

Data File Used in this Analysis:

```
# Math 3070 - 1          Ventricular Data          May 18, 2011
# Treibergs
#
# From Dalgaard, Introductory Statistics in R, Springer, 2008
# Data from ISwR package, available from CRAN,
# dataset is called "thuesen"
# Source: Altman, Practical Statistics for Medical Research,
# Chapman & Hall 1991.
# Data shows ventricular shortening velocity and blood glucose
# for 24 type 1 diabetic patients
#
# Variables
#
# blood.glucose    fasting blood glucose (mmol / l)
# short.velocity   mean circumferential shortening velocity (% / s)
#
blood.glucose short.velocity
    15.3         1.76
    10.8         1.34
     8.1         1.27
    19.5         1.47
     7.2         1.27
     5.3         1.49
     9.3         1.31
    11.1         1.09
     7.5         1.18
    12.2         1.22
     6.7         1.25
     5.2         1.19
    19.0         1.95
    15.1         1.28
     6.7         1.52
     8.6         NA
     4.2         1.12
    10.3         1.37
    12.5         1.19
    16.1         1.05
    13.3         1.32
     4.9         1.03
     8.8         1.12
     9.5         1.70
```

R Session:

R version 2.11.1 (2010-05-31)
Copyright (C) 2010 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
Natural language support but running in an English locale
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

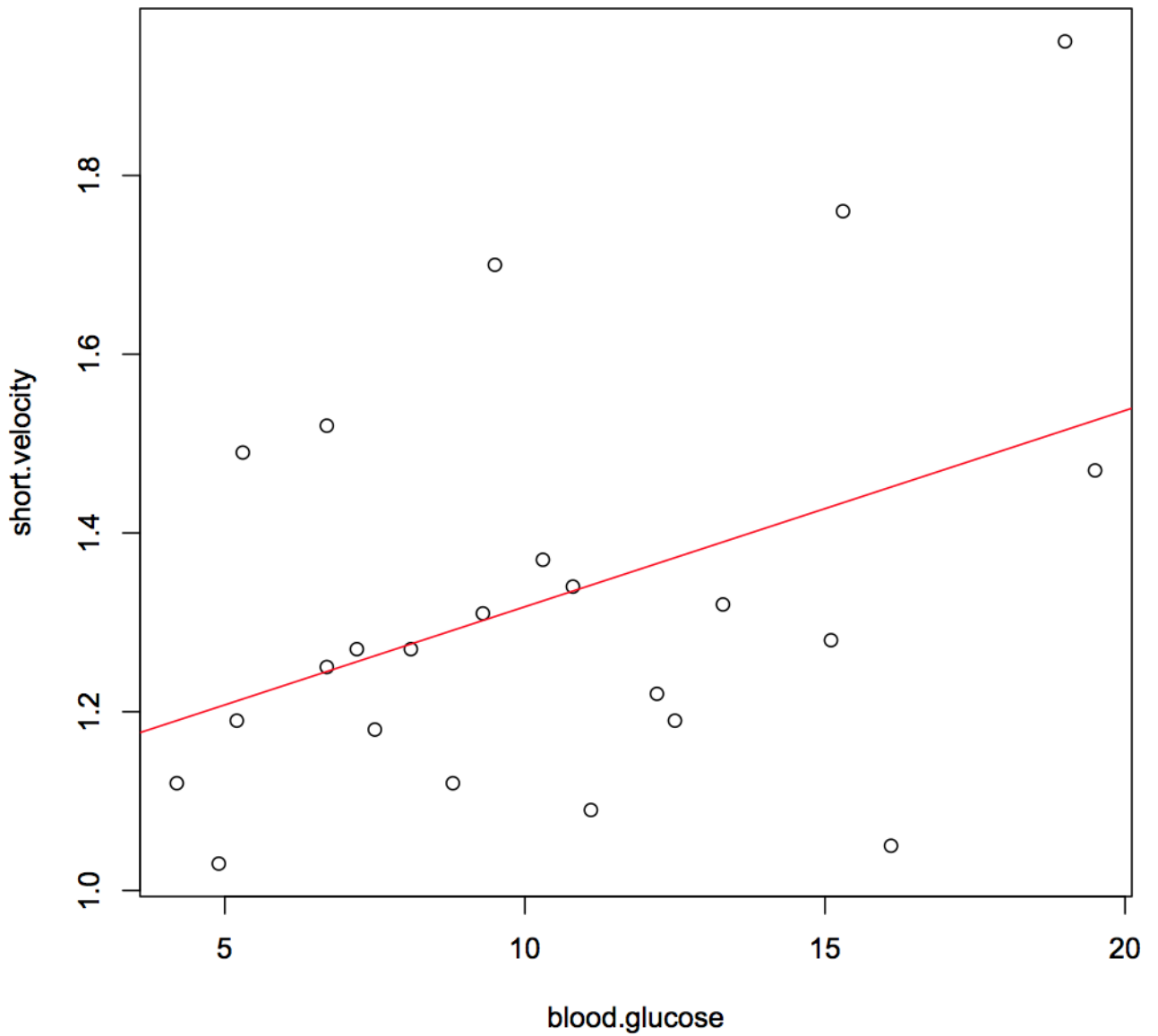
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.34 (5589) i386-apple-darwin9.8.0]

```
> # The following data set is available from CRAN in their
> # ISwR package that can be downloaded. Then run
> # > library(ISwR)
> # The dataset can be attached
> # > attach(thuesen)
> #
> # For convenience I have copied the data into a file.
> tt <- read.table("M3074VentricularData.txt",header=TRUE)
> tt
  blood.glucose short.velocity
1          15.3           1.76
2          10.8           1.34
3           8.1           1.27
4          19.5           1.47
5           7.2           1.27
6           5.3           1.49
7           9.3           1.31
8          11.1           1.09
9           7.5           1.18
10         12.2           1.22
11           6.7           1.25
12           5.2           1.19
13         19.0           1.95
14         15.1           1.28
15           6.7           1.52
16           8.6            NA
17           4.2           1.12
18         10.3           1.37
19         12.5           1.19
20         16.1           1.05
21         13.3           1.32
22           4.9           1.03
```

```
23         8.8         1.12
24         9.5         1.70
> attach(tt)
>
> ##### SCATTERPLOT OF blood.glucose VS short.velocity #####
> plot(blood.glucose,short.velocity,
+ main="Scatterplot of blood.glucose vs. short.velocity")
> # ADD THE BEST FITTING LINE FOR VISUAL EFFECT
> abline(lm(short.velocity ~ blood.glucose), col=2)
```

Scatterplot of blood.glucose vs. short.velocity



```

> ##### CORREATION COEFFICIENT FOUND BY HAND #####
>
> # Note that one of the data points is NA. Thus we look at the
> # remaining records in the dataset. Call the new dataset tt2.
>
> tt2 <- tt[!is.na(short.velocity),];tt2
  blood.glucose short.velocity
1          15.3          1.76
2          10.8          1.34
3           8.1          1.27
4          19.5          1.47
5           7.2          1.27
6           5.3          1.49
7           9.3          1.31
8          11.1          1.09
9           7.5          1.18
10         12.2          1.22
11          6.7          1.25
12          5.2          1.19
13         19.0          1.95
14         15.1          1.28
15          6.7          1.52
17          4.2          1.12
18         10.3          1.37
19         12.5          1.19
20         16.1          1.05
21         13.3          1.32
22          4.9          1.03
23          8.8          1.12
24          9.5          1.70
>
> Compute the means and correlations.
> xbar <- mean(tt2[,1]);xbar
[1] 10.37391
> ybar <- mean(tt2[,2]);ybar
[1] 1.325652
> V11 <- sum((tt2[,1]-xbar)^2)
> V12 <- sum((tt2[,1]-xbar)*(tt2[,2]-ybar))
> V22 <- sum((tt2[,2]-ybar)^2)
> c(V11,V12,V22)
[1] 429.704348  9.437391  1.193365
>
> # Compute the correlation coefficient.
>
> r <- V12/sqrt(V11*V22) ; r
[1] 0.4167546

> ##### HYPOTHESIS TESTS ON r AND CI FOR r #####
>
> # The hypothesis test on r is to convert to T which
> # satisfies t-distribution with n-2 df. Number obs

```

```

>
> n <- length(tt2[,1]); n
[1] 23
>
> T <- r*sqrt(n-2)/sqrt(1-r^2); T
[1] 2.100957
>
> #
> tcrit <- qt(0.05/2,n-2,lower.tail=FALSE);tcrit
[1] 2.079614
>
> # As T falls outside of the interval -tcrit to tcrit, we reject H0: rho=0 in favor
> # of Ha: rho\ne 0 at 5% level. Indeed, the p-value for the two sided test is
>
> 2*pt(T,n-2,lower.tail=FALSE)
[1] 0.04789591
>
> # To get the CI, we estimate Fischer's Z (called V) which has approx normal dist
>
> V <- log((1+r)/(1-r))/2; V
[1] 0.443758
>
> zcrit <- qnorm(0.05/2,lower.tail=FALSE);zcrit
[1] 1.959964
>
> # The CI for V is the interval from c1 to c2
> c1 <- V-zcrit/sqrt(n-3)
> c2 <- V+zcrit/sqrt(n-3)
> c(c1,c2)
[1] 0.005496737 0.882019278

> # Solving back for the corresponding interval for r yields
>
> c((exp(2*c1)-1)/(exp(2*c1)+1),(exp(2*c2)-1)/(exp(2*c2)+1))
[1] 0.005496682 0.707429479
>
> ##### CANNED TEST AND CI FOR r #####
> # The built in routine gives the same numbers as our calculation "by hand."
>
> cor.test(blood.glucose,short.velocity)

```

Pearson's product-moment correlation

```

data: blood.glucose and short.velocity
t = 2.101, df = 21, p-value = 0.0479
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.005496682 0.707429479
sample estimates:
      cor
0.4167546

```