ST502 Project

Halid Kopanski and Justin Feathers

Part I

Framingham Heart Study

The Framingham data set contains the diastolic blood pressure of 300 smokers and nonsmokers. For this study we will assume the following:

H_0: μ_1 - $\mu_2 = 0$

H_A: μ_1 - $\mu_2 \neq 0$

The null hypothesis being there is no difference in the blood pressure between smokers and non smokers. We believe that there is a difference and this paper will prove if we have enough evidence to reject the null hypothesis.

Normality of Data

Plotting the density of total data (see Appendix I: Figure 1A), we can that it closely follows a normal distribution, Density plots of the split data exhibit similar forms (see Appendix I: Figure 1B). We can also calculate the kurtosis and the skew of the data, which are 3.812, 0.88 for kurtosis and skew respectively, and see that they are quite close to values typically found in normally distributed data. Additionally, the calculated values of sample standard deviation, $\frac{IQR}{1.35}$, and $\frac{MAD}{0.675}$ are 22.89, 22.22, and 21.48. The similarities between the three values indicate that the data is not influenced by outliers. Therefore, we will assume the data and the subsequent split data to be normal.

Statistical Analysis

In the first analysis we will be assuming equal variances, thereby allowing us to use pooled sample variance (see Appendix I: Equation 1).

The pooled sample variance of the data is 510 using a degree of freedom value 298. In this case, we reject the null hypothesis because the calculated p value, 0.0041, is smaller than the chosen α value of 0.05. In terms of t values, our observed t value of -3.04 is smaller than the t value, -1.97 for a two sided α of 0.05.

When computing the observed t value using the assumption that the population variances are not equivalent (variance smoker is 352.2 and variance nonsmoker is 562.1), a value of -2.9 is obtained. Comparing that the two sided α of -1.98 with 158 degrees of freedom (as computed using the Satterthwaite Approximation see Appendix I: Equation 4). The observed t value is less than the chosen α value of 0.05. In case, there is sufficient evidence to reject the null hypothesis in favor of the alternative.

The observed 95% confidence intervals are -15.08 to -3.23 for pooled sample variance and -15.4 to -2.91 for non pooled sample variance. It can be seen that 0 does not fall into the 95% confidence interval in either case. Therefore the null hypothesis can be rejected.

In all three case, there is sufficient evidence to reject the null hypothesis and to support the alternative at an α level of 0.05.

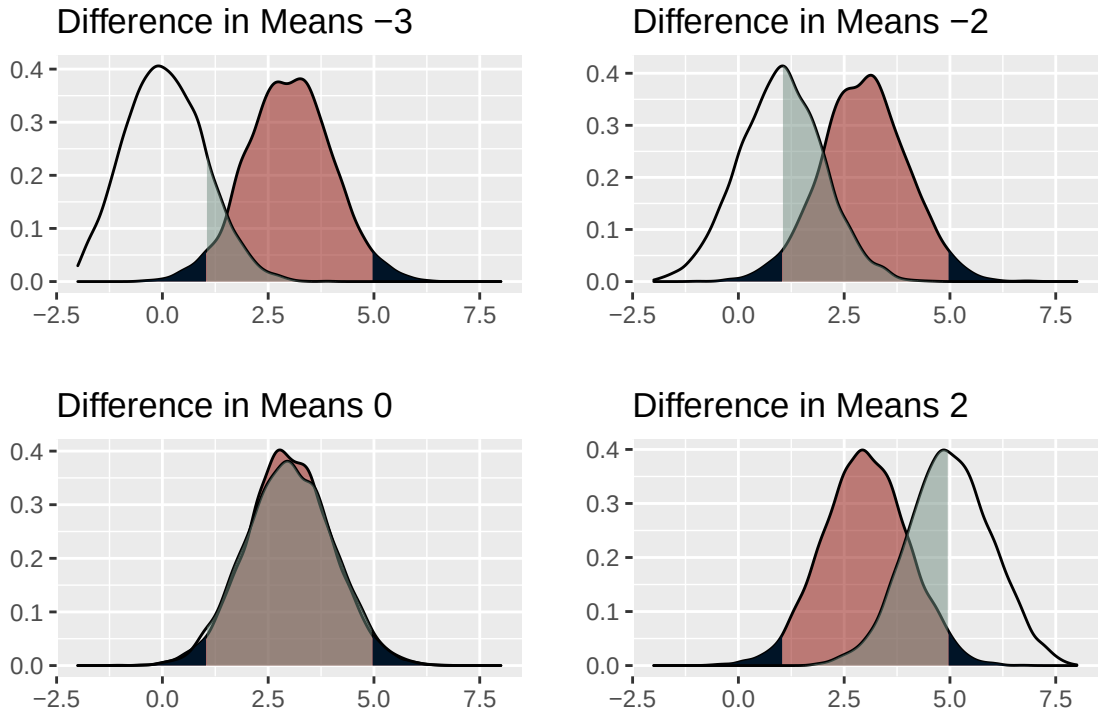
Part II

Introduction

In Part I we were able to see how hypothesis testing allowed to find enough evidence to challenge the "status quo" argument that smoking has no affect on the blood pressure of a person. What we found, through a number of robust statistical methods, that the difference in means, calculated to be 9.1577778, was significant to not have arisen by chance. That within an α of 5%, we are confident that the two groups do exhibit measurable differences. In the following sections, we will explore, through simulated data, the concept of randomness in data and what steps we can take to mitigate that randomness. All simulated date will be generated using the built in rnorm() function in R.

Power of Hypothesis Testing

It is important to briefly discuss the concept of power in hypothesis testing. Power describes the probability of not committing a Type II error, which is to not reject the null hypothesis when there was in actuality enough evidence to do so. The probability of a Type II error is represented by β. Power is the complement to that probability (1 - β). The value of β is the portion of the alternate distribution that is within the null hypothesis non rejection region limits. Below are a number of plots to help depict this concept. In each of the plots the shaded red is the null distribution and its location stays constant (mean = 3). The dark tails represent the rejection region of the null distribution. The outline distribution represents the "true" alternative distribution. Within that distribution is the shaded light blue region which is equal to β. It can be seen that as the difference between the two distributions shrinks the value of β increases and the power shrinks. Power reaches minimum and β reaches maximum when the two distributions are the same.



Simulation Study

Below, various scenarios were simulated by creating 2 normally distributed data sets of various mean, variances, and sample sizes. Each scenario was repeated a thousand times. For each of those repeats a hypothesis test was conducted and the number of times the null hypothesis was rejected was recorded. See Appendix III for results. All hypothesis testing was conducted assuming different values of variance, regardless of the actual variance values.

The following table includes the values of each parameter used in the simulation:

Parameter	Possible Values
μ_0	0, 4, 5, 6, 10
σ_0^2	1, 4, 9
n_0	10, 30, 70
μ_A	5
σ_A^2	1
n_A	10, 30, 70

It was noted that the smaller difference between the alternative and null mean values were, the less likely the null hypothesis was rejected. Larger values of variance also reduced the number of rejected null hypothesis. In both cases, the more overlap between the two distributions, the less powerful the test. One way that can increase the power of the t-test is to increase the sample size. For a specific example, we can look at test cases #12 and #27. In both cases the distribution means were 6 and 5 for the null and alternative hypotheses respectively, along with variances of 1 for both distributions. The only difference between the two cases was the sample size ($n_{10} = 10$ and $n_{129} = 70$). This increase in sample size resulted in almost a 6 fold increase in power (169 vs 958 rejected nulls). Test cases 69 and 114 use different combinations of, but lower than 70, sample sizes than the aforementioned two. While they exhibited increased signs of power, they did not get as high as test case 129. It stands to reason that large variances and small deltas in mean can be mitigated by increasing the sample size accordingly.

Appendix I: Equations and Figures

Figures

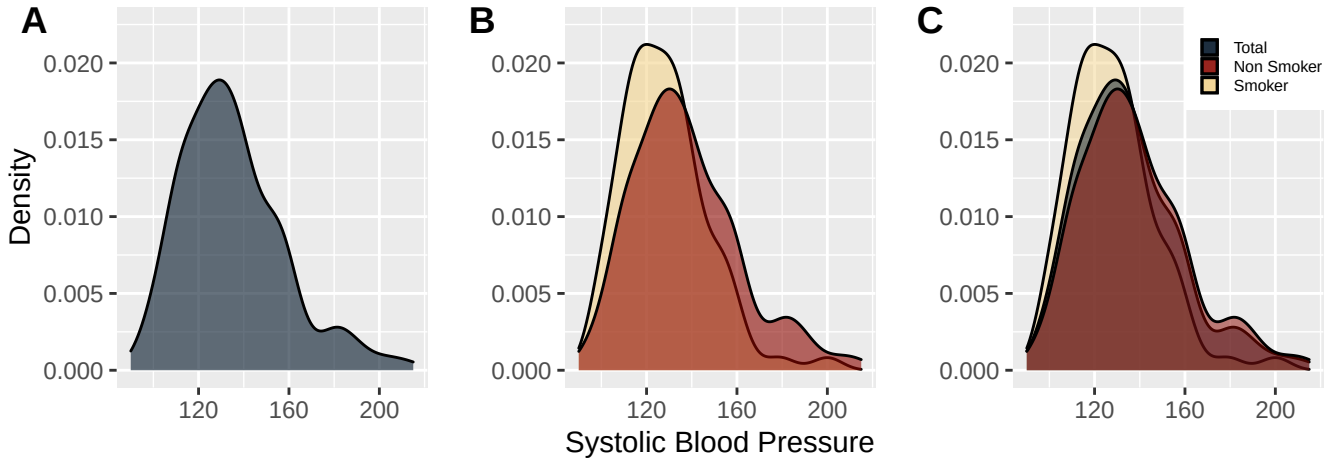


Figure 1: Data plots

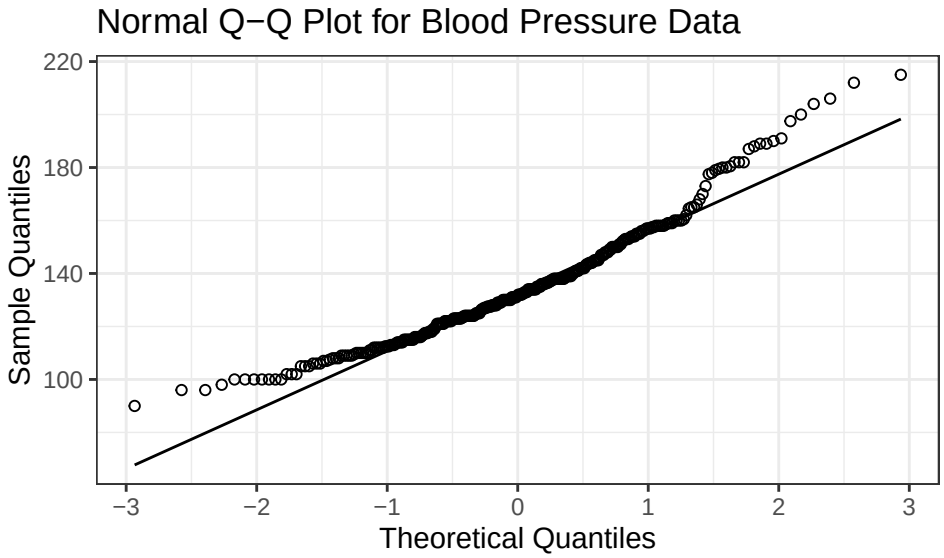


Figure 2: Q-Q Plot

Equations:

Equation 1: Pooled Sample Variance:

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}$$

Equation 2: Observed T Statistic (pooled variance):

$$t_{obs} = \frac{\mu_1 - \mu_2}{S_p\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

Equation 3: Observed T Statistic (distinct variance):

$$t_{obs} = \frac{\mu_1 - \mu_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}}$$

Equation 4: Satterthwaite Approximation:

$$\nu = \frac{\left(\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}\right)^2}{\frac{\left(\frac{S_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{S_2^2}{n_2}\right)^2}{n_2 - 1}}$$

Appendix II: Part I Results

Data Mean: 134.935
Data Sample Variance: 524.0852258

Smoker Mean: 128.0666667
Smoker Variance: 352.2117117

Nonsmoker Mean: 137.2244444
Nonsmoker Variance: 562.1447123

Difference in mean: -9.1577778

Pooled Sample Variance: 510.0136987
Nonpooled Sample Variance: 9.9936838

CI Pooled: -15.08, -3.23
CI Nonpooled: -15.4, -2.91

Appendix III: Part II Results

Test Case	μ_1	σ_1^2	n_1	μ_2	σ_2^2	n_2	Test Results
1	0	1	10	5	1	10	1000
2	4	1	10	5	1	10	574
3	5	1	10	5	1	10	50
4	6	1	10	5	1	10	565
5	10	1	10	5	1	10	1000
6	0	4	10	5	1	10	1000
7	4	4	10	5	1	10	281
8	5	4	10	5	1	10	51
9	6	4	10	5	1	10	297
10	10	4	10	5	1	10	1000
11	0	9	10	5	1	10	993
12	4	9	10	5	1	10	181
13	5	9	10	5	1	10	60
14	6	9	10	5	1	10	169
15	10	9	10	5	1	10	997
16	0	1	30	5	1	10	1000
17	4	1	30	5	1	10	782
18	5	1	30	5	1	10	51
19	6	1	30	5	1	10	777
20	10	1	30	5	1	10	1000
21	0	4	30	5	1	10	1000
22	4	4	30	5	1	10	276
23	5	4	30	5	1	10	4
24	6	4	30	5	1	10	252
25	10	4	30	5	1	10	1000
26	0	9	30	5	1	10	1000
27	4	9	30	5	1	10	89
28	5	9	30	5	1	10	3
29	6	9	30	5	1	10	96
30	10	9	30	5	1	10	1000
31	0	1	70	5	1	10	1000
32	4	1	70	5	1	10	835
33	5	1	70	5	1	10	55
34	6	1	70	5	1	10	847
35	10	1	70	5	1	10	1000
36	0	4	70	5	1	10	1000
37	4	4	70	5	1	10	242
38	5	4	70	5	1	10	1
39	6	4	70	5	1	10	251
40	10	4	70	5	1	10	1000
41	0	9	70	5	1	10	1000
42	4	9	70	5	1	10	23
43	5	9	70	5	1	10	1
44	6	9	70	5	1	10	40
45	10	9	70	5	1	10	1000
46	0	1	10	5	1	30	1000
47	4	1	10	5	1	30	773
48	5	1	10	5	1	30	54
49	6	1	10	5	1	30	780
50	10	1	10	5	1	30	1000
51	0	4	10	5	1	30	1000
52	4	4	10	5	1	30	543
53	5	4	10	5	1	30	143
54	6	4	10	5	1	30	540

Test Case	μ_1	σ_1^2	n_1	μ_2	σ_2^2	n_2	Test Results
55	10	4	10	5	1	30	1000
56	0	9	10	5	1	30	1000
57	4	9	10	5	1	30	423
58	5	9	10	5	1	30	220
59	6	9	10	5	1	30	408
60	10	9	10	5	1	30	1000
61	0	1	30	5	1	30	1000
62	4	1	30	5	1	30	975
63	5	1	30	5	1	30	47
64	6	1	30	5	1	30	963
65	10	1	30	5	1	30	1000
66	0	4	30	5	1	30	1000
67	4	4	30	5	1	30	659
68	5	4	30	5	1	30	46
69	6	4	30	5	1	30	681
70	10	4	30	5	1	30	1000
71	0	9	30	5	1	30	1000
72	4	9	30	5	1	30	400
73	5	9	30	5	1	30	58
74	6	9	30	5	1	30	399
75	10	9	30	5	1	30	1000
76	0	1	70	5	1	30	1000
77	4	1	70	5	1	30	994
78	5	1	70	5	1	30	60
79	6	1	70	5	1	30	995
80	10	1	70	5	1	30	1000
81	0	4	70	5	1	30	1000
82	4	4	70	5	1	30	773
83	5	4	70	5	1	30	6
84	6	4	70	5	1	30	778
85	10	4	70	5	1	30	1000
86	0	9	70	5	1	30	1000
87	4	9	70	5	1	30	409
88	5	9	70	5	1	30	10
89	6	9	70	5	1	30	412
90	10	9	70	5	1	30	1000
91	0	1	10	5	1	70	1000
92	4	1	10	5	1	70	838
93	5	1	10	5	1	70	50
94	6	1	10	5	1	70	832
95	10	1	10	5	1	70	1000
96	0	4	10	5	1	70	1000
97	4	4	10	5	1	70	625
98	5	4	10	5	1	70	205
99	6	4	10	5	1	70	621
100	10	4	10	5	1	70	1000
101	0	9	10	5	1	70	1000
102	4	9	10	5	1	70	580
103	5	9	10	5	1	70	339
104	6	9	10	5	1	70	550
105	10	9	10	5	1	70	1000
106	0	1	30	5	1	70	1000
107	4	1	30	5	1	70	994
108	5	1	30	5	1	70	52
109	6	1	30	5	1	70	997
110	10	1	30	5	1	70	1000
111	0	4	30	5	1	70	1000

Test Case	μ_1	σ_1^2	n_1	μ_2	σ_2^2	n_2	Test Results
112	4	4	30	5	1	70	864
113	5	4	30	5	1	70	122
114	6	4	30	5	1	70	868
115	10	4	30	5	1	70	1000
116	0	9	30	5	1	70	1000
117	4	9	30	5	1	70	630
118	5	9	30	5	1	70	174
119	6	9	30	5	1	70	635
120	10	9	30	5	1	70	1000
121	0	1	70	5	1	70	1000
122	4	1	70	5	1	70	1000
123	5	1	70	5	1	70	60
124	6	1	70	5	1	70	999
125	10	1	70	5	1	70	1000
126	0	4	70	5	1	70	1000
127	4	4	70	5	1	70	968
128	5	4	70	5	1	70	52
129	6	4	70	5	1	70	958
130	10	4	70	5	1	70	1000
131	0	9	70	5	1	70	1000
132	4	9	70	5	1	70	746
133	5	9	70	5	1	70	59
134	6	9	70	5	1	70	765
135	10	9	70	5	1	70	1000

Appendix IV: Code

```
knitr::opts_chunk$set(echo = TRUE)

#Use required packages
library(tidyverse) #for plots and data manipulation
library(cowplot) #aligning plots
library(gridExtra)
library(scales)

df_data <- read_csv("framingham_data.csv") # Read in data
df_data$index <- seq(nrow(df_data)) # Add an index column

#df_data %>% summary # Summarize Data

# Split data into smoker and nonsmoker
df_smoker <- df_data %>% filter(currentSmoker == 1)
df_nonsmoker <- df_data %>% filter(currentSmoker == 0)

#Create a sample variance function to ensure proper calculation
sample_variance <- function(x, sampling = TRUE){
    if (sampling == TRUE){
        sum((x - mean(x))^2) / (length(x) - 1)
    } else if(sampling == FALSE) {
        sum((x - mean(x))^2) / (length(x))
    }
}
#Create pooled sample variance function
f_pooled_variance <- function(x, y){
    ((length(x) - 1) * sample_variance(x) +
     (length(y) - 1) * sample_variance(y)) /
    (length(x) + length(y) - 2)
}

# Skewness function
skew_function <- function(x){
    mean((x - mean(x))^3) / sqrt(sample_variance(x))^3
}

# kurtosis function
kurt_function <- function(x){
    mean((x - mean(x))^4) / sqrt(sample_variance(x))^4
}

# Create a Satterthawaite Approximation Function

satterth <- function(s1, s2, n1, n2){
    term1 <- s1/n1
    term2 <- s2/n2
    nu <- (term1 + term2)^2 / ((term1^2/(n1 - 1)) + (term2^2/(n2 - 1)))
    return(floor(nu))
}


#Plot and compare split data

#options(repr.plot.width = 6, repr.plot.height = 4, repr.plot.res = 150)
```

```r
plot_colors <- c("#001427","#708d81","#f4d58d","#bf0603","#8d0801")
y_limits <- c(0, 0.0225)

total_data <- ggplot(df_data) + geom_density(aes(sysBP),
                                             fill = plot_colors[1],
                                             alpha = 0.6) +
                        ylim(y_limits) + ylab("Density") + xlab("")

sep_data <- ggplot() + geom_density(data = df_smoker, aes(sysBP),
                                    fill = plot_colors[3], alpha = 0.6) +
                       geom_density(data = df_nonsmoker, aes(sysBP),
                                    fill = plot_colors[5], alpha = 0.6) +
                       ylim(y_limits) + ylab("") + xlab("Systolic Blood Pressure")

plot_3 <- ggplot() + geom_density(data = df_smoker, aes(sysBP,
                                    fill = plot_colors[3]), alpha = 0.5) +
                       geom_density(data = df_data, aes(sysBP,
                                    fill = plot_colors[1]), alpha = 0.5) +
                       geom_density(data = df_nonsmoker, aes(sysBP,
                                    fill = plot_colors[5]), alpha = 0.5) +
                       ylim(y_limits) + ylab("") + xlab("") +
                       scale_fill_manual("",
                                         values = plot_colors[c(1, 5, 3)],
                                         labels = c("Total", "Non Smoker", "Smoker")) +
                       theme(legend.position = c(0.8, 0.9),
                             legend.text = element_text(size = 6),
                             legend.key.height = unit(0.25, 'cm'),
                             legend.key.width = unit(0.25, 'cm'))

#plot_grid(total_data, sep_data, plot_3, align = 'vh',
          #hjust = -1,nrow = 2, ncol = 2)

data_kurtosis <- kurt_function(df_data$sysBP)
data_skew <- skew_function(df_data$sysBP)
data_IQR <- as.numeric(quantile(df_data$sysBP, probs = 0.75)) -
       as.numeric(quantile(df_data$sysBP, probs = 0.25))
data_MAD <- median(abs(df_data$sysBP - median(df_data$sysBP)))
data_samVar <- sample_variance(df_data$sysBP)

eIQR <- data_IQR / 1.35
eMAD <- data_MAD / 0.675

# Q-Q Plot

data_qqplot <-
ggplot(df_data, aes(sample = sysBP)) +
stat_qq(shape = 1) + stat_qq_line() +
ggtitle("Normal Q-Q Plot for Blood Pressure Data") +
xlab("Theoretical Quantiles") +
ylab("Sample Quantiles")

# Common values for analysis

alpha <- 0.05

mu_smoker <- mean(df_smoker$sysBP)
var_smoker <- sample_variance(df_smoker$sysBP)
```

```r
n_smoker <- length(df_smoker$sysBP)

mu_nonsmoker <- mean(df_nonsmoker$sysBP)
var_nonsmoker <- sample_variance(df_nonsmoker$sysBP)
n_nonsmoker <- length(df_nonsmoker$sysBP)

# Two Sample T-test - Pooled Sample Variance - P-value

dof_1 <- (n_smoker + n_nonsmoker - 2)

p_sample_var_1 <- f_pooled_variance(df_smoker$sysBP,
                                    df_nonsmoker$sysBP)


t_obs_1 <- (mu_smoker - mu_nonsmoker) / (sqrt(p_sample_var_1) * sqrt(1/n_smoker + 1/n_nonsmoker))

t_stat_1 <- qt(alpha / 2, dof_1)

p_value_obs_1 <- dt(t_obs_1, dof_1)

#Two Sample T-test - Difference Variance Sample Variance - P-value

dof_2 <- satterth(var_smoker, var_nonsmoker, n_smoker, n_nonsmoker)

np_sample_var_2 <- (var_nonsmoker/n_smoker + var_nonsmoker/n_nonsmoker)

t_obs_2 <- (mu_smoker - mu_nonsmoker) / (sqrt(var_nonsmoker/n_smoker + var_nonsmoker/n_nonsmoker))

t_stat_2 <- qt(alpha / 2, dof_2)

p_value_obs_2 <- dt(t_obs_2, dof_2)

# Confidence Limits

diff_mu <- mu_smoker - mu_nonsmoker

#Pooled Sample variance

CL_pooled <- t_stat_1 * (sqrt(p_sample_var_1/n_smoker + p_sample_var_1/n_nonsmoker))

#Non pooled Sample variance

CL_nonpooled <- t_stat_2 * (sqrt(var_nonsmoker/n_smoker + var_nonsmoker/n_nonsmoker))

CI_pooled <- round(c(diff_mu + CL_pooled, diff_mu - CL_pooled), 2)

CI_nonpooled <-round(c(diff_mu + CL_nonpooled, diff_mu - CL_nonpooled), 2)


#Part II
#Introduction

options(repr.plot.width = 12, repr.plot.height = 5, repr.plot.res = 150)
set.seed(100)

null_mean <- 3
alt_means <- c(0, 1, 3, 5)
plot_list <- list()
```

```r
#plot_colors <- c("#072ac8","#1e96fc","#a2d6f9","#fcf300","#ffc600")

for(i in 1:length(alt_means)){

    sim1 <- rnorm(5000, null_mean, sqrt(1))
    sim2 <- rnorm(5000, alt_means[i], sqrt(1))

    alpha1 <- qnorm(0.025, null_mean, sqrt(1))
    alpha2 <- qnorm(0.975, null_mean, sqrt(1))

    df_set <- tibble("H0" = sim1, "HA" = sim2)

    title_string <- sprintf("Difference in Means %i", (alt_means[i] - null_mean))

    plot_list[[i]]  <-
    ggplot(data = df_set) + geom_density(aes(H0), alpha = 0.5, fill = plot_colors[5]) +
                            geom_area(
                                aes(x = stage(H0, after_scale = oob_censor(x, c(-Inf, alpha1)
                                                                           )
                                              )
                                   ),
                                stat = "density", fill = plot_colors[1]
                                 ) +
                            geom_area(
                                aes(x = stage(H0, after_scale = oob_censor(x, c(alpha2, Inf)
                                                                           )
                                              )
                                   ),
                                stat = "density", fill = plot_colors[1]
                                     ) +
                            geom_density(aes(HA), alpha = 0.5) +
                            geom_area(
                                aes(x = stage(HA, after_scale = oob_censor(x, c(alpha1, alpha2)
                                                                           )
                                              )
                                   ),
                                stat = "density", fill = plot_colors[2], alpha = 0.5
                                     ) +
                            xlim(-2, 8) + xlab("") + ylab("") + ggtitle(title_string)

                                }

do.call(grid.arrange, plot_list)
#Part II
set.seed(1)

alpha <- 0.05

test_function <- function (x, y, pooled = FALSE){

    mu_1 <- mean(x)
    var_1 <- sample_variance(x, sampling = TRUE)

    mu_2 <- mean(y)
    var_2 <- sample_variance(y, sampling = TRUE)

    #Calculate the pooled sample variance
```

```
    pooled_sample <- ((length(x) - 1) * var_1 + (length(y) - 1) * var_2) / (length(x) + length(y) - 2)

    #calculate the observed t statistic
    if (pooled == TRUE){

        cal_sigma <- (sqrt(pooled_sample/length(x) + pooled_sample/length(y)))

        ttest <- (mu_1 - mu_2) / cal_sigma
        dof <- length(x) + length(y) - 2 #Determine degrees of freedom

        } else {

        cal_sigma <- (sqrt(var_1/length(x) + var_2/length(y)))

        ttest <- (mu_1 - mu_2) / cal_sigma
        dof <- satterth(var_1, var_2, length(x), length(y))
    }

    #Determine whether or not the null hypothesis can be rejected (1 = rejected, 0 = not rejected)
    verdict <- !between(ttest, qt(alpha / 2, dof), qt(1 - alpha / 2, dof))

    #Return calculated values
    return(c(mu_1, var_1, mu_2, var_2, ttest, cal_sigma, dof,verdict))
}



mu1 <- c(0, 4, 5, 6, 10)
var1 <- c(1, 4, 9)
n1 <- c(10, 30, 70)

mu2 <- 5
var2 <- 1
n2 <- c(10, 30, 70)

sim_test <- function(x_mu, x_var, x_n, y_mu, y_var, y_n, pooled = TRUE){

    sim_data_results <- matrix(rep(0, 8), ncol = 8)

    for (i in 1:1000){

        sim_set1 <- rnorm(x_n, x_mu, sqrt(x_var))
        sim_set2 <- rnorm(y_n, y_mu, sqrt(y_var))

        sim_data_results <- rbind(sim_data_results, test_function(sim_set1, sim_set2, pooled))

        #print(sim_data_results)
    }

    df_sim_data <- data.frame(sim_data_results[2 : nrow(sim_data_results),])
    colnames(df_sim_data) = c("Null Mean", "Null Variance", "Alternate Mean", "Alternate Variance",
                              "T statistic", "Calculated Variance", "DoF", "Null Reject")
    return(df_sim_data)
}

# HA: mean = 5, var = 1
df_combo <- data.frame(expand.grid(mu1, var1, n1, mu2, var2, n2))
df_combo2 <- cbind(1:nrow(df_combo), df_combo, rep(0, nrow(df_combo)))
```

```r
colnames(df_combo2) <- c("Test Case", "mu1", "var1", "n1", "mu2", "var2", "n2", "Test Results")

test_results <- list()

for (i in 1:nrow(df_combo)){
    test_results[[i]] <- do.call(sim_test, as.list(as.numeric(df_combo[i,])))
    df_combo2[i, 8] <- sum(as.data.frame(test_results[i])[,8])
    }

#df_combo2
# Plotting
options(repr.plot.width = 6, repr.plot.height = 4, repr.plot.res = 150)

plot_grid(total_data, sep_data, plot_3, align = 'vh',
          hjust = -1,nrow = 1, ncol = 3, labels = c("A", "B", "C"))
data_qqplot + theme_bw()
knitr::kable(df_combo2, col.names = c("Test Case",
                                      "$\\mu_1$",
                                      "$\\sigma_1^2$",
                                      "$n_1$",
                                      "$\\mu_2$",
                                      "$\\sigma_2^2$",
                                      "$n_2$",
                                      "Test Results"),
             escape = FALSE)
```