II2202: Advanced Quantitative tools

G. Q. Maguire Jr.
&
Marilyn E. Noz

2011.09.16

Independent versus **dependant** variables

- Independent variable – a variable that you can **change**

- Dependant variable
  - A response or outcome
  - This is what you will **measure**

Types of data

- Nominal data: unordered groups, e.g., male/female, left-handed/right-handed, …
- Ordinal data: rank ordered; the difference between item numbered n and n+i does not tell you anything other than that one is ranked ahead of the other, e.g. Top 500 Universities, top 10 protocols in bytes, …
- Interval data: continuous ranges mapped to some scale, without a clear zero
- Ratio data: like interval data but with a clear absolute zero value

Metrics

Type of data	Example Metrics	Common statistics
Nominal data	Success/failure	Frequencies, Chi-square
Ordinal data	Ranking	Frequencies, Chi-square, Wilcoxian rank sum tests, Spearman rank correlation
Interval data	Likert scale, System Useability Scale,	All descriptive statistics (average, median, std. dev., …), Student's t-test, ANOVA, correlation, regression, …
Ratio data	Task completion time, packet inter-arrival time, …	All of the previous + geometric mean

Adapted from Table 2.3 on page 23 of [1]

Measures of Central Tendency

Three most common measures are:

Mean	arithmetic average
Median	mid point of the distribution (half the values are larger and half are smaller)
Mode	most common value

Selecting participants

- **Random** sampling
- **Systematic** sampling – e.g. every 3rd person
- **Stratified** sampling – based upon a representative subset
- **Samples of convenience**
  - Who an you get?
  - Are they representative of the target population?

Sample size

- What is the goal?

  Is the difference expected to be large or small?

- What is an acceptable margin of error?

Within-subjects versus between-subjects

- Within-subjects
  - Also known as repeated-measures
  - The same subject, but repeated measurements
- Between-subjects
  - Comparing results of subject_i with subject_k
  - Avoids carry-over effects (where the subject learns from one trial and this causes a difference in subsequent trials)
- Mixed design

Counterbalancing

To avoid carryover effects vary the order of the tasks:

- Randomize order
- Sets of predefined orders – subject is randomly assigned to one of these sets

Error bars

Why show error bars?

- To convey to the viewer the expected range of values that might be expected
- Between the whiskers is the total confidence interval (CI) within which you are working
  - This might be shown as 90%, 95%, or 99%
  - These correspond to 10%, 5%, and 1% probability that the true value is **outside** this range

Error Bars in R

Use the package “gplots”

Reference manual “gplots.pdf” gives instructions for using plotCI - also available from help(plotCI) after the package has been loaded.

CI = confidence interval

For a good set of example code with plots drawn – the plots are at the end – see

http://www.oga-lab.net/RGM2/func.php?rd_id=plotrix:plotCI

Error Bars in R – example – read in data and format for finding CI

```
# error in distance difference in scans 2 and 3 (both trials):
# normal distribution
# error difference from expected

errorbars1 <- function()   (create simple function)
{
library(gplots)

expected1 <- read.csv(file="Hip-phantom-scan-procedure-series-
        1-2-3a.csv",header=TRUE,sep=",");
phant2 <- read.csv(file="hip_stats2.csv",header=TRUE,sep=",");
phant2a <-
        read.csv(file="hip_stats2a.csv",header=TRUE,sep=",");
phant3 <- read.csv(file="hip_stats3.csv",header=TRUE,sep=",");
phant3a <-
        read.csv(file="hip_stats3a.csv",header=TRUE,sep=",");
```

```
x1 <- expected1[3:15,9] - phant2[3:15, 20]
x2 <- expected1[3:15,9] - phant2a[3:15, 20]
x3 <- expected1[3:15,9] -  phant3[3:15, 20]
x4 <- expected1[3:15,9] - phant3a[3:15, 20]

# First make an x5 that exists as a vector
x5<-c(1:28)
x5[1:7] <- x1[1:7]
x5[8:14] <- x2[1:7]
x5[15:21] <- x3[1:7]
x5[22:28] <- x4[1:7]
print(x5)
```

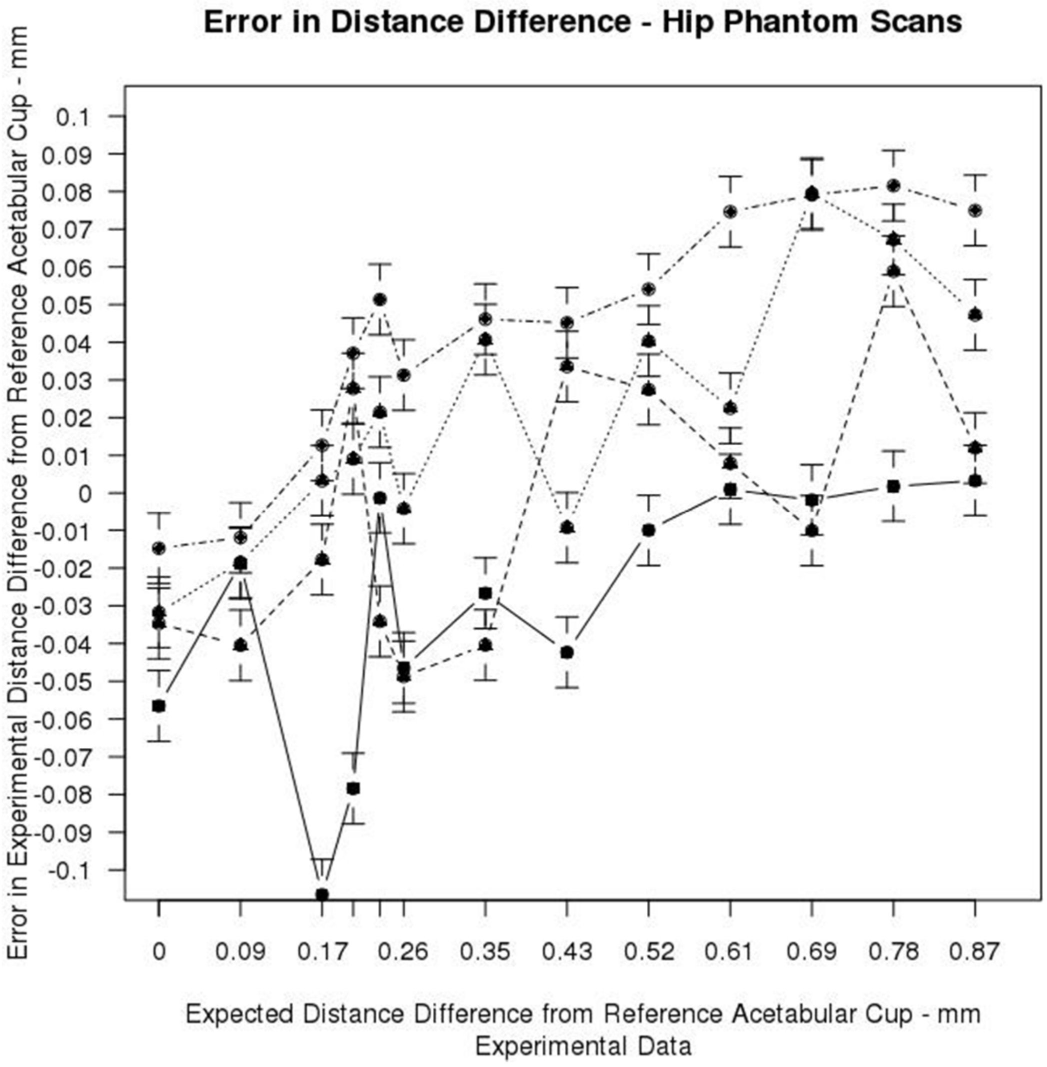
Error Bars in R – find 99% CI

```
meanex <- mean(x5)
print("mean")
print(meanex)
SDex <- (sqrt(var(x5)))
print("SD")
print(SDex)
upperCI <- meanex +(2.58*SDex/(sqrt(length(x5))))
print("upperCI")
print(upperCI)
lowerCI <- meanex - (2.58*SDex/(sqrt(length(x5))))
print("lowerCI")
print(lowerCI)
totalCI <- upperCI - lowerCI
print("totalCI")
print(totalCI)
```

Error Bars in R – Plot Graph with error bars

```
plot(expected1[3:15,9], x1, type="b", ylab="Error in Experimental Distance Difference from
     Reference Acetabular Cup - mm", xlab="Expected Distance Difference from
     Reference Acetabular Cup - mm",  ylim=c(-0.1, 0.1), xlim=c(0.0,0.9), main="Error in
     Distance Difference - Hip Phantom Scans",sub="Experimental Data", axes=F,pch=15,
     lty=1)

points(expected1[3:15,9], x2, type="b",pch=17, lty=2)
points(expected1[3:15,9], x3, type="b",pch=17, lty=3)
points(expected1[3:15,9], x4, type="b",pch=18, lty=4)

plotCI(expected1[3:15,9], x1, totalCI,  pch=21, pt.bg=par("bg"), add=TRUE)
plotCI(expected1[3:15,9], x2, totalCI,  pch=21, pt.bg=par("bg"), add=TRUE)
plotCI(expected1[3:15,9], x3, totalCI,  pch=21, pt.bg=par("bg"), add=TRUE)
plotCI(expected1[3:15,9], x4, totalCI,  pch=21, pt.bg=par("bg"), add=TRUE)

par(mfrow = c(1, 1))
# Note for docs on plotCI see gplots.pdf and web site given above
box()
axis(1, at=expected1[2:15,9], labels=round(expected1[2:15,9],digits=2), las=1)
axis(2, at=seq(-0.1, 0.1, 0.01),labels=round(seq(-0.1, 0.1, 0.01), digits=2), las=2)
}
```

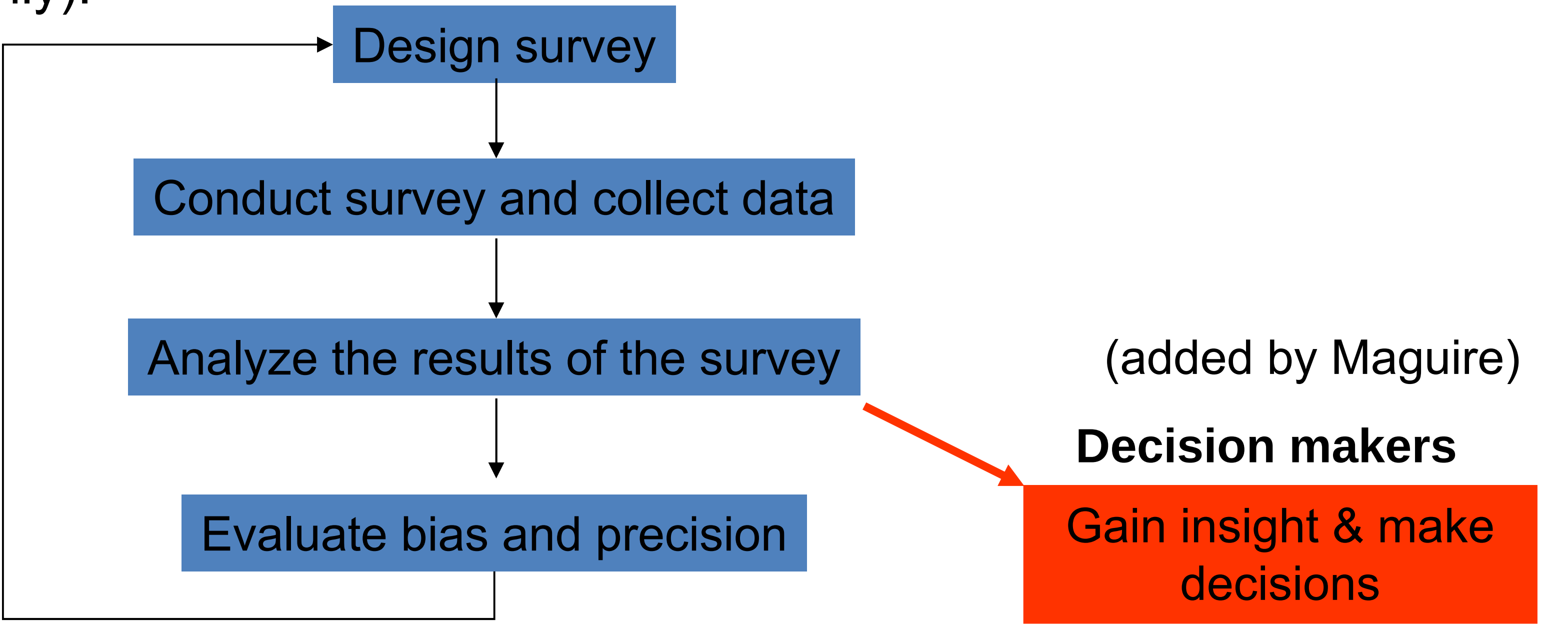
Error Bars in R – Resulting Plot



(Starting) Quantitative analysis of survey data

Overview

Gillian Raab, Professor of Applied Statistics at Napier University, shows the process of carrying out surveys as view by a statistician (roughly):


Design survey
Conduct survey and collect data
Analyze the results of the survey
Evaluate bias and precision
(added by Maguire)
Decision makers
Gain insight & make decisions

Adapted from the figure on his slide 7 in “Background to P|E|A|S project”, 9 September 2004, http://www2.napier.ac.uk/depts/fhls/peas/workshops/workshop1presentationGR.ppt

Objective

- What is the object of the survey?
  - Finding a **predictive** model
  - Finding **hidden** relationships
  - Segmenting a population into **strata**
  - **Visualizing** responses
    (e.g., Distance from a park versus frequency of visits to this park)
  - Making a **decision** (e.g., where to put a park)
- What is (are) the research question(s)?

Considerations when designing studies

Ken Kelley and Scott E. Maxwell state:

"At a minimum, the following points must be considered when designing studies in the behavioral, educational, and social sciences:

(a) the question(s) of interest must be determined;

(b) the population of interest must be identified;

(c) a sampling scheme must be devised;

(d) selection of independent and dependent measures must occur;

(e) a decision regarding experimentation versus observation must be made;

(f) statistical methods must be chosen so that the question(s) of interest can be answered in an appropriate and optimal way;

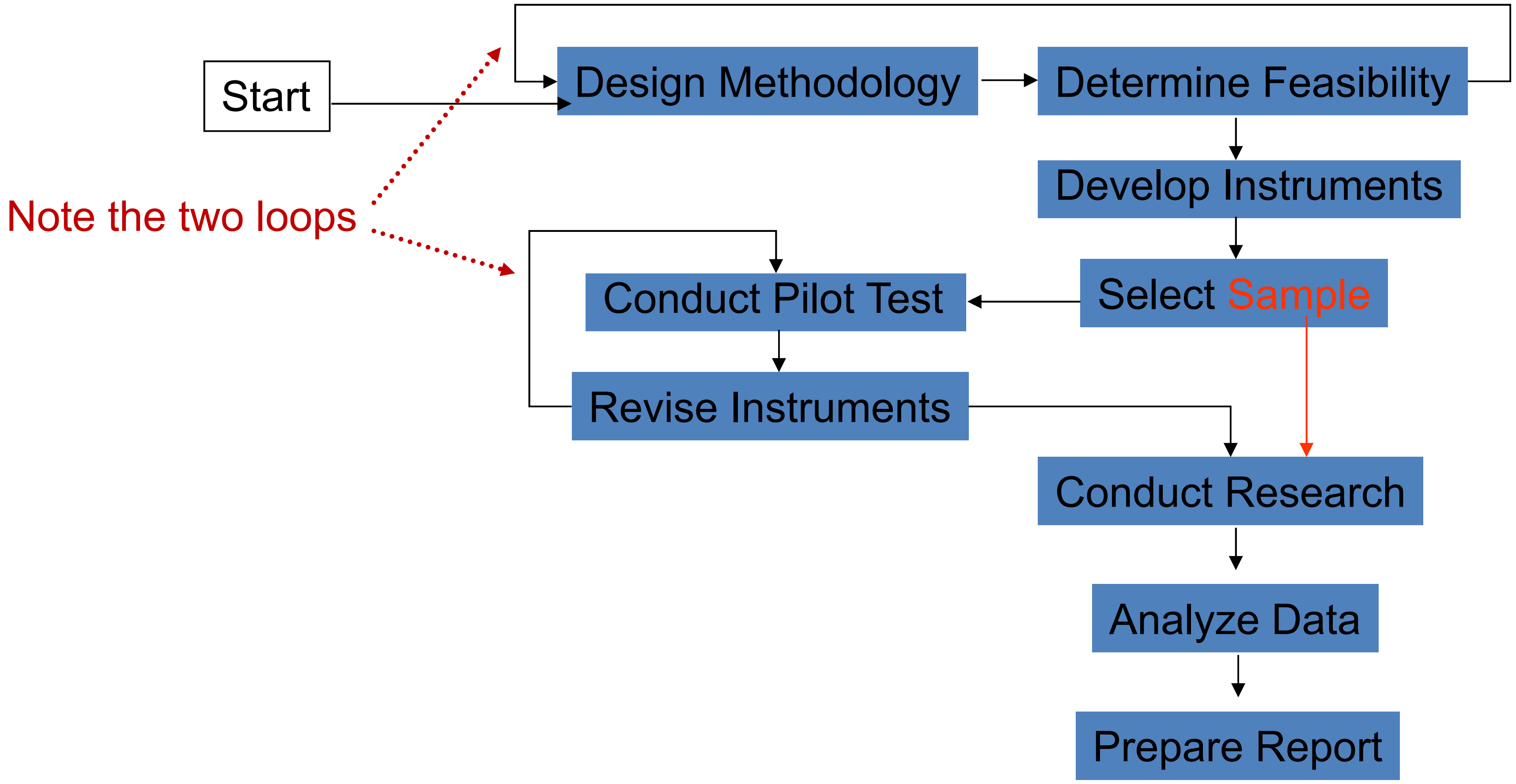
(g) sample size planning must occur so that an appropriate sample size given the particular scenario, as defined by points a through f, can be used;

(h) the duration of the study and number of measurement occasions need to be considered;

(i) the financial cost (and feasibility) of the proposed study calculated."

Ken Kelley and Scott E. Maxwell, Sample Size Planning with Applications to Multiple Regression: Power and Accuracy for Omnibus and Targeted Effects, In P. Alasuuta, L. Bickman, & J. Brannen (Editors), Hand book of social research methods. Sage, Newbury Park, CA, USA, 2008, pp. 166-192
http://nd.edu/~kkelley/publications/chapters/Kelley_Maxwell_Chapter_SSMR_2008.pdf

Questionnaire Research Flow Chart



Note the two loops

Adapted from pg. 3 of David S. Walonick, *A Selection from Survival Statistics*, StatPac,Inc. Bloomington, MN, USA, 14 August 2010, ISBN 0-918733-11-1, http://www.statpac.com/surveys/

Sampling methods

- *Probability*
  - Random sampling & systematic sampling (every Nth person) ⇒ equal probability of selection ↓
  - Sampling proportional to size (PPS) – concentrates on the largest segments of the population
  - Stratified sampling (members of each stratum (a sub-population) share some characteristic)
  - Advantage: can calculate sampling error
- *Nonprobability*
  - Accidental, Haphazard, convenience sampling ⇒ these might not be representative of the target population
  - Purposeful – sampling with a purpose in mind
    - Modal instance sampling –focused on 'typical' case
    - Expert sampling – choosing experts for your samples
    - Quota sampling - proportional vs. non-proportional
    - Heterogeneity sampling – to achieve diversity in samples
    - Snowball sampling – get recommendations of other to sample, from your samples

For further details of Nonprobability sampling see: William M.K. Trochim, The Research Methods Knowledge Base, 2nd Edition, webpage: Nonprobability Sampling, Last Revised: 10/20/2006 http://www.socialresearchmethods.net/kb/sampnon.php

Sample size

Choosing the size of your sample is related to your expected signal to noise ration and your desired confidence.

Statisticians speak about **statistical power**, for details see http://www.socialresearchmethods.net/kb/power.php

See also:

Ken Kelley and Scott E. Maxwell, Sample Size Planning with Applications to Multiple Regression: Power and Accuracy for Omnibus and Targeted Effects, In P. Alasuuta, L. Bickman, & J. Brannen (Editors), Hand book of social research methods. Sage, Newbury Park, CA, USA, 2008, pp. 166-192 http://nd.edu/~kkelley/publications/chapters/Kelley_Maxwell_Chapter_SSMR_2008.pdf

S. E. Maxwell, K. Kelley, and J. R. Rausch. Sample size planning for statistical power and accuracy in parameter estimation. Annual Review of Psychology, 59, 2008, pages 537-563. http://nd.edu/~kkelley/publications/articles/Maxwell_Kelley_Rausch_2008.pdf

K. Kelley and S.E. Maxwell, Sample size for multiple regression: Obtaining regression coefficients that are accurate, not simply significant. Psychological Methods, 8(3), 2003, pages 305-321. http://nd.edu/~kkelley/publications/articles/Kelley_Maxwell_2003.pdf

K. Kelley, S.E. Maxwell, and J.R. Rausch, Obtaining power or obtaining precision: Delineating methods of sample-size planning. Evaluation and the Health Professions, 26(3), 2003, pages 258-287. http://nd.edu/~kkelley/publications/articles/Kelley_Maxwell_Rausch_2003.pdf

K. Kelley, K. Lai, and P-J Wu. Using R for data analysis: A best practice for research. In J. Osbourne (Ed.), Best practices in quantitative methods, Sage, Newbury Park, CA, USA, 2008, pages 535-572. http://nd.edu/~kkelley/publications/chapters/Kelley_Lai_Wu_Using_R_2008.pdf

Getting started with data analysis

Assuming that the survey has already be conducted and that the data has been entered into a computer system, what is the next step?

- **Preliminary analysis**
  - Descriptive statistics
- **Exploratory data analysis**
  - Plots (points, lines, scatterplots, …), histograms, …

Types of analysis

- Design-based analysis
  - In this approach randomness is **induced** by the random selection of sample or the assignment of samples to a subset
  - Choice of a statistical model can be used for model-based inference
- Model-based analysis

  In this approach randomness is because of the **innate** randomness in the measurements (in the case of surveys – these are the responses)

Modeling techniques

- Prediction, classification (using neural networks, bayesian networks, trees, …), regression
- Clustering, segmentation
- Fitting to an *apriori* model
- Factor analysis, principle components analysis

Weights

When we have samples, we need to make sure that these samples are representative of the total population – to do this we may need to establish weights

For details of weights see:

James R. Chromy and Savitri Abeyasekera, "Statistical analysis of survey data", Chapter XIX, In Household Sample Surveys in Developing and Transition Countries. United Nations, New York, NY,2005. http://www.cpc.unc.edu/projects/addhealth/data/guides/weight1.pdf

Significance

Significance is a statistical term indicating *your confidence* in your conclusion that a real difference exists or that a relationship actually exists, i.e., that the result is unlikely to be due simply to chance.

If your hypothesis states a direction of this difference – use a **One-Tailed** significance test, otherwise use a **Two-Tailed** significance test.

Note: Significant **does not** imply important, interesting, or meaningful!

Similarly not all observations that are not statistically significant are unimportant, uninteresting, …

Testing for significance

1. Decide on your significance level α
2. Calculate your statistical value p
3. If p < α, then the result is significant, else it is not significant

An alternative view is:

$$\text{confidence} = (\text{signal/noise}) * \sqrt{\text{sample size}}$$

For details of the above equation see: David L. Sackett, Why randomized controlled trials fail but needn't:2. Failure to employ physiological statistics, or the only formula a clinician-trialist is ever likely to need (or understand!). Canadian MedicalAssociation Journal (CMAJ), 165(9):1226-37, 30 October 2001PubMedID (PMID): 11706914 http://www.cmaj.ca/cgi/content/full/165/9/1226

See also: Understanding Hypothesis Testing: Example #1, Department of Statistics, West Virginia University, last modified 4 April 2000 http://www.stat.wvu.edu/SRS/Modules/HypTest/exam1.html

Next steps

1. Search the literature and read extensively
2. Consult a statistician to get help with your statistical analysis
   (In most cases this is going to cost you money, but can save you a lot of time and effort.)

Selected Topics in Quantitative analysis

Power

Parametric and non-Parametric Data

If your data is “normally distributed”, then it can be examined using “parametric” statistics such as mean, variance, standard deviation etc.

In R:

```
meandata<- mean(data_set)

variance.data<-var(data_set)

std.data<-(sqrt(var(data_set)))
```

Normal or Parametric Data

```
data.new ← Read.table(“tmp.data.head”, header=TRUE)
t.test(data.new[,2])
        One Sample t-test
data:  data.new[, 2]
t = 2876.076, df = 41, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 11.05714 11.07268
sample estimates:
mean of x
 11.06491
```

What does this mean? The data is highly significantly different from zero.

Normal or Parametric Data - 2

Try using a mu (mean) different from zero:

```
t.test(data.new[,2], mu= 11.066)
```

```
One Sample t-test

data:  data.new[, 2]
t = 1.2767, df = 41, p-value = 0.069
alternative hypothesis: true mean is not equal to 11.06
95 percent confidence interval:
 11.05714 11.07268
sample estimates:
mean of x
 11.06491
```

This is much better. Now p > 0.05 meaning the data is not significantly different from 11.06.

Two Sample Student’s t-test

```
data.new2 ← read.table(“tmp.data.head2”,
  +header = TRUE)
```

```
t.test(data.new[,2], data.new2[,2],
  +paired=TRUE)
```

Get values.

Non-Parametric Data

It is necessary to use the non-parametric statistics when the data is not normal.

Usually the Mann-Whitney U test is recommended.

In R can use the Wilcoxon Signed Rank Test which is very similar.

```
wilcox.test(data.new[,2], mu=11.066)
    Wilcoxon signed rank test with continuity correction

data:  data.new[, 2]
V = 518, p-value = 0.4092
alternative hypothesis: true location is not equal to 11.066

Warning message:
In wilcox.test.default(dov.head[, 2], mu = 11.066) :
  cannot compute exact p-value with ties
```

Statistical Power

The power level of one or more data sets describes how good the inferences drawn from your data are likely to be.

For example, you make 119 measurements of some quantity in millimeters Then you determine your data is best analyzed using a one sample Student's t-test, and you find that the mean of your sample measurements is 11.06 mm.

Now the question to ask is:
How likely is it that **someone else** who performs these measurements will also get a mean of 11.06 mm?

This is the "power" of your inference.

Statistical Power

In order to calculate power, it is necessary to calculate the effective sample size or "effect size" which is generally categorized as **small**, **medium**, or **large**.

In other words you need know how big the effect is expected to be. This you have to do by examining your data.

If the effect is small, more samples are needed to achieve the usually acceptable power level of at least 80%.

Statistical Power - Effect Size

The effect size associated with your data, is dependent on the type of test that is involved in analyzing the sample - ANOVA, Student's t-test, proportions, chisq (and also, when two tests are being compared, if the samples being analyzed are even or uneven). Once this is known, then the effect size can be calculated.

When the effect size has been determined, then the numbers of samples you have can be used to see what the power level of your data is. It is also possible to state the power level that you would like to have and compute the number of samples that you would need to achieve this power level.

In R, the package to obtain effect size and power is “pwr”.

Statistical Power – Calculate Effect Size

```
# enter data, examine it to determine which test is best for
# analysis (Student's t-test, ANOVA, proportions ...), and
# determine whether the effect is small, medium, or large.

# Read in the data:

data.new <- read.table("tmp.head_all", header=TRUE)

# examine data and decide on Student’s t-test
# invoke the library to be used

library(pwr)

cohen.ES(test="t", size="medium")
```

```
     Conventional effect size from Cohen (1982)
                    test = t
                    size = medium
                    effect.size = 0.5
```

Statistical Power–Calculate Power Given Sample Size

```
pwr.t.test(n = 119, d = 0.2, sig.level = 0.05, power = NULL,
    +type = "one.sample", alternative = "two.sided")
```

```
          One-sample t test power calculation
                    N = 119
                    d = 0.2
            sig.level = 0.05
                Power = 0.5808414
          alternative = two.sided
```

effect size

58% power level with 119 samples; this is **unacceptable**

We would like to have 80% or 90% power level, thus we must obtain more samples.

But if the effect size is **medium**, i.e., d = 0.5, then the power = 0.9997192, which is acceptable

Statistical Power–Calculate Sample Size Given Power

```
pwr.t.test(n = NULL, d = 0.2, sig.level = 0.05,
   +power = 0.90, type = "one.sample",
   +alternative = "two sided”)
```

```
One-sample Student's t-test power calculation
                    N = 264.6137
                    d = 0.2
                    sig.level = 0.05
                    power = 0.9
                    alternative = two.sided
```

effect size

For 90% power level, 265 samples are required if the effect size is small. If the power level is reduced to 80% power level, then fewer samples would be needed.

If effect size is **medium**, d = 0.5, the number of samples needed would only be only 44, which would clearly be a easier number of samples to obtain.

Writing functions in R

Simple Functions in R

One of the nice features in R is the ability to:

- Write new functions in R
- Write functions in R that perform many R functions in a sequence
- Call R functions from a C program
- Call C programs from an R function

Function to determine kidney function using a radionuclidic method:

```
GFR1 <- function(dose, standard, filtert, time)
{
print("Dose (counts/min-ml) is:", quote = FALSE); print(dose)
print("standard volume (ml) is:", quote = FALSE); print(standard)
print("Filter (counts/min-ml) at T is:", quote = FALSE); print(filtert)
print("Time (minutes) is:", quote = FALSE); print(time)

filt <- filtert * 0.94
dose1 <- dose * standard
first <- (-0.278 * time) + 119.1 + (2405./time)
print("A is:", quote = FALSE); print(first)
third <- first * logb(dose1/filt)
print("first term is:", quote = FALSE); print(third)
fourth <- (2.866 * time) - 1222.9 - (16820./time)
print("B is:", quote = FALSE); print(fourth)
five <- (third + fourth);
print("GFR1 is :", quote = FALSE); print(five)
}
```

Note that the first four lines suggest the **units** that must be used for a successful result!

Inference – Check data normality

```
eda.shape <- function(x)
{
        par(mfrow = c(2, 2))
        hist(x)
        boxplot(x)
        iqd <- summary(x)[5] - summary(x)[2]
        plot(density(x, width = 2 * iqd), xlab = "x",
+               ylab = "", type = "l")
        qqnorm(x, pch = 1)
        qqline(x)
        invisible()
}
```

Arrange layout of multigraph

Line continuation with “+”

Normality Tests

The qqnorm plot produces a quantile-quantile plot which has the data being tested on one axis and the corresponding quantiles of a standard normal distribution on the other.

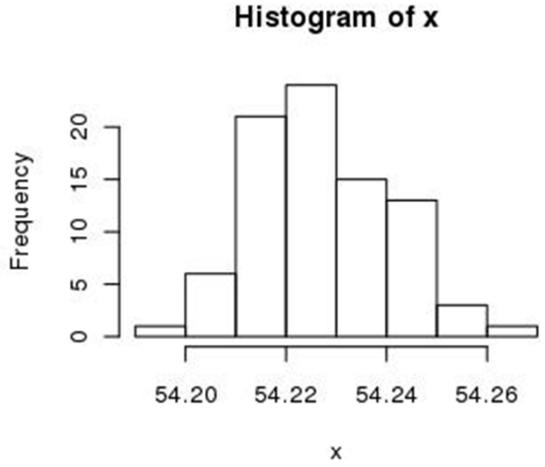
The density plot is a smooth version of the histogram; i.e., smooth estimates of the population frequency or probability density. Using “width=2iqd” in density, sets the degrees of smoothness of the density plot a good way.

Histogram and density = best picture of the population shape

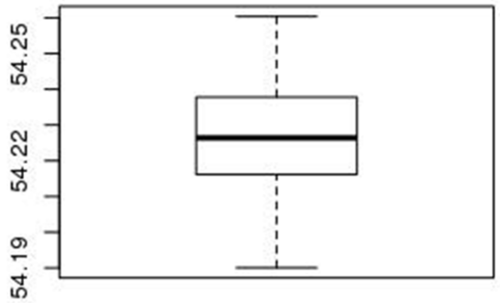
The qqnorm and boxplot = best picture of outliers

Result of simple check for Normality

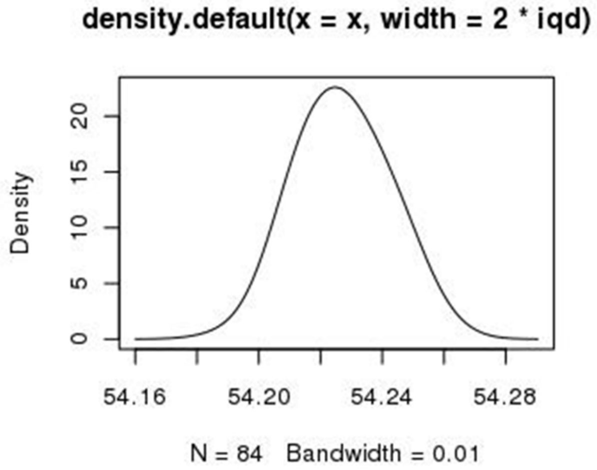
Histogram



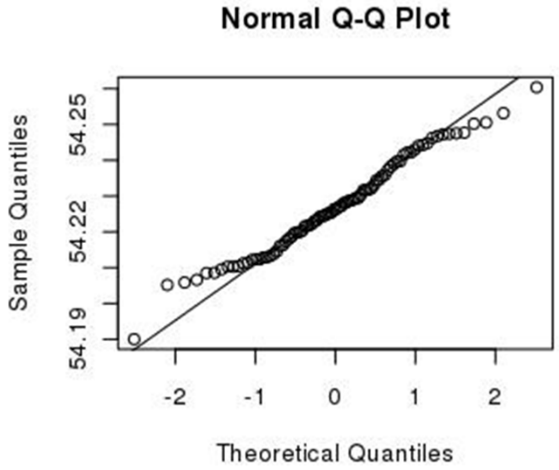
Boxplot



Plot of density



Quantiles-Quantiles plot



Inference #2 – put a title above a multiple graph

```
eda.shape <- function(x, y, z)
{
```

x is the data

y is the main title

z is the Y label for the box plot
and the X label for the histogram & density plot

```
        par(oma=c(0,0,5,0))
```

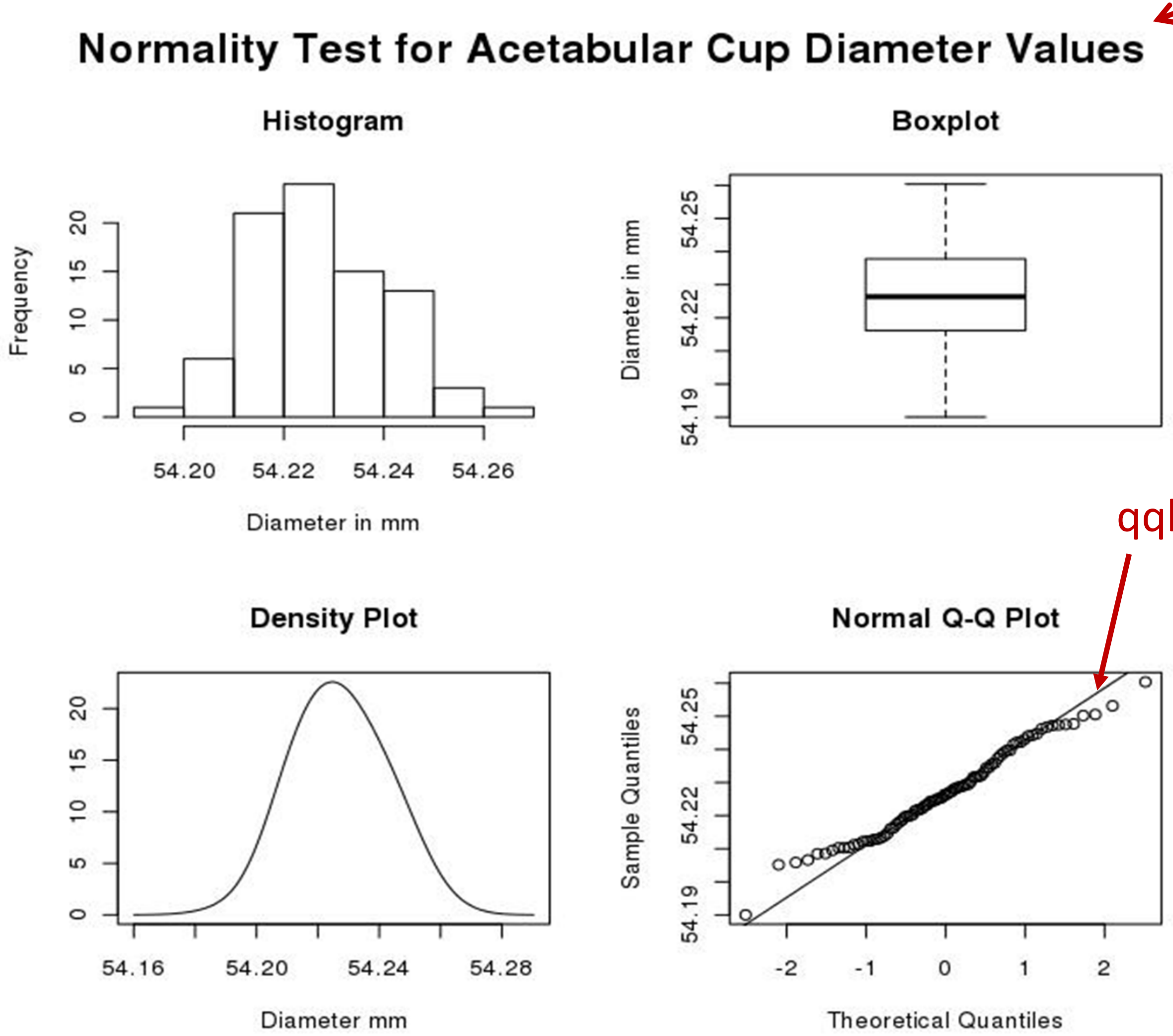
Add space for main title

```
        par(mfrow = c(2, 2))
         hist(x,xlab=z, main="Histogram")
         boxplot(x, ylab=z, main="Boxplot")
         iqd <- summary(x)[5] - summary(x)[2]
         plot(density(x, width = 2 * iqd), xlab = z, ylab = "", type = "l",
+                main = "Density Plot")
         qqnorm(x, pch = 1)
         qqline(x)
         invisible()
         mtext(y, side=3, outer=T, cex= 1.2)
```

Add main title

```
}
```

Result



main title

qqline

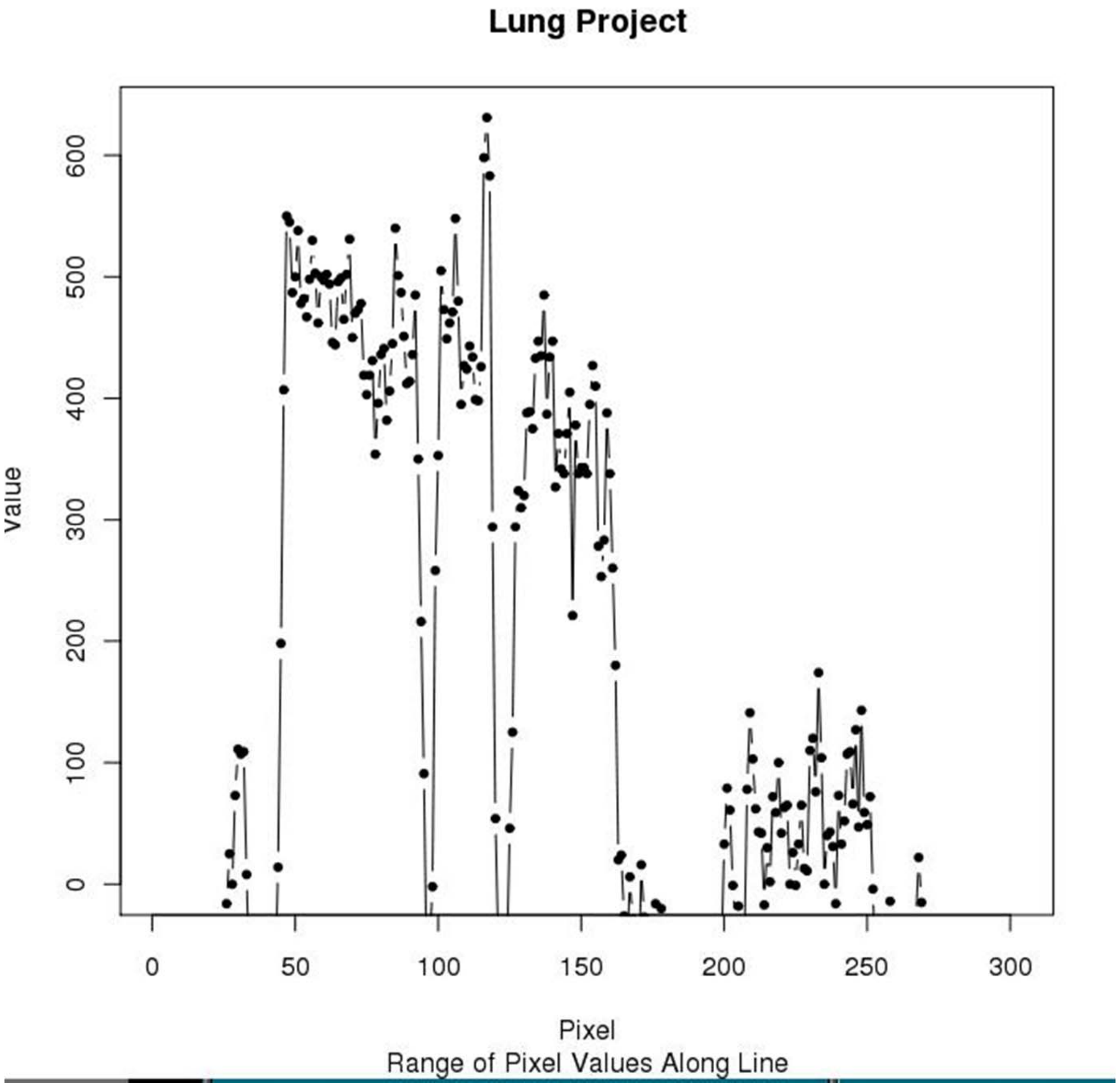
R Plot function:

```
rplot<-function(filename) {
print(filename)
x11(xpos=-50)
ybuff <- (scan(filename))
ymaxvalue <- max(ybuff)
yminvalue <- min(ybuff)
plot(ybuff, ylab="Value", xlab="Pixel", type="b",
+ylim=range(yminvalue,ymaxvalue), main="Lung
+Project", sub="Range of Pixel Values Along
+Line",pch=20)
system("sleep 5")
}
```

Call This Function from a C Program

In the C program:

```
/* open file for R instructions */
    sprintf(str,"%s/rfile.input", Dxmenu_Dir);
    if ( (fpR = creat(str, MODE)) == -1) {
      fprintf(stderr, "Cannot open file for R instructions.\n");
      return;
    }
 /* load the  file to be plotted */
    sprintf(datafilename, "%s_%d_%d_profile", name, slice,where);
/* Put this in the file opened above  and close the file*/
    sprintf(str, "rplot(\"%s\")\n", datafilename);
    write(fpR,str,sizeof(char)*strlen(str));
    close(fpR);
 /* run the R program */
   sprintf(str, "/bin/csh %s/rfile.sh %s", Path1, Path2);
   system(str);
  }
```


Lung Project
Value
0
100
200
300
400
500
600
0
50
100
150
200
250
300
Pixel
Range of Pixel Values Along Line

Form of rfile.input and rfile.sh

rfile.input:

```
rplot("lung_case_three_256_profile")
```

rfile.sh:

```
# shell script for making a plot
echo $0 $1
# remove the output file before you try to write it
#
/bin/rm $1/rfile.output
#
# Run R from C program getting input from a command file
# and write the output to another file
#
/usr/bin/R -q --no-save <$1/rfile.input >$1/rfile.output
```

NOTE: the input/output directories could be different.

Call a unix program from an R function

```
invitro.cals ← function(string, flag, string2, string3)
{
# string is the directory location of all the files to be used
# string2 is the directory location of all the files if a radionuclide is being modeled
# string3 is the spine data to be used for the bone marrow sample calculation (see 22)
print("flag:", quote = FALSE)
print(flag)
thalf <- scan(paste(string, "std.decay.time", sep = ""))
filename <- paste(string, "decay.time", sep = "")
if(unix(paste("test_file", filename), output = F)) {
            thalf.model <- scan(paste(string, "decay.time", sep = ""))
}
else {
            thalf.model <- scan(paste(string, "std.decay.time", sep = ""))
}
if((flag == 0) || (flag == 5) || (flag == 10) || (flag == 15)) {
  filename <- paste(string, "counts.blood", sep = "")
    if(unix(paste("test_file", filename), output = F)) {
                    starter.file <- dget(paste(string, "counts.blood", sep = ""))
                    flag.invitro <- 1
    }
   else { ….
               }
           }   …..
```

Here is the call to the program

“if” Statements and “for” Loops

```
….

  if(flag.invitro == 0) {
     for(a in 1:count.len)
        counts[a, 1] <-
(as.numeric(starter.file[a]/(RF*(as.numeric(std))))*100000
     dput(counts, paste(string, "counts.blood", sep = ""))
   }
  else {
   for(a in 1:count.len)
     counts[a, 1] <- (as.numeric(starter.file[a]))
  }
  ……
}
```

Plot Formatting

```
cup.measures <- function()
{
phant1 <- read.csv(file="hip_stats1.csv",header=TRUE,sep=",")
diameter1 <- ((phant1[2:15, 10])*2)

# Plot the numbers 1-14 (on x) against the diameter (on y)
# choose labels on the x and y axis
# choose  limits for the x and y axis
# choose a main and sub title
# choose a plotting type – lines “l”, symbols “p”, or both “b”
# choose a symbol type – a number indicates a built in symbol
# or one can indicate a symbol by pch=”sym”, e.g., pch=”ö”
# choose a line type – a number of line types are available by number

plot(c(1:14),diameter1,xlab="Individual Scans",ylab="Diameter in mm",
+ylim=c(54.18,  54.27), xlim=c(0,15),main="Acetabular Cup Diameter",
+sub="Experimental Data", type="b",pch=7, lty=1,axes=F)
```

Add Labels to the Points

```
# load library to plot labels
library(plotrix)

# Get labels
plotlabels <- phant1$labelr[2:15]

# plot labels
thigmophobe.labels(c(1:14),
diameter1,plotlabels,col="darkblue", font=2)

# label color is darkblue
# label font is bold
```

Add Another Plot to This One

```
phant1a <- read.csv(file="hip_stats1a.csv",header=TRUE,sep=",")
diameter1a <- ((phant1a[2:15, 10])*2)

# Plot
# Note: different symbol and different line type
points(c(1:14), diameter1a, type="b", pch=9, lty=2)

# Get labels
plotlabels <- phant1a$labelr[2:15]

# plot labels
thigmophobe.labels(c(1:14), diameter1a, plotlabels, col="darkgreen",
    +font=2)

# continue adding as many plots as wanted
# note that one can minutely control every aspect of a plot
# use 'help(par)' for all the gory details
```

Do Some Statistics and Add to Plot

```
# do mean and SD *2
total_cup <- c(diameter1, diameter1a, ...)
meanc <- mean(total_cup)
medianc <- median(total_cup)
SD <- sqrt(var(total_cup))
SD2 <- SD * 2
meanplus <- meanc + SD2
meanminus <- meanc - SD2

# add to plot
ylmean<-meanc + 0.003
text(0.2,ylmean,"Mean", srt=0, crt=0)
points(c(0:41),rep(meanc,42),type="l", lty = 1)

ylmedian<-medianc - 0.003
text(0.4,ylmedian,"Median", srt=0, crt=0)
points(c(0:41),rep(medianc,42),type="l", lty = 1)

ylup <- meanplus + 0.004
Text(1.3,ylup,"Mean Plus 2SD", srt=0, crt=0)
points(c(0:41),rep(meanplus,42),type="l", lty = 1)

yldn <- meanminus + 0.004
text(1.5,yldn,"Mean Minus 2SD", srt=0, crt=0)
points(c(0:41),rep(meanminus,42),type="l", lty = 1)
```

Finish Plot

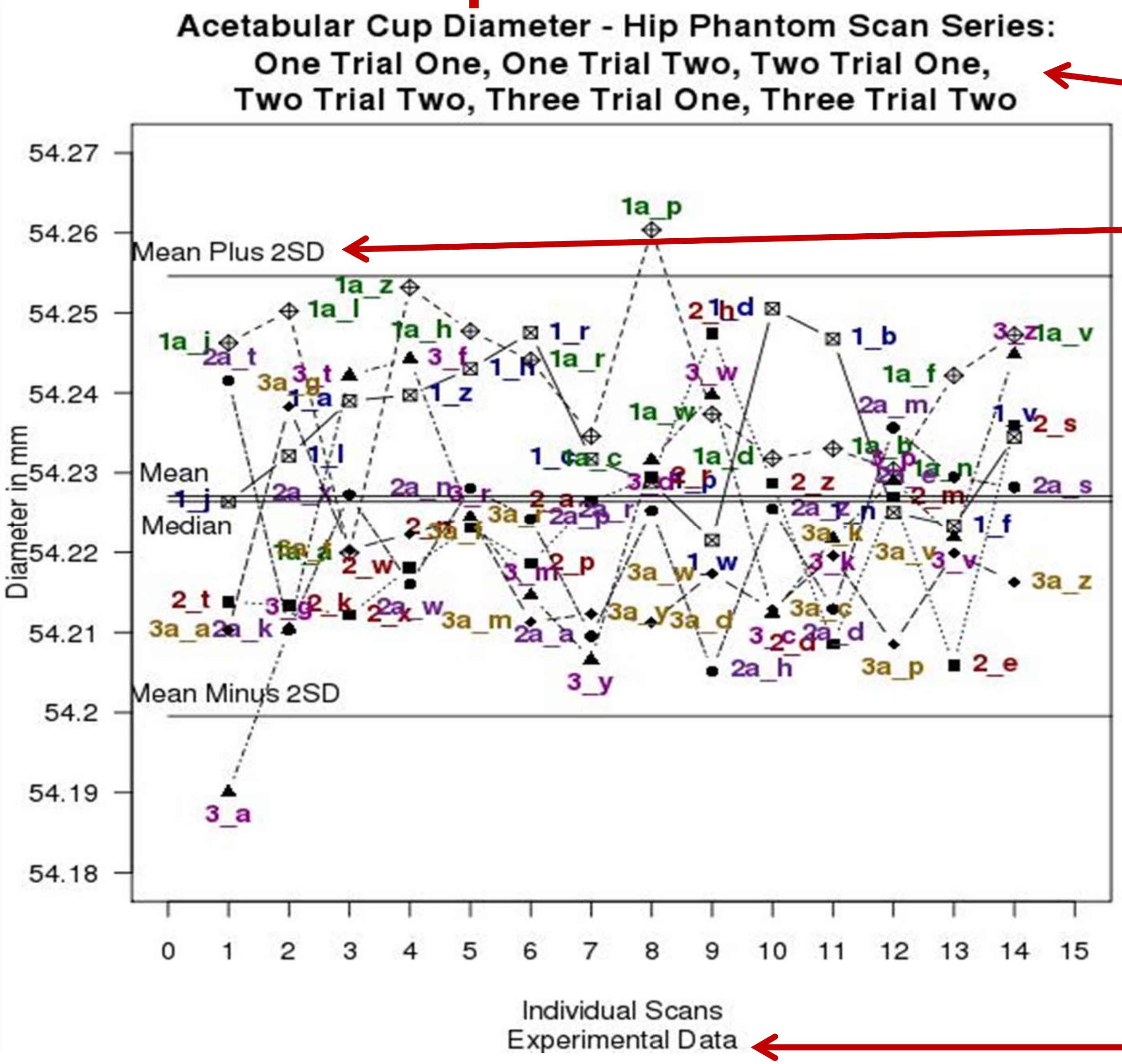
```
# fix the axes and tick marks
# first draw a box
box()

# Now fix the x-axis indicated by “1”
# indicate where to draw the tick marks
# indicate the labels to be used
# indicate the orientation of the labels – parallel, horizontal,
# perpendicular, vertical,
axis(1, at=c(0:15),labels=c(0:15), las=1)
```

las = 1 sets orientation parallel

```
# Now fix the y-axis
axis(2, at=seq(54.18, 54.27, 0.01),
   +labels=round(seq(54.18, 54.27, 0.01), digits=2), las=2)
```

las = 2 sets orientation horizontal

Example Finished Plot



main title

ylup Text

subtitle

Figure Legends

For some plots it might be necessary to add a legend. This can be placed inside or outside the actual plot. The format of a legend can be:

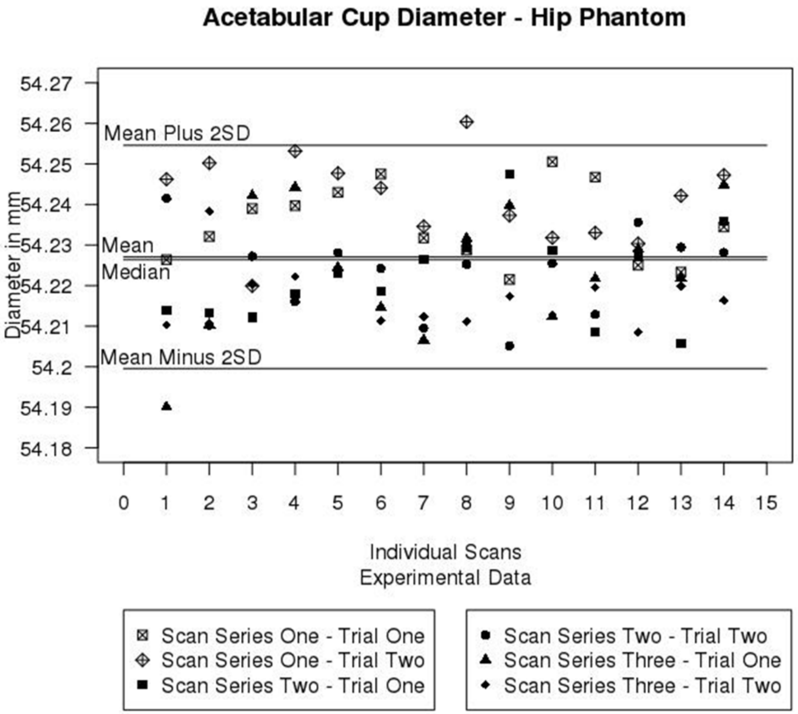
```
# place legend at x,y where these coordinates are derived from the graph

legend(x=tmp.u[1], y=tmp.u[4], legend=list("Scan Series One - Trial One","Scan Series One – Trial +Two", "Scan Series Two
      - Trial One", "Scan Series Two - Trial Two", "Scan Series Three - Trial +One", "Scan Series Three - Trial Two"),
      pch=c(7,9,15,16,17,18))
```

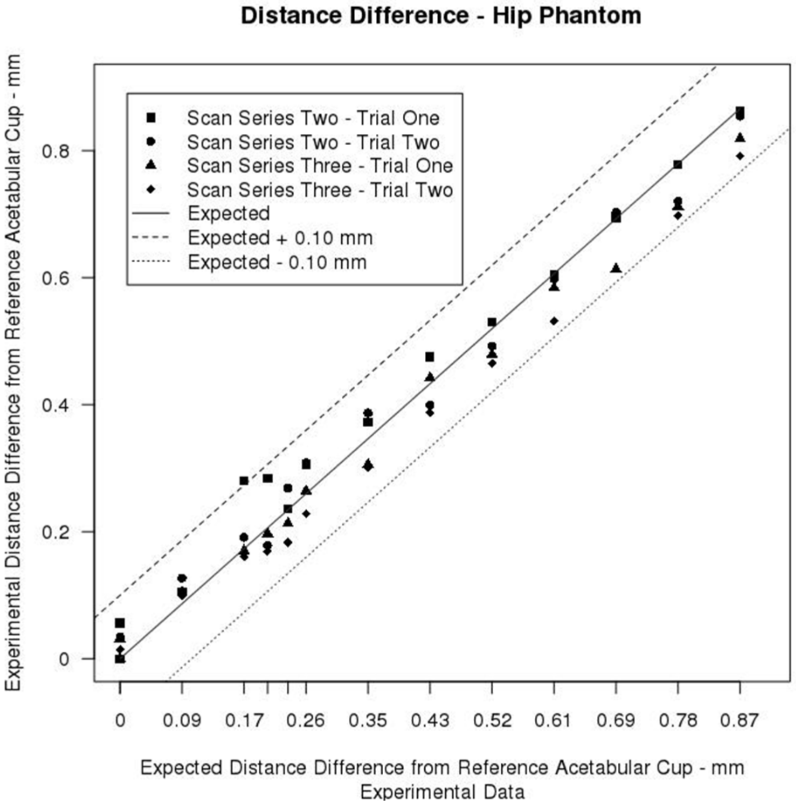
```
# break the above legend into two pieces and place outside the graph

legend(x=0.0, y=54.14, legend=list("Scan Series One - Trial One","Scan Series One - Trial Two",
      +"Scan Series Two - Trial One"), pch=c(7,9,15))

legend(x=8.0, y=54.14, legend=list("Scan Series Two - Trial Two", "Scan Series Three - Trial One",
      +"Scan Series Three - Trial Two"), pch=c(16,17,18))
```

```
# place the legend at an interactive point
#l ocator reads the position of the graphics cursor when the (first) mouse button is pressed

legend(locator(), legend=list("Scan Series One - Trial One","Scan Series One - Trial Two", "Scan + Series Two - Trial One",
      "Scan Series Two - Trial Two", "Scan Series Three - Trial One", "Scan +Series Three - Trial Two"),
      pch=c(7,9,15,16,17,18))
```

```
# use lines and points in graph and indicate which is which:

legend(x=0.01, y = 0.89, legend=list("Scan Series Two - Trial One", "Scan Series Two - Trial Two",
      +"Scan Series Three - +Trial One", "Scan Series Three - Trial Two", "Expected","Expected + 0.10 +mm",
      “Expected - 0.10 mm"), lty=c(-1,-1,-1,-1,1,2,3), pch=c(15,16,17,18,-1,-1,-1)
```

Example Plots





Remarks

Notice that in the previous set of slides, the example functions were just a set of functions which already existed in R.

It is convenient to work in an editor like emacs, try things out, find all the components needed to do the job and then save the set as an R function (e.g., cup.measures).

Data analysis

Data Analysis – Linear Regression

Read in data and set up factors:

```
Measure <- factor(LETTERS[1:6])

Measure
# [1] A B C D E F
# Levels: A B C D E F

pat_dat <-scan("tmp.patient_t1")
# Read 42 items
# pat_dat
# [1]  2.97 -6.54  1.17  0.20  0.66 -0.59  1.62 -8.20 -1.11  0.14  1.98  2.14 1.41 -7.68  0.79
      -0.16 -0.70 -1.24
# [19] 1.11 -3.52  3.21 -0.02 -0.28  1.04 1.67 -6.24  1.36  0.35  0.74  1.09  0.07 -0.73
      1.32 -0.41 -0.57  0.62
# [37]  1.96 -1.07  2.78  0.97  0.57  0.05

Patient<-factor(c(rep(1,6), rep(2,6), rep(3,6), rep(4,6), rep(5,6), rep(6,6), rep(7,6)))
# Patient
# [1] 1 1 1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 4 4 4 4 4 4 5 5 5 5 5 5 6 6 6 6 6 6 7 7 7 7 7 7
# Levels: 1 2 3 4 5 6 7
```

Data Analysis – Linear Regression

Make a data frame:

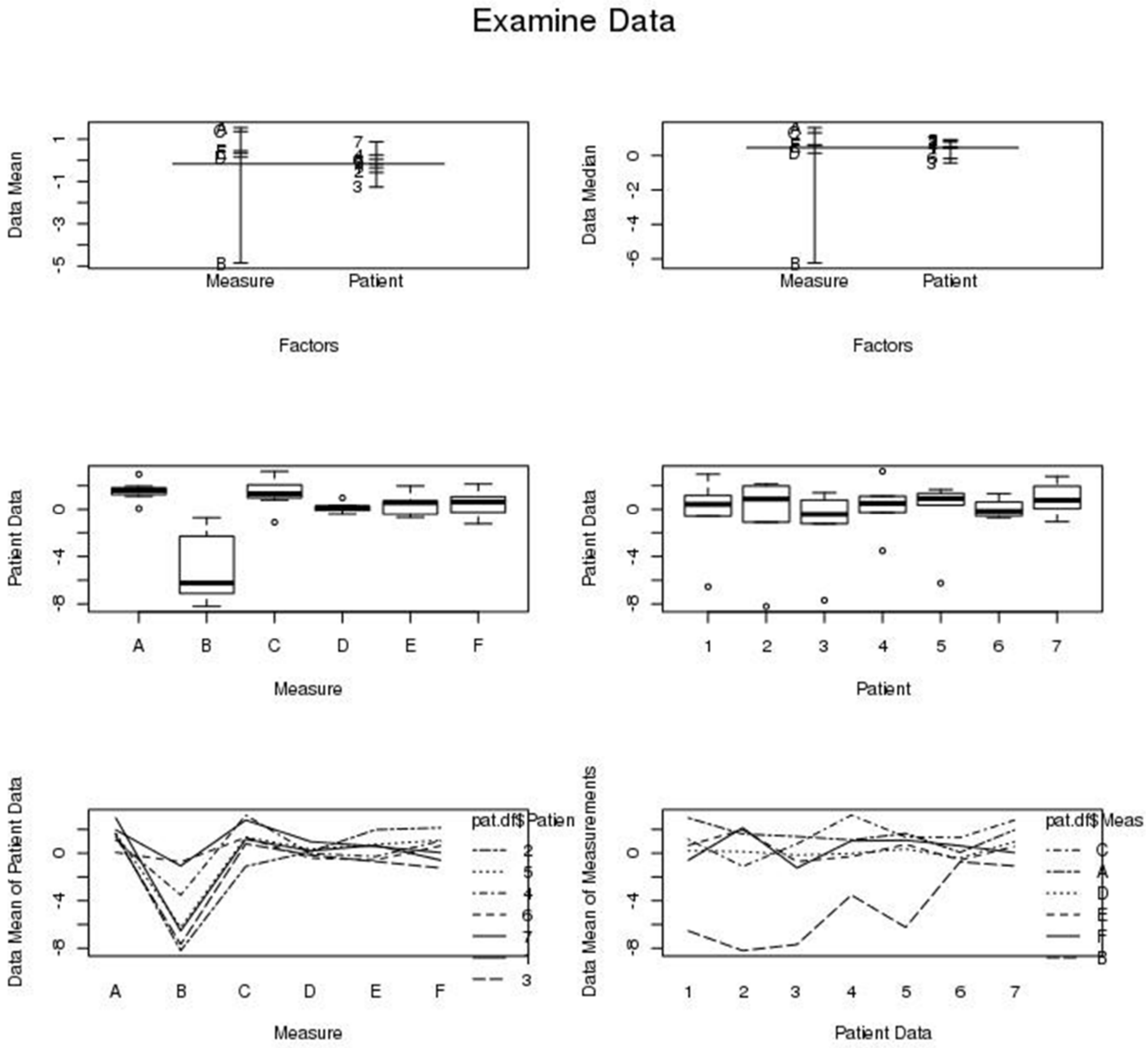
```
pat.df <-data.frame(Measure,Patient,pat_dat)

pat.df

   Measure Patient pat_dat
1       A       1    2.97
2       B       1   -6.54
3       C       1    1.17
4       D       1    0.20
5       E       1    0.66
6       F       1   -0.59
7       A       2    1.62
8       B       2   -8.20
9       C       2   -1.11
10      D       2    0.14
11      E       2    1.98
12      F       2    2.14
13      A       3    1.41
14      B       3   -7.68
15      C       3    0.79
16      D       3   -0.16
….
```

Data Analysis – Linear Regression

Graphically examine the data:

```
par(mfrow=c(3,2))
plot.design(pat.df)
plot.design(pat.df,fun=median)
plot(pat_dat~Measure+Patient, data=pat.df)
interaction.plot(pat.df$Measure,
+            pat.df$Patient,  +pat.df$pat_dat)
interaction.plot(pat.df$Patient,
+            pat.df$Measure,  +pat.df$pat_dat)
```

Data Analysis – Linear Regression



Data Analysis – Linear Regression

From the curves: there is no interaction between the patients and the measurements, but there is difference in the measurements for each patient

```
aov.pat1 <- aov(pat_dat~Measure*Patient, pat.df)
summary(aov.pat1)
```

“*” means interaction between these factors, i.e. Measure **depends upon** Patient

```
                Df   Sum Sq Mean Sq
Measure          5   195.768  39.154
Patient          6    16.301    2.717
Measure:Patient 30   71.452   2.382
```

Df = degrees of freedom

“+” means difference between these factors i.e. Measure is **independent** of Patient

```
aov2.pat1 <- aov(pat_dat~Measure+Patient, pat.df)
summary(aov2.pat1)
```

F test

```
           Df  Sum Sq Mean Sq   F value    Pr(>F)
Measure     5  195.768   39.154  16.4391   8.162e-08 ***
Patient     6   16.301    2.717   1.1407   0.3634
Residuals  30  71.452   2.382
```

Measurements are significantly different (as Pr < 0.05)

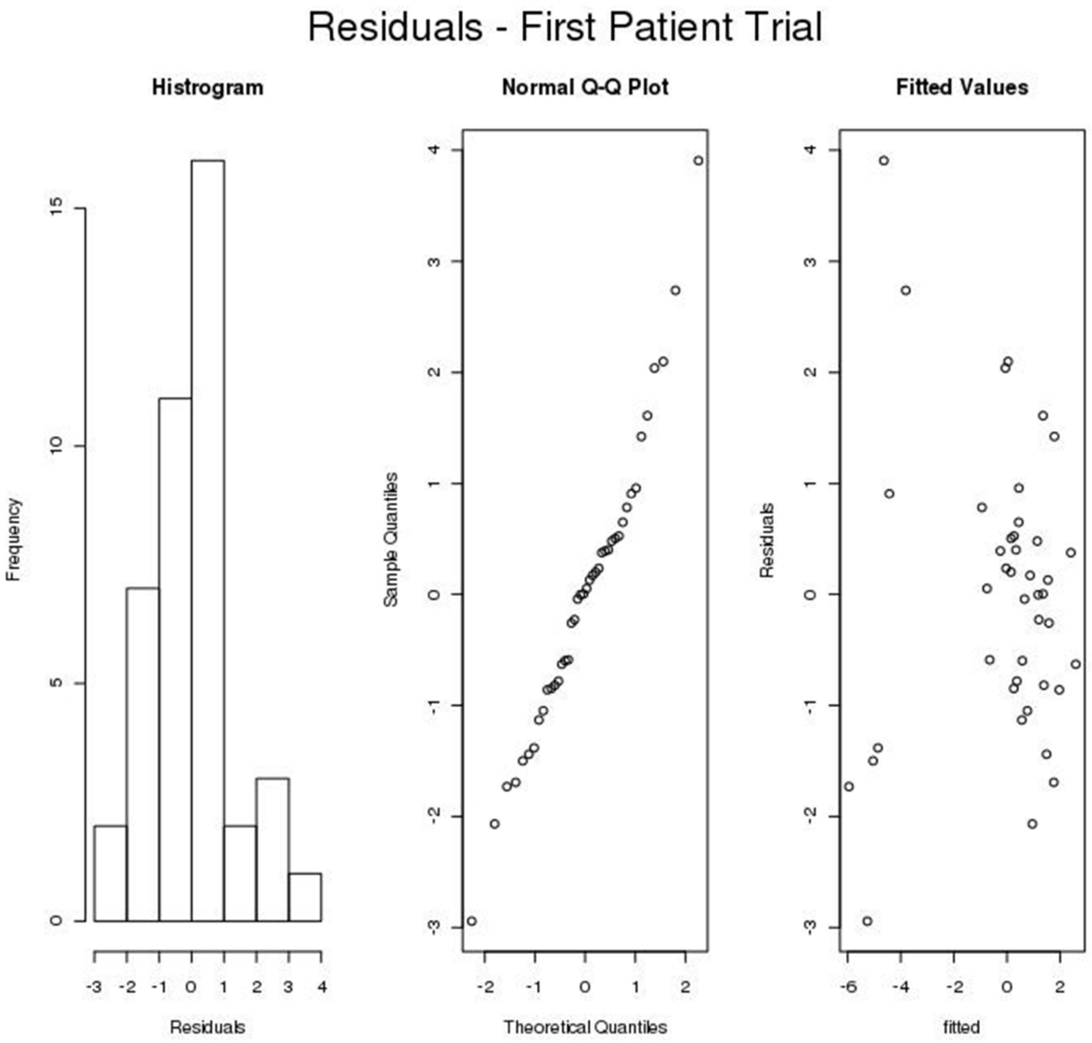
Patients are not significantly different (as Pr > 0.05)

Data Analysis – Linear Regression

Examine the residuals graphically:

```
par(oma=c(0,0,5,0))
par(mfrow=c(1,3))
hist(resid(aov2.pat1), main = "Histrogram",
+     xlab="Residuals", ylab = "Frequency")
qqnorm(resid(aov2.pat1))
plot(fitted(aov2.pat1), resid(aov2.pat1),
+     xlab = "fitted", ylab = "Residuals",
+     main = "Fitted Values")
mtext("Residuals - First Patient Trial", side=3,
+        outer=TRUE, cex=1.5)
```

Plots of the residuals



Multiple Plots 1

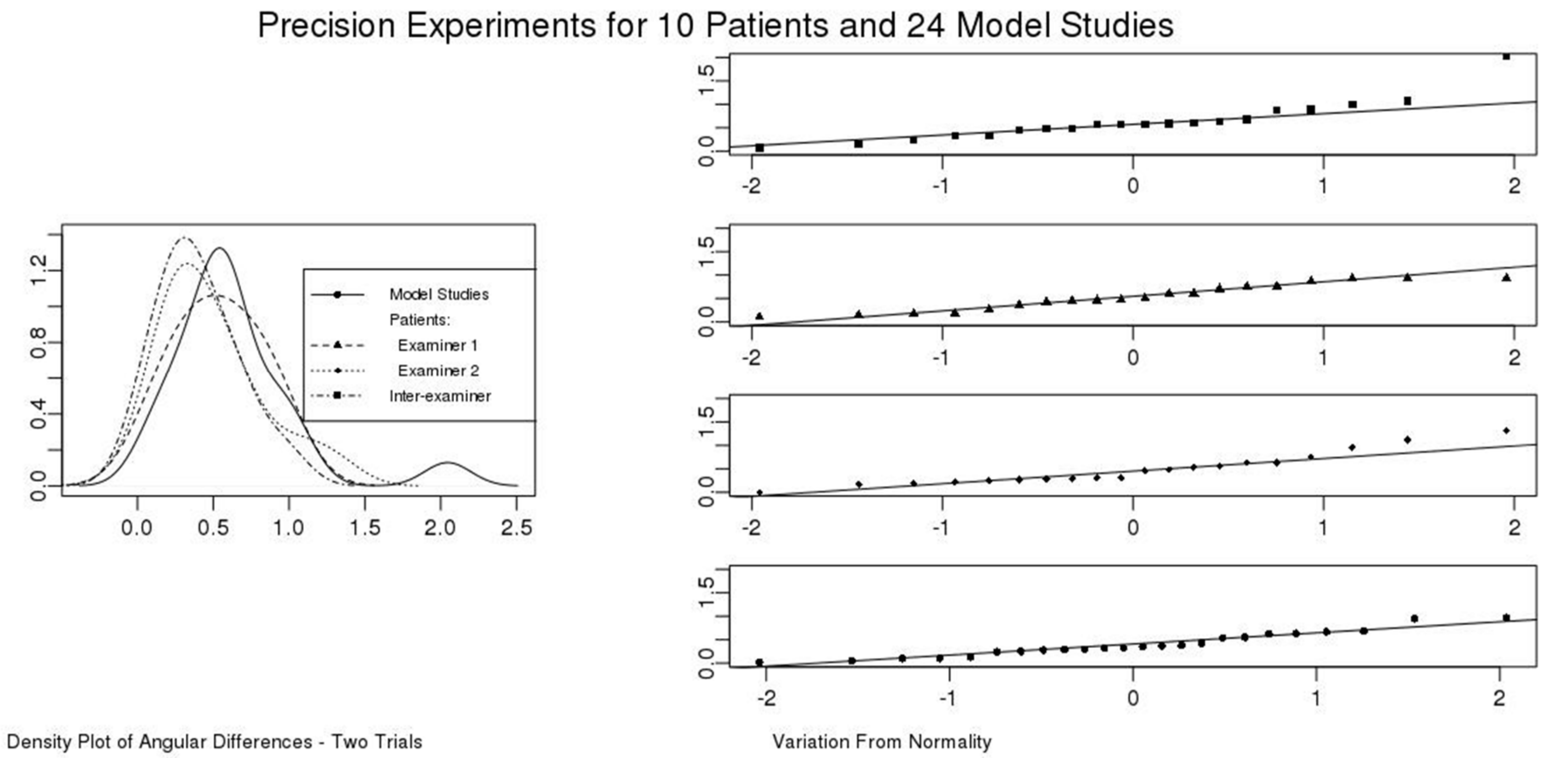
```
eda10.shape <- function(x,y,t,s)
{
          par(oma = c(10,0,5,0))
          par(fig = c(0.1, 0.4, 0.25, 0.75), mar = c(2,0.5,0.5,0.5), mgp = c(0,0.5,0))
#         par(fig = c(0.1, 0.4, 0.25, 0.75))
          iqd <- summary(x)[5] - summary(x)[2]
          plot(density(x, width = 2 * iqd), xlab = "", type = "l", ylim=c(0, 1.4), pch=15,
+       lty = 1, ylab = "", main = "")
          iqd <- summary(y)[5] - summary(y)[2]
          points(density(y, width = 2 * iqd), type = "l", xlab = "", ylab = "", pch=17,  lty = 2)
          iqd <- summary(t)[5] - summary(t)[2]
          points(density(t, width = 2 * iqd), type = "l", pch=18, xlab = "", ylab = "", lty = 3)
          iqd <- summary(s)[5] - summary(s)[2]
          points(density(s, width = 2 * iqd), type = "l", pch=16, xlab = "", ylab = "", lty = 4)
      legend(locator(1), legend=list("Model Studies", "Patients:", "  Examiner 1",
+     "  Examiner 2", "Inter-examiner"), lty=c(1,-1,2,3,4), pch=c(16,-1,17,18,15), cex=0.70)
      mtext("Density Plot of Angular Differences - Two Trials", side = 1, outer = T,
+     cex = 0.85, at = 0.2)
      mtext("Precision Experiments for 10 Patients and 24 Model Studies",
+     side = 3, outer = T, cex = 1.5)
#
```

Multiple Plots 2

```
     par(fig=c(0.5,1, 0.75, 1),mar = c(2,0.5,0.5,0.5),mgp=c(0,0.5,0), new=TRUE)
     qqnorm(x, pch = 15, xlab ="", ylab="", cex = 0.75, main="",ylim=c(0, 2.0))
     qqline(x)
     invisible()
     par(fig=c(0.5,1, 0.5, 0.75),new=TRUE)
     qqnorm(y, pch = 17, xlab ="", ylab="", cex = 0.75,main="",ylim=c(0, 2.0))
     qqline(y)
     invisible()
     par(fig=c(0.5,1, 0.25, 0.5),new=TRUE)
     qqnorm(t, pch = 18, xlab ="", ylab="", cex = 0.75, main="",ylim=c(0, 2.0))
     qqline(t)
     invisible()
     par(fig=c(0.5,1, 0, 0.25),new=TRUE)
     qqnorm(s, pch = 16, xlab ="", ylab="", cex = 0.75, main="",ylim=c(0, 2.0))
     qqline(s)
     invisible()
   mtext("Variation From Normality", side = 1, outer = T, cex = 0.85, at = 0.6)
}
```

Use as: eda10.shape(file1,file2,file3, file4)

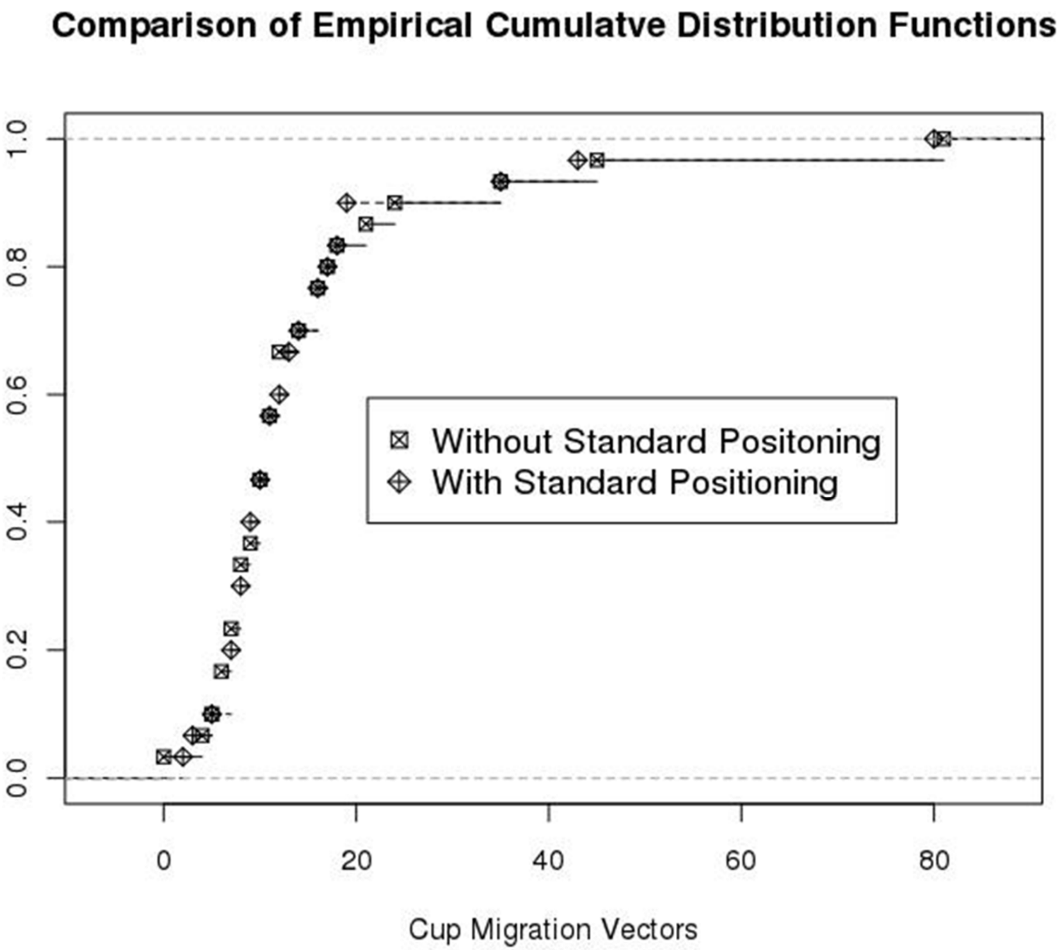
EXAMPLE:
eda10.shape(scan("lot_hen_pat"), scan("lotta_pat1-2"), scan("hen_pat1-2"), scan("tmp.lotta1-2"))



Multiple ECDF functions

```
 plot(ecdf(scan("tmp.after_std")),main="Comparison of
+    Empirical Cumulatve Distribution Functions", xlab = "Cup
+    Migration Vectors", ylab= "", sub = "", pch = 7)

plot(ecdf(scan("tmp.before_std")), add=TRUE, pch=9, lty = 2)


legend(locator(1), c("Without Standard Positoning","With
+    Standard Positioning"), pch=c(7,9), cex = 1.2)
```

Result with multiple ECDFs



Using Excel for statistics and plotting

Experiment 1 (again)

Captured packets using Wireshark during a long (2150.12 second) VoIP call

⇒ at least: 107,505 RTP packets in each direction

⇒429 RTCP packets in one direction

From network to local user agent

Raw output fom Microsoft Excel 2010 (Beta)

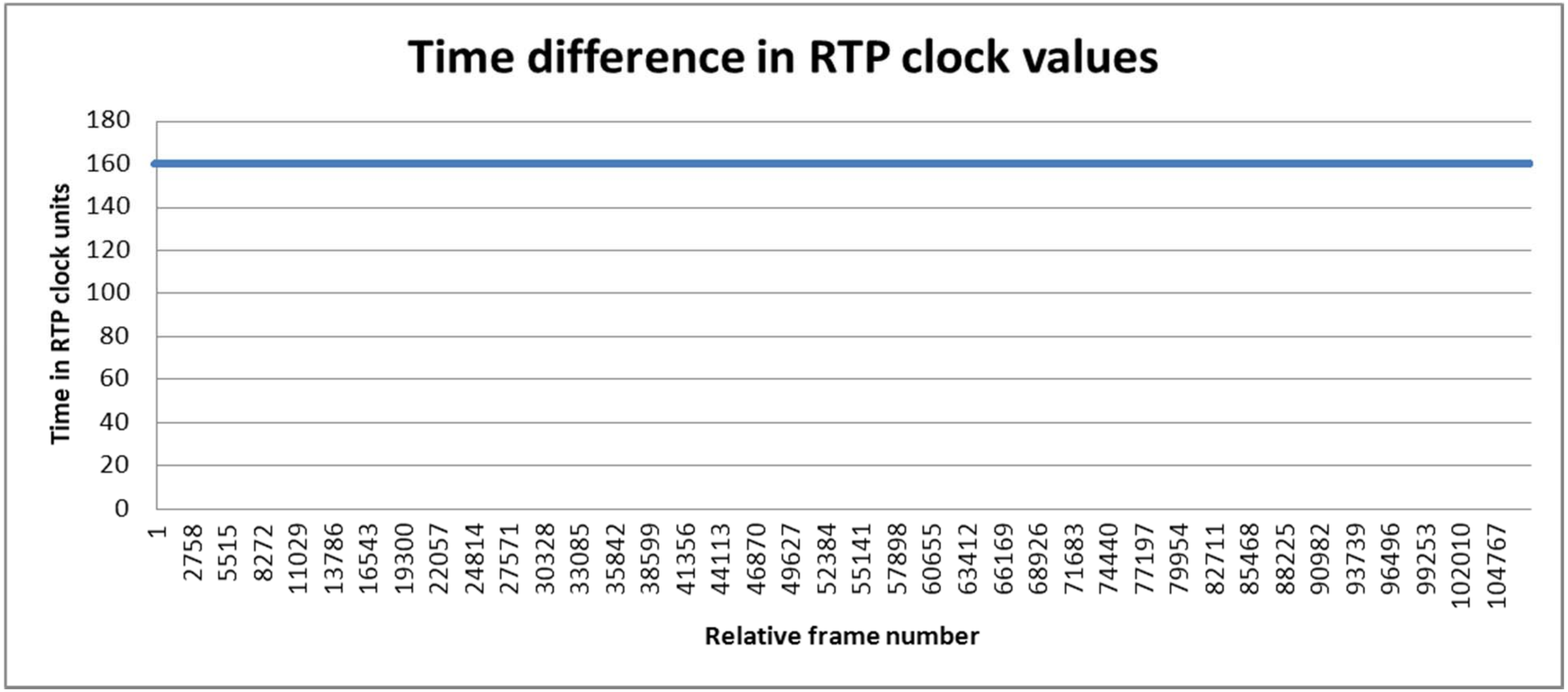
Difference in RTP clock from previous sample

Mean	160
Standard Error	0
Median	160
Mode	160
Standard Deviation	0
Sample Variance	0
Kurtosis	#DIV/0!
Skewness	#DIV/0!
Range	0
Minimum	160
Maximum	160
Sum	17200960
Count	107506
Confidence Level(95.0%)	0

Inter-arrival times (in seconds) of RTP packets

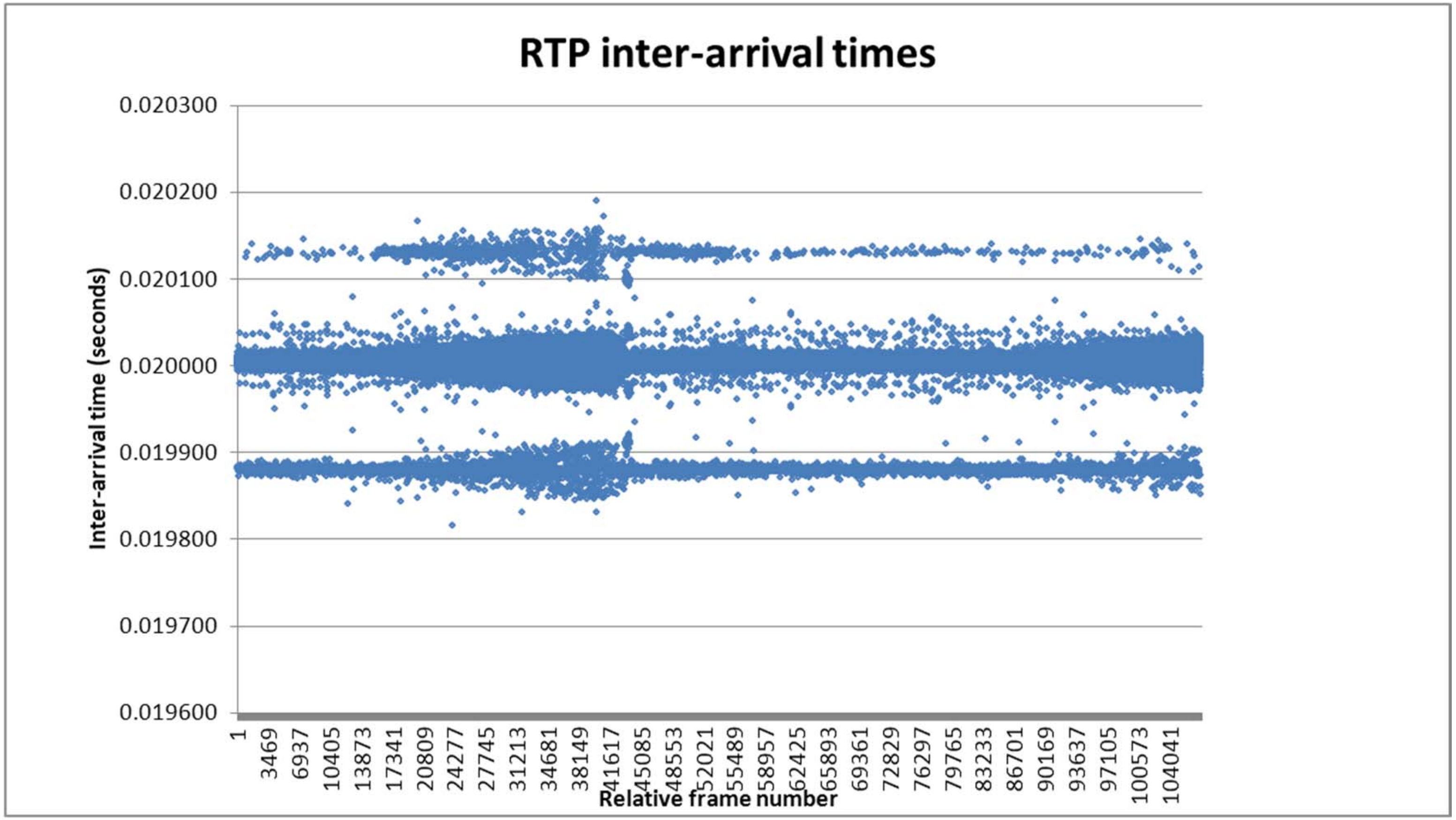
Mean	0.019999999
Standard Error	9.28526E-08
Median	0.020004
Mode	0.020005
Standard Deviation	3.04446E-05
Sample Variance	9.26874E-10
Kurtosis	12.36652501
Skewness	-2.054662184
Range	0.000374
Minimum	0.019815
Maximum	0.020189
Sum	2150.11991
Count	107506
Confidence Level(95.0%)	1.8199E-07

First look at the RTP clock differences

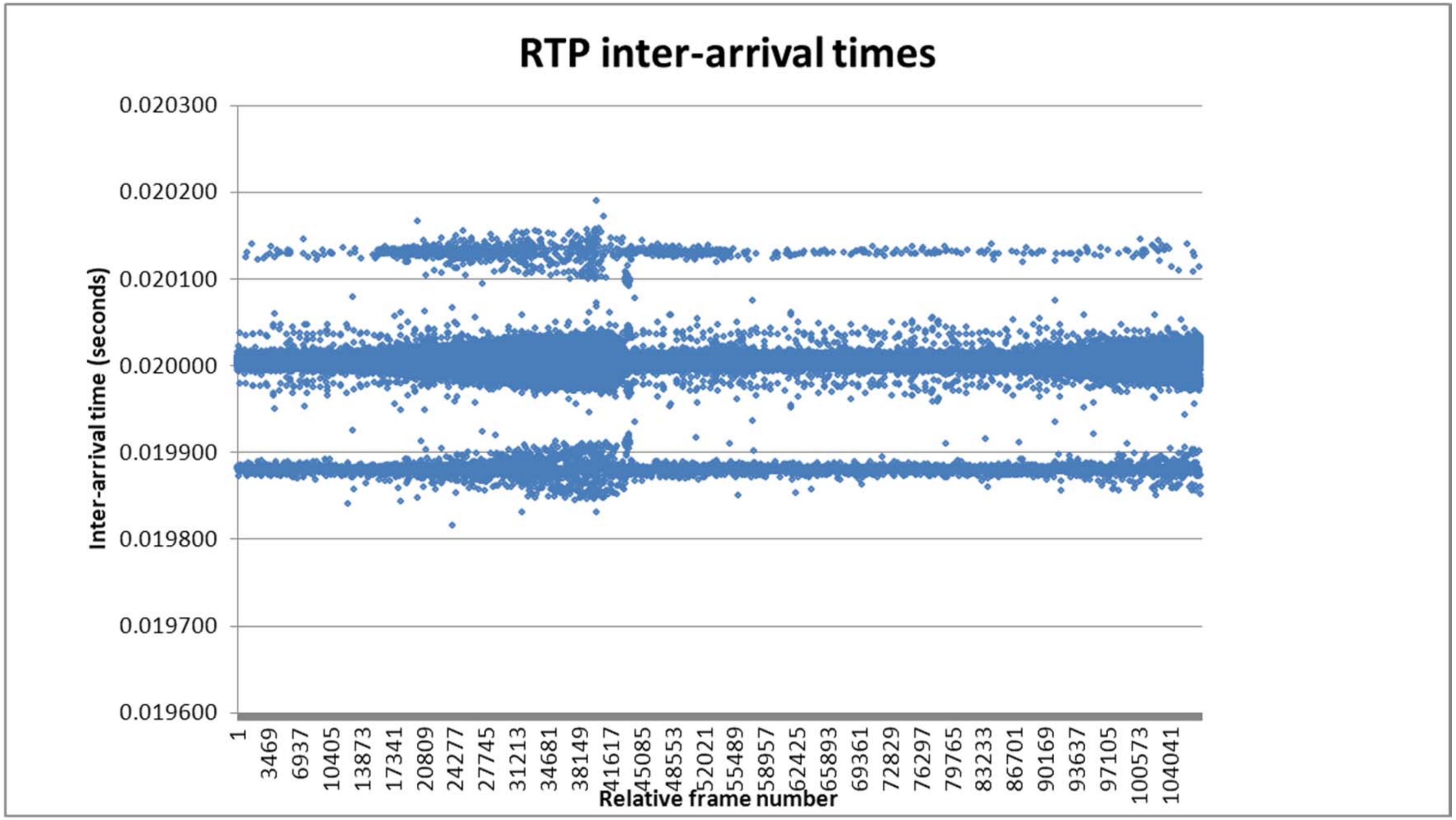


Conclusion: 160 audio samples per frame, with a frame time of 0.20 ms
⇒ 8 K sample/second sampling rate – consistent with ITU-T G.711 PCMA encoding

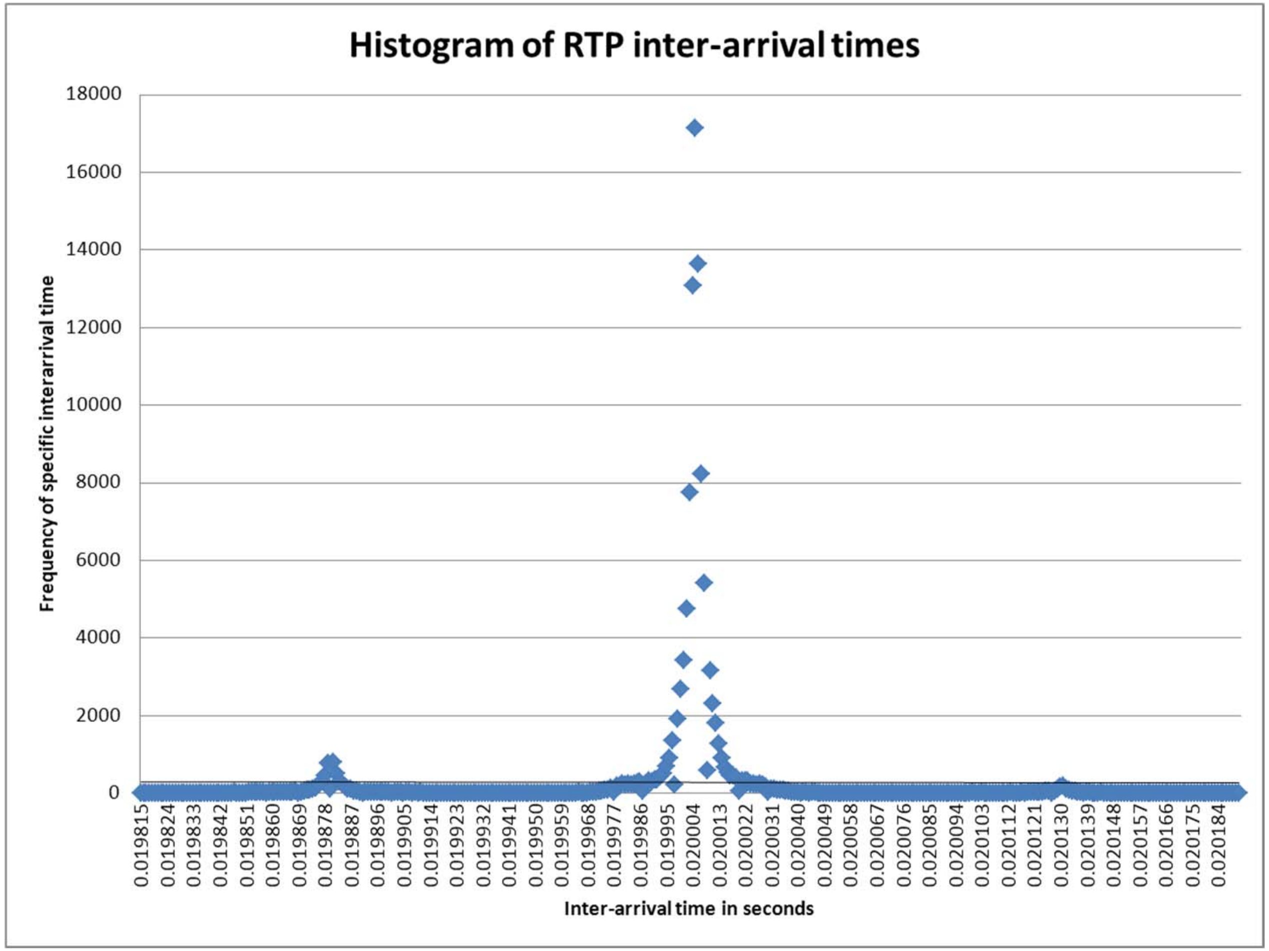
RTP inter-arrival times as measured by Wireshark



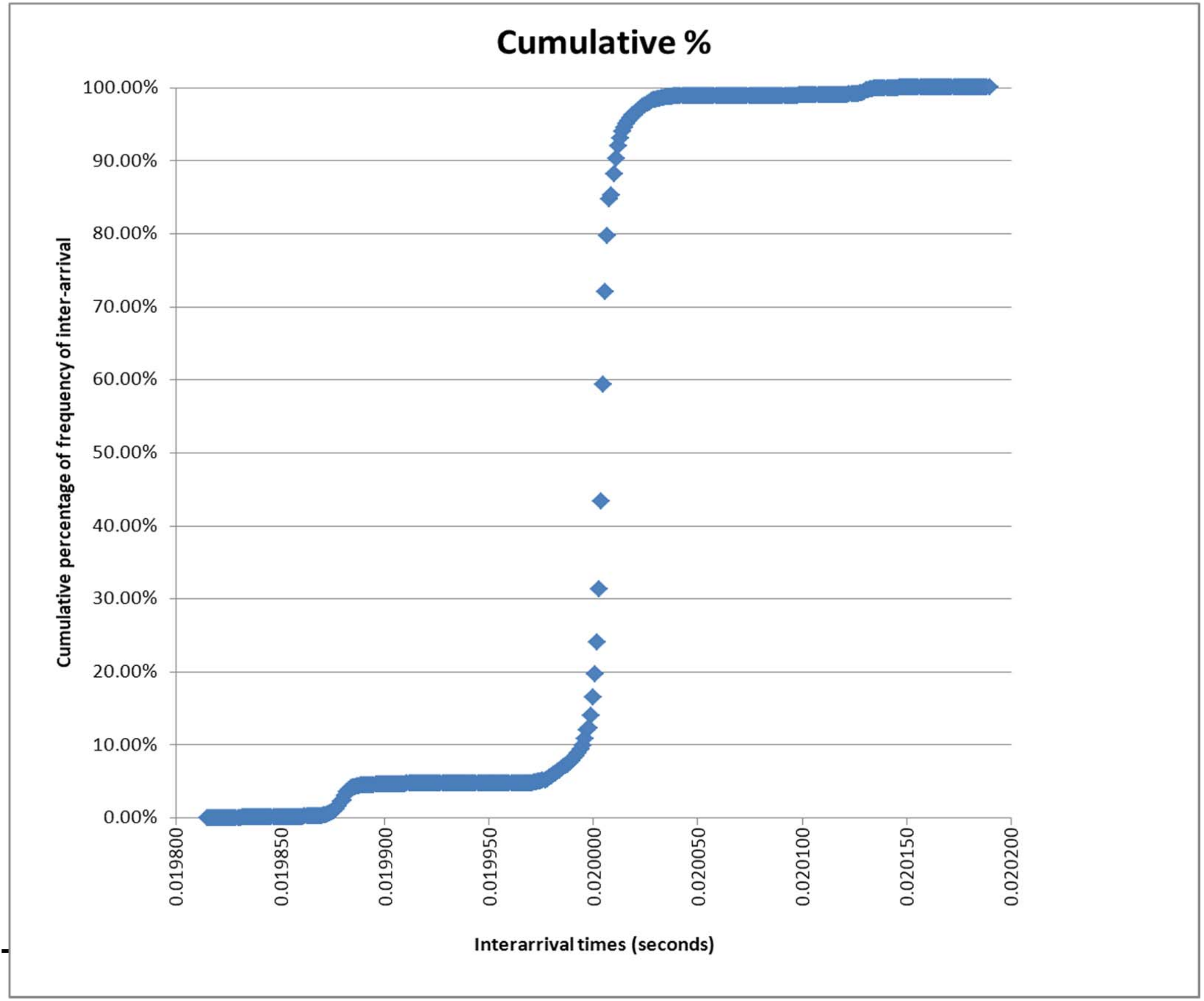
Re-scale vertical axis



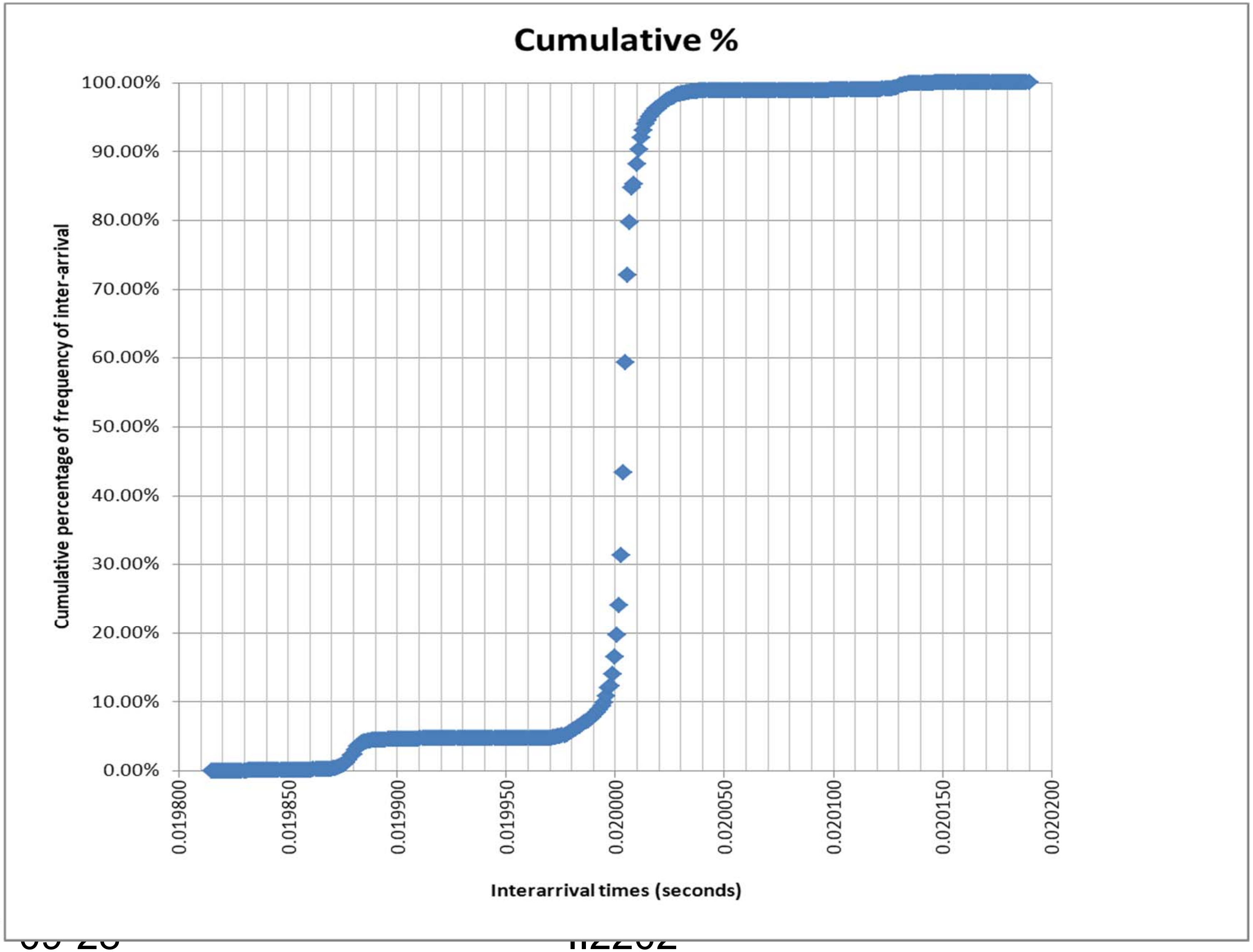
Compute histogram



Plot as a Cumulative Distibution



Add grid lines



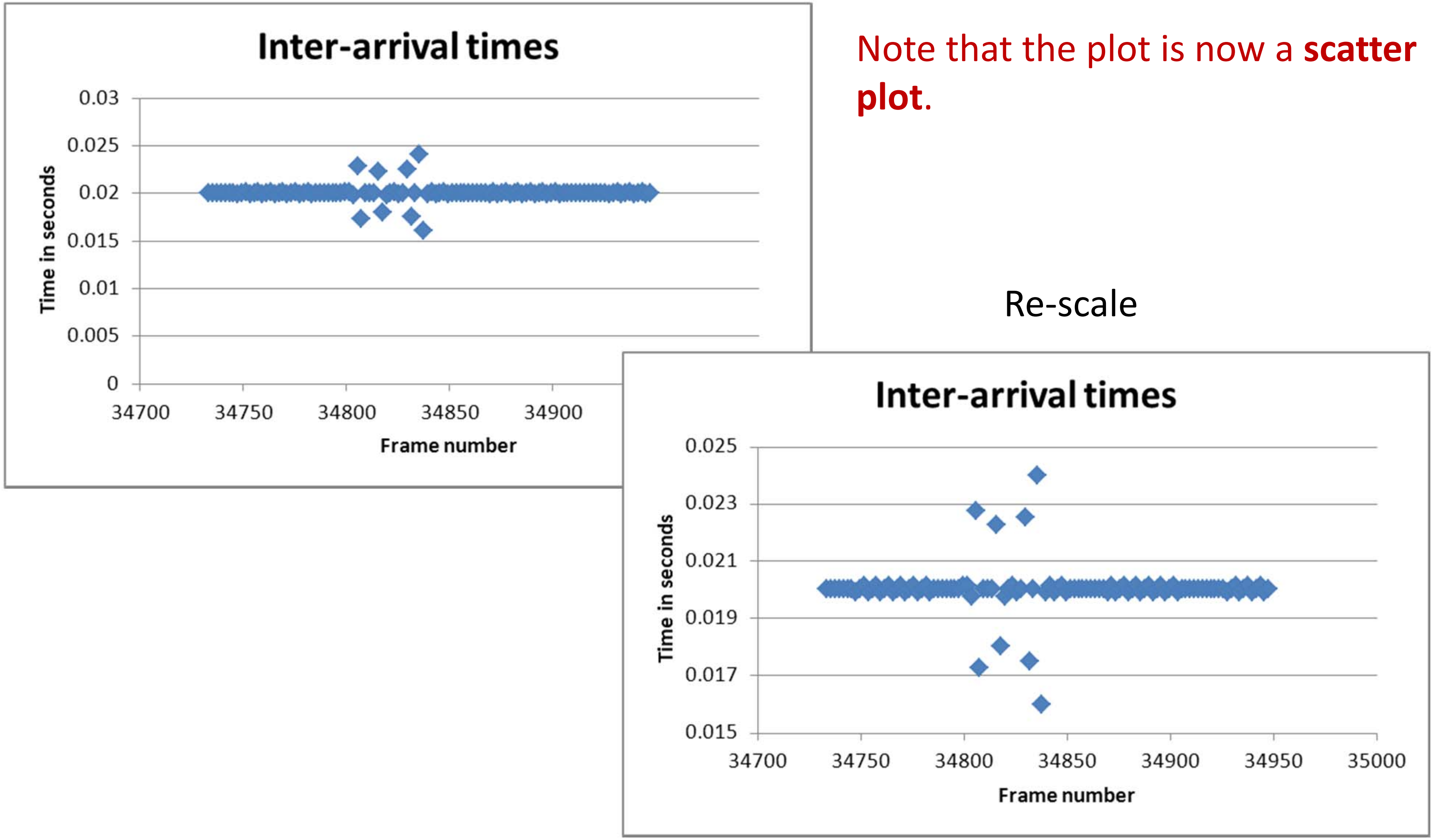
As numbers - near median

	seconds	frequency	Cumulative %
	0.019995	687	9.92%
	0.019996	895	10.75%
	0.019997	1334	11.99%
	0.019998	209	12.18%
	0.019999	1898	13.95%
Mean	0.020000	2671	16.44%
	0.020001	3403	19.60%
	0.020002	4747	24.02%
	0.020003	7742	31.22%
Median	0.020004	13059	43.37%
Mode	0.020005	17121	59.30%
	0.020006	13630	71.98%
	0.020007	8211	79.62%
	0.020008	5404	84.64%
	0.020009	570	85.18%
	0.020010	3158	88.11%
	0.020011	2305	90.26%
	0.020012	1787	91.92%
	0.020013	1262	93.09%
	0.020014	886	93.92%

With varying numbers of samples

Descriptive Statistics	First 100	First 1K	First 10K	First 100K
Mean	0.02000071	0.020000066	0.020000004	0.02
Standard Error	2.12714E-06	7.53406E-07	2.51164E-07	9.69855E-08
Median	0.020005	0.020004	0.020004	0.020004
Mode	0.020005	0.020005	0.020005	0.020005
Standard Deviation	2.12714E-05	2.38248E-05	2.51164E-05	3.06695E-05
Sample Variance	4.52471E-10	5.67621E-10	6.30831E-10	9.40618E-10
Kurtosis	28.87137928	21.46428225	19.07376827	12.23083198
Skewness	-5.453831468	-4.509853108	-3.831289593	-2.003065575
Range	0.000135	0.000252	0.000277	0.000374
Minimum	0.01988	0.019872	0.019868	0.019815
Maximum	0.020015	0.020124	0.020145	0.020189
Sum	2.000071	20.000066	200.000044	1999.999951
Count	100	1000	10000	100000
Confidence Level(95.0%)	4.2207E-06	1.47844E-06	4.92331E-07	1.9009E-07

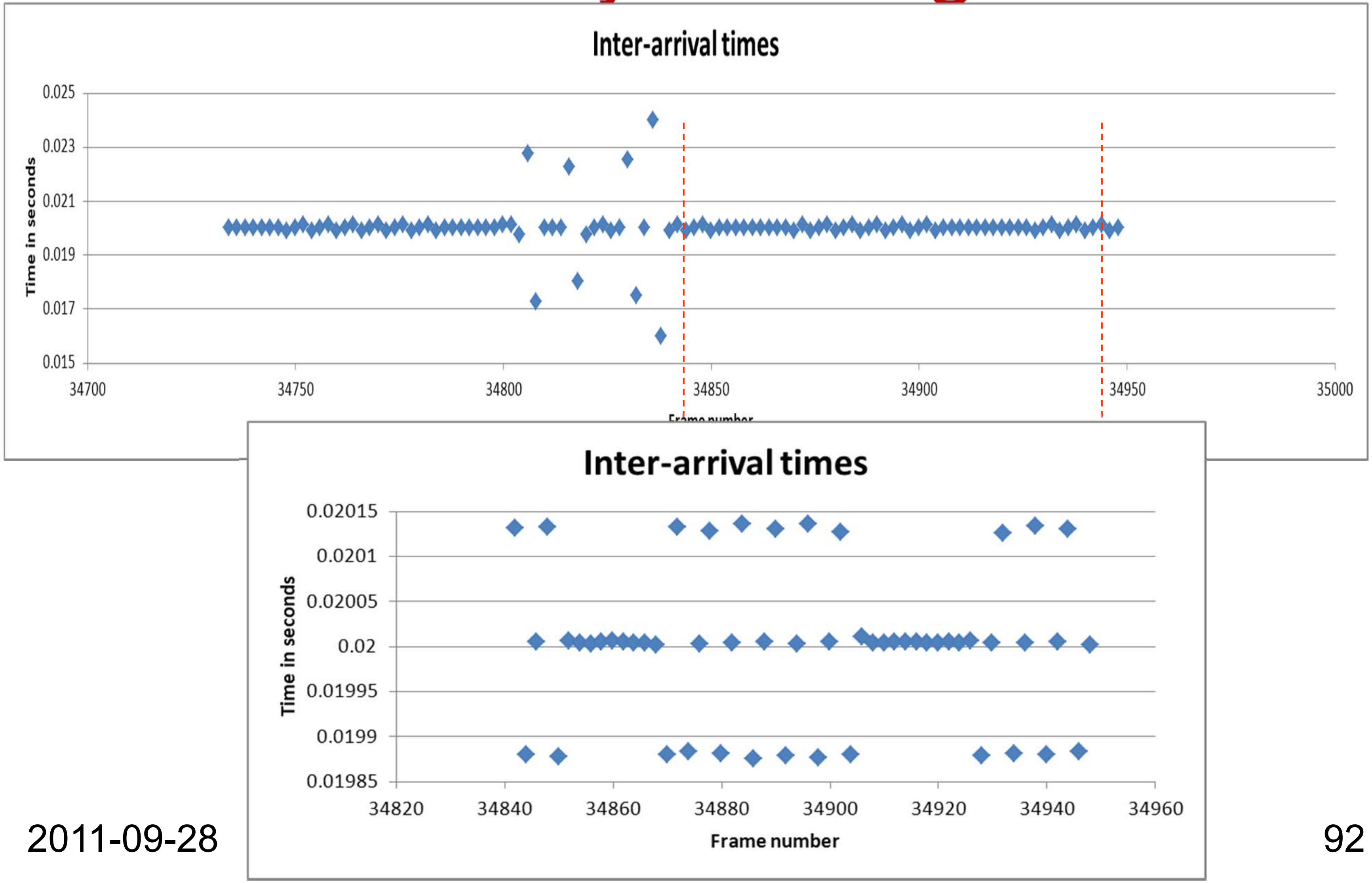
Zooming in on behavior



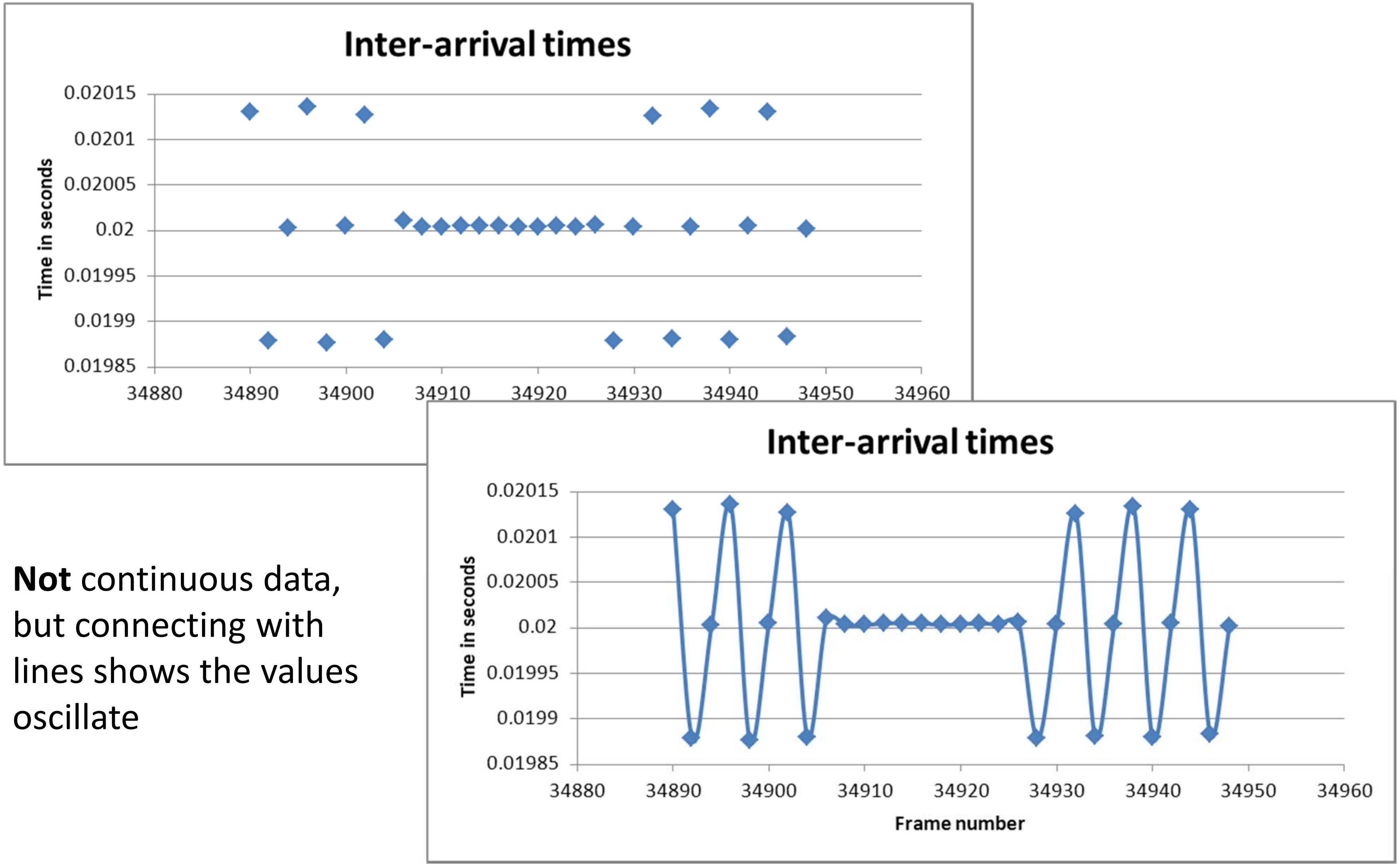
Note that the plot is now a **scatter plot**.

Re-scale

Looking in more detail at a relatively “flat” region

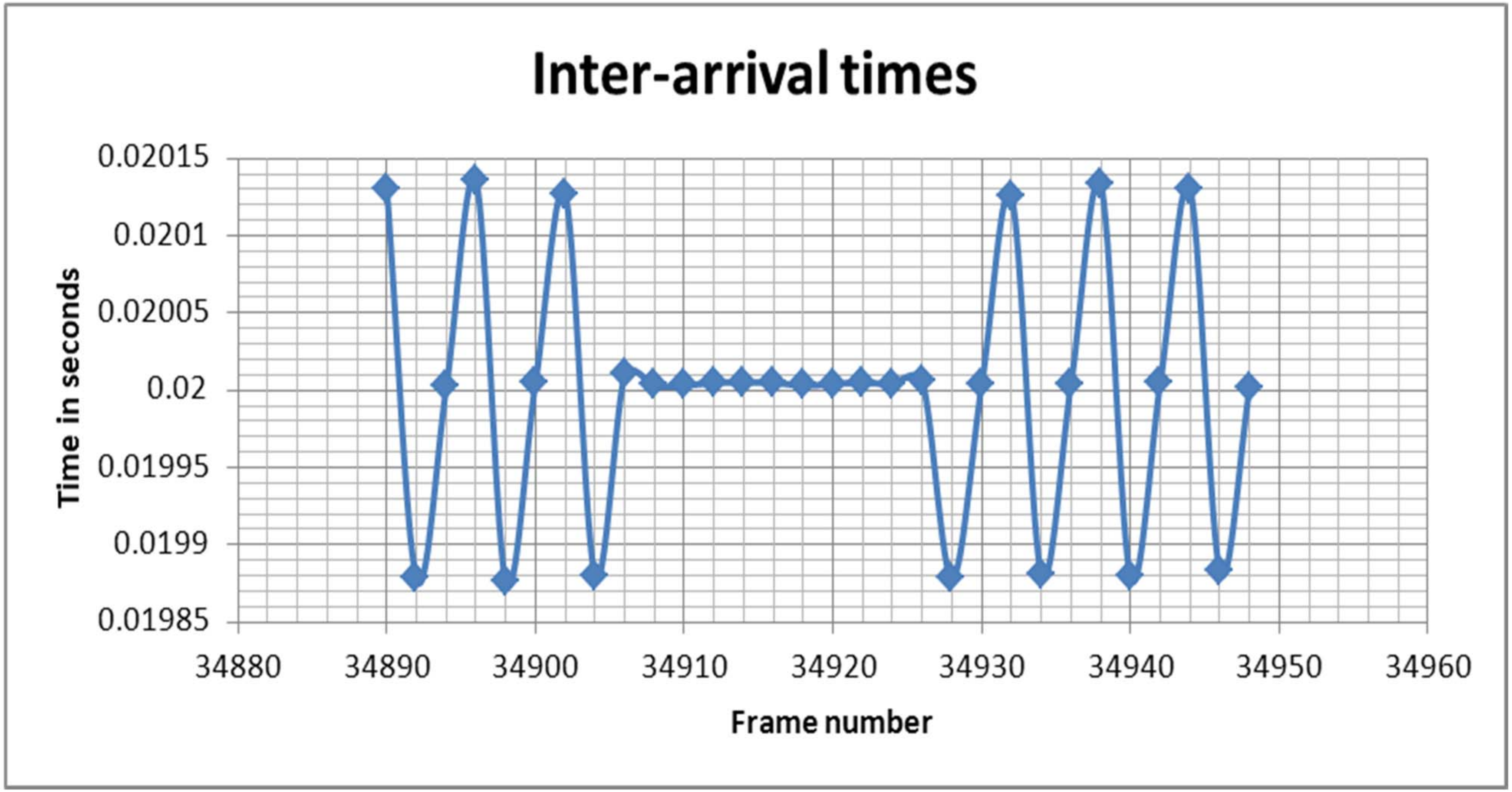


Is there some pattern?

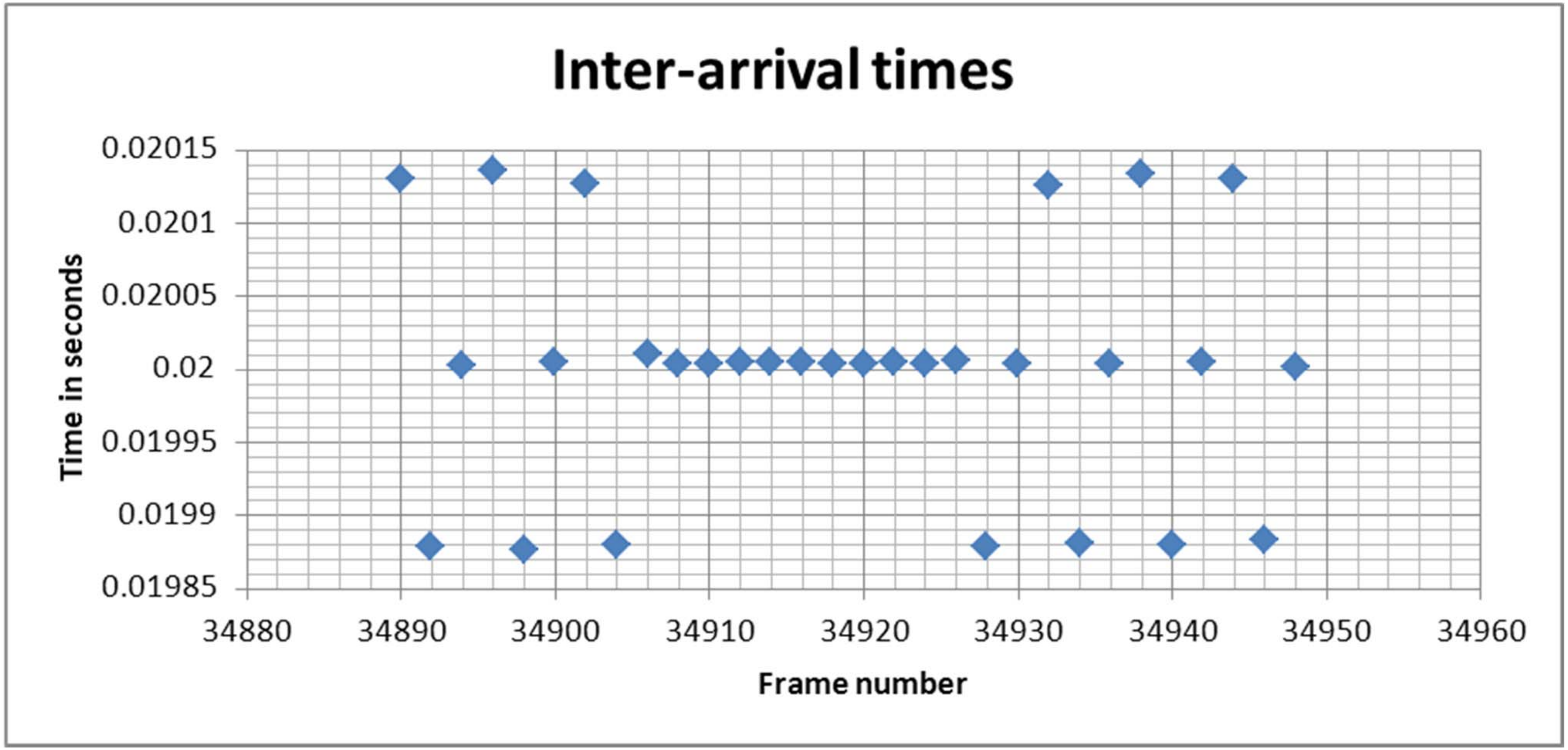


Not continuous data, but connecting with lines shows the values oscillate

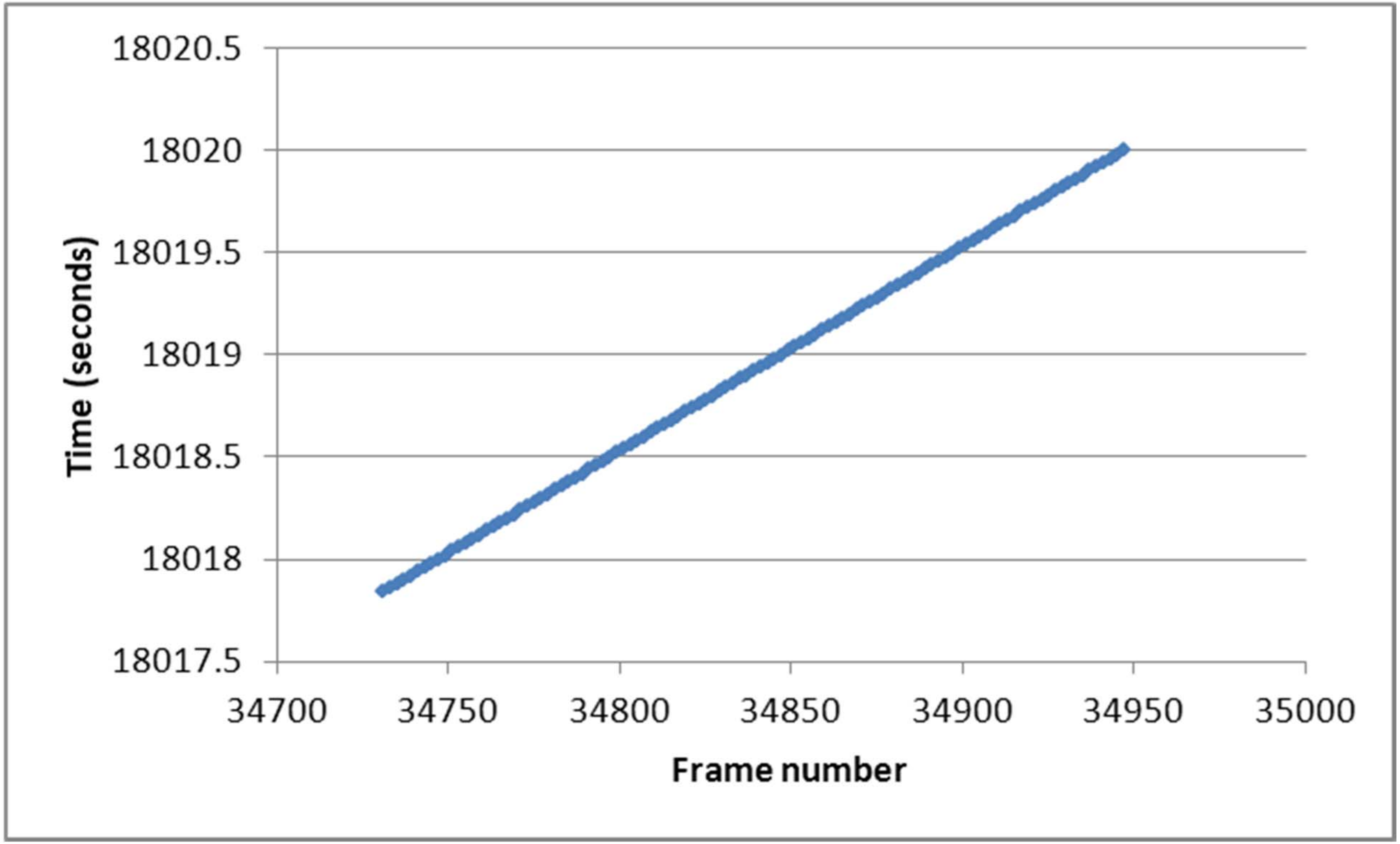
Adding grid lines



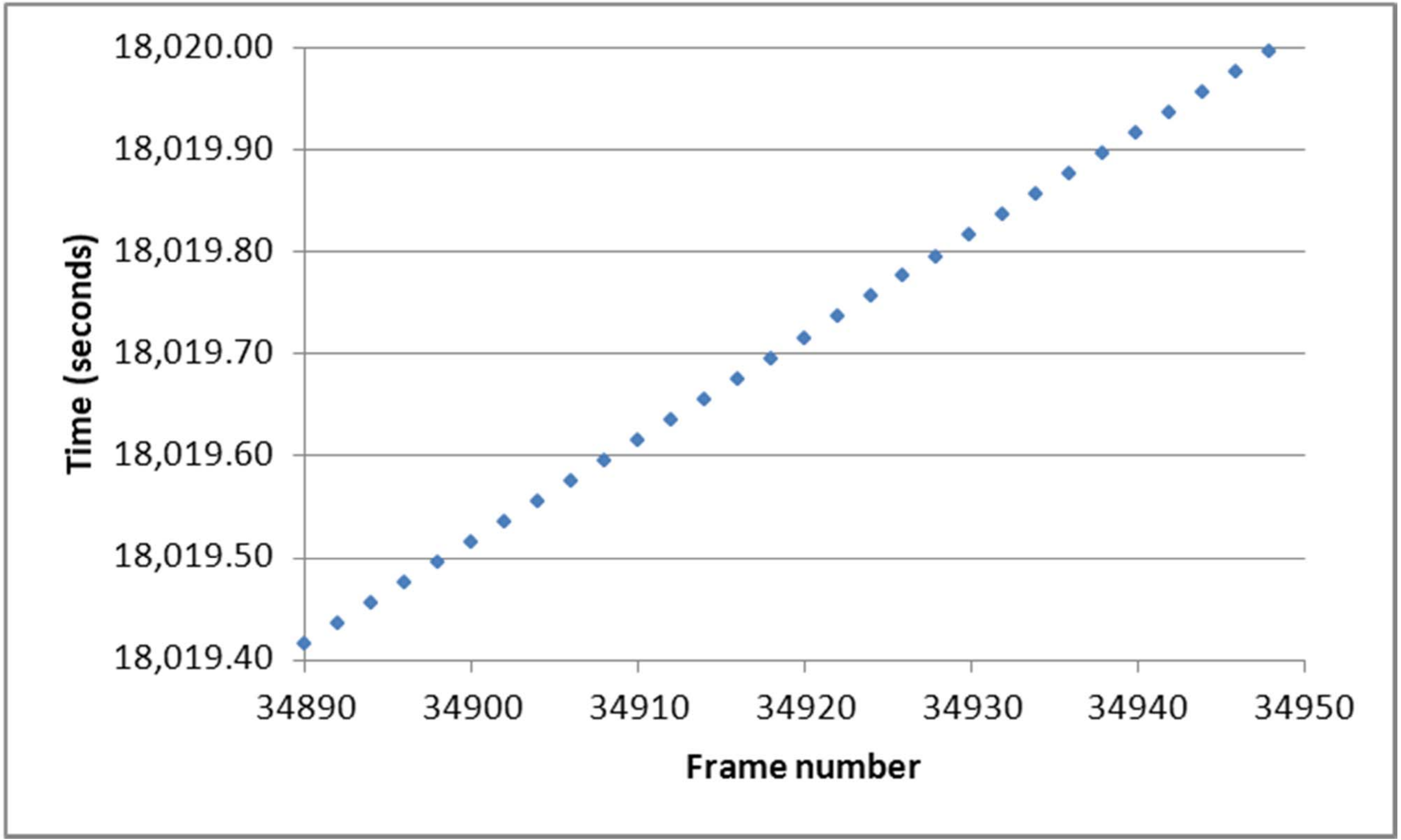
Are grid lines alone sufficient?



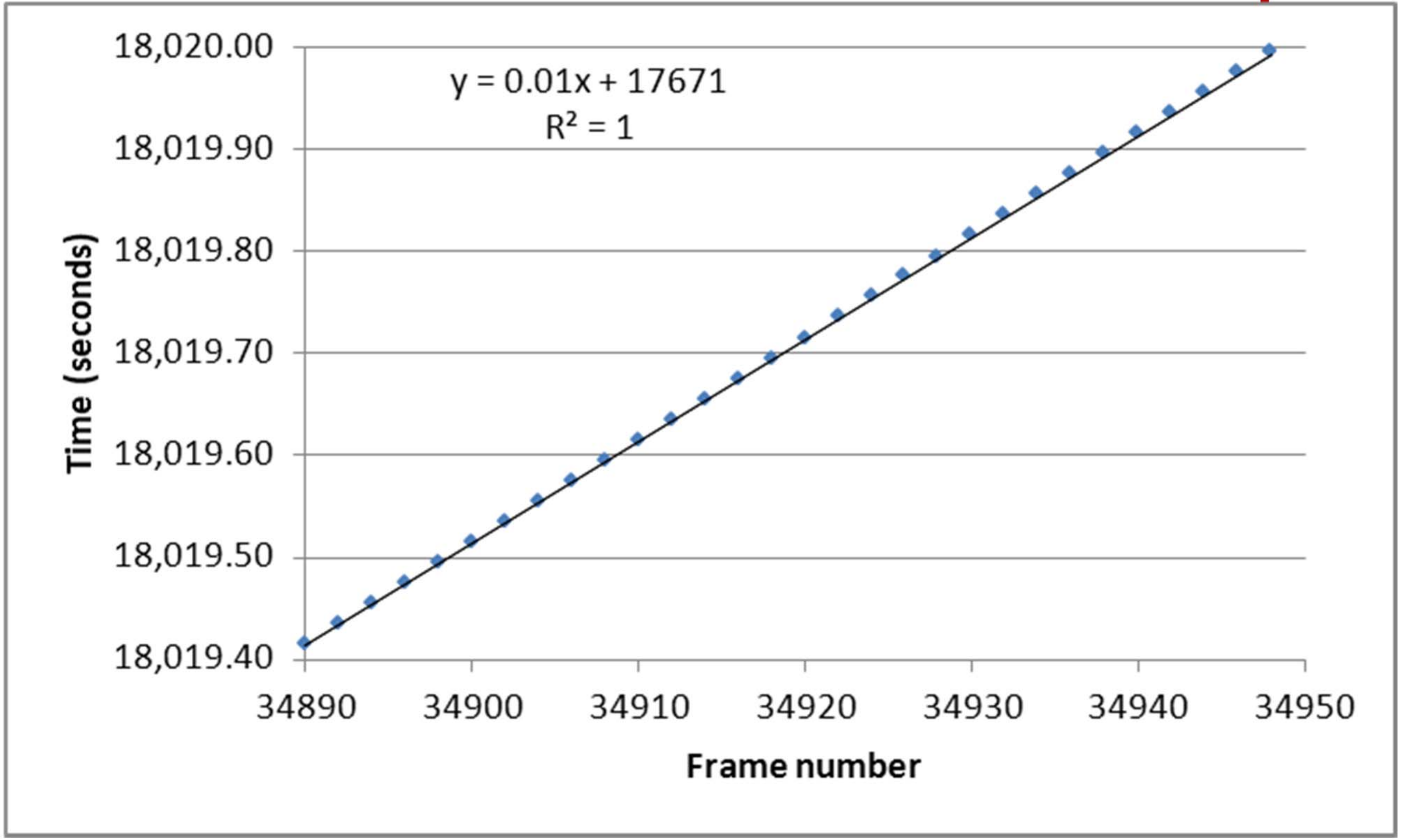
Scatter plots of frame # versus time



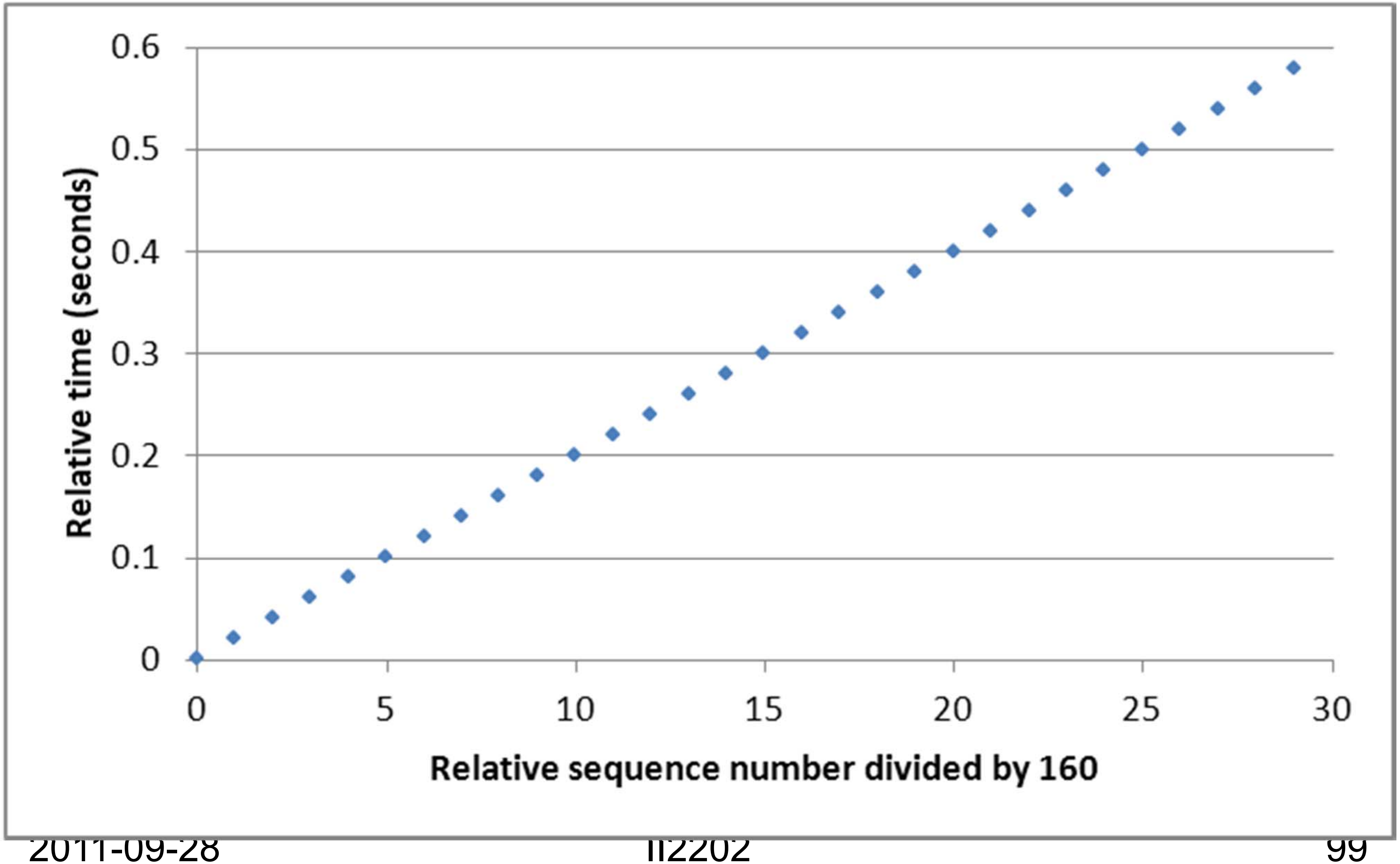
Zoom in on last few samples



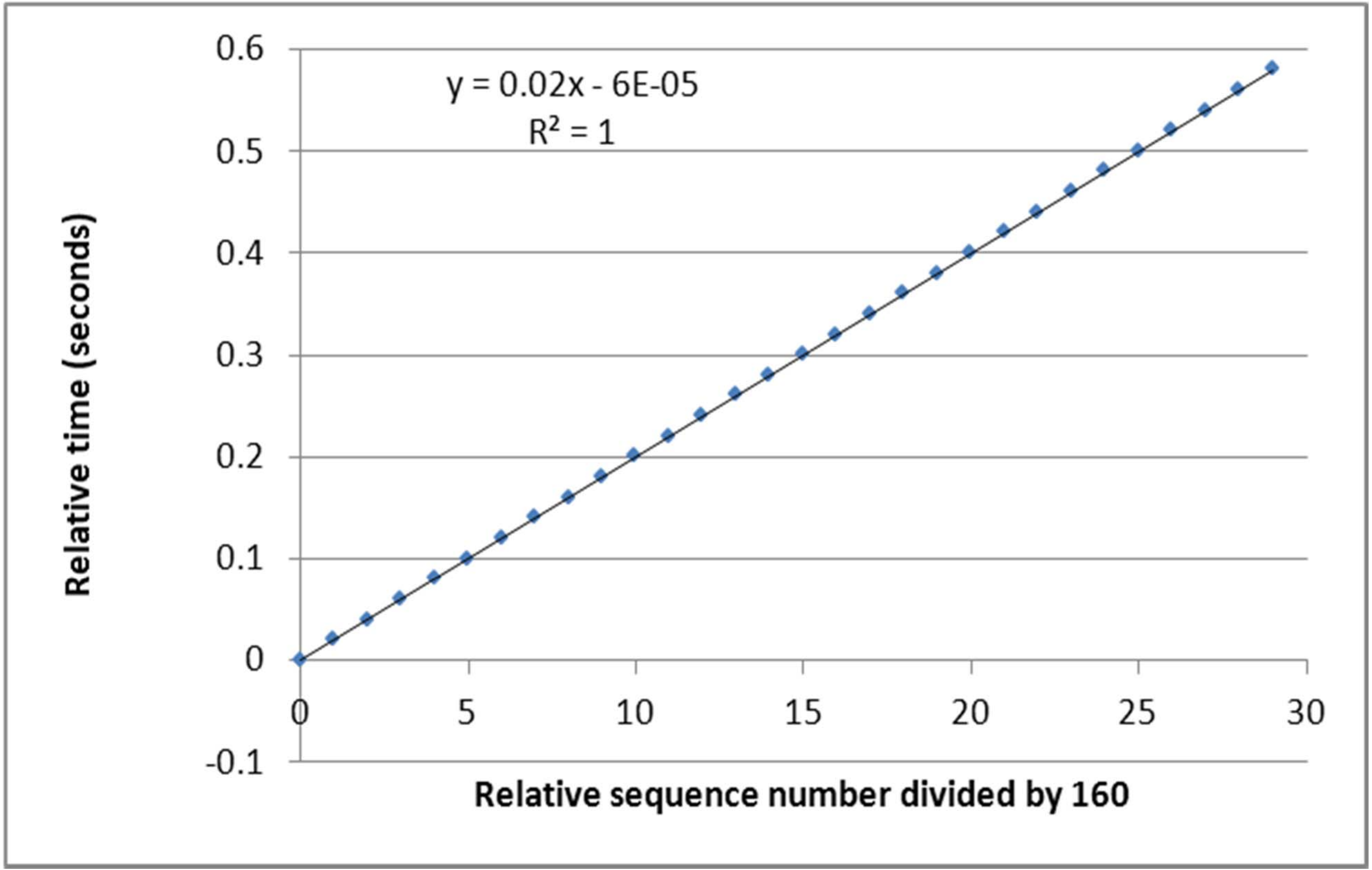
Add a trendline and show eqn.

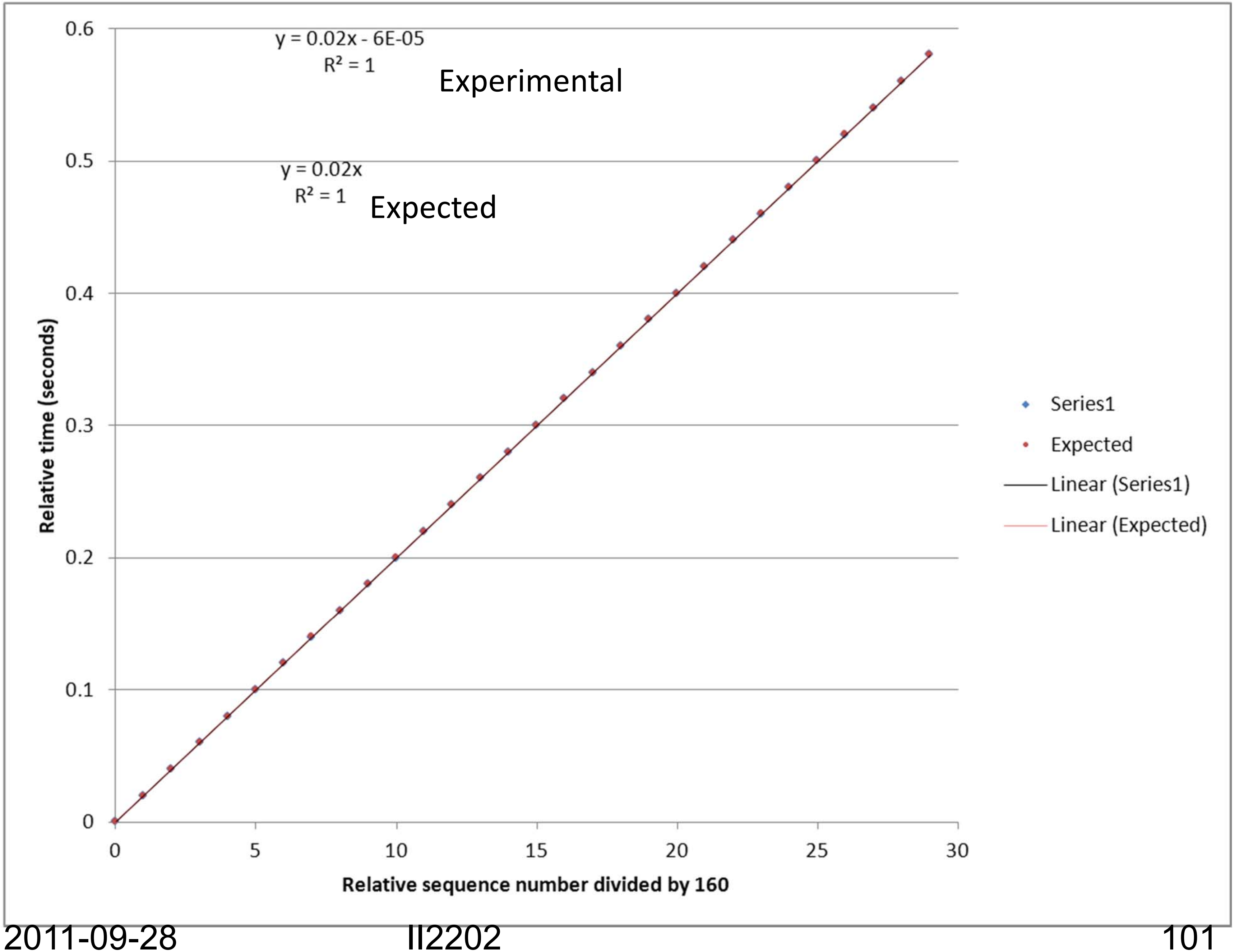


Computing new axis

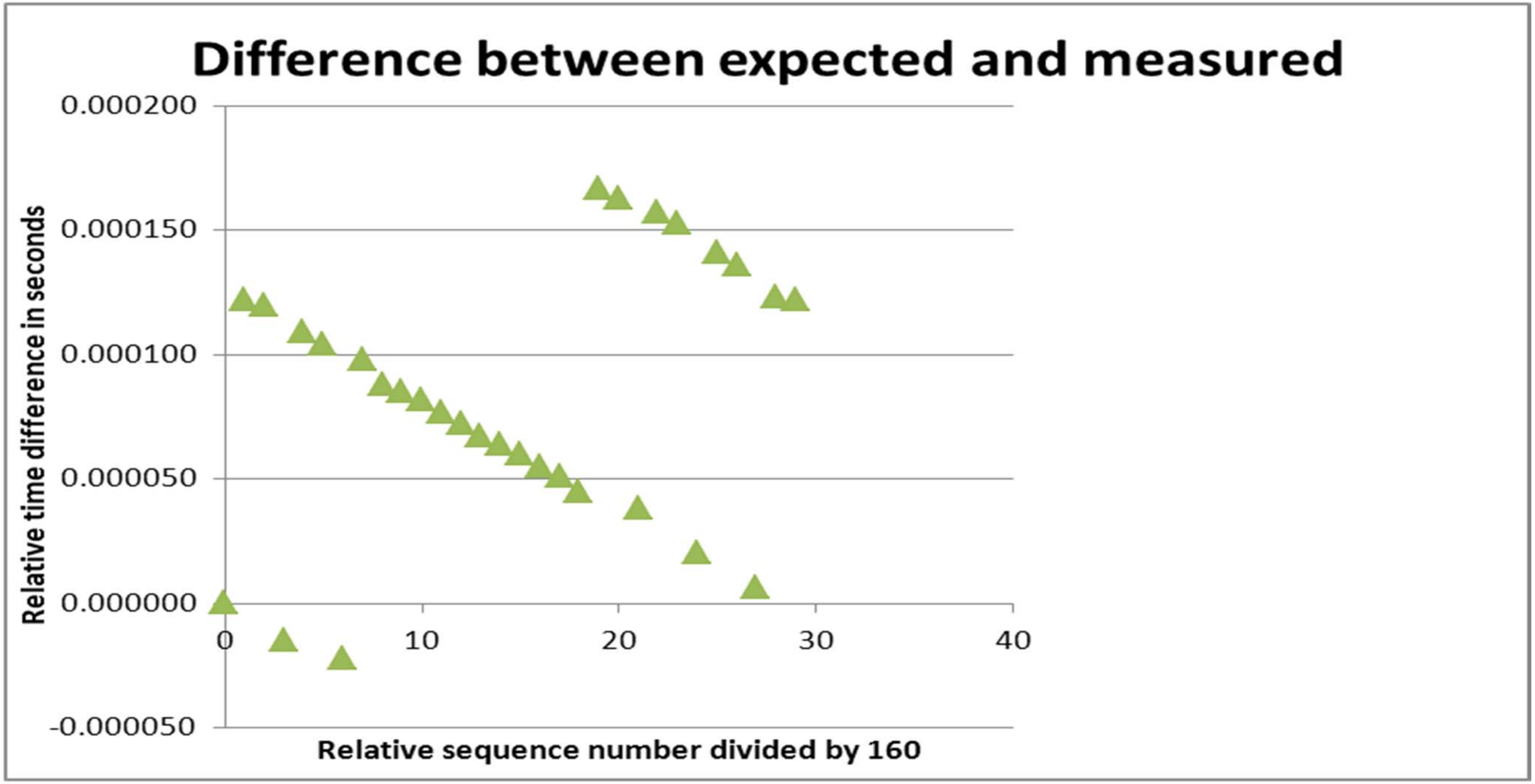


Now add the trendline

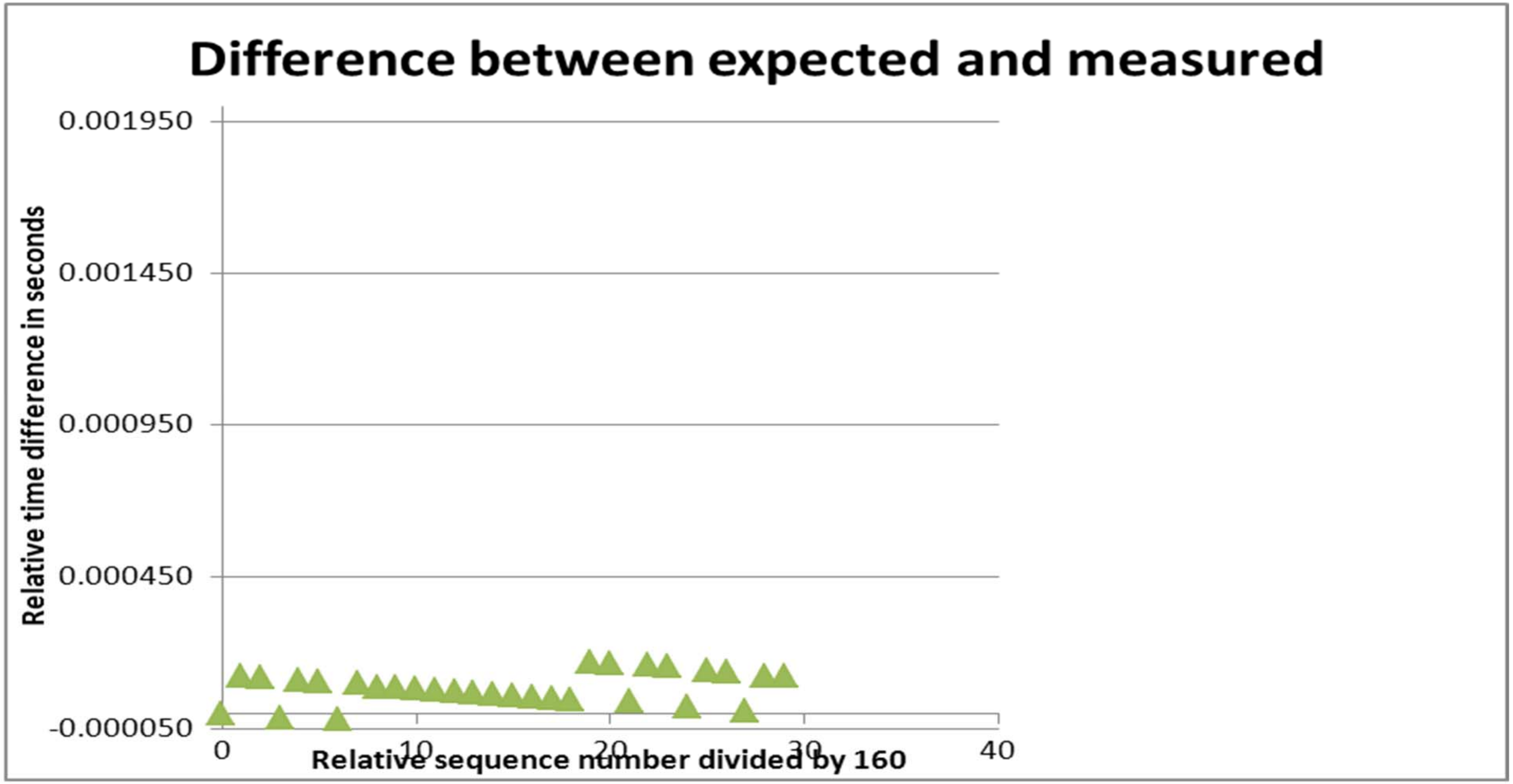


y = 0.02x - 6E-05
R² = 1
Experimental
y = 0.02x
R² = 1
Expected
0.6
0.5
0.4
0.3
0.2
0.1
0
Relative time (seconds)
0
5
10
15
20
25
30
Relative sequence number divided by 160
Series1
Expected
Linear (Series1)
Linear (Expected)

How does the measured data differ from the expected data?



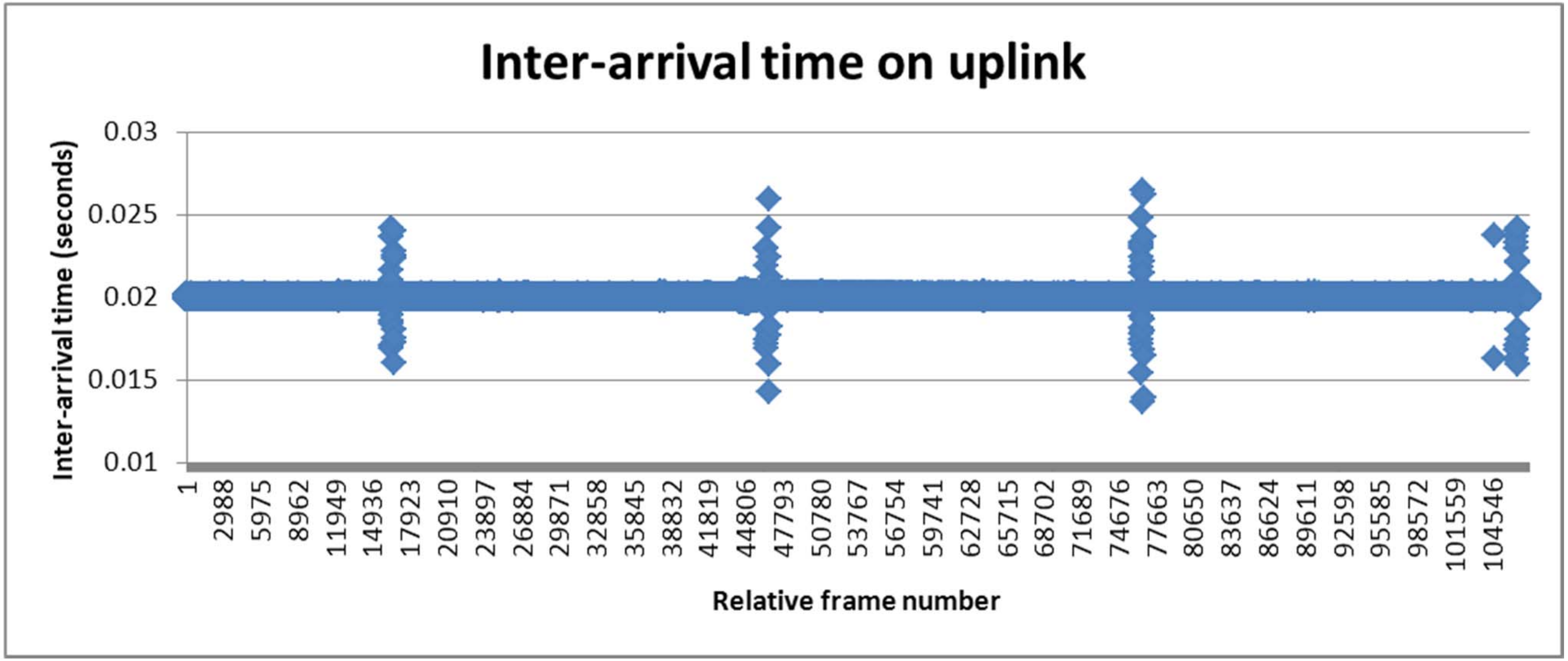
Does the difference matter? Plot scaled to 1/10 of the inter-arrival time period



For traffic in the opposite direction

Mean	0.020000275
Standard Error	3.6743E-07
Median	0.020004
Mode	0.020005
Standard Deviation	0.000120472
Sample Variance	1.45135E-08
Kurtosis	670.0855429
Skewness	0.482218958
Range	0.012759
Minimum	0.013625
Maximum	0.026384
Sum	2150.109545
Count	107504
Confidence Level(95.0%)	7.20157E-07

Uplink inter-arrival times



What is going on?

Note the spikes near:		time in seconds	difference in time in seconds
16453		329.06	
46682		933.64	604.58
76657		1533.14	599.5
106512		2130.24	597.1

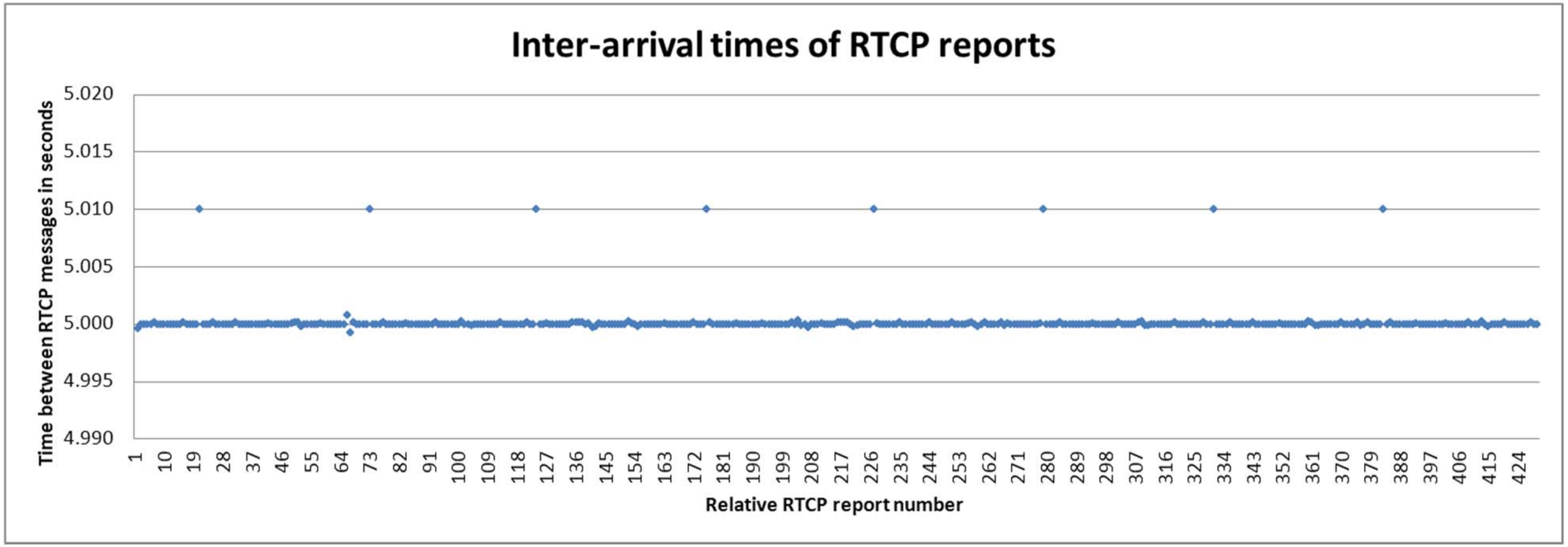
Q: What happens roughly every 600 seconds?

A: DHCP requests

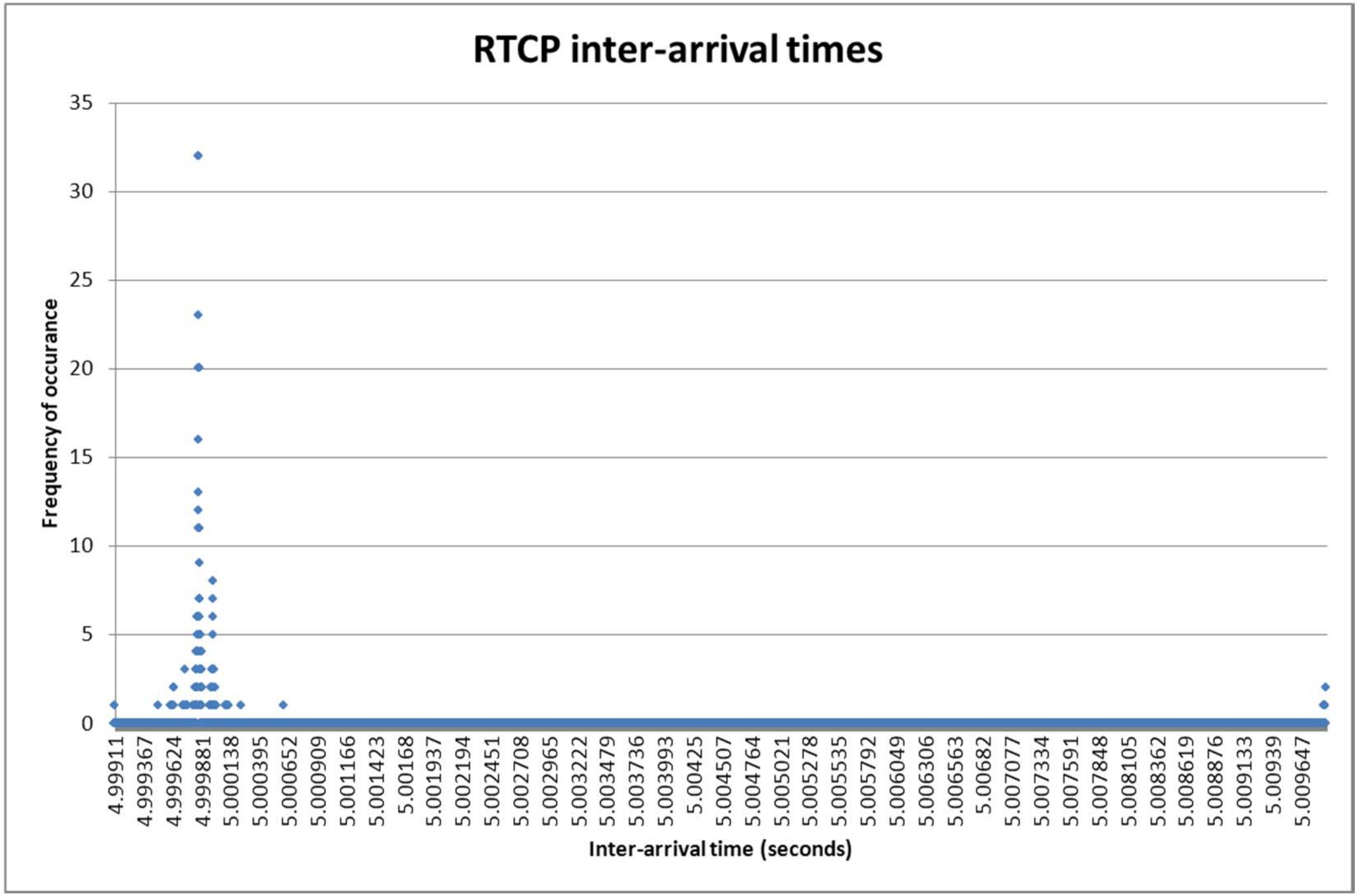
RTCP descriptive statistics

Mean	5.00006104
Standard Error	6.54393E-05
Median	4.999861
Mode	4.99986
Standard Deviation	0.001355399
Sample Variance	1.83711E-06
Kurtosis	48.80806181
Skewness	7.096344028
Range	0.010758
Minimum	4.99911
Maximum	5.009868
Sum	2145.026186
Count	429
Confidence Level(95.0%)	0.000128622

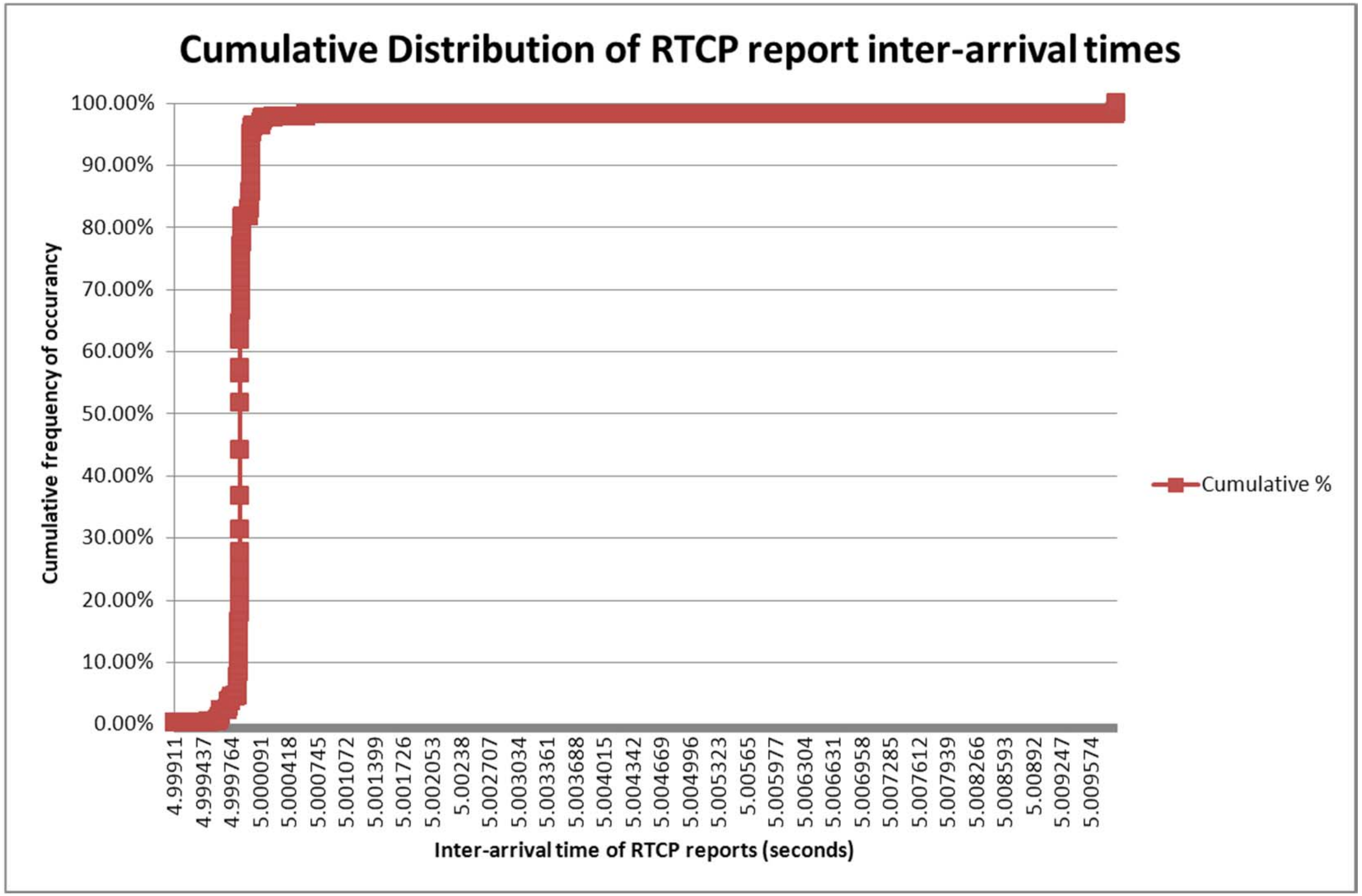
Plot of inter-arrival times of RTCP reports



Histogram of RTCP inter-arrivals



RTCP CDF


Cumulative Distribution of RTCP report inter-arrival times
Cumulative frequency of occurancy
100.00%
90.00%
80.00%
70.00%
60.00%
50.00%
40.00%
30.00%
20.00%
10.00%
0.00%
4.99911
4.999437
4.999764
5.000091
5.000418
5.000745
5.001072
5.001399
5.001726
5.002053
5.00238
5.002707
5.003034
5.003361
5.003688
5.004015
5.004342
5.004669
5.004996
5.005323
5.00565
5.005977
5.006304
5.006631
5.006958
5.007285
5.007612
5.007939
5.008266
5.008593
5.00892
5.009247
5.009574
Inter-arrival time of RTCP reports (seconds)
Cumulative %

References

[1] Tom Tullis and Bill Albert, “Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics”, Morgan-Kaufmann, 2008, ISBN 978-0-12-373558-4

[2] R Graphics Gallery, http://addictedtor.free.fr/graphiques/

[3] Hadley Wickham, ggplot2: Elegant Graphics for Data Analysis (Use R),Springer; 2nd Printing. August 7, 2009, 216 pages, ISBN-10: 0387981403 and ISBN-13: 978-0387981406, website for the book: http://had.co.nz/ggplot2/book/

[4] Hadley Wickham, website of Hadley Wickham, Rice University, Houston TX, USA, 2010,last accessed Wed 15 Sep 2010 04:51:27 PM CEST, http://had.co.nz/

[5] Dong-Yun Kim, "MAT 356 R Tutorial, Spring 2004", web page, Department of Mathematics, Illinois State University, Normal, IL, USA, last modified: 14 January 2004 07:51:38 AM CET, http://math.illinoisstate.edu/dhkim/rstuff/rtutor.html

[6] Frank McCown, Producing Simple Graphs with R, web page, Computer Science Department, Harding University, Searcy, AR, USA, last modified: 06/08/2008 01:06:21, http://www.harding.edu/fmccown/r/

[7] Michael Wexler, R GUIs, web page, last modified Wed 08 Sep 2010 05:02:06 PM CEST, http://www.nettakeaway.com/tp/?s=R (VP of Web Analytics at Barnes and Noble.com)

[8] Dennis R. Mortensen, Yahoo! Web Analytics 9.5 Launched. Visual.revenue blog,New York City, Tuesday, April 28, 2009, http://visualrevenue.com/blog/2009/04/yahoo-web-analytics-95-launched.html

[9] Julian J. Faraway, “Linear Models with R”, Chapman & Hall/CRC Texts in Statistical Science, 2005, 242 pages, ISBN 0-203-50727-4

Additional R References

http://svn.r-project.org/R/trunk/src/library/stats/R/ecdf.R

General plotting of different distributions in R:

Vito Ricci, Fitting Distributions with R, http://cran.r-project.org/doc/contrib/Ricci-distributions-en.pdf

Yet more references

See the course web page:

http://www.ict.kth.se/courses/II2202/