Using the **R Code** and **Git Example** Environments with **knitr**

Alan's Modifications and Notes

November 4, 2015

1 Introduction

This is a test of the **R Code** and **Git Example** environments. By the way, this document was last compiled Wednesday, November 04, 2015 - 01:41:04 PM.

1.1 Simple Arithmetic

R Code 1.1

```
1 + 1
```

```
[1] 2
```

1.2 Generate Random Data

R Code 1.2

```
set.seed(13)
x <- rnorm(100)
```

Find the standard deviation of `x`.

R Code 1.3

```
sd(x) # standard deviation
```

```
[1] 0.9508399
```

Note that **R Code** 1.2 and 1.3 are hyperlinked! The standard deviation of `x` is computed in **R Code** 1.3 and is 0.9508399.

1.3 Graphs and Environments

R Code 1.4

```
set.seed(41)
junk <- rnorm(10000)
MEAN <- mean(junk)
MEAN

[1] 0.006226888
```

The mean of the junk is 0.0062269. Note: It seems that an error is thrown if a code chunk with a graph and `rcode` is executed at the same time. Work around is as shown below. That is, hide the figure when showing the code...then show the figure with a separate code chunk. Note that Figure 1 is hyperlinked!

```
library(ggplot2)
ggplot(data = mtcars) +
  geom_density(aes(x = mpg), fill = "pink") +
  theme_bw() +
  labs(x = "miles per gallon", y = "", title = "$\\alpha + \\beta = \\delta$")
```

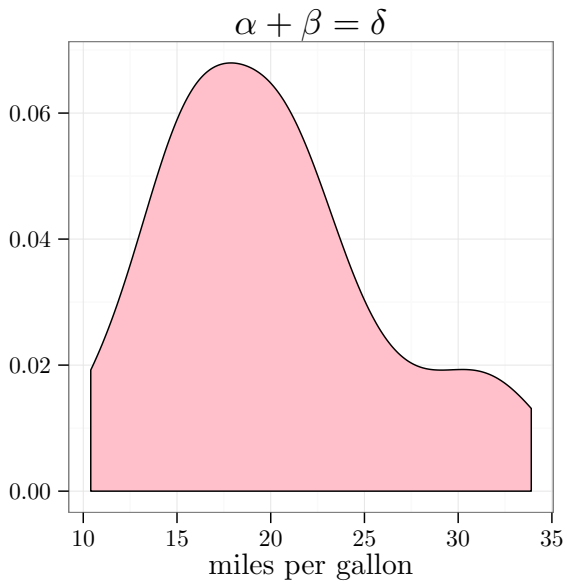


Figure 1: This is where you explain your graph

2 Git Stuff

When working with OSX, one may want to change `engine = 'sh'` to `engine = 'bash'`.

Git Example 2.1

```
git config --list
```

```
user.name=Alan Arnholt
user.email=arnholtat@appstate.edu
credential.helper=cache --timeout=10000000
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
remote.origin.url=https://github.com/STAT-ATA-ASU/STT5811ClassRepo.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.master.remote=origin
branch.master.merge=refs/heads/master
```

Look at **R Code** 1.1 on page 1 to add $1 + 1$ and get the answer 2. The output from **Git Example** 2.1 shows how my machine is configured. **Git Example** 2.2 shows the log.

Git Example 2.2

```
git log --pretty=oneline -3
```

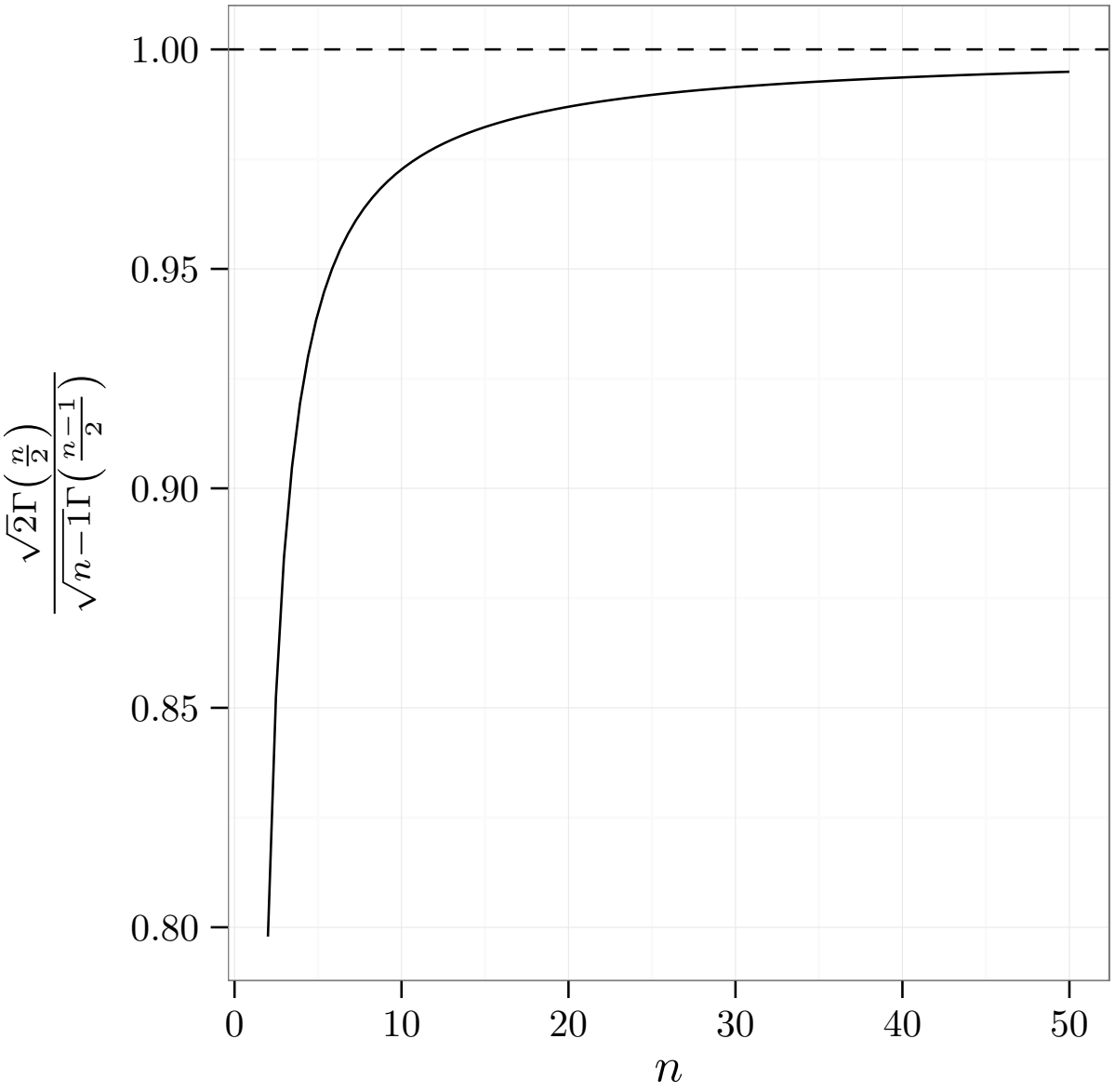
```
1667a1de882476bb1f8d0cf07cca5a1cdee1cdd1 t26
71c4e4a2debabc0ef39966f37dfb9d07d00e32b7 t2
10b6b277e52ec157c66a990c1ac21df6e2a2130b t1
```

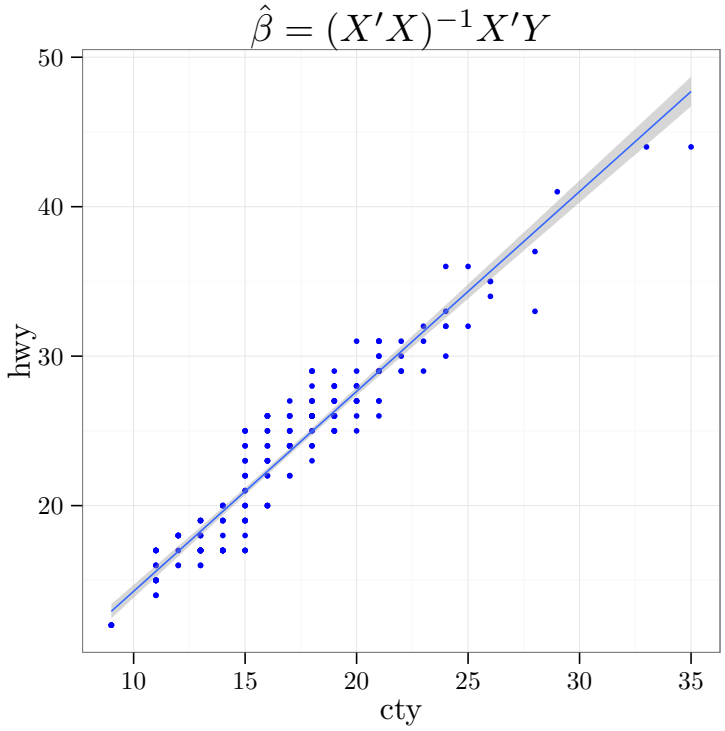
3 Using LaTeX in Graphs

How about some more LaTeX in a `ggplot2` graph.

R Code 3.1

```
f <- function(x){sqrt(2/(x - 1))*gamma(x/2)/gamma((x - 1)/2)}
library(ggplot2)
p <- ggplot(data.frame(x = c(2, 50)), aes(x = x))
p + stat_function(fun = f) +
  labs(x = "$n$", y = "$\\frac{\\sqrt{2}\\Gamma\\left(\\frac{n}{2}\\right)}
       {\\sqrt{n-1}\\Gamma\\left(\\frac{n-1}{2}\\right)}$\n") +
  theme_bw() +
  geom_hline(yintercept = 1, lty = "dashed")
```





- R version 3.2.2 (2015-08-14), `x86_64-pc-linux-gnu`
- Locale: `LC_CTYPE=en_US.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=en_US.UTF-8`, `LC_COLLATE=en_US.UTF-8`, `LC_MONETARY=en_US.UTF-8`, `LC_MESSAGES=en_US.UTF-8`, `LC_PAPER=en_US.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=en_US.UTF-8`, `LC_IDENTIFICATION=C`
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: ggplot2 1.0.1, knitr 1.11, tikzDevice 0.8.1
- Loaded via a namespace (and not attached): codetools 0.2-14, colorspace 1.2-6, digest 0.6.8, evaluate 0.8, filehash 2.3, formatR 1.2.1, grid 3.2.2, gtable 0.1.2, highr 0.5.1, labeling 0.3, magrittr 1.5, MASS 7.3-44, munsell 0.4.2, plyr 1.8.3, proto 0.3-10, Rcpp 0.12.1, reshape2 1.4.1, scales 0.3.0, stringi 1.0-1, stringr 1.0.0, tools 3.2.2