Statistics 135 – Lab Project

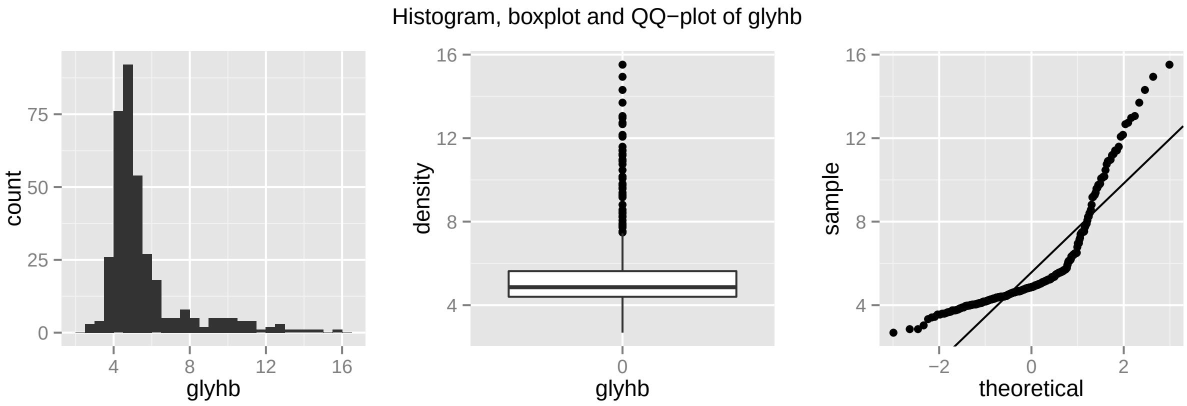
April 29, 2015

1 Background

```
library("ggplot2")
library("grid")
library("gridExtra")

data.df <- na.omit(read.csv("diabetes.csv"))
```

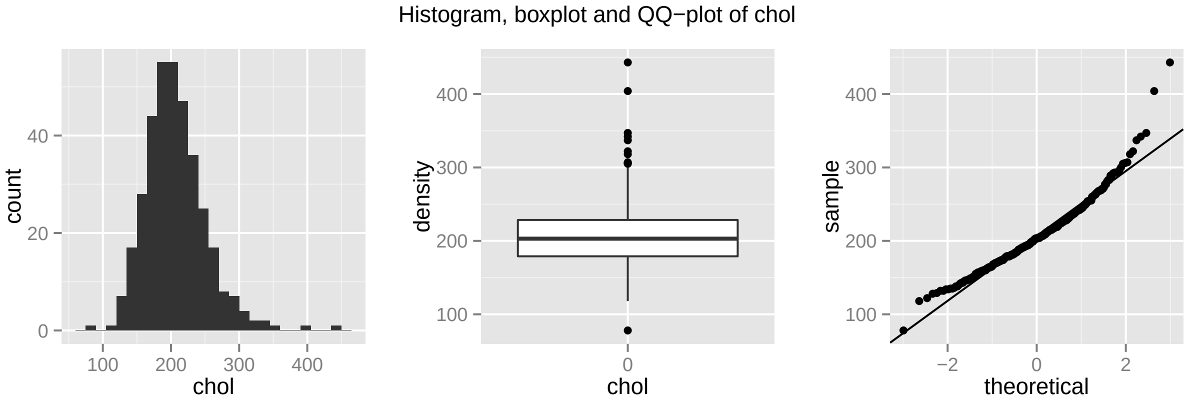
2 Accessing Data, Visualization and Summarization

1.
```
glyhb.histogram <- ggplot(data.df) +
  geom_histogram(aes(x=glyhb), binwidth=0.5)

glyhb.boxplot <- ggplot(data.df) +
  geom_boxplot(aes(x=factor(0), y=glyhb)) +
  labs(x="glyhb", y="density")

glyhb.qqplot <- ggplot(data.df) +
  stat_qq(aes(sample=glyhb)) +
  geom_abline(aes(intercept=mean(glyhb), slope=sd(glyhb)))

grid.arrange(glyhb.histogram, glyhb.boxplot, glyhb.qqplot, ncol=3,
             main="Histogram, boxplot and QQ-plot of glyhb")
```



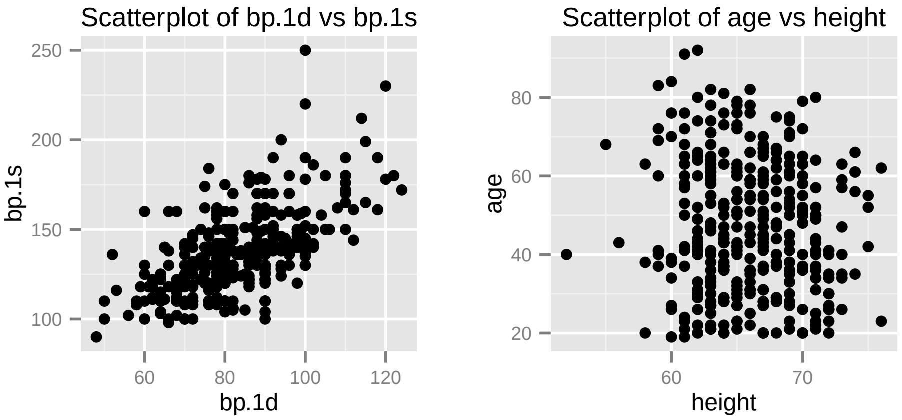
2.
```
chol.histogram <- ggplot(data.df) +
  geom_histogram(aes(x=chol), binwidth=15)

chol.boxplot <- ggplot(data.df) +
```

```r
  geom_boxplot(aes(x=factor(0), y=chol)) +
  labs(x="chol", y="density")

chol.qqplot <- ggplot(data.df) +
  stat_qq(aes(sample=chol)) +
  geom_abline(aes(intercept=mean(chol), slope=sd(chol)))

grid.arrange(chol.histogram, chol.boxplot, chol.qqplot, ncol=3,
             main="Histogram, boxplot and QQ-plot of chol")
```



3.

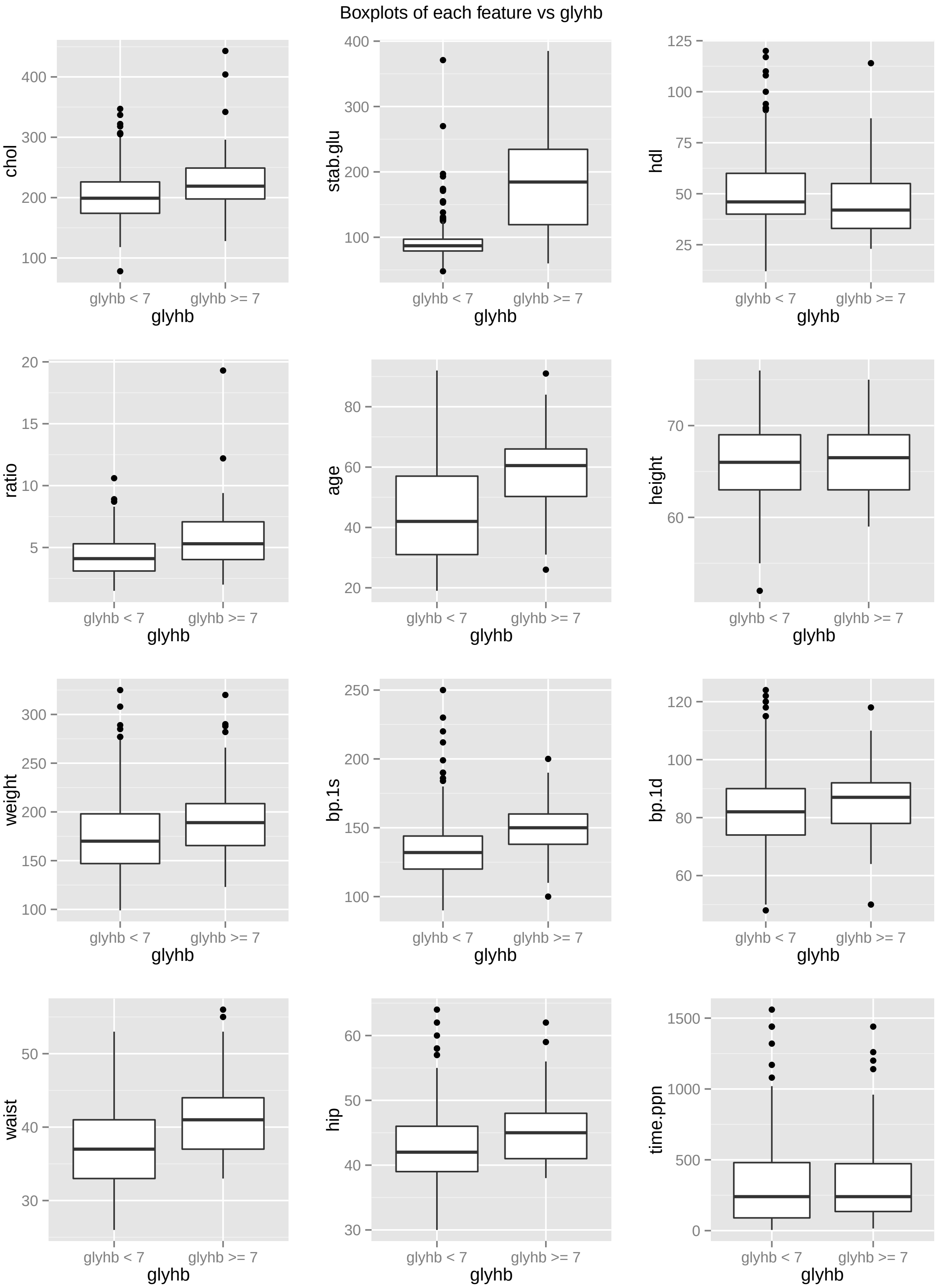
```r
bp.1d.bp.1s.scatterplot <- ggplot(data.df) +
  geom_point(aes(x=bp.1d, y=bp.1s)) +
  labs(title="Scatterplot of bp.1d vs bp.1s") +
  theme(text=element_text(size=8.5))

height.age.scatterplot <- ggplot(data.df) +
  geom_point(aes(x=height, y=age)) +
  labs(title="Scatterplot of age vs height") +
  theme(text=element_text(size=8.5))

grid.arrange(bp.1d.bp.1s.scatterplot, height.age.scatterplot, ncol=2)
```



4.

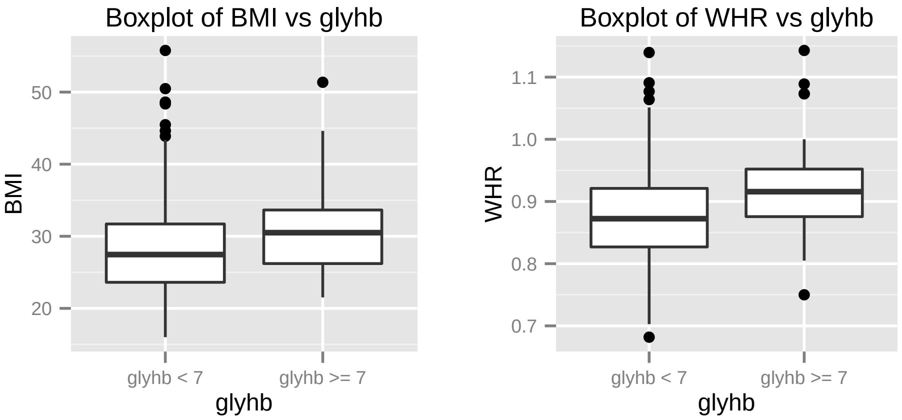
```r
data.df$glyhb.cond <- NA
data.df[data.df$glyhb>=7, ]$glyhb.cond <- "glyhb >= 7"
data.df[data.df$glyhb<7, ]$glyhb.cond <- "glyhb < 7"
```

```r
features <- c("chol", "stab.glu", "hdl", "ratio", "age", "height", "weight",
              "bp.1s", "bp.1d", "waist", "hip", "time.ppn")
feature.boxplots = list()
for (feature in features) {
  feature.boxplot <- ggplot(data.frame(glyhb.cond=data.df$glyhb.cond,
                                       feature=data.df[[feature]])) +
    geom_boxplot(aes(x=glyhb.cond, y=feature)) +
    labs(x="glyhb", y=feature)
  feature.boxplots[[feature]] <- feature.boxplot
}

do.call(grid.arrange, c(feature.boxplots, ncol=3,
                        main="Boxplots of each feature vs glyhb"))
```



5.

```
data.df$BMI <- 703*data.df$weight/data.df$height^2
data.df$WHR <- data.df$waist/data.df$hip

features <- c("BMI", "WHR")
feature.boxplots = list()
for (feature in features) {
  feature.boxplot <- ggplot(data.frame(glyhb.cond=data.df$glyhb.cond,
                                       feature=data.df[[feature]])) +
    geom_boxplot(aes(x=glyhb.cond, y=feature)) +
    labs(title=paste(c("Boxplot of", feature, "vs glyhb"), collapse=" "),
         x="glyhb", y=feature) +
    theme(text=element_text(size=8.5))
  feature.boxplots[[feature]] <- feature.boxplot
}

do.call(grid.arrange, c(feature.boxplots, ncol=2))
```



6.

3 Parametric Inferece

1.

$$X \sim Gamma(\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$$

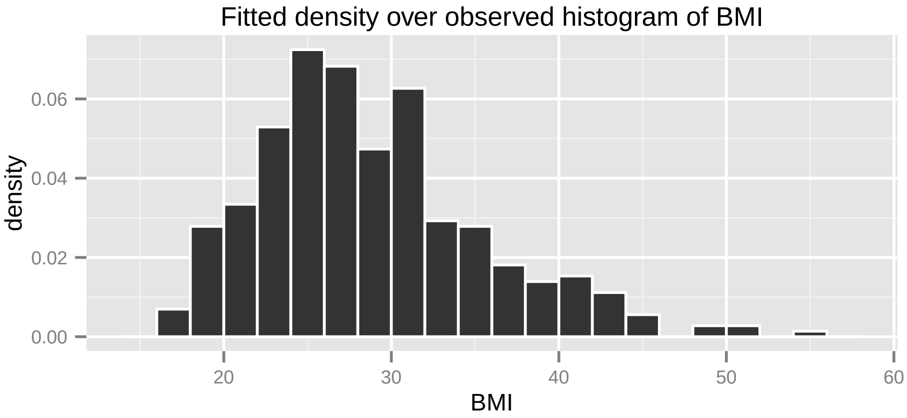
$$E(X) = \frac{\alpha}{\beta}$$

$$\begin{aligned} E(X^2) &= Var(X) + [E(X)]^2 \\ &= \frac{\alpha}{\beta^2} + \left(\frac{\alpha}{\beta}\right)^2 \\ &= \frac{\alpha(\alpha+1)}{\beta^2} \end{aligned}$$

$$\begin{cases} E(X) = \frac{\alpha}{\beta} \\ E(X^2) = \frac{\alpha(\alpha+1)}{\beta^2} \end{cases} \implies \begin{cases} \alpha = \frac{[E(X)]^2}{Var(x)} \\ \beta = \frac{E(X)}{Var(x)} \end{cases} \implies \begin{cases} \hat{\alpha}_{MOM} = \frac{\overline{X}_n}{\frac{1}{n}\sum_{i=1}^n (X_i - \overline{X}_n)^2} \\ \hat{\beta}_{MOM} = \frac{\overline{X}_n^2}{\frac{1}{n}\sum_{i=1}^n (X_i - \overline{X}_n)^2} \end{cases}$$

```
gamma.bootstrap <- function(x) {
  alpha.mom <- mean(x)^2/var(x)
  beta.mom <- mean(x)/var(x)

  bootstrap <- sapply(1:1000, function(i) {
    samples <- sample(x, length(x), replace=TRUE)
    alpha.sample <- mean(samples)^2/var(samples)
    beta.sample <- mean(samples)/var(samples)
    return(c(alpha.sample, beta.sample))
  })

  CIs <- sapply(1:2, function(i) {
    CI <- quantile(bootstrap[i, ], probs=c(0.025, 0.975))
    return(CI)
  })
  colnames(CIs) <- c("alpha", "beta")

  return(CIs)
}

CI.BMI <- gamma.bootstrap(data.df$BMI)

################################################################################
CI.BMI

##           alpha     beta
## 2.5%  15.87960 0.548766
## 97.5% 21.68973 0.757444

################################################################################
```

```
ggplot(data.df) +
  geom_histogram(aes(x=BMI, y=..density..), binwidth=2, col="white") +
  stat_function(fun=function(x)
    dgamma(x, shape=alpha.mom, rate=beta.mom), col="white") +
  labs(title="Fitted density over observed histogram of BMI") +
  theme(text=element_text(size=8.5))
```



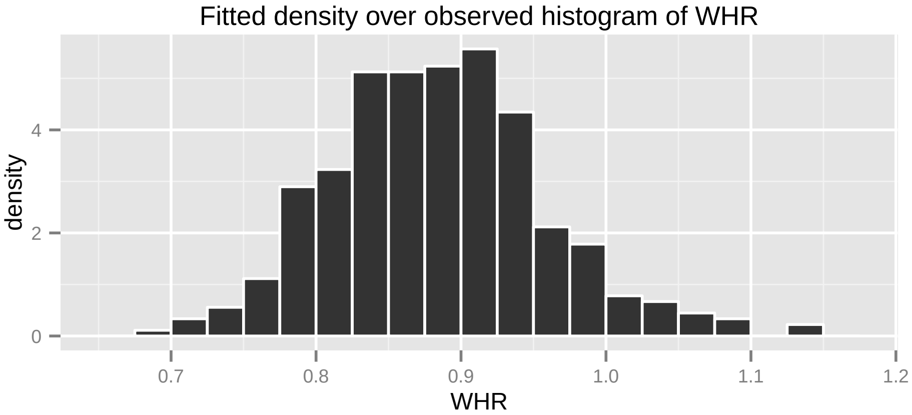
2.

$$\hat{\mu}_{MLE} = \overline{X}_n$$

$$\hat{\sigma}^2_{MLE} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X}_n)^2$$

```
normal.bootstrap <- function(x) {
  mu.mle <- mean(x)
  sigma.mle <- sd(x)

  bootstrap <- sapply(1:1000, function(i) {
    samples <- sample(x, length(x), replace=TRUE)
    mu.sample <- mean(samples)
    sigma.sample <- sd(samples)
    return(c(mu.sample, sigma.sample))
  })

  CIs <- sapply(1:2, function(i) {
    CI <- quantile(bootstrap[i, ], probs=c(0.025, 0.975))
    return(CI)
  })
  colnames(CIs) <- c("mu", "sigma")

  return(CIs)
}

CI.WHR <- normal.bootstrap(data.df$WHR)

################################################################################
CI.WHR

##              mu      sigma
## 2.5%  0.8742232 0.06755450
## 97.5% 0.8898956 0.08004401

################################################################################
```

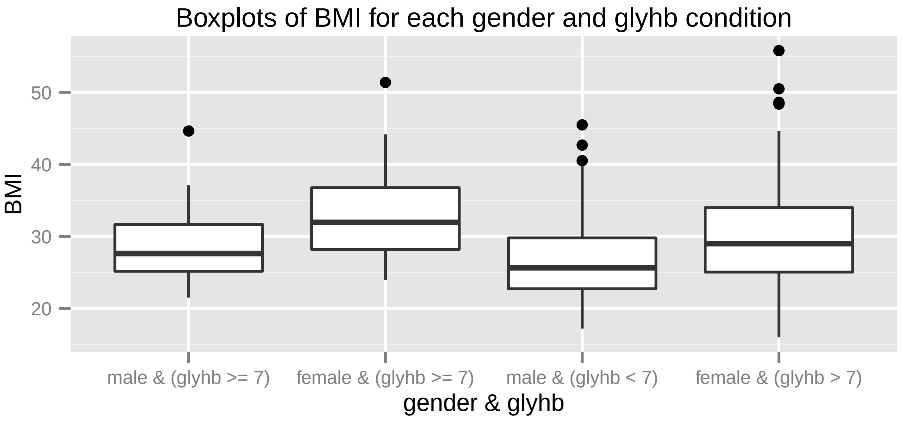
```
ggplot(data.df) +
  geom_histogram(aes(x=WHR, y=..density..), binwidth=0.025, col="white") +
  stat_function(fun=function(x)
    dnorm(x, mean=mu.mle, sd=sigma.mle), col="white") +
  labs(title="Fitted density over observed histogram of WHR") +
  theme(text=element_text(size=8.5))
```



3.

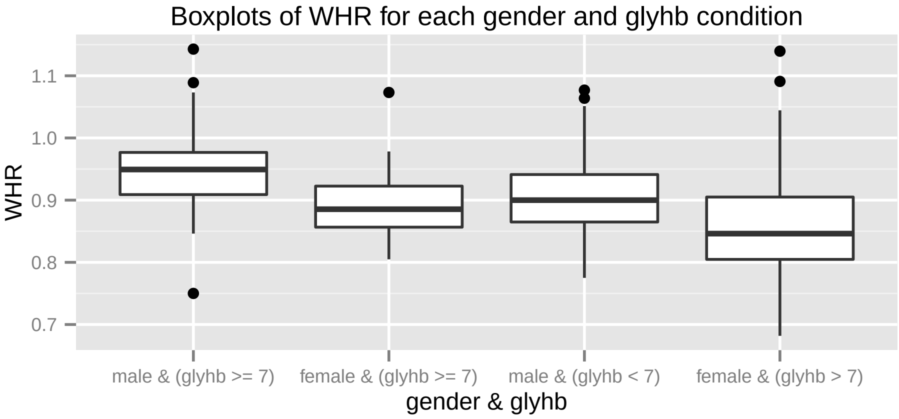
```
data.df$gender.glyhb.cond <- NA
data.df[data.df$gender=="male" & data.df$glyhb>=7, ]$gender.glyhb.cond <-
  "male & (glyhb >= 7)"
data.df[data.df$gender=="female" & data.df$glyhb>=7, ]$gender.glyhb.cond <-
  "female & (glyhb >= 7)"
data.df[data.df$gender=="male" & data.df$glyhb<7, ]$gender.glyhb.cond <-
  "male & (glyhb < 7)"
data.df[data.df$gender=="female" & data.df$glyhb<7, ]$gender.glyhb.cond <-
  "female & (glyhb < 7)"

conditions <- c("male & (glyhb >= 7)", "female & (glyhb >= 7)",
                "male & (glyhb < 7)", "female & (glyhb < 7)")

gamma.bootstrap2 <- function(x) {
  alpha.mom <- mean(x)^2/var(x)
  beta.mom <- mean(x)/var(x)

  bootstrap <- sapply(1:1000, function(i) {
    samples <- sample(x, length(x), replace=TRUE)
    mu.sample <- mean(samples)
    sigma.sample <- sd(samples)
    return(c(mu.sample, sigma.sample))
  })

  CIs <- sapply(1:2, function(i) {
    CI <- quantile(bootstrap[i, ], probs=c(0.025, 0.975))
    return(CI)
  })
  colnames(CIs) <- c("mu", "sigma")

  return(CIs)
}

CIs.BMI <- lapply(conditions, function(x) {
  gender.glyhb.df <- subset(data.df, gender.glyhb.cond==x)
  CIs.BMI <- gamma.bootstrap2(gender.glyhb.df$BMI)
  return(CIs.BMI)
})
names(CIs.BMI) <- conditions

################################################################################
CIs.BMI

## $`male & (glyhb >= 7)`
##             mu    sigma
## 2.5%  26.95963 3.418566
## 97.5% 30.80594 7.070036
##
## $`female & (glyhb >= 7)`
##             mu    sigma
## 2.5%  30.90487 4.660985
## 97.5% 35.69416 8.293779
##
## $`male & (glyhb < 7)`
##             mu   sigma
## 2.5%  25.48804 4.72097
## 97.5% 27.38754 6.40592
##
## $`female & (glyhb < 7)`
##             mu    sigma
## 2.5%  28.76716 6.216621
## 97.5% 30.76864 7.802709

################################################################################
```

```
ggplot(data.df) +
  geom_boxplot(aes(x=factor(gender.glyhb.cond, levels=conditions), y=BMI)) +
  labs(title="Boxplots of BMI for each gender and glyhb condition",
       x="gender & glyhb") +
  theme(text=element_text(size=8.5))
```



```
CIs.WHR <- lapply(conditions, function(x) {
  gender.glyhb.df <- subset(data.df, gender.glyhb.cond==x)
  CIs.WHR <- normal.bootstrap(gender.glyhb.df$WHR)
  return(CIs.WHR)
})
names(CIs.WHR) <- conditions

################################################################################
CIs.WHR

## $`male & (glyhb >= 7)`
##              mu      sigma
## 2.5%  0.9170921 0.04852926
## 97.5% 0.9807403 0.10435834
##
## $`female & (glyhb >= 7)`
##              mu      sigma
## 2.5%  0.8745839 0.04175829
## 97.5% 0.9151449 0.07790498
##
## $`male & (glyhb < 7)`
##               mu      sigma
## 2.5%  0.8945964 0.05586194
## 97.5% 0.9163876 0.07088215
##
## $`female & (glyhb < 7)`
##               mu      sigma
## 2.5%  0.8445274 0.06136025
## 97.5% 0.8646460 0.07883802

################################################################################
```

```
ggplot(data.df) +
  geom_boxplot(aes(x=factor(gender.glyhb.cond, levels=conditions), y=WHR)) +
  labs(title="Boxplots of WHR for each gender and glyhb condition",
       x="gender & glyhb") +
  theme(text=element_text(size=8.5))
```



4 Testing

1.

2.

3.

4.

5.

6.

5 Regression

1.

2.

3.

4.

5.

6.

7.