Experiment 3: kB for Gain 500

Madeleine Allen Edward Piper

2/9/2019

```
vin1<-read.csv("V1.CSV")
names(vin1)<-c("x", "y")
vin2<-read.csv("V2.CSV")
names(vin2)<-c("x", "y")
vin3<-read.csv("V3.CSV")
names(vin3)<-c("x", "y")
vin4<-read.csv("V4.CSV")
names(vin4)<-c("x", "y")
vin5<-read.csv("V5.CSV")
names(vin5)<-c("x", "y")
vout1<-read.csv("O1.CSV")
names(vout1)<-c("x", "y")
vout2<-read.csv("O2.CSV")
names(vout2)<-c("x", "y")
vout3<-read.csv("O3.CSV")
names(vout3)<-c("x", "y")
vout4<-read.csv("O4.CSV")
names(vout4)<-c("x", "y")
vout5<-read.csv("O5.CSV")
names(vout5)<-c("x", "y")
```

```
vin1_error<- sd(vin1$y, na.rm=TRUE)/sqrt(length(vin1$y[!is.na(vin1$y)]))
vin2_error<-sd(vin2$y, na.rm=TRUE)/sqrt(length(vin2$y[!is.na(vin2$y)]))
vin3_error<- sd(vin3$y, na.rm=TRUE)/sqrt(length(vin3$y[!is.na(vin3$y)]))
vin4_error<- sd(vin4$y, na.rm=TRUE)/sqrt(length(vin4$y[!is.na(vin4$y)]))
vin5_error<- sd(vin5$y, na.rm=TRUE)/sqrt(length(vin5$y[!is.na(vin5$y)]))

vout1_error<-sd(vout1$y[35:44], na.rm=TRUE)/sqrt(length(vout1$y[!is.na(vout1$y)]))
vout2_error<- sd(vout2$y[37:54], na.rm=TRUE)/sqrt(length(vout2$y[!is.na(vout2$y)]))
vout3_error<- sd(vout3$y[36:56], na.rm=TRUE)/sqrt(length(vout3$y[!is.na(vout3$y)]))
vout4_error<- sd(vout4$y[35:53], na.rm=TRUE)/sqrt(length(vout4$y[!is.na(vout4$y)]))
vout5_error<- sd(vout5$y[35:53], na.rm=TRUE)/sqrt(length(vout5$y[!is.na(vout5$y)]))
```

```
m_in<- (vin1$y+vin2$y+vin3$y+vin4$y+vin5$y)/5
#take average of vouts
m_vout <- data.frame(Frequency = vout1$x, Volts = (vout1$y+vout2$y+vout3$y+vout4$y+vout5$y)/5)
```

Recording the resistor values measured during lab.

Experiment 3: Johnson Noise - Boltzmann Constant

```
experiment3data1<-read.csv("/Users/mallen/Documents/128AL/JohnsonNoise128AL/experiment3data2.csv")
```

Calculate Vmeas, V, and Vsystem

```
Vsys<- experiment3data1[1,7] #first row 7th column
VsysError <- experiment3data1[1,9]
```

```
Vmeask1<- (experiment3data1[2,7])
Vmeask10<-experiment3data1[3,7]
Vmeask20 <-experiment3data1[4,7]
Vmeask32<-experiment3data1[5,7]
Vmeask48<-experiment3data1[6,7]
Vmeask100<-experiment3data1[7,7]

Vmeas<-c(Vmeask1, Vmeask10, Vmeask20, Vmeask32, Vmeask48, Vmeask100)

VmeasError<-sqrt((sum(experiment3data1[2:7,9])^2))

V<- sqrt(-Vsys^2+Vmeas^2)
Verror<- sqrt(VmeasError^2+ VsysError^2)
```

Calculating G

```
gain <- data.frame(Frequency = vout1$x, Gain = (m_vout[2]/m_in))

capacitance <-87.875*(10^-12)
capacitanceError <-.594*(10^-12)
#df is just the x componenent

riemanSum <- function(fa,fb){
  area <-0.5*(125)*(fb-fa)+fa*125
  return(area)
}

C = capacitance
integrand <- data.frame(
  gain[2]^2/(1+(2*pi*C*vin1$x*short)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k1)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k10)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k20)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k35)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k48)^2),
  gain[2]^2/(1+(2*pi*C*vin1$x*k100)^2)
  )
area <- data.frame(
  G1 =0,
  G2 =0,
  G3 =0,
  G4 =0,
  G5 =0,
  G6 =0,
  G7 =0

  )
for(i in 1:length(integrand))
{
    for(l in 1:398)
    {
```

```
      if(is.na(integrand[l+1,i]))
      {
        break
      }
      else
      {
        area[i] <- area[i]+ riemanSum(integrand[l,i],integrand[l+1,i])
      }
    }
}
```

So this returns a gain value G for each resistor (called “area”)

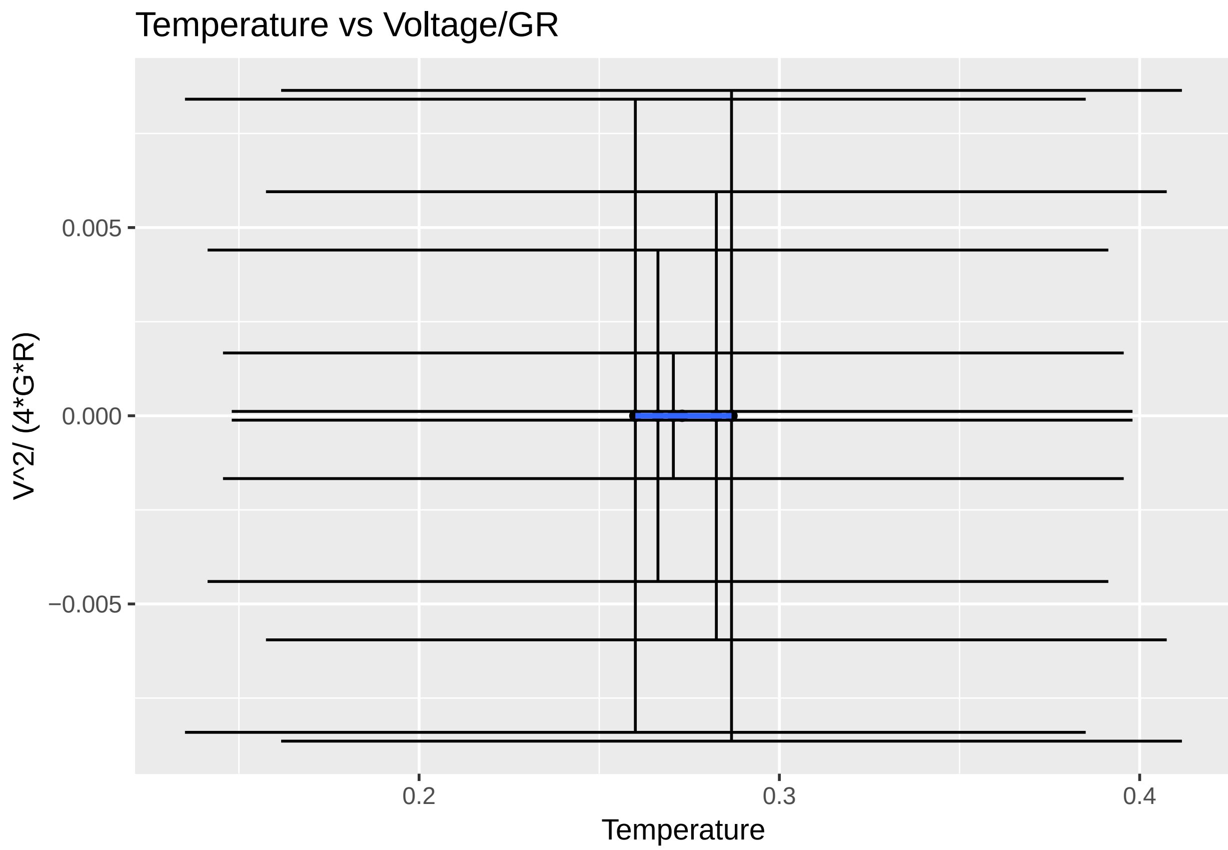
```
#prepare the data for graphing
kb<- 1.38064852 *10^-23 #m2 kg s-2 K-1

area2<-area[2:7]/100 #take away the short's data
resistors2 <-resistors[1:6]
y_value2<- (V^2)/(4*resistors2*area2)
y2<- unlist(y_value2, use.names=FALSE)

#calculate the temperatures
Temperature<- ((V^2)/(4*kb*area2*resistors2))/100
Temperature2<-unlist(Temperature, use.names = FALSE)
fit <- lm(y2~Temperature2)
```

```
#Temperature<- ((V^2)/(4*kb*area2*resistors2))/100
#Temperature2<-unlist(Temperature, use.names = FALSE)

#Temperature3Error<-Temperature3*sqrt((V^2/Verror^2)+(resistors2/resistorserror)^2)

#1st degree fit
Temperature2<-Temperature2/500
Temperature2[2]<- Temperature2[2]/10
print(Temperature2)
```

```
## [1] 0.2867199 0.2730080 0.2600258 0.2705773 0.2825070 0.2662982
```

```
Temperature2Error<-Temperature2*sqrt((V^2/Verror^2)+(resistors2/resistorserror)^2)
```

```
library(ggplot2)
qplot((Temperature2),(y2))+geom_errorbar(aes(x=(Temperature2), ymin=(y2-Temperature2Error2), ymax=(y2+T
  geom_smooth(method="lm", se=FALSE, fullrange=TRUE, level=0.95)+labs(title = "Temperature vs Voltage/G
```

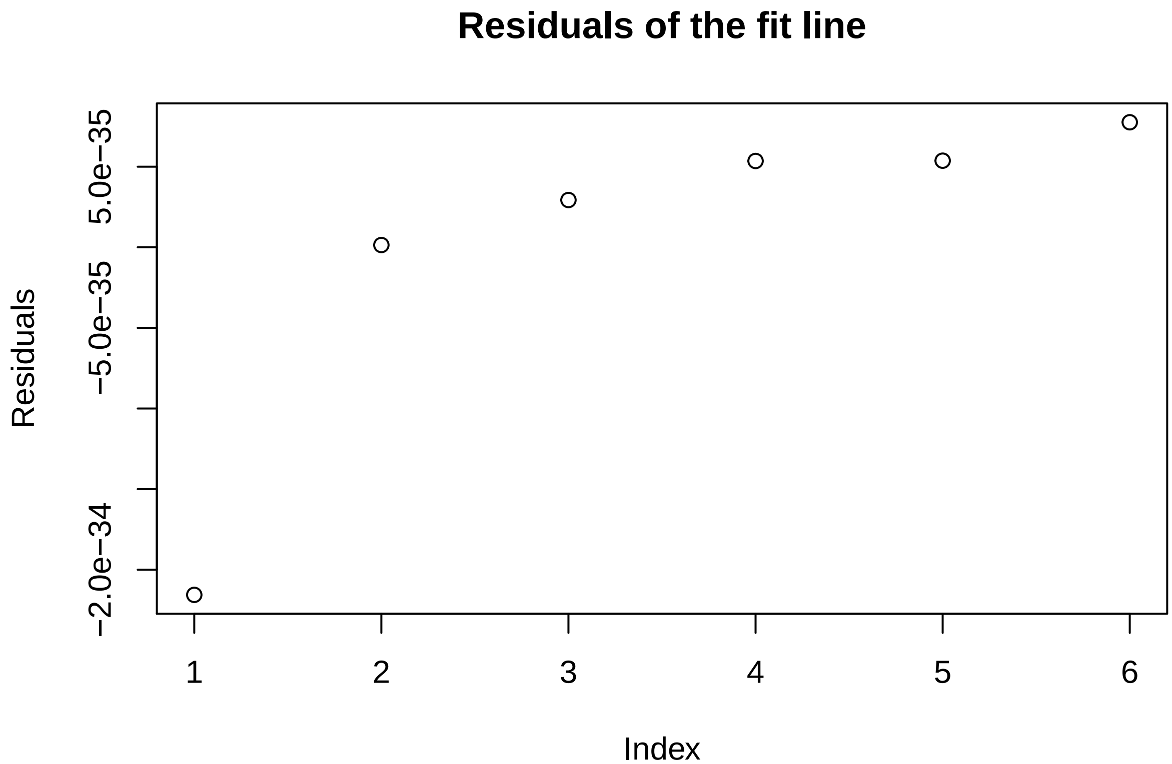



```
summary(fit)
```

```
## Warning in summary.lm(fit): essentially perfect fit: summary may be
## unreliable
```

```
##
## Call:
## lm(formula = y2 ~ Temperature2)
##
## Residuals:
##          1          2          3          4          5          6
## -2.156e-34  1.415e-36  2.933e-35  5.356e-35  5.373e-35  7.757e-35
##
## Coefficients:
##                Estimate Std. Error    t value Pr(>|t|)
## (Intercept)  -3.145e-34  6.191e-35 -5.080e+00  0.00708 **
## Temperature2  1.381e-21  1.084e-37  1.274e+16  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.216e-34 on 4 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.622e+32 on 1 and 4 DF,  p-value: < 2.2e-16
```

```
plot(fit$residuals, main = "Residuals of the fit line", ylab= "Residuals")
```


Residuals of the fit line
Residuals
5.0e−35
−5.0e−35
−2.0e−34
1
2
3
4
5
6
Index