Inferences for Regression (Chapter 23)

Patrick Frenett, Vickie Ip, and Nicholas Horton (nhorton@amherst.edu)

July 17, 2017

Introduction and background

This document is intended to help describe how to undertake analyses introduced as examples in the Fourth Edition of *Intro Stats* (2013) by De Veaux, Velleman, and Bock. More information about the book can be found at http://wps.aw.com/aw_deveaux_stats_series. This file as well as the associated R Markdown reproducible analysis source file used to create it can be found at https://nhorton.people.amherst.edu/is4.

This work leverages initiatives undertaken by Project MOSAIC (http://www.mosaic-web.org), an NSF-funded effort to improve the teaching of statistics, calculus, science and computing in the undergraduate curriculum. In particular, we utilize the `mosaic` package, which was written to simplify the use of R for introductory statistics courses. A short summary of the R needed to teach introductory statistics can be found in the mosaic package vignettes (http://cran.r-project.org/web/packages/mosaic). A paper describing the mosaic approach was published in the *R Journal*: https://journal.r-project.org/archive/2017/RJ-2017-024.

Note that some of the figures in this document may differ slightly from those in the IS4 book due to small differences in datasets. However in all cases the analysis and techniques in R are accurate.

Chapter 23: Inferences for Regression

Section 23.1: The population and the sample

```
library(mosaic); library(readr)
BodyFat <- read_csv("https://nhorton.people.amherst.edu/sdm4/data/Body_fat_complete.csv")
```

```
dim(BodyFat)
```

```
## [1] 250  16
```

```
glimpse(BodyFat)
```

```
## Observations: 250
## Variables: 16
## $ Body Density <dbl> 1.0708, 1.0853, 1.0414, 1.0751, 1.0340, 1.0502, 1...
## $ PctBF        <dbl> 12.3, 6.1, 25.3, 10.4, 28.7, 20.9, 19.2, 12.4, 4....
## $ Age          <int> 23, 22, 22, 26, 24, 24, 26, 25, 25, 23, 26, 27, 3...
## $ Weight       <dbl> 154.25, 173.25, 154.00, 184.75, 184.25, 210.25, 1...
## $ Height       <dbl> 67.75, 72.25, 66.25, 72.25, 71.25, 74.75, 69.75, ...
## $ Neck         <dbl> 36.2, 38.5, 34.0, 37.4, 34.4, 39.0, 36.4, 37.8, 3...
## $ Chest        <dbl> 93.1, 93.6, 95.8, 101.8, 97.3, 104.5, 105.1, 99.6...
## $ Abdomen      <dbl> 85.2, 83.0, 87.9, 86.4, 100.0, 94.4, 90.7, 88.5, ...
## $ waist        <dbl> 33.54331, 32.67717, 34.60630, 34.01575, 39.37008,...
## $ Hip          <dbl> 94.5, 98.7, 99.2, 101.2, 101.9, 107.8, 100.3, 97....
## $ Thigh        <dbl> 59.0, 58.7, 59.6, 60.1, 63.2, 66.0, 58.4, 60.0, 6...
## $ Knee         <dbl> 37.3, 37.3, 38.9, 37.3, 42.2, 42.0, 38.3, 39.4, 3...
```

```
## $ Ankle        <dbl> 21.9, 23.4, 24.0, 22.8, 24.0, 25.6, 22.9, 23.2, 2...
## $ Bicep        <dbl> 32.0, 30.5, 28.8, 32.4, 32.2, 35.7, 31.9, 30.5, 3...
## $ Forearm      <dbl> 27.4, 28.9, 25.2, 29.4, 27.7, 30.6, 27.8, 29.0, 3...
## $ Wrist        <dbl> 17.1, 18.2, 16.6, 18.2, 17.7, 18.8, 17.7, 18.8, 1...
```

We can confirm the coefficients from the model on page 645.

```
BodyFatmod <- lm(PctBF ~ waist, data=BodyFat)
coef(BodyFatmod)
```

```
## (Intercept)       waist
##  -42.734134    1.699972
```

Section 23.2: Assumptions and conditions

We can regenerate the output and figures for the example on pages 647-651.

```
msummary(BodyFatmod)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -42.73413    2.71651  -15.73   <2e-16 ***
## waist         1.69997    0.07431   22.88   <2e-16 ***
##
## Residual standard error: 4.713 on 248 degrees of freedom
## Multiple R-squared:  0.6785, Adjusted R-squared:  0.6772
## F-statistic: 523.3 on 1 and 248 DF,  p-value: < 2.2e-16
```

```
rsquared(BodyFatmod)
```

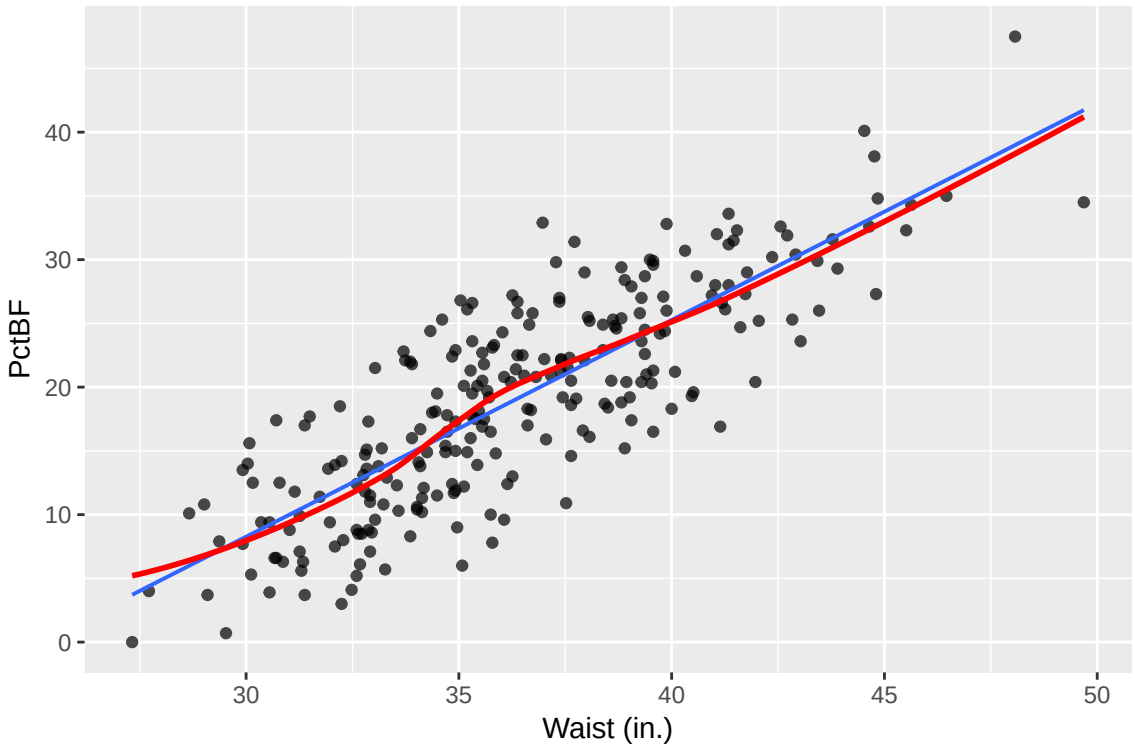
```
## [1] 0.6784564
```

```
confint(BodyFatmod)    # see page 755
```

```
##                  2.5 %    97.5 %
## (Intercept) -48.084497 -37.38377
## waist         1.553603   1.84634
```

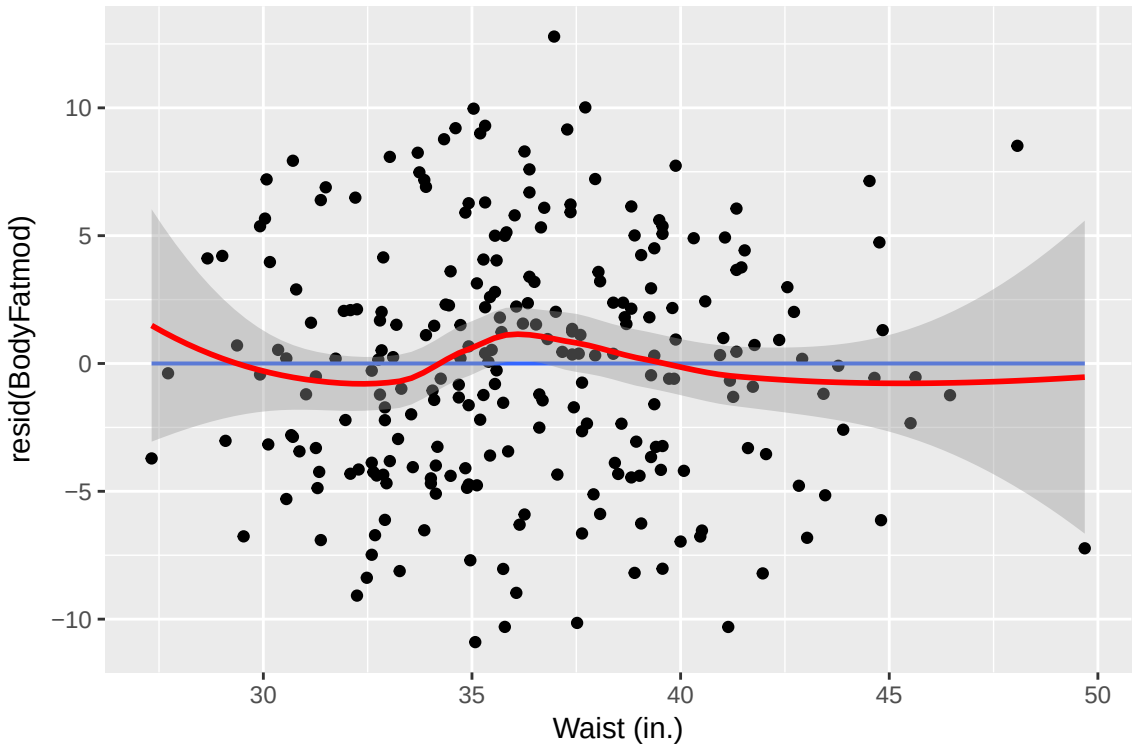
```
# Figure 23.4
gf_point(PctBF ~ waist,data=BodyFat,alpha=0.7) %>%
  gf_labs(x="Waist (in.)") %>%
  gf_lm() %>%
  gf_smooth(col="red", se=FALSE)
```

```
## `geom_smooth()` using method = 'loess'
```



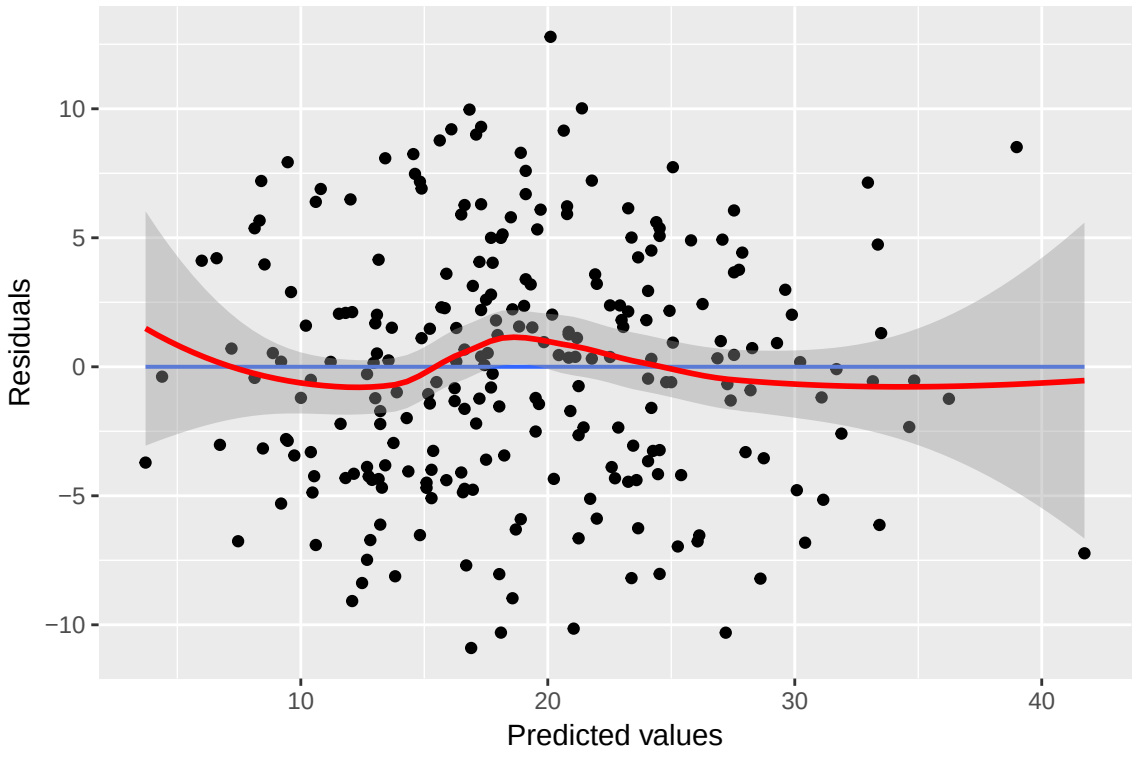

```
# Figure 23.5
gf_point(resid(BodyFatmod) ~ waist,data=BodyFat) %>%
  gf_labs(x="Waist (in.)") %>%
  gf_lm() %>%
  gf_smooth(col="red")
```

```
## `geom_smooth()` using method = 'loess'
```

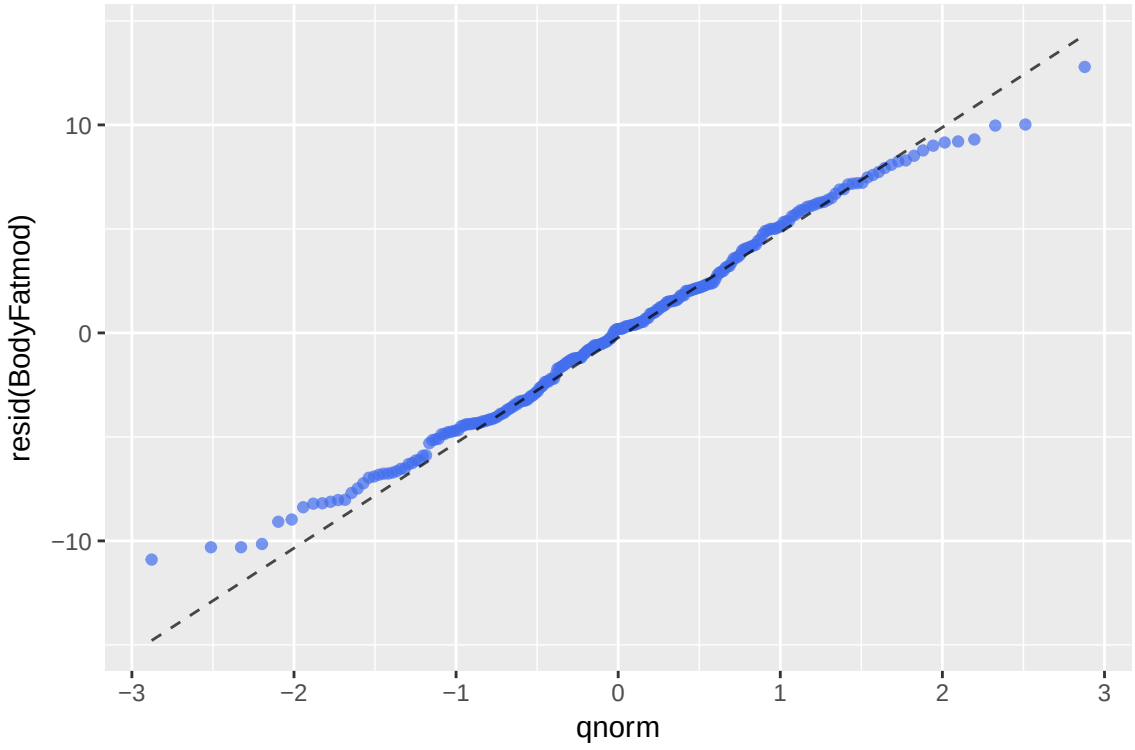



```
# equiv of Figure 23.6 note that Figure 23.6 refers to the diamonds dataset
gf_point(resid(BodyFatmod) ~ fitted(BodyFatmod),data=BodyFat) %>%
  gf_labs(x="Predicted values", y="Residuals") %>%
  gf_lm() %>%
  gf_smooth(col="red")
```

```
## `geom_smooth()` using method = 'loess'
```



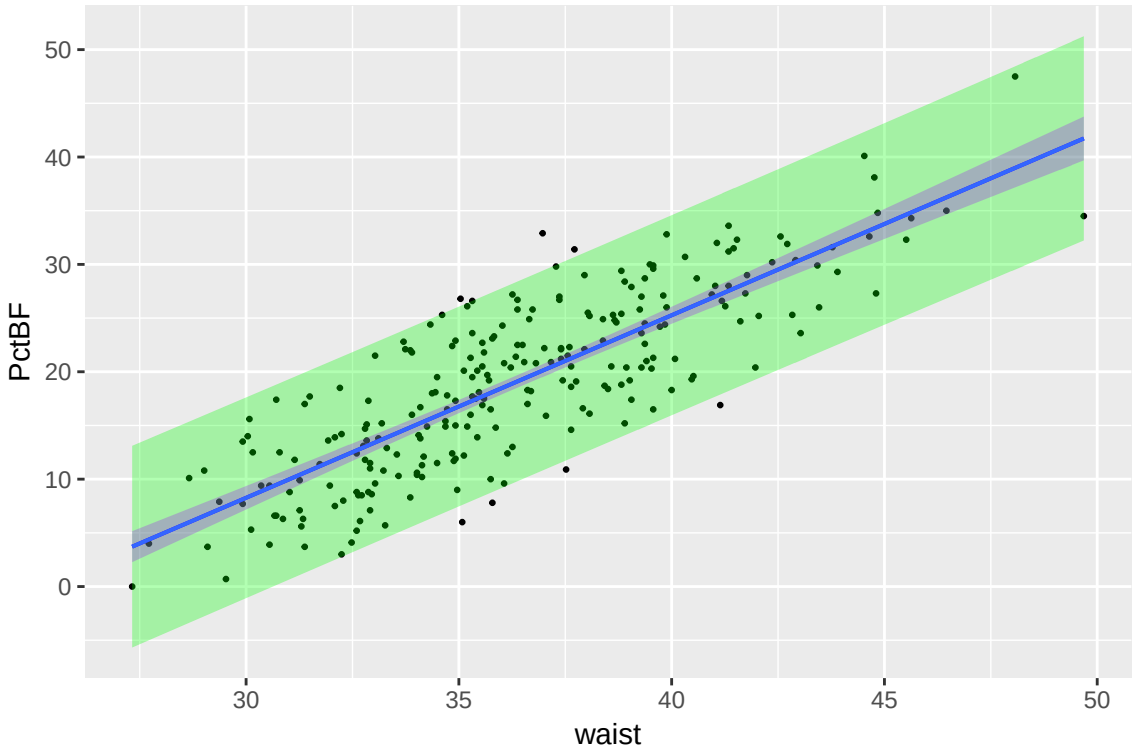

```
# Figure on bottom of page 650
gf_qq(~ resid(BodyFatmod), col="royalblue2", alpha=0.7) %>%
  gf_qqline() %>%
  gf_labs(x = "qnorm", y="resid(BodyFatmod)")
```



Section 23.6: Confidence intervals for predicted values

We can reproduce Figure 23.12 (page 662) using the `gf_lm(interval = )` function.

```
gf_point(PctBF ~ waist, data=BodyFat, cex=0.5) %>%
  gf_lm(interval = "prediction", col="blue", fill = "green") %>%
  gf_lm(interval = "confidence", fill = "purple")
```




```
Craters <- read.csv("https://nhorton.people.amherst.edu/sdm4/data/Craters.csv")
dim(Craters)
```

```
## [1] 168   4
```

```
Craters <- mutate(Craters,
                  logDiam = log(Diam.km.),
                  logAge = log(age..Ma.))
Cratermod <- lm(logDiam ~ logAge, data=Craters)
df_stats(~ logAge, data=Craters) # note example in book has n=39
```

```
##         min       Q1  median      Q3      max     mean       sd   n
## 1 -9.808177 3.608111 4.82391 5.94985 7.783224 3.761161 3.464722 168
##   missing
## 1       0
```

```
confpred <- predict(Cratermod, interval="confidence")
intpred <- predict(Cratermod, interval="prediction")
```

```
## Warning in predict.lm(Cratermod, interval = "prediction"): predictions on current data refer to _futu
```

```
select(Craters, -Name) %>% head(., 3)
```

```
##                             Location Diam.km. age..Ma.   logDiam
## 1                      Kansas, U.S.A.    0.015  1.0e-03 -4.199705
## 2 Western Australia,        Australia    0.024  2.7e-01 -3.729701
## 3                             Russia    0.027  5.5e-05 -3.611918
##      logAge
## 1 -6.907755
## 2 -1.309333
## 3 -9.808177
```

```
head(confpred, 3)
```

```
##           fit        lwr        upr
## 1 -2.15354532 -2.7661543 -1.5409363
## 2 -0.06385606 -0.3990184  0.2713063
## 3 -3.23616841 -4.0010756 -2.4712612
```

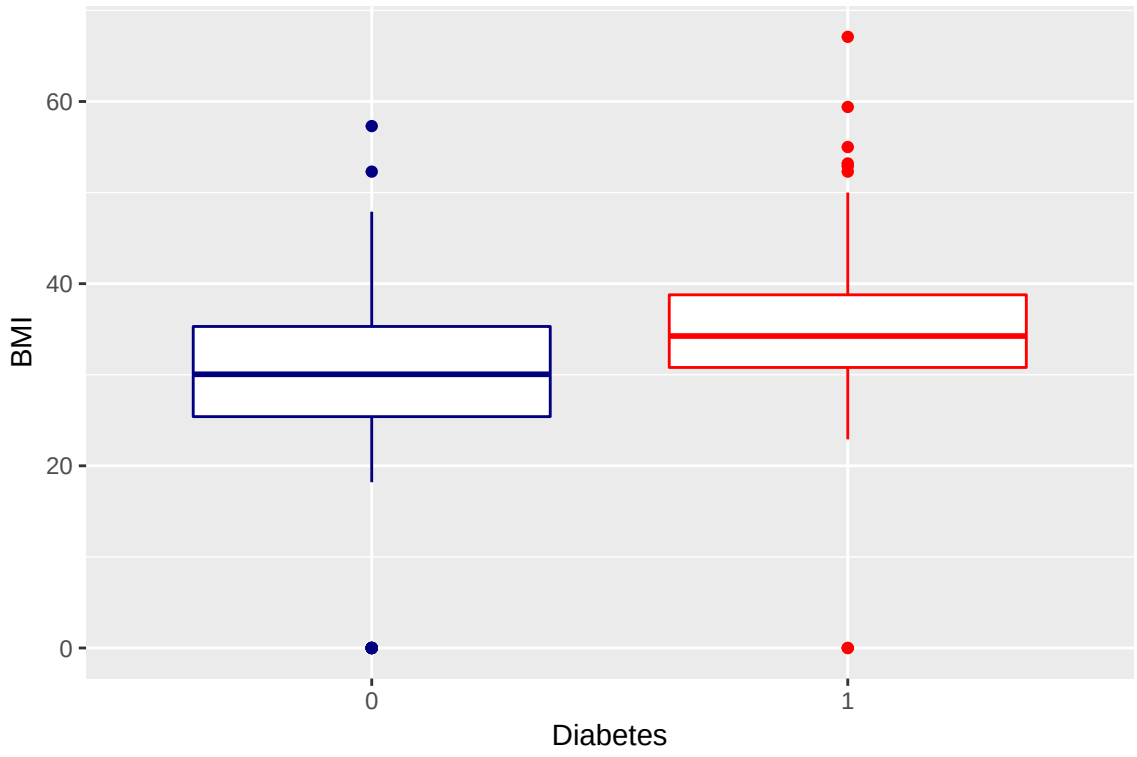
```
head(intpred, 3)
```

```
##           fit       lwr        upr
## 1 -2.15354532 -4.675010  0.3679190
## 2 -0.06385606 -2.532626  2.4049141
## 3 -3.23616841 -5.798897 -0.6734403
```

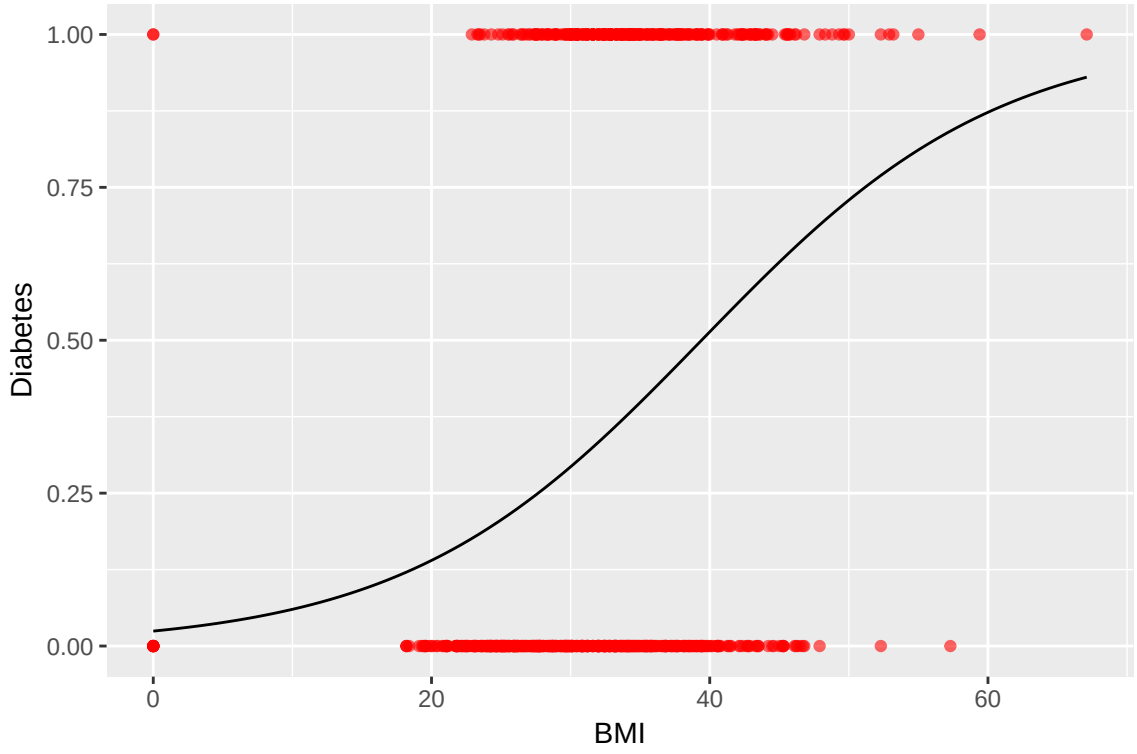
Section 23.7: Logistic regression

The Pima Indian dataset example is given on pages 663-667.

```
Pima <- read_csv("https://nhorton.people.amherst.edu/sdm4/data/Pima_Indians_Diabetes.csv")
Diabetes <- filter(Pima, BMI>0)  # get rid of missing values for BMI
gf_boxplot(BMI ~ as.factor(Diabetes), col = c("navy", "red"), data=Pima) %>%
  gf_labs(x="Diabetes")
```




```
pimamod <- glm(Diabetes ~ BMI, family="binomial", data=Pima)
f2 <- makeFun(pimamod)
gf_point(Diabetes ~ BMI, data=Pima, alpha=0.6, col="red") %>%
  gf_fun(pimamod)
```



```r
msummary(pimamod)
```

```
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.68641    0.40896  -9.014  < 2e-16 ***
## BMI          0.09353    0.01205   7.761 8.45e-15 ***
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 920.71  on 766  degrees of freedom
## AIC: 924.71
##
## Number of Fisher Scoring iterations: 4
```