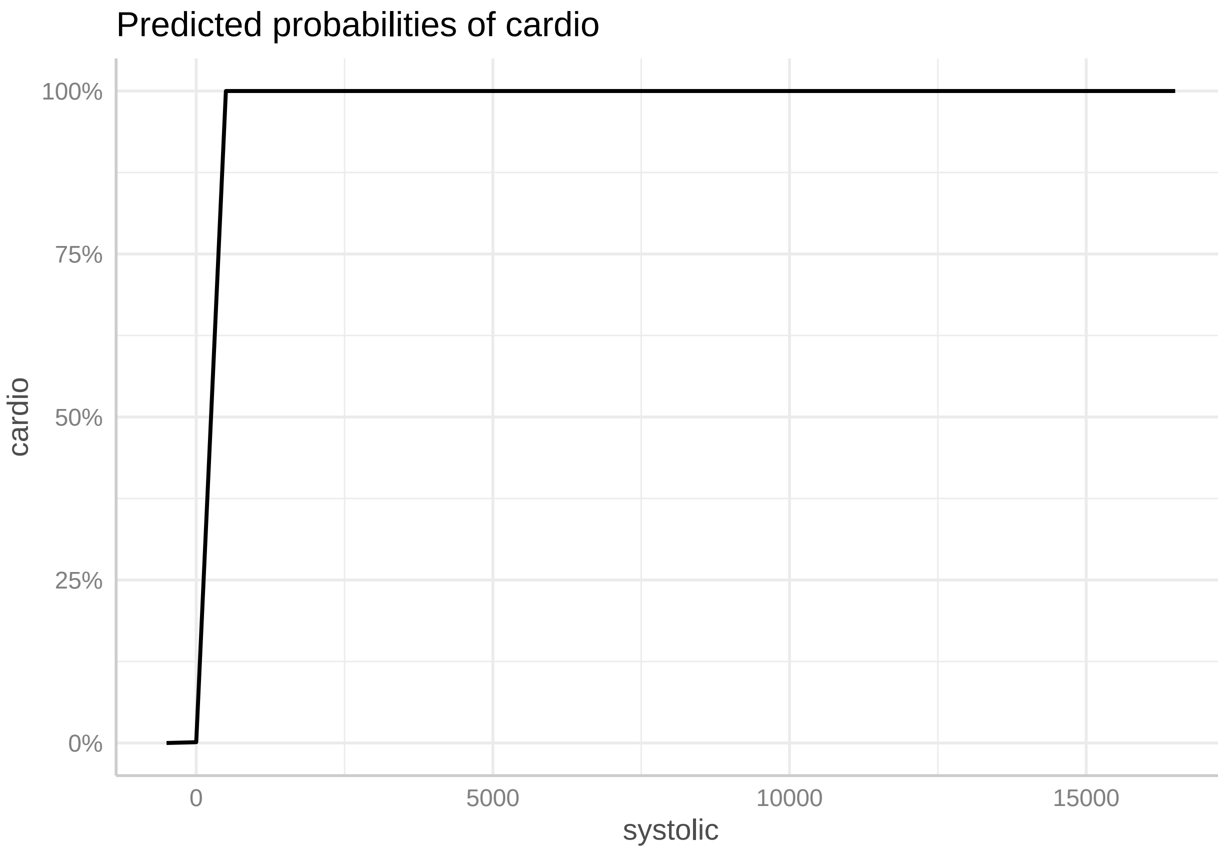

```r
plot(ggpredict(probit.model,"diastolic"))
```

```
## Data were 'prettified'. Consider using `terms="diastolic [all]"` to get
##   smooth plots.
```

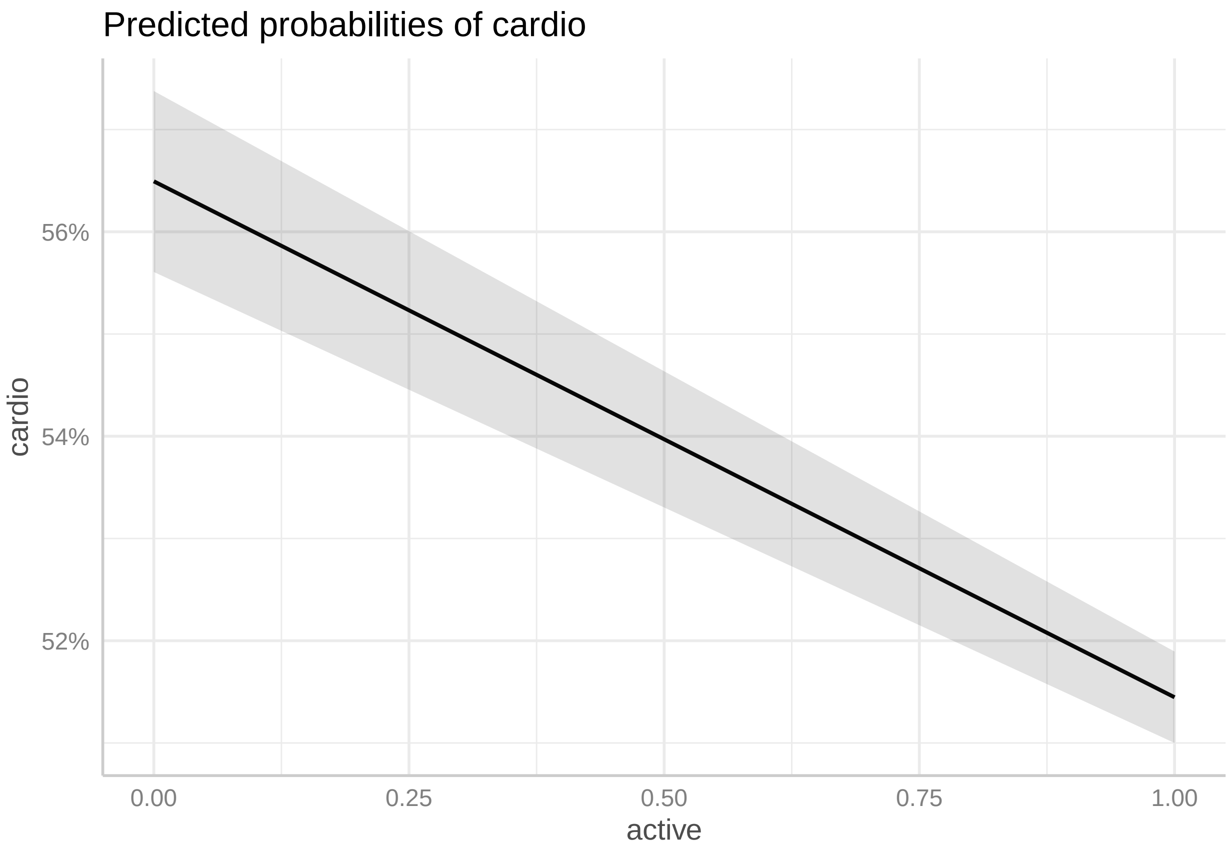



```
plot(ggpredict(probit.model,"systolic"))
```

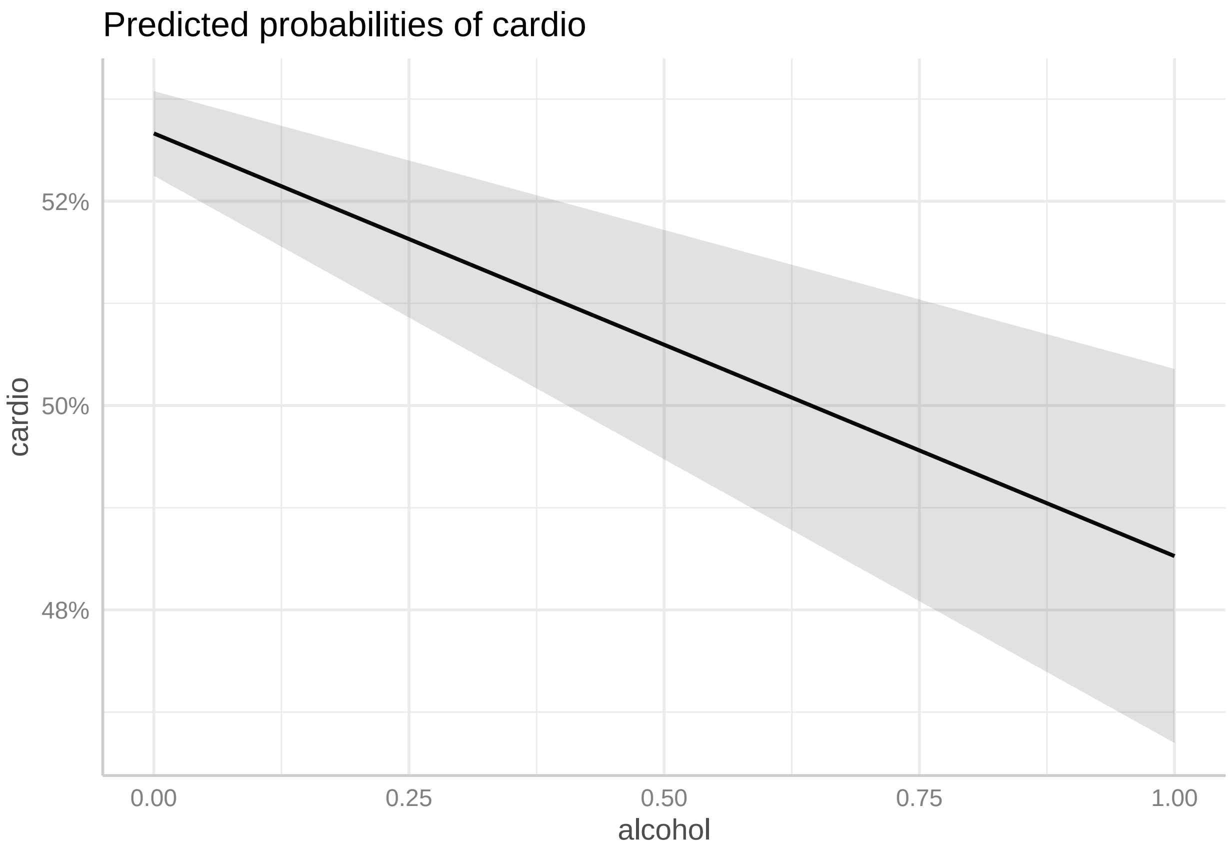
```
## Data were 'prettified'. Consider using `terms="systolic [all]"` to get
##   smooth plots.
```



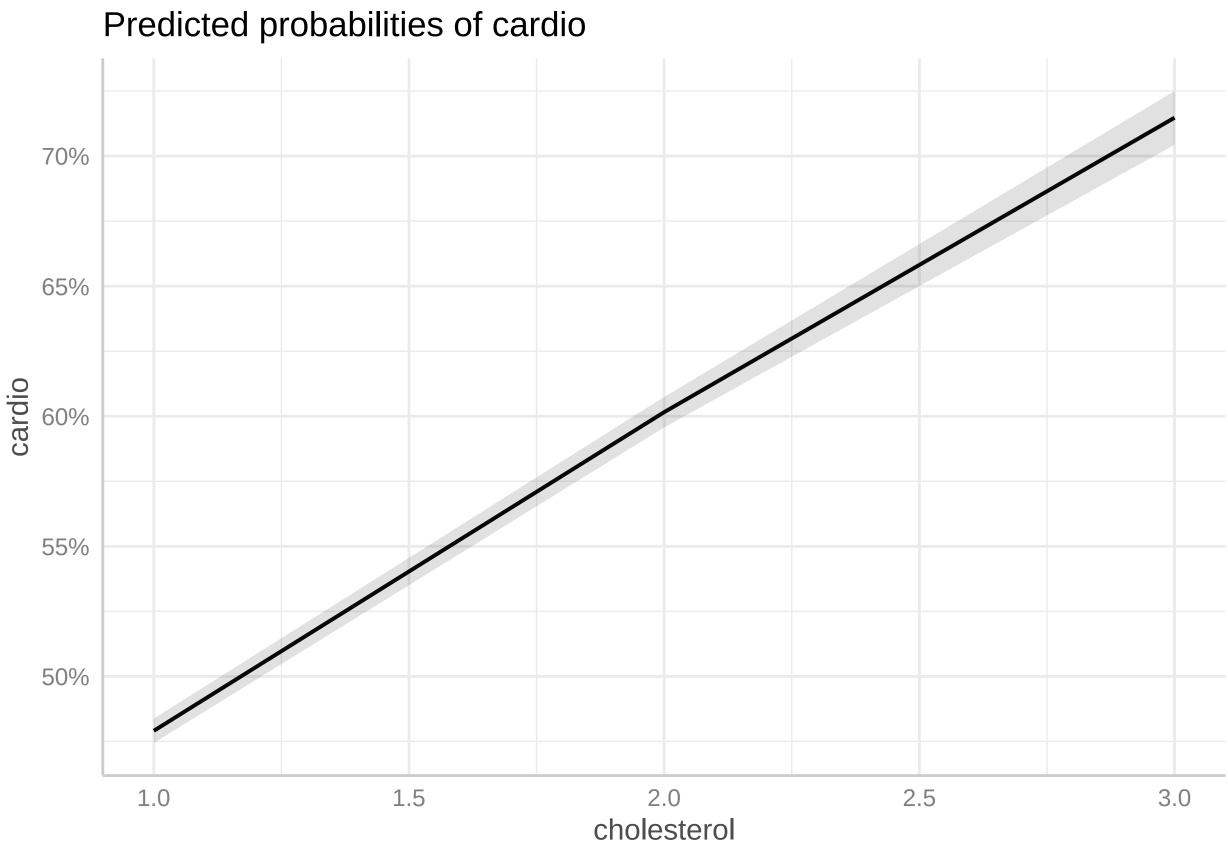

```
plot(ggpredict(probit.model,"active"))
```



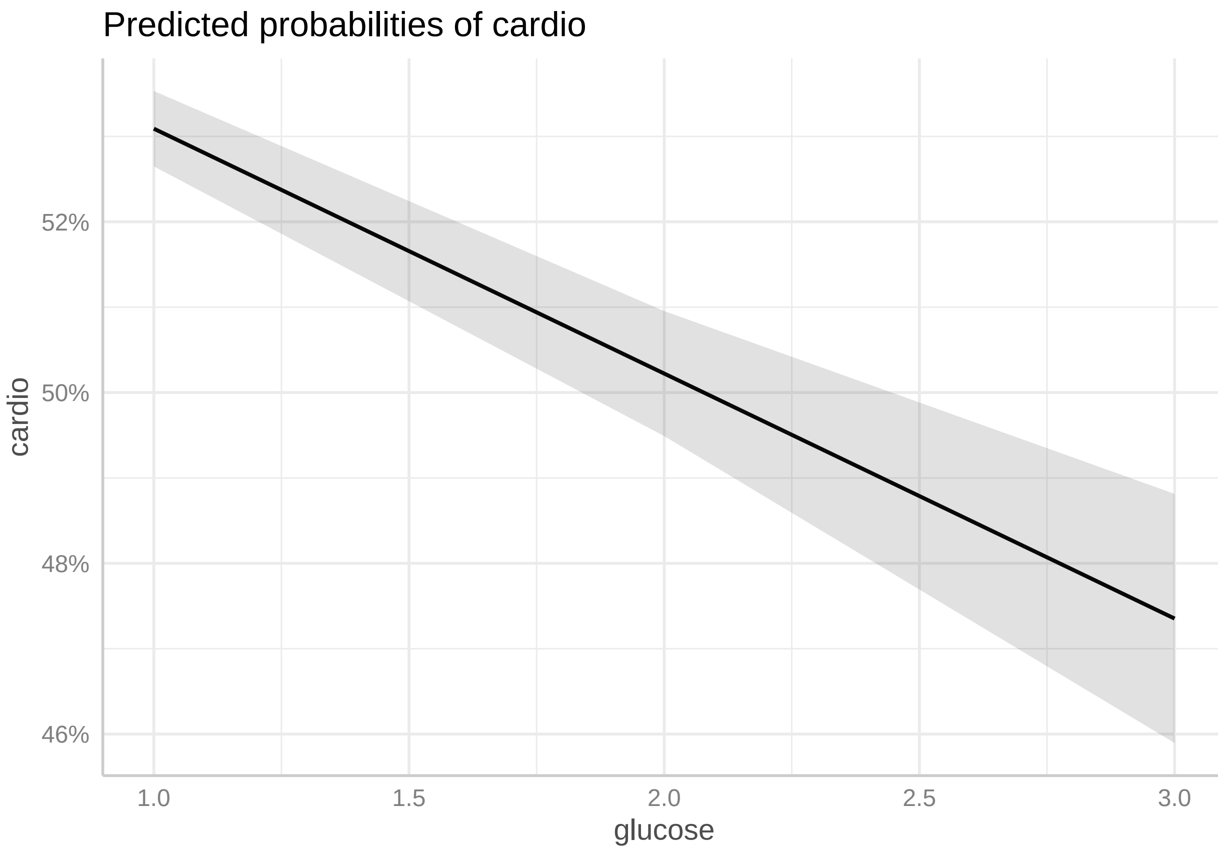

```
plot(ggpredict(probit.model,"alcohol"))
```



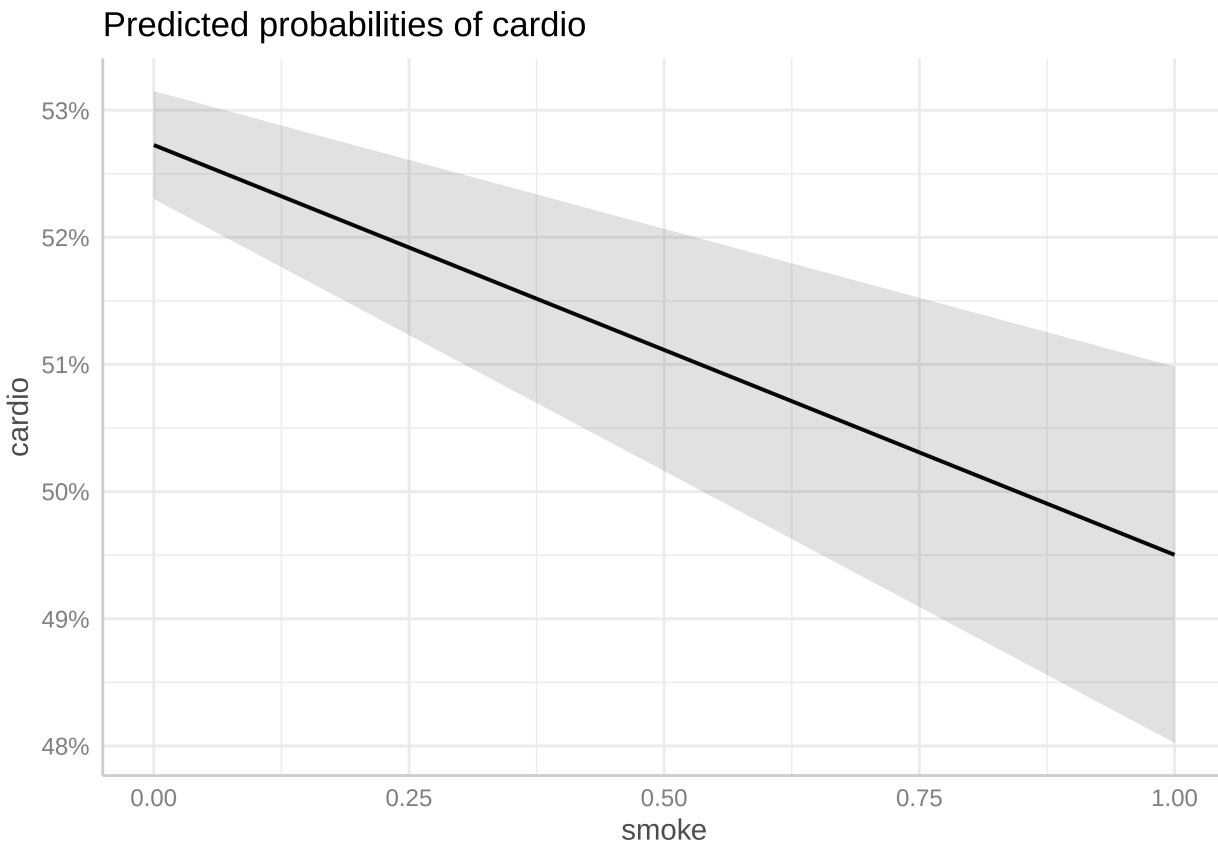

```
plot(ggpredict(probit.model,"cholesterol"))
```




```
plot(ggpredict(probit.model,"glucose"))
```




```
plot(ggpredict(probit.model,"smoke"))
```




```
#Looking at the probit model predicted values, we compared the same variables to see how
#it affected the values. The same conclusion can be made about the correlation between
#predictors and having cardiovascular disease. But something we found interesting was that
#for the height predicted values, the error band of the confidence interval seemed to be a
#bit larger than the linear probability model. We believe that although this may be the case,
#we found our probit model to have the best accuracy for predicting cardiovascular disease
#within the patient, as it balances out the values for each predictor more accurately. For
#the weight, the predicted probabilities were a bit higher at 50kg, around 0.45, and 0.63
#at around 100kg.

#Logit Model
logit.model <- glm(cardio~gender+age+height+weight+systolic+diastolic+cholesterol+
                 glucose+smoke+alcohol+active, family=binomial(link="logit"),data = heart)
```

```
## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(logit.model)
```

```
##
## Call:
## glm(formula = cardio ~ gender + age + height + weight + systolic +
##     diastolic + cholesterol + glucose + smoke + alcohol + active,
##     family = binomial(link = "logit"), data = heart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -8.4904  -0.9638  -0.0976   0.9897   4.6641
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.510e+00  2.142e-01 -39.725  < 2e-16 ***
## gender       1.532e-02  2.107e-02   0.727    0.467
## age          1.488e-04  3.553e-06  41.889  < 2e-16 ***
## height      -5.732e-03  1.231e-03  -4.656 3.22e-06 ***
## weight       1.535e-02  6.594e-04  23.275  < 2e-16 ***
## systolic     3.953e-02  6.053e-04  65.314  < 2e-16 ***
## diastolic    3.001e-04  6.734e-05   4.456 8.37e-06 ***
## cholesterol  5.233e-01  1.499e-02  34.917  < 2e-16 ***
## glucose     -1.186e-01  1.700e-02  -6.978 2.99e-12 ***
## smoke       -1.316e-01  3.317e-02  -3.968 7.26e-05 ***
## alcohol     -1.691e-01  4.021e-02  -4.204 2.62e-05 ***
## active      -2.098e-01  2.105e-02  -9.967  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 97041  on 69999  degrees of freedom
## Residual deviance: 80920  on 69988  degrees of freedom
## AIC: 80944
##
## Number of Fisher Scoring iterations: 25
```

```r
logit.pred.model <- ifelse(fitted(logit.model) > 0.5, 1, 0)
table(logit.pred.model, cardio)
```

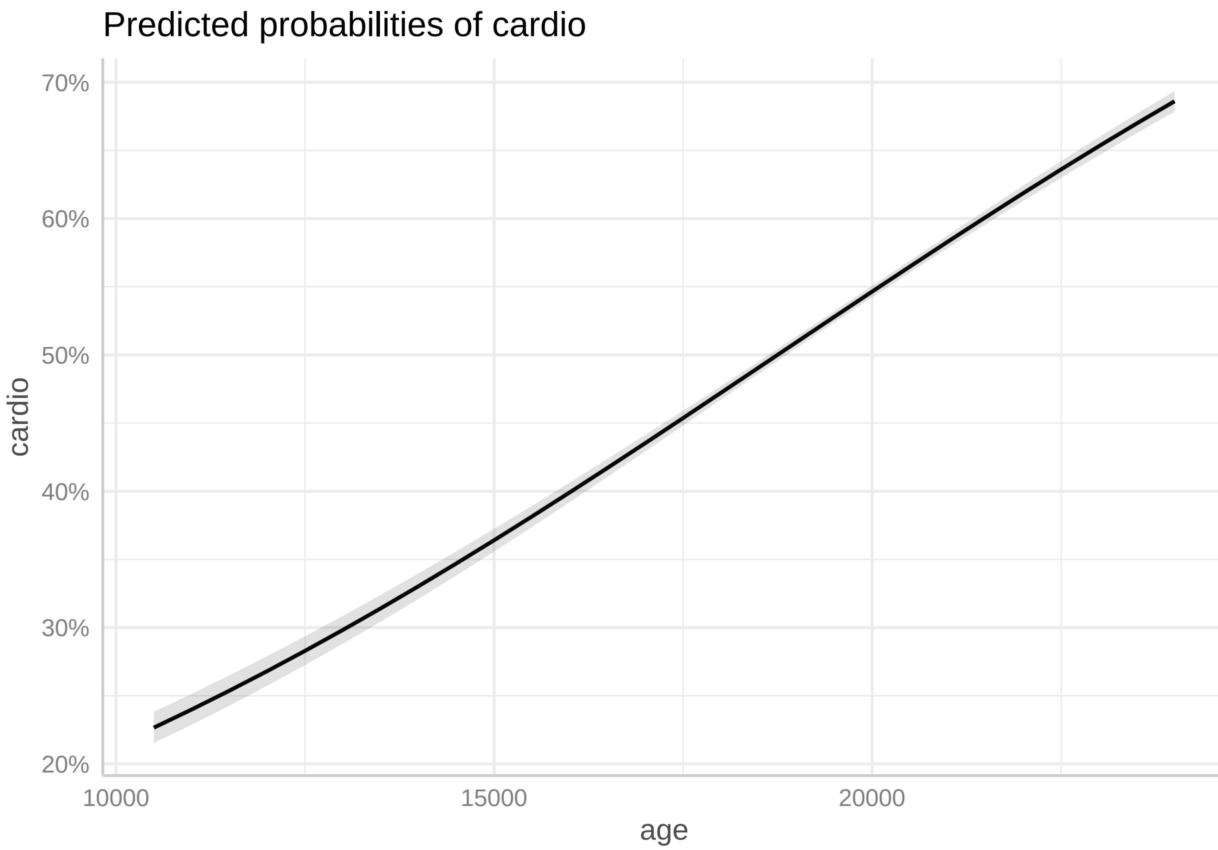
```
##                 cardio
## logit.pred.model     0     1
##                0 26777 11264
##                1  8244 23715
```

```r
mean(logit.pred.model == cardio)
```

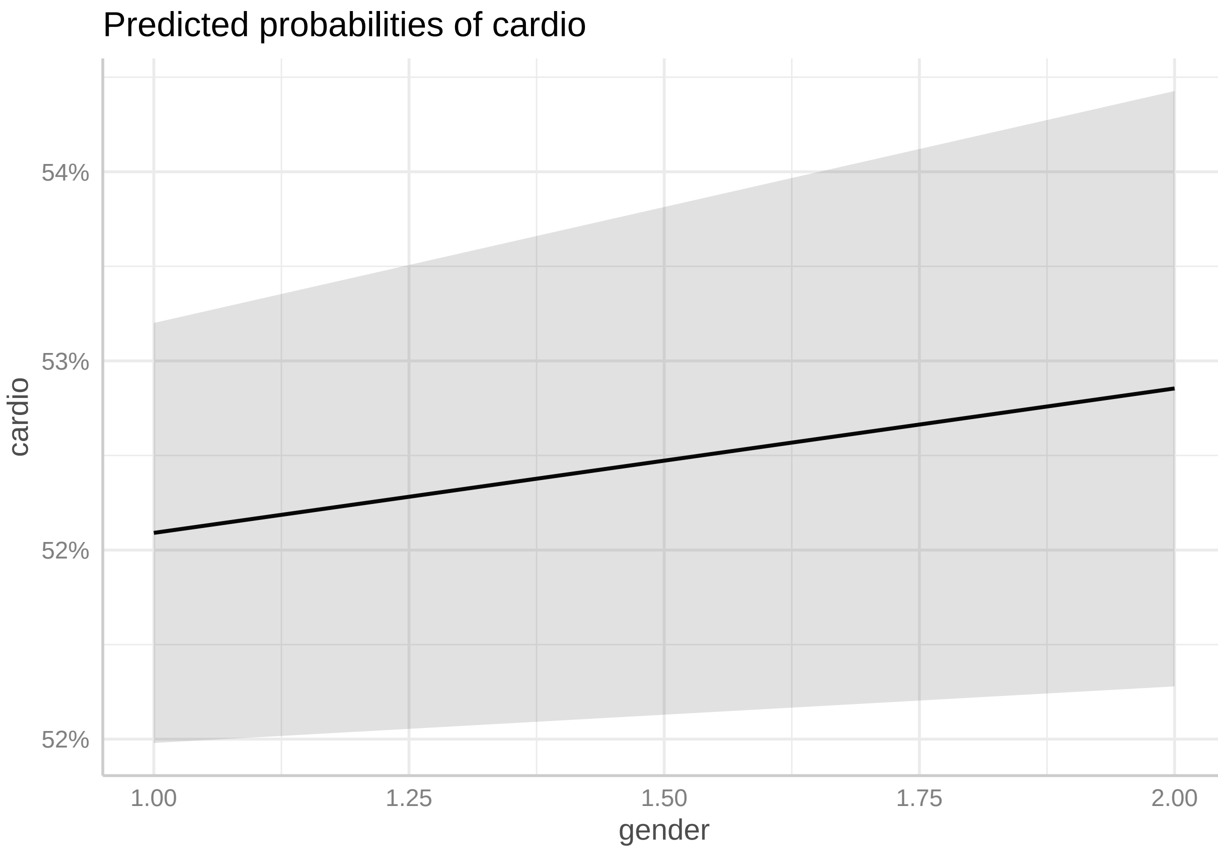
```
## [1] 0.7213143
```

```r
plot(ggpredict(logit.model,"age"))
```

```
## Data were 'prettified'. Consider using `terms="age [all]"` to get smooth
##   plots.
```

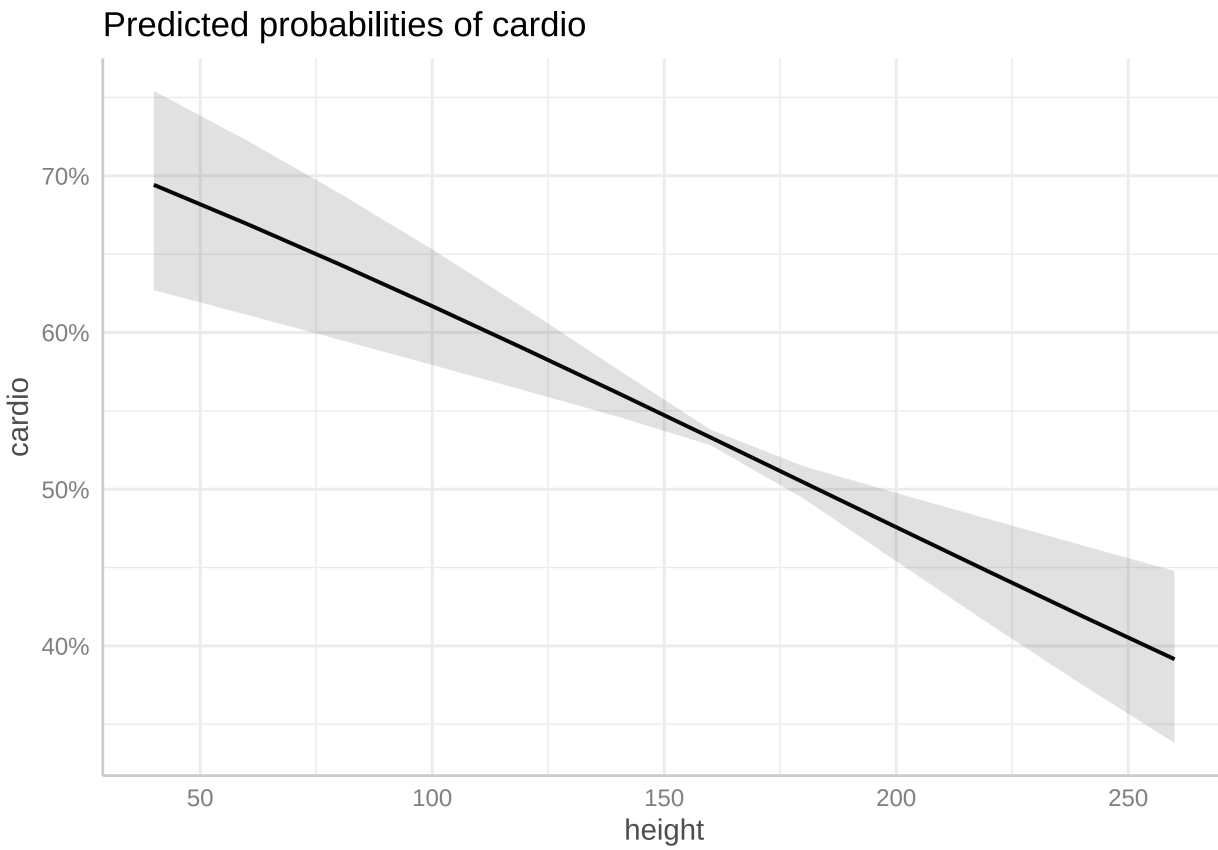



```
plot(ggpredict(logit.model,"gender"))
```



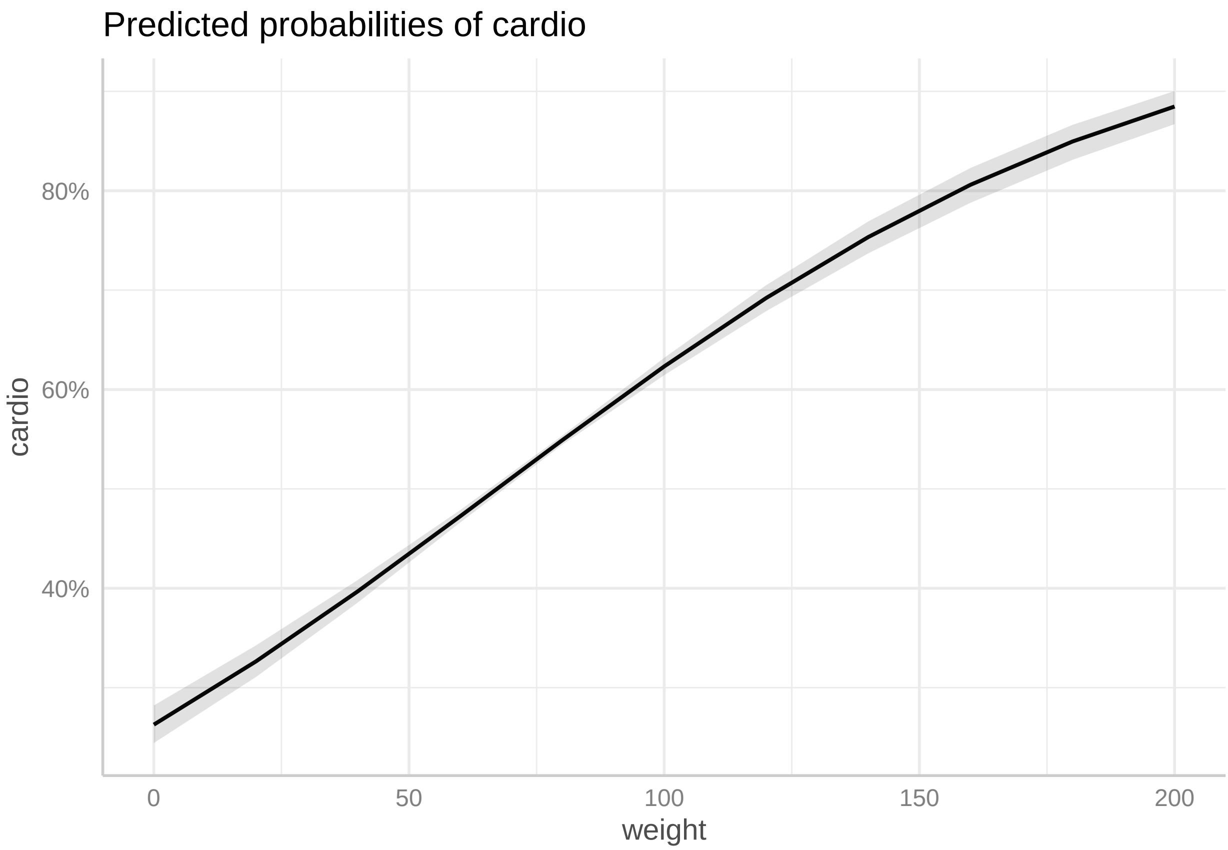

```
plot(ggpredict(logit.model,"height"))
```

```
## Data were 'prettified'. Consider using `terms="height [all]"` to get
##   smooth plots.
```



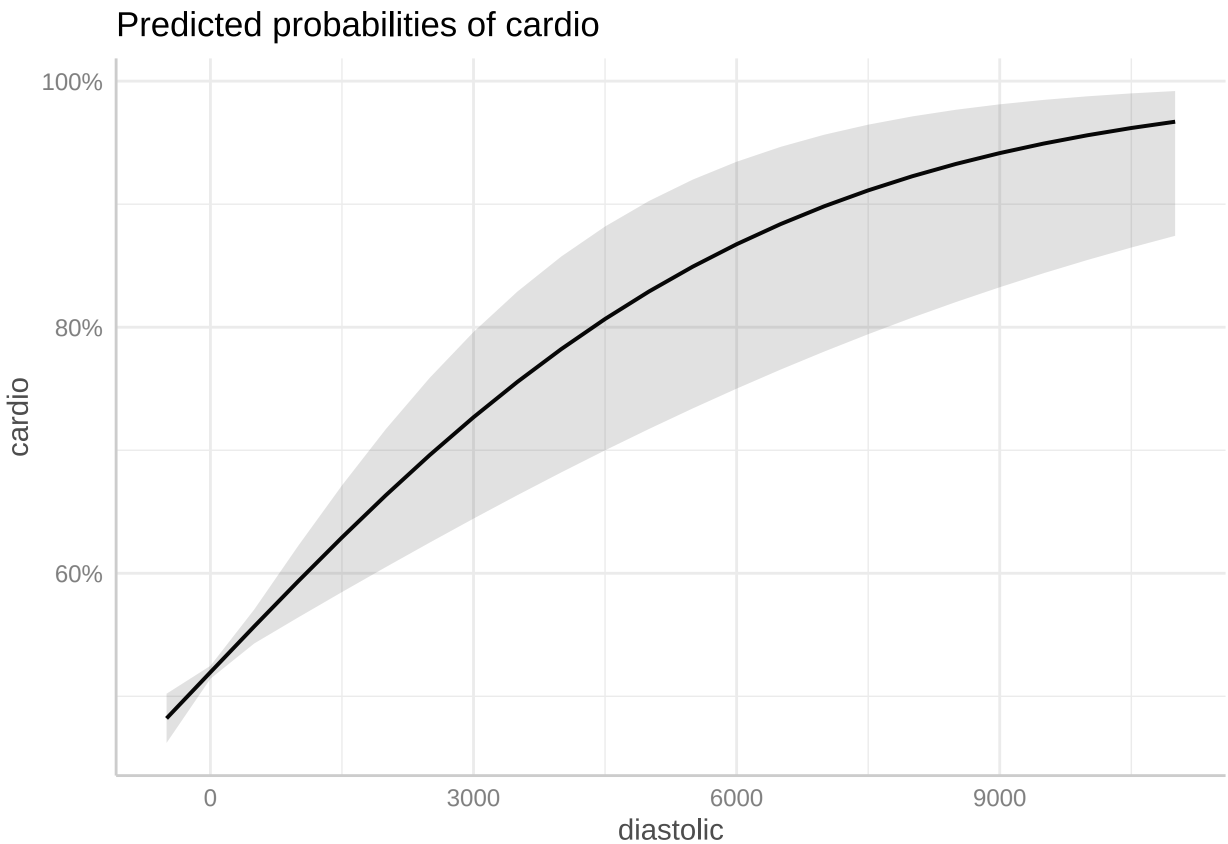

```
plot(ggpredict(logit.model,"weight"))
```

```
## Data were 'prettified'. Consider using `terms="weight [all]"` to get
##   smooth plots.
```



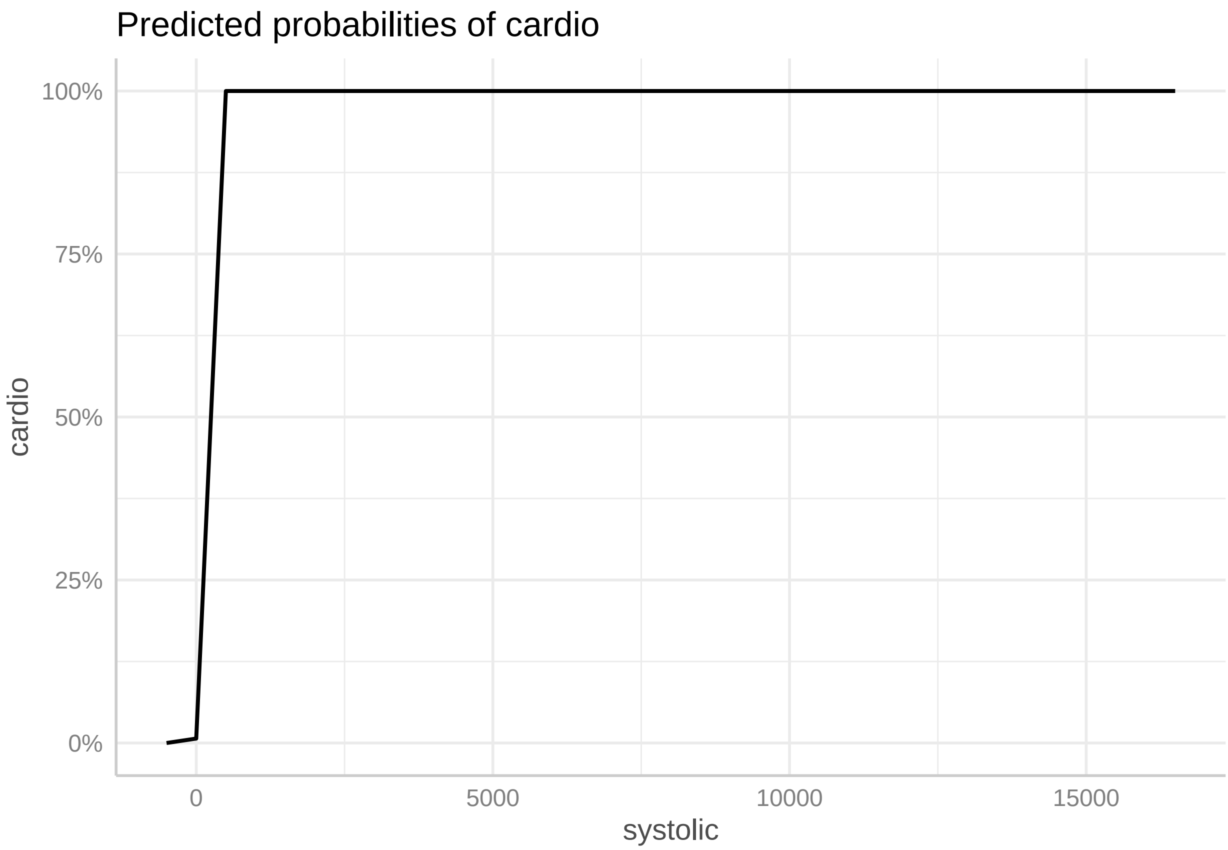

```
plot(ggpredict(logit.model,"diastolic"))
```

```
## Data were 'prettified'. Consider using `terms="diastolic [all]"` to get
##   smooth plots.
```

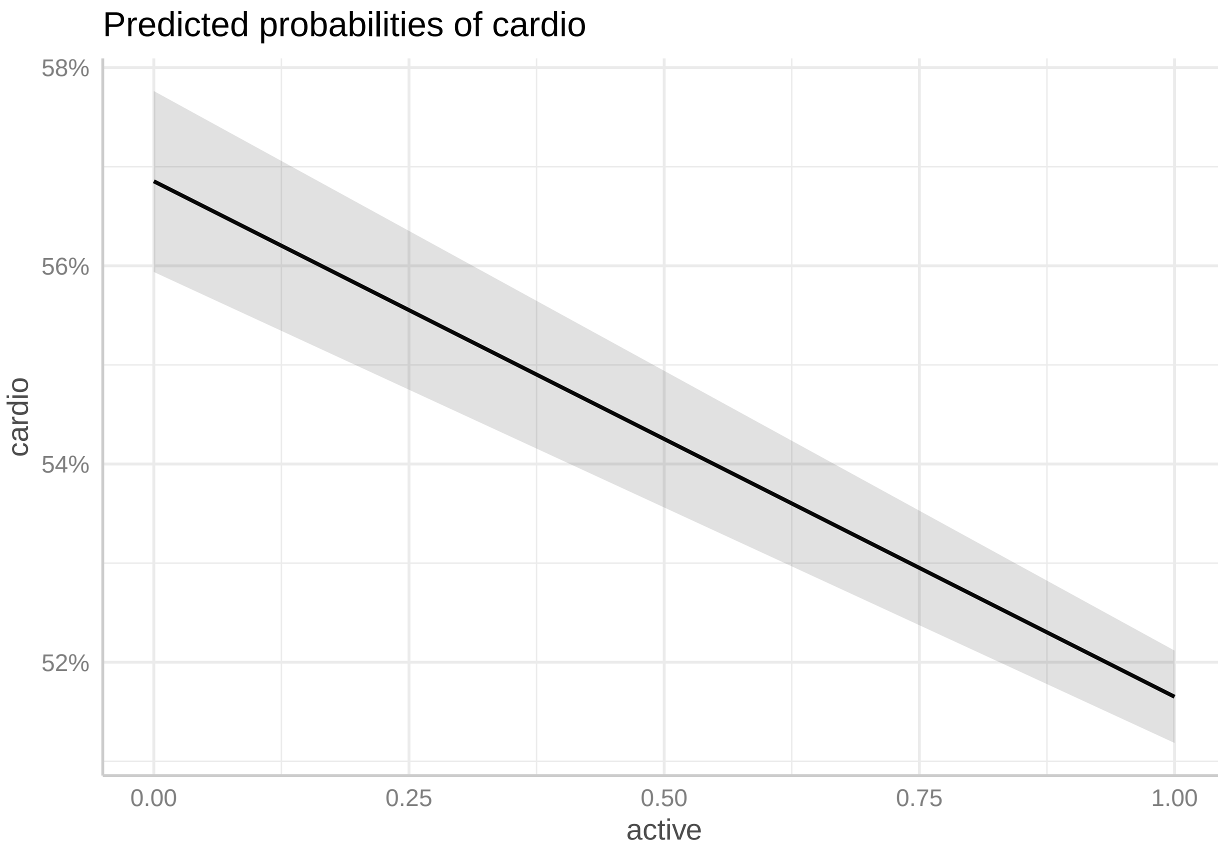



```
plot(ggpredict(logit.model,"systolic"))
```

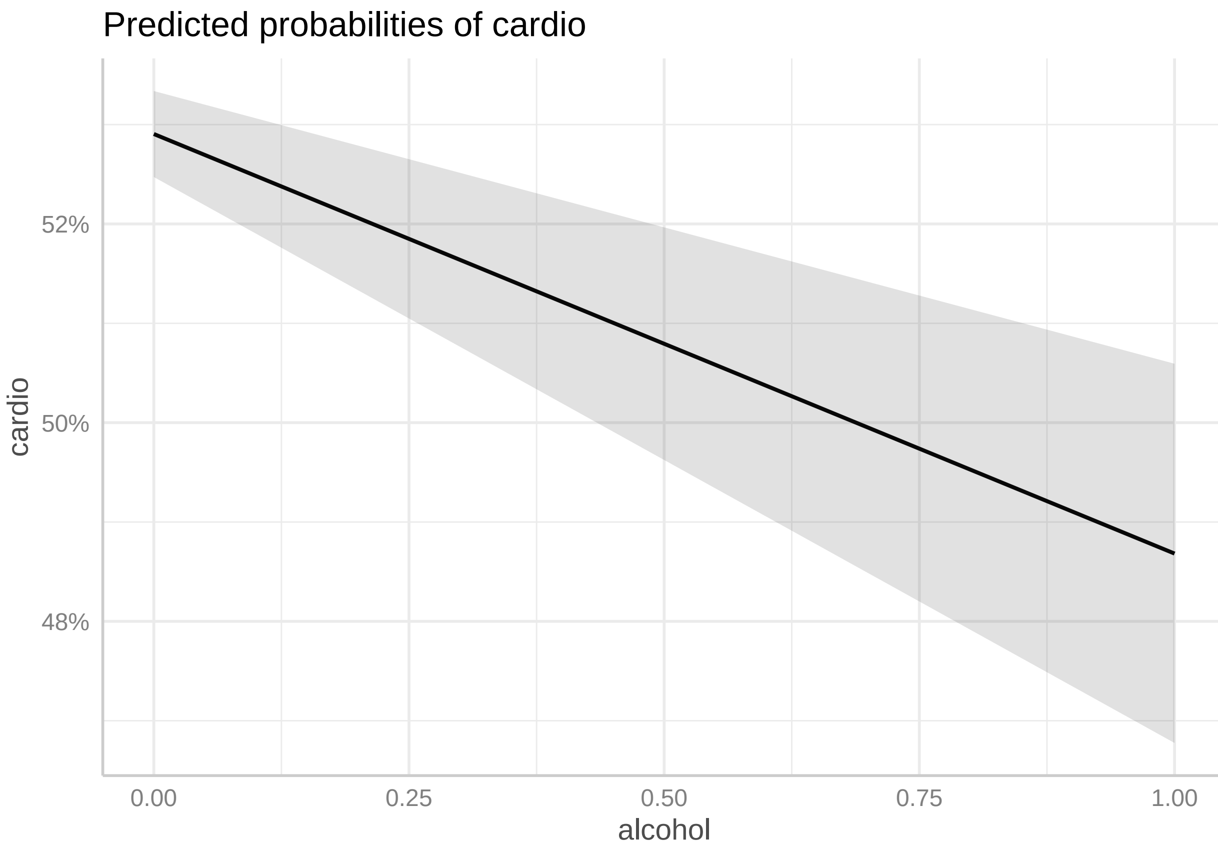
```
## Data were 'prettified'. Consider using `terms="systolic [all]"` to get
##   smooth plots.
```



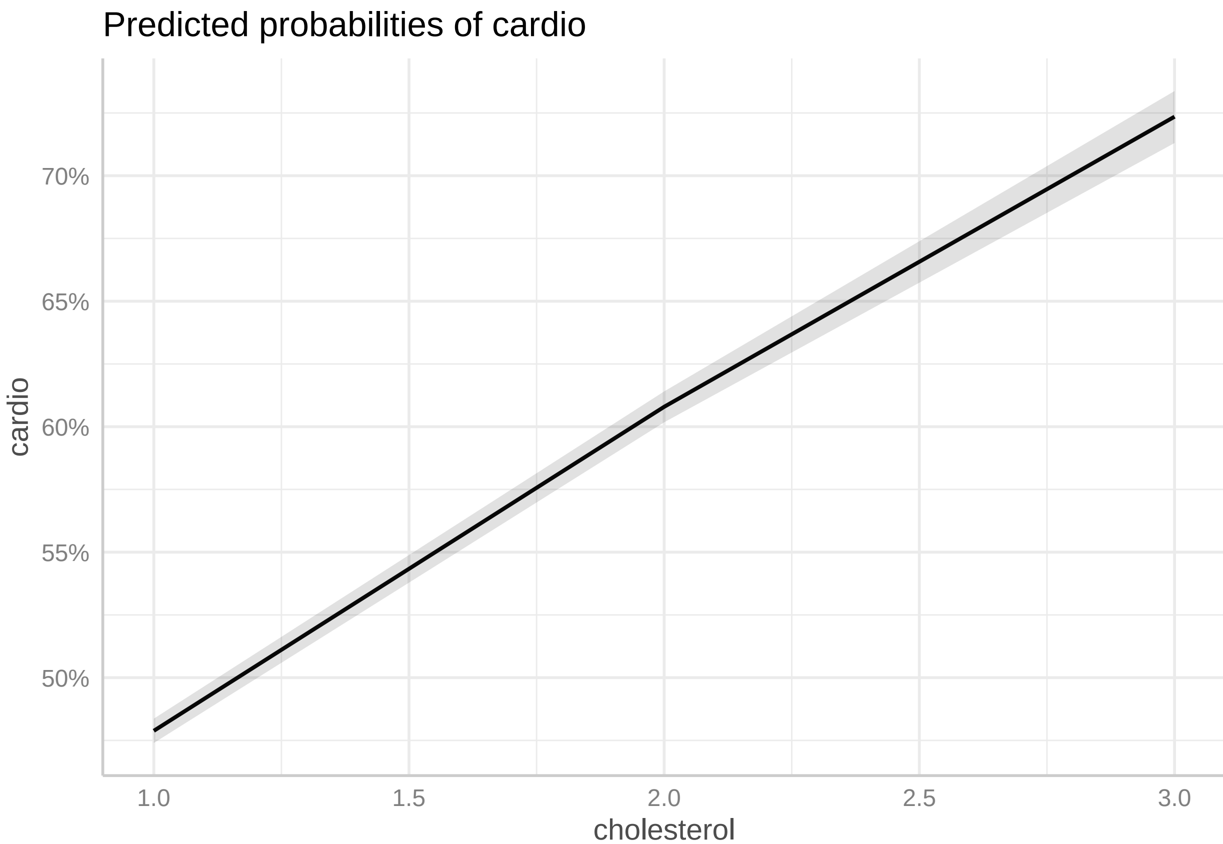

```
plot(ggpredict(logit.model,"active"))
```



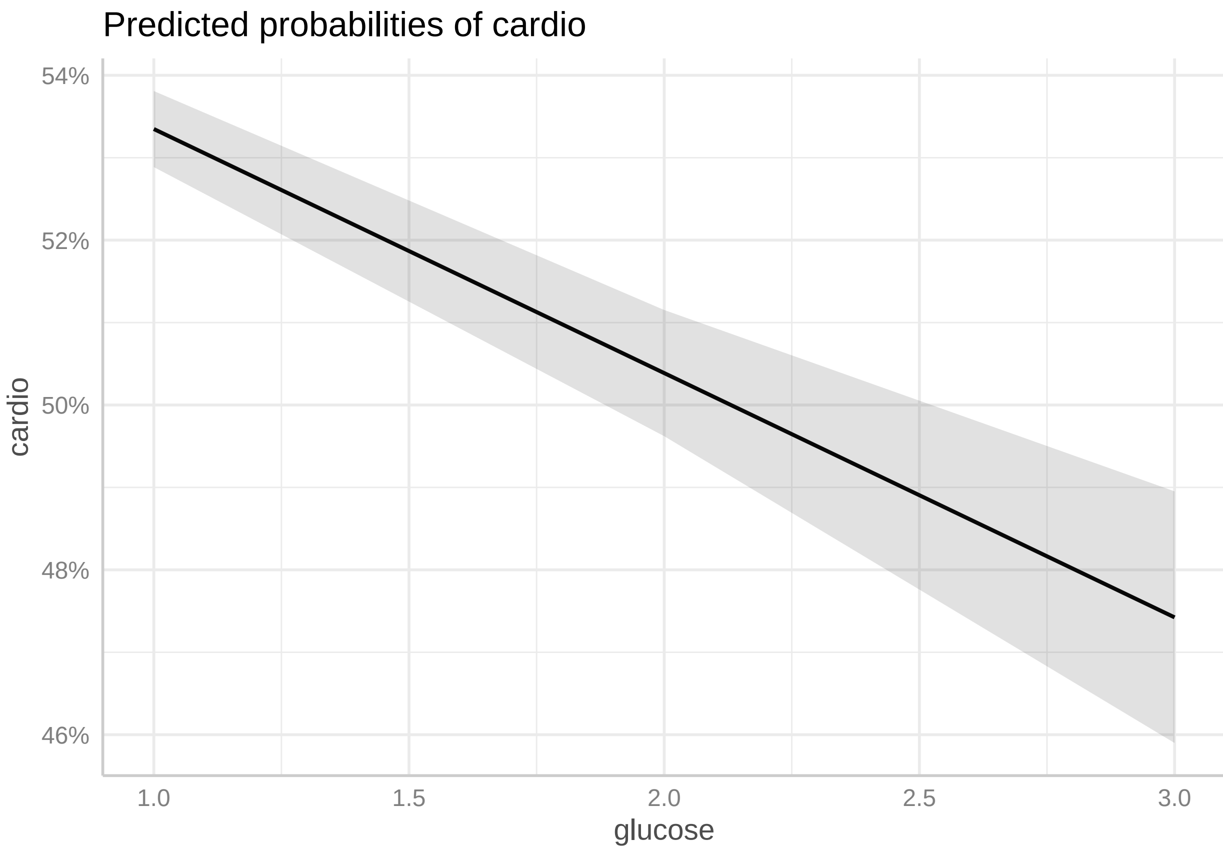

```
plot(ggpredict(logit.model,"alcohol"))
```



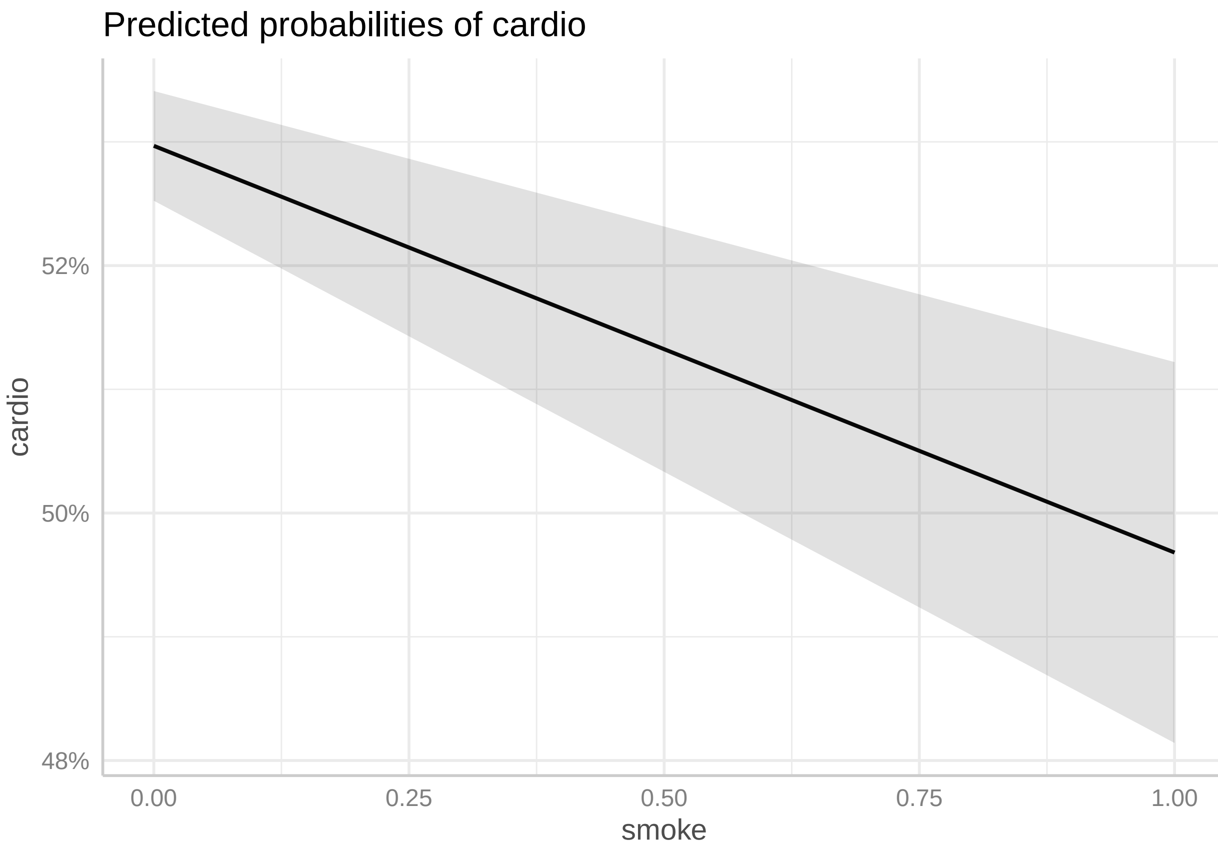

```
plot(ggpredict(logit.model,"cholesterol"))
```




```
plot(ggpredict(logit.model,"glucose"))
```




```
plot(ggpredict(logit.model,"smoke"))
```




```
#Looking at the logit predicted values, we saw that the predicted values for cardiovascular
#disease was a couple % points larger for height at 150kg and had around the same
#predicted value around 0.40 for 250cm or greater, with the same error band for
#confidence interval. The accuracy for weight is around the same as both linear
#probability and Probit model.




#Sensitivity Specificity
inTraining <- createDataPartition(cardio, p = 0.75, list = FALSE)
training <- heart[inTraining,]
testing <- heart[-inTraining,]

train_control <- trainControl(method = "cv", number = 5)

probit_model <- train(as.factor(cardio)~., data = training, method = "glm", family = "binomial", trCont
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
pred_cardio = predict(probit_model, newdata = testing)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
confusionMatrix(data = pred_cardio, reference=as.factor(testing$cardio))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 6788 2797
##          1 1986 5929
##
##                Accuracy : 0.7267
##                  95% CI : (0.72, 0.7333)
##     No Information Rate : 0.5014
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4532
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.7736
##             Specificity : 0.6795
##          Pos Pred Value : 0.7082
##          Neg Pred Value : 0.7491
##              Prevalence : 0.5014
##          Detection Rate : 0.3879
##    Detection Prevalence : 0.5477
##       Balanced Accuracy : 0.7266
##
##        'Positive' Class : 0
##
```

```
#After testing all three models, we found that the probit model was the preferred
#model as it was able to predict cardiovascular disease slightly better than the logit
#model. When we ran the prediction model for all three models the probit models gave
#more true zeros and ones. When we look at the confusion matrix we see that the accuracy
#of the testing data was slightly lower than the baseline accuracy. The model is a
#borderline good model as there were over 70,000 observations in the data set. With
#such a large sample size we could say that the model was good, but it is lower than
#the baseline accuracy. This would also make it a poor model as it did not beat the
```

```r
#accuracy of the dominant class baseline accuracy. Our model is able to predict the one
#class at 76% accuracy, which is higher than the baseline accuracy. For the zero class the
#accuracy is lower than the baseline accuracy. This would mean that the model is better at
#being able to predict whether somebody has cardiovascular disease compared to somebody
#not having cardiovascular disease.
```

```r
#Question 2 Part 4
mean(age)
```

```
## [1] 53.33949
```

```r
mean(active)
```

```
## [1] 0.8037286
```

```r
mean(alcohol)
```

```
## [1] 0.05377143
```

```r
mean(cardio)
```

```
## [1] 0.4997
```

```r
mean(cholesterol)
```

```
## [1] 1.366871
```

```r
mean(diastolic)
```

```
## [1] 96.63041
```

```r
mean(gender)
```

```
## [1] 0.6504286
```

```r
mean(glucose)
```

```
## [1] 1.226457
```

```r
mean(height)
```

```
## [1] 164.3592
```

```r
mean(smoke)
```

```
## [1] 0.08812857
```

```r
mean(systolic)
```

```
## [1] 128.8173
```

```r
mean(weight)
```

```
## [1] 74.20569
```

```r
#all variables mean
pred_cardio.mean <- data.frame(cardio=0.50, age=53.34, active=.80, alcohol=0.05,
                               cholesterol=1.37, diastolic=96.63, gender=0.65, glucose=1.23,
                               height=164.36, smoke=0.09, systolic=128.82, weight=74.21)
predict(probit.model, pred_cardio.mean, type="response", se.fit=TRUE)
```

```
## $fit
##          1
## 0.04343607
```

```
##
## $se.fit
##           1
## 0.003931662
##
## $residual.scale
## [1] 1
```

```r
#10% increase
pred_cardio.mean10 <- data.frame(cardio=0.50, age=58.67, active=.88, alcohol=0.055,
                                 cholesterol=1.51, diastolic=106.29, gender=0.72, glucose=1.35,
                                 height=180.80, smoke=0.10, systolic=141.70, weight=81.63)
predict(probit.model, pred_cardio.mean10, type="response", se.fit=TRUE)
```

```
## $fit
##          1
## 0.08632775
##
## $se.fit
##           1
## 0.007118865
##
## $residual.scale
## [1] 1
```

```r
#20% increase
pred_cardio.mean20 <- data.frame(cardio=0.50, age=64.01, active=.96, alcohol=0.06,
                                 cholesterol=1.61, diastolic=115.96, gender=0.78, glucose=1.48,
                                 height=197.23, smoke=0.11, systolic=154.58, weight=89.05)
predict(probit.model, pred_cardio.mean20, type="response", se.fit=TRUE)
```

```
## $fit
##         1
## 0.1518603
##
## $se.fit
##          1
## 0.01193075
##
## $residual.scale
## [1] 1
```

```r
#30%increase
pred_cardio.mean30 <- data.frame(cardio=0.50, age=69.34, active=1, alcohol=0.07,
                                 cholesterol=1.78, diastolic=125.62, gender=0.85, glucose=1.60,
                                 height=213.67, smoke=0.12, systolic=167.47, weight=96.47)
predict(probit.model, pred_cardio.mean30, type="response", se.fit=TRUE)
```

```
## $fit
##         1
## 0.2527153
##
## $se.fit
##         1
## 0.0186626
##
```

```
## $residual.scale
## [1] 1
```

```
#We used the logit model as a preferred model and for the first prediction we looked at
#the chance of getting cardiovascular disease given all the variables were set at the mean.
#We found that the chance was four percent, so we decided to increase the means by ten
#percent for each following prediction. We found that as we increased the means of all
#the variables that the chance of getting cardiovascular disease.  This makes sense as
#things like blood pressure and weight increase are known to increase the chance of some
#type of cardiovascular disease. However, we could have changed the values of different
#predictors, by increasing some of them, while decreasing others, to see if it would
#improve the accuracy of our prediction, and maybe the predictions may have potentially
#decreased the chances of getting cardiovascular disease.
```