# Supporting Information

## Censoring trace-level environmental data: statistical analysis considerations to limit bias

Barbara Jane George[1*], Leslie Gains-Germain[2], Kristin Broms[2], Kelly Black[2], Marschall Furman[3], Michael D. Hays[4], Kent W. Thomas[1], Jane Ellen Simmons[1]

[1] Center for Public Health and Environmental Assessment, Office of Research and Development, U.S. EPA, Research Triangle Park, North Carolina 27711, United States

[2] Neptune and Company, Inc., Lakewood, Colorado 80215, United States

[3] Oak Ridge Institute for Science and Education (ORISE) Research Participant at U.S. EPA, Office of Research and Development, Center for Public Health and Environmental Assessment, Research Triangle Park, North Carolina 27711, United States

[4] Center for Environmental Measurement and Modeling, Office of Research and Development, U.S. EPA, Research Triangle Park, North Carolina 27711, United States

*Address correspondence to B.J. George, CPHEA/ORD/U.S. EPA, 109 T.W. Alexander Dr., Research Triangle Park, NC 27711 USA. Telephone: (919) 541-4551. E-mail: george.bj@epa.gov

## Table of Contents

```
/* SAS code to resample stove observations */
/* Author: Barbara Jane George, PhD, PStat */
/* US Environmental Protection Agency      */
/* August 2020                             */

%let job=groups;


**********;
data stoves;
input
 @1  analyte $25.
 @26 result
 @36 MDL
 @47 CCLV
 @56 stove $17.;
;

*list;
cards;
Dibenzo[a,h]anthracene   0.05      0.023      0.10     Solgas/Repsol
Dibenzo[a,h]anthracene   0.03      0.023      0.10     Solgas/Repsol
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Solgas/Repsol
Dibenzo[a,h]anthracene   0.05      0.023      0.10     Solgas/Repsol
Dibenzo[a,h]anthracene   0.19      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.11      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.08      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.12      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.02      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.06      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.1       0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Butterfly Model
Dibenzo[a,h]anthracene   0.12      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.32      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.15      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.04      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.52      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.05      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.15      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.03      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   0.04      0.023      0.10     Eco Chula XXL
Dibenzo[a,h]anthracene   1         0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   1.16      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   1.46      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.01      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.87      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   1.31      0.023      0.10     Jiko Poa Rocket
```

```
Dibenzo[a,h]anthracene   0.08      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   1.31      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.02      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.4       0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.75      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.04      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   2.94      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.89      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.07      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.6       0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   1.12      0.023      0.10     Jiko Poa Rocket
Dibenzo[a,h]anthracene   0.04      0.023      0.10     Jiko Poa Rocket
;
ods csv file="Stove data for DBA.csv";
 proc means data=stoves n mean std;
 class stove;
 var result;
 run;

 proc print data=stoves;
 run;
ods csv close;

**********;
data benzo;
set stoves;

if stove in ("Eco Chula XXL","Butterfly Model");


/* https://blogs.sas.com/content/iml/2018/06/20/bootstrap-method-example-sas.html */
/* Sample with replacement from each stratum. First sort by the STRATA variable.  */
proc sort data=benzo;
by stove result;
run;

/* Then perform stratified sampling with replacement */
proc surveyselect data=benzo out=BootSamples noprint seed=123
                  method=urs              /* with replacement */
                  OUTHITS                 /* use OUTHITS option when you do not want
a frequency variable */
                  samprate=1 reps=10;     /* 10 resamples */
                  strata stove;           /* sample N1 from first group and N2 from
second */
run;


**********;
data butter eco;
set BootSamples(drop=Replicate NumberHits ExpectedHits SamplingWeight);

if stove in ("Eco Chula XXL") then output eco;
else if stove in ("Butterfly Model") then output butter;

**********;
```

```
data butter;   *stove type 2;
set butter;

if _n_ <=31;   *resampled observations to concatenate to original sample;

**********;
data eco;      *stove type 1  - long-tailed;
set eco;

if _n_ <=30;   *resampled observations to concatenate to original sample;

**********;
data buttereco;
set benzo(in=in1) butter eco;


if stove in ("Eco Chula XXL") then do;
 stove2="Stove Type 1 (n=15)";
 stove3="Stove Type 1 (n=45)";
 stove4="Stove Type 1";
end;
else if stove in ("Butterfly Model") then do;
 stove2="Stove Type 2 (n=9) ";
 stove3="Stove Type 2 (n=40)";
 stove4="Stove Type 2";
end;

if in1 then orig='Yes';
else orig='No ';


proc sort data=buttereco;
by stove4;

ods csv file="Print (&job) with resamples data=buttereco.csv";
 proc print data=buttereco;
 run;
ods csv close;


/* https://blogs.sas.com/content/iml/2016/03/09/comparative-panel-overlay-histograms-
sas.html */
/* https://stats.idre.ucla.edu/sas/code/generating-multiline-axis-labels-in-sas-proc-
sgplot/  */

ods rtf file="SGPanel (&job) resampled data=buttereco.rtf";
ods graphics / width=400px height=400px;

proc sgpanel data=buttereco noautolegend;
panelby stove3 / rows=2 layout=rowlattice novarname;
histogram result / binwidth=0.03 binstart=0 scale=count ;
colaxis min=0 label="DBA (ng/(*ESC*){unicode '03BC'x}L)";
density result / type=kernel(c=2);
run;
```

```
ods graphics off;
ods rtf close;



ods rtf file="SGPanel (&job) orig sample data=buttereco.rtf";
ods graphics / width=400px height=400px;

proc sgpanel data=buttereco noautolegend;
panelby stove2 / rows=2 layout=rowlattice novarname;
histogram result / binwidth=0.03 binstart=0 scale=count;
colaxis min=0 label="DBA (ng/(*ESC*){unicode '03BC'x}L)";
density result / type=kernel(c=2);
where orig='Yes';
run;

ods graphics off;
ods rtf close;


**********;
data superbenz;
set benzo(in=in1) buttereco(in=in2) ;

if stove in ("Eco Chula XXL") then do;
 stove2="Stove Type 1 (n=15)";
 stove3="Stove Type 1 (n=45)";
 stove4="Stove Type 1";
end;
else if stove in ("Butterfly Model") then do;
 stove2="Stove Type 2 (n=9) ";
 stove3="Stove Type 2 (n=40)";
 stove4="Stove Type 2";
end;

if in1      then resampled='Original Sample (n=24)';
else if in2 then resampled='Resampled Data (n=85) ';


ods rtf file="SGPanel (&job) both sets data=superbenz.rtf";
ods graphics / width=400px height=400px;

proc sgpanel data=superbenz noautolegend;
panelby stove4 resampled/ rows=2 layout=lattice novarname;
histogram result / binwidth=0.03 binstart=0 scale=count ;
colaxis min=0 label="DBA (ng/(*ESC*){unicode '03BC'x}L)";
density result / type=kernel(c=2);
run;

ods graphics off;
ods rtf close;
```

```
# R code for testing group differences
# Author: Barbara Jane George, PhD, PStat
# US Environmental Protection Agency
# August 2020

#setwd ("")
getwd()

library(dplyr)
library(EnvStats)
library(NADA)
library(fitdistrplus)
library(ggplot2)


data <- read.csv("Print (groups) with resamples data=buttereco.csv")

## num of samples below the CCLV
sum(data$result < data$CCLV)

## Create new vectors based on censoring
results <- data$result
CCLV <- data$CCLV
resultsCCLV <- ifelse(results<CCLV,CCLV,results)
result_halfCCLV <- ifelse(results<CCLV,CCLV/2,results)
stove <- data$stove
stove4 <- data$stove4
censored <- ifelse(results < data$CCLV, TRUE, FALSE)

### AIC for distribution fitting -- before & after censoring
fitg <- fitdist(results, "gamma")
fitln <- fitdist(results, "lnorm")
fitn <- fitdist(results, "norm")
fitg$aic
fitln$aic
fitn$aic

results_cens <- data.frame(left = ifelse(censored == TRUE, 0, results),
                           right = ifelse(censored == TRUE, data$CCLV, results))
aic_lnorm <- fitdistcens(results_cens, distr = "lnorm")$aic
aic_norm  <- fitdistcens(results_cens, distr = "norm")$aic
aic_gamma <- fitdistcens(results_cens, distr = "gamma")$aic
aic_lnorm
aic_norm
aic_gamma


CCLV <- unique(data$CCLV)
DL <- CCLV


## MLE for results censored at CCLV
estResult3 <- cenmle(resultsCCLV,censored,stove ) #NADA cenmle assumes lognormal,
tests equality of medians, reference cell
summary(estResult3)
```

```
## MLE for results not censored
nocensor <- ifelse(results < data$CCLV, FALSE, FALSE)
estResult4 <- cenmle(results,nocensor,stove ) #NADA cenmle assumes lognormal, tests
equality of medians, reference cell
summary(estResult4)


results_ggplot <- data.frame(results,stove4)
p <- ggplot(results_ggplot, aes(x = factor(stove4), y = results, color =
factor(stove4))) +
     geom_stripchart(test.text=TRUE, digits=3,
                     test.text.params = list(p.value.digits=4)) +           labs(x =
NULL, y =expression(paste("Uncensored DBA (ng/",mu,"L)")))
p
tiff("stripchart (07-30-20) results.tiff", width=5, height=5, units='in',
res=300,compression='lzw')
plot(p)
dev.off()


results_halfCCLV <- data.frame(result_halfCCLV,stove4)
pp <- ggplot(results_halfCCLV, aes(x = factor(stove4), y = result_halfCCLV, color =
factor(stove4))) +
      geom_stripchart(test.text=TRUE, digits=3,
                     test.text.params = list(p.value.digits=4)) +              labs(x
= NULL, y = expression(paste("DBA (ng/",mu,"L) with CCLV/2 substitution")))
pp
tiff("stripchart (07-30-20) results_halfCCLV.tiff", width=5, height=5, units='in',
res=300,compression='lzw')
plot(pp)
dev.off()


analyte_dat_eco <- data %>% filter(stove == "Eco Chula XXL")
result_eco <- analyte_dat_eco$result
DL_eco <- ifelse(result_eco<DL,DL,result_eco)
halfDL_eco <- ifelse(result_eco<DL,0.5*DL,result_eco)
halfDL_mean_eco <- mean(halfDL_eco)
halfDL_sd_eco <- sd(halfDL_eco)
halfDL_cv_eco <- halfDL_sd_eco/halfDL_mean_eco
halfDL_n_eco <- length(halfDL_eco)

analyte_dat_butterfly <- data %>% filter(stove == "Butterfly Model")
result_butterfly <- analyte_dat_butterfly$result
DL_butterfly <- ifelse(result_butterfly<DL,DL,result_butterfly)
halfDL_butterfly <- ifelse(result_butterfly<DL,0.5*DL,result_butterfly)
halfDL_mean_butterfly <- mean(halfDL_butterfly)
halfDL_sd_butterfly <- sd(halfDL_butterfly)
halfDL_cv_butterfly <- halfDL_sd_butterfly/halfDL_mean_butterfly
halfDL_n_butterfly <- length(halfDL_butterfly)

full_mean_eco <- mean(result_eco)
full_sd_eco <- sd(result_eco)
```

```
full_cv_eco <- full_sd_eco/full_mean_eco
full_MLE_eco <- elnormAlt(result_eco,method = "mle")$parameters
full_MLE_mean_eco <- full_MLE_eco[1]
full_MLE_sd_eco <- full_MLE_eco[2] * full_MLE_eco[1]
full_MLE_cv_eco <- full_MLE_eco[2]

censored_eco <- ifelse(result_eco<DL, TRUE, FALSE)
MLE_eco <- elnormAltCensored(DL_eco, censored_eco, method = "mle")$parameters
MLE_mean_eco <- MLE_eco[1]
MLE_sd_eco <- MLE_eco[2] * MLE_eco[1]
MLE_cv_eco <- MLE_eco[2]
MLE_n_eco <- length(result_eco)

full_mean_butterfly <- mean(result_butterfly)
full_sd_butterfly <- sd(result_butterfly)
full_cv_butterfly <- full_sd_butterfly/full_mean_butterfly
full_MLE_butterfly <- elnormAlt(result_butterfly,method = "mle")$parameters
full_MLE_mean_butterfly <- full_MLE_butterfly[1]
full_MLE_sd_butterfly <- full_MLE_butterfly[2] * full_MLE_butterfly[1]
full_MLE_cv_butterfly <- full_MLE_butterfly[2]

censored_butterfly <- ifelse(result_butterfly<DL, TRUE, FALSE)
MLE_butterfly <- elnormAltCensored(DL_butterfly, censored_butterfly, method =
"mle")$parameters
MLE_mean_butterfly <- MLE_butterfly[1]
MLE_sd_butterfly <- MLE_butterfly[2] * MLE_butterfly[1]
MLE_cv_butterfly <- MLE_butterfly[2]
MLE_n_butterfly <- length(result_butterfly)


full_mean_eco
full_mean_butterfly
full_MLE_mean_eco
full_MLE_mean_butterfly
halfDL_mean_eco
halfDL_mean_butterfly
MLE_mean_eco
MLE_mean_butterfly

full_sd_eco
full_sd_butterfly
full_MLE_sd_eco
full_MLE_sd_butterfly
halfDL_sd_eco
halfDL_sd_butterfly
MLE_sd_eco
MLE_sd_butterfly

full_cv_eco
full_cv_butterfly
full_MLE_cv_eco
full_MLE_cv_butterfly
halfDL_cv_eco
halfDL_cv_butterfly
MLE_cv_eco
```

```
MLE_cv_butterfly

halfDL_n_eco
halfDL_n_butterfly
MLE_n_eco
MLE_n_butterfly


## apply log transformation to concentration data
results_fullsamp_log <- data.frame(reslog = log(results), stove4)
results_halfCCLV_log <- data.frame(reslog = log(result_halfCCLV), stove4)

results_ggplot <- data.frame(results,stove4)
p <- ggplot(results_fullsamp_log, aes(x = factor(stove4), y = reslog, color =
factor(stove4))) +
     geom_stripchart(test.text=TRUE, digits=3,
                     test.text.params = list(p.value.digits=4)) +
     labs(x = NULL, y =expression(paste("log(DBA) Uncensored, Full Data")))
p
tiff("stripchart log(DBA) results.tiff", width=5, height=5, units='in',
res=300,compression='lzw')
plot(p)
dev.off()


results_halfCCLV <- data.frame(result_halfCCLV,stove4)
pp <- ggplot(results_halfCCLV_log, aes(x = factor(stove4), y = reslog, color =
factor(stove4))) +
      geom_stripchart(test.text=TRUE, digits=3,
                     test.text.params = list(p.value.digits=4)) +
      labs(x = NULL, y = expression(paste("log(DBA) with CCLV/2 substitution")))
pp
tiff("stripchart log(DBA) results_halfCCLV.tiff", width=5, height=5, units='in',
res=300,compression='lzw')
plot(pp)
dev.off()
```

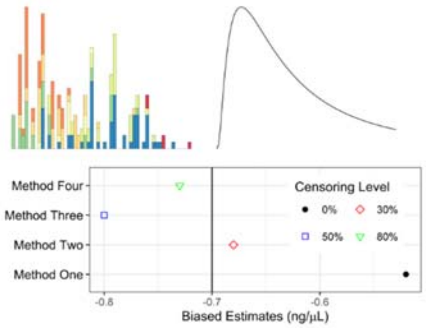
```
# R code for example TOC graphic
# Author: Barbara Jane George, PhD, PStat
# US Environmental Protection Agency
# August 2020

# use Paint, File, Properties to examine TIFF DPI and dimensions in inches or pixels

#setwd("")
getwd()

library(ggplot2)
require('magick')

##########
########## Biased Estimates example
##########

Average <- c(-0.52,-0.68,-0.8,-0.73)
method <- c('Method One','Method Two','Method Three','Method Four')
method <- factor(method,levels = c('Method One','Method Two','Method Three','Method
Four'))

CensoringLevel <- c('0%','30%','50%','80%')
dd <- data.frame(Average,method,CensoringLevel)

means <- data.frame(Parameter=c("Mean"), parm_mean=c(-0.7))
qq <- ggplot(dd,aes(Average,method)) +
      geom_vline(aes(xintercept=parm_mean),means) +

geom_point(size=2,aes(Average,method,colour=factor(CensoringLevel),shape=factor(Censo
ringLevel))) +
      scale_shape_manual(values=c(16,5,0,6)                    , name = "Censoring
Level", labels=c("0%","30%","50%","80%")) +
      scale_colour_manual(values=c("black","red","blue","green"), name = "Censoring
Level", labels=c("0%","30%","50%","80%")) +
      theme_bw() +  guides(shape=guide_legend("Censoring Level",nrow=2,byrow=TRUE)) +
#, size=FALSE
      theme(legend.position=c(0.95, 0.95),legend.justification = c("right",
"top"),axis.text.y=element_text(size=rel(1.2),colour=("black"))) +
###,axis.title.x = element_text(angle=0,vjust=6,hjust=1)) +
      labs(x = expression(paste("Biased Estimates (ng/",mu,"L)")),y=NULL)

qq

tiff("Biased Estimates Example.tiff", width = 5, height = 2, units = 'in', res =
1200)
plot(qq)
dev.off()


##########
########## Top Row examples
##########
```

```
###Adapted from http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-
Code.html

# Histogram on a Continuous Variable
theme_set(theme_classic())
gg <- ggplot(mpg, aes(displ)) + scale_fill_brewer(palette = "Spectral") +
      geom_histogram(aes(fill=class),
                    binwidth = .1,
                    col="black",
                    size=.1) +
      theme(axis.title=element_blank(),
            axis.text=element_blank(),
            axis.ticks=element_blank(),
            legend.position="none",
            axis.line=element_blank())
ggsave("top_left.pdf", width = 5, height = 4)


set.seed(123)
hh <- data.frame(x = rlnorm(1000, meanlog = -0.5, sdlog = 1))
ggplot(hh) +
  stat_function(fun = "dlnorm",
                args = list(meanlog = -1, sdlog = 1)) +
        theme(axis.title=element_blank(),
           axis.text=element_blank(),
           axis.ticks=element_blank(),
           legend.position="none",
           axis.line=element_blank())
ggsave("top_right.pdf", width = 5, height = 4)


##########
########## white sliver for padding
##########


tiff("sliver.tiff", width = 0.2, height = 1, units="in", res=300)
dev.off()


##########
########## blue arrow
##########

#https://ggplot2.tidyverse.org/reference/geom_segment.html
df2 <- expand.grid(
  lineend = c('round'),
  linejoin = c('mitre'),
  stringsAsFactors = FALSE
)
df2 <- data.frame(df2, y = 1:1)
aa <- ggplot(df2, aes(x = 1, y = y,xend = 1.25, yend = y, )) +       #label =
paste(lineend, linejoin)
  theme(panel.background = element_rect(fill = NA),axis.text =
element_blank(),axis.ticks = element_blank()) +
```

```
  geom_segment(
     colour = "blue",
     lineend = df2$lineend, linejoin = df2$linejoin,
     size = 4, arrow = arrow(length = unit(0.3, "inches"))
  ) +
  #geom_text(hjust = 'outside', nudge_x = -0.2) +
  xlim(0.5, 2)
aa

#pdf("arrow.pdf")
#plot(aa)
#dev.off()

ggsave("arrow.pdf")


#used Adobe Acrobat to edit, crop horizontal arrow
#used Adobe Acrobat to edit, select, rotate, crop vertical arrow



##########
########## magick
##########

### https://cran.r-project.org/web/packages/magick/vignettes/intro.html
### image_scale(image, "200"): resize proportionally to width: 200px
### image_scale(image, "x200"): resize proportionally to height: 200px
### stackoverflow.com/questions/44960520/join-images-with-data-frames-in-r


#####
##### top
#####

measurements = image_read_pdf("top_left.pdf")
simulations  = image_read_pdf("top_right.pdf")

#arrowhoriz   = image_read_pdf("arrow - horizontal.pdf")
#arrowdown    = image_read_pdf("arrow - vertical3.pdf")

#top = c(measurements,arrowhoriz,simulations,arrowdown)  #if including arrows
top = c(measurements,simulations)

# concatenate them left-to-right (use 'stack=T' to concatenate top-to-bottom)
side_by_side = image_append(top, stack=F)
#top2         = image_scale(side_by_side, "730")  #if including arrows
top2         = image_scale(side_by_side, "640")  #see transform in magick.pdf

image_write(top2, density=300, path = "top_example.tiff", format = "tiff")
#density=300 needed for 300 dpi
#top3        = image_read("top_example.tiff")


# use Paint, File, Properties to examine TIFF DPI and dimensions in inches or pixels
```

```
#####
##### bottom
#####

bias         = image_read("Biased Estimates Example.tiff")
sliver       = image_read("sliver.tiff")

#bottom = c(sliver,sliver,sliver,bias)
bottom = c(bias)

# concatenate them left-to-right (use 'stack=T' to concatenate top-to-bottom)
bottom_pair  = image_append(bottom, stack=F)
#bottom2      = image_scale(bottom_pair, "730")
bottom2      = image_scale(bottom_pair, "670")

image_write(bottom2, path = "bottom_example.tiff", density=300, format = "tiff")
#bottom3     = image_read("bottom_example.tiff")


# concatenate top & bottom (use 'stack=T' to concatenate top-to-bottom)
all = c(top2,bottom2)
vertical = image_append(all , stack=TRUE)
image_write(vertical, path = "TOC Graphic example.tiff",density=300, format = "tiff")
#,density=300)
```

**TOC Graphic example.tiff**



Notes

- Open with Paint, File, Properties to examine TIFF dimensions in inches or pixels and DPI
- After pasting in DOCX, right click, choose Size and Position for dimensions
- **arrow.pdf** is created but not incorporated above. Comments in the code describe using Adobe Acrobat to edit, crop, and rotate.