

# PDS-ASSIGNMENT\_3

## STEP-A

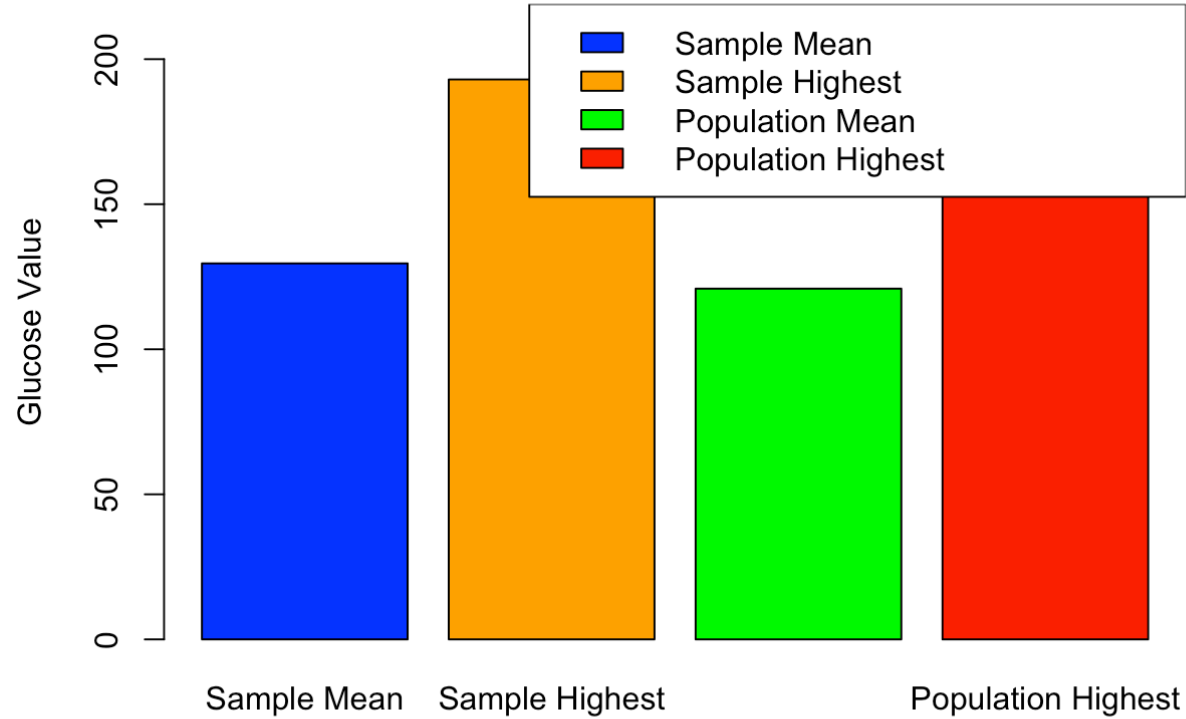
```
> library(readr)
> diabetes <- read_csv("~/Downloads/diabetes.csv")
Rows: 768 Columns: 9
— Column specification —————
Delimiter: ","
dbl (9): Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age,...
```

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.  
> View(diabetes)

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0
5	0	137	40	35	168	43.1	2.288	33	1
6	5	116	74	0	0	25.6	0.201	30	0
7	3	78	50	32	88	31.0	0.248	26	1
8	10	115	0	0	0	35.3	0.134	29	0
9	2	197	70	45	543	30.5	0.158	53	1
10	8	125	96	0	0	0.0	0.232	54	1
11	4	110	92	0	0	37.6	0.191	30	0
12	10	168	74	0	0	38.0	0.537	34	1
13	10	139	80	0	0	27.1	1.441	57	0
14	1	189	60	23	846	30.1	0.398	59	1
15	5	166	72	19	175	25.8	0.587	51	1
16	7	100	0	0	0	30.0	0.484	32	1
17	0	118	84	47	230	45.8	0.551	31	1

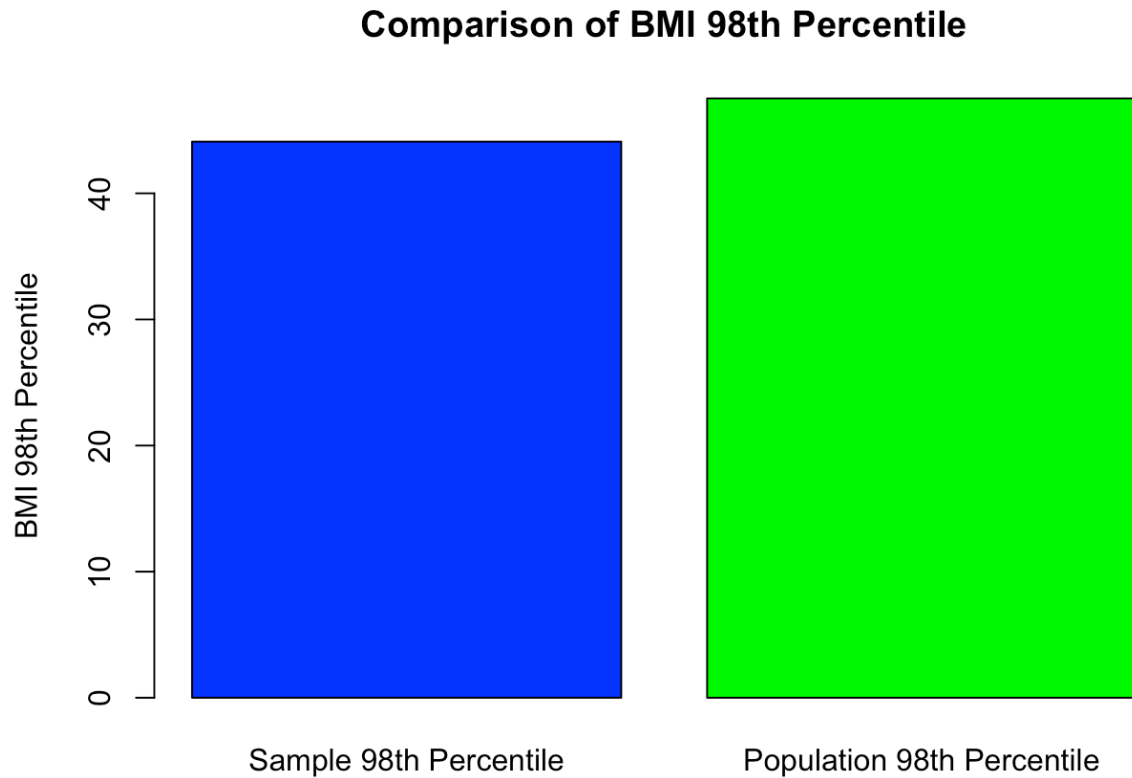
```
> set.seed(42)
> sample_data <- diabetes[sample(nrow(diabetes),25), ]
> sample_mean_glucose <- mean(sample_data$Glucose)
> sample_highest_glucose <- max(sample_data$Glucose)
> population_mean_glucose <- mean(diabetes$Glucose)
> population_highest_glucose <- max(diabetes$Glucose)
> barplot(c(sample_mean_glucose,sample_highest_glucose,population_mean_glucose,population_highest_glucose),
names.arg = c("Sample Mean", "Sample Highest","Population Mean","Population Highest"),col= c("blue","orange",
"green","red"), ylim = c(0, max(sample_highest_glucose,population_highest_glucose) + 20), ylab = "Glucose
value", main = "Comparison of Glucose Statistics ")
> legend("topright", legend = c("Sample Mean", "Sample Highest", "Population Mean", "Population Highest"), f
ill = c("blue","orange","green","red"))
> |
```

# Comparison of Glucose Statistics



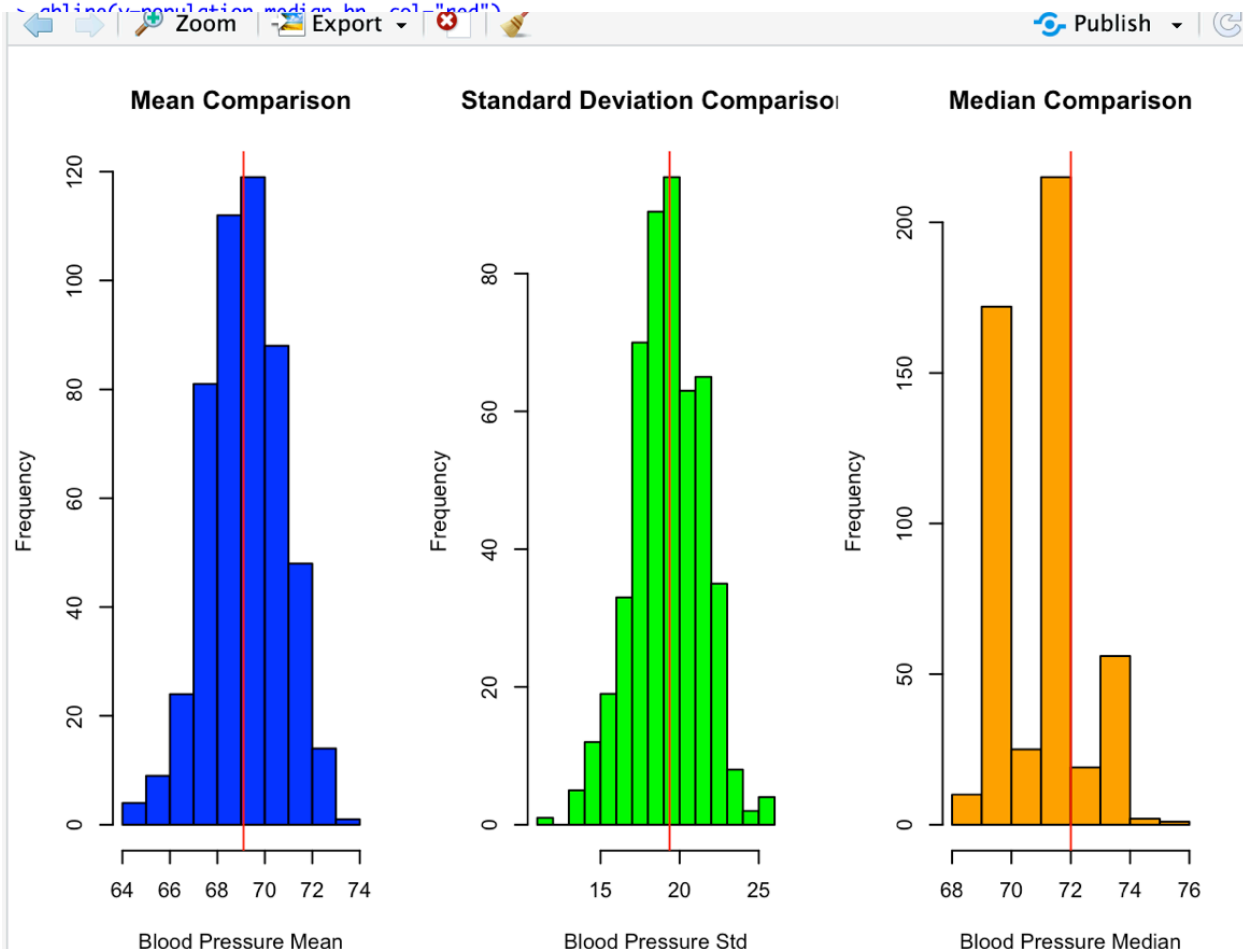
## STEP-B




```
> sample_98th_percentile <- quantile(sample_data$BMI, 0.98)
> population_98th_percentile <- quantile(diabetes$BMI, 0.98)
> barplot(c(sample_98th_percentile,population_98th_percentile),
  names.arg = c("Sample 98th Percentile","Population 98th Percentile"),
  col = c("blue", "green"),main = "Comparison of BMI 98th Percentile",
  ylab = "BMI 98th Percentile ")
```



## STEP-C

```
>
> bootstrap_sampling <- function(diabetes, n_samples=500, sample_size=150) {
+   sample_statistics <- matrix(nrow=n_samples, ncol=3)
+
+   for (i in 1:n_samples) {
+     sample <- sample(diabetes, size=sample_size, replace=TRUE)
+     sample_statistics[i, 1] <- mean(sample)
+     sample_statistics[i, 2] <- sd(sample)
+     sample_statistics[i, 3] <- quantile(sample, 0.5)
+   }
+
+   return(sample_statistics)
+ }
> blood_pressure_population <- diabetes$BloodPressure
> bootstrap_samples <- bootstrap_sampling(blood_pressure_population)
> population_mean_bp <- mean(blood_pressure_population)
> population_std_bp <- sd(blood_pressure_population)
> population_median_bp <- quantile(blood_pressure_population, 0.5)
> par(mfrow=c(1, 3))
> hist(bootstrap_samples[,1], main="Mean Comparison", xlab="Blood Pressure Mean", col="blue")
> abline(v=population_mean_bp, col="red")
>
> hist(bootstrap_samples[,2], main="Standard Deviation Comparison", xlab="Blood Pressure Std", col="green")
> abline(v=population_std_bp, col="red")
>
> hist(bootstrap_samples[,3], main="Median Comparison", xlab="Blood Pressure Median", col="orange")
> abline(v=population_median_bp, col="red")
```



Data		
bootstrap_samples	num [1:500, 1:3] 68.9 70.9 69.7 70.9 69.5 ...	
▶ diabetes	768 obs. of 9 variables	
▶ sample_data	25 obs. of 9 variables	
Values		
blood_pressure_popul...	num [1:768] 72 66 64 66 40 74 50 0 70 96 ...	
population_98th_perc...	Named num 47.5	
population_highest_g...	199	
population_mean_bp	69.10546875	
population_mean_gluc...	120.89453125	
population_median_bp	Named num 72	
population_std_bp	19.3558071706448	
sample_98th_percenti...	Named num 44.1	
sample_highest_gluco...	193	
sample_mean_glucose	129.6	
Functions		
bootstrap_sampling	function (diabetes, n_samples = 500, sample_size = 150)	