|  |  |
| --- | --- |
| OFFICE SYMBOL | DATE |
|  |  |

MEMORANDUM FOR Analysts Who Produce Memoranda From R

SUBJECT: Making Awesome RMarkdown Army Memoranda

# This is a readme and rmarkdown template for making U.S. Army memoranda based on AR 25-50.

## Versions and requirements. This was built using:

|  |  |
| --- | --- |
| Software | Version |
| R | 3.5.3 |
| rmarkdown | 1.13 |
| knitr | 1.23 |
| pander | 0.6.3 |
| RStudio | 1.1.383 |
| Pandoc | 1.19.2.1 (comes installed with RStudio) |
| MS Word | 2016 |
| ArmyMemoTemplate.docx | 0.1 |

## References.

|  |  |
| --- | --- |
| Reference | Link |
| AR 25-50 | <https://armypubs.army.mil/epubs/DR_pubs/DR_a/pdf/web/r25_50.pdf> |
| rmarkdown documentation | <https://rmarkdown.rstudio.com> |
| knitr home | <https://yihui.name/knitr/> |
| knitr on the cran | <https://cran.r-project.org/web/packages/knitr/knitr.pdf> |
| pander home | <https://www.r-project.org/nosvn/pandoc/pander.html> |
| rmarkdown word guide | <https://bookdown.org/yihui/rmarkdown/word-document.html> |
| pandoc documentation | <https://pandoc.org> |
| R for Data Science, R Markdown | <https://r4ds.had.co.nz/r-markdown.html> |
| Inumerable heros of the opensource community | <https://www.google.com> |

## Known Issues. There are at least three known issues: headers, centering, and cross referencing.

### Headers. The headers in the document will render based on what is written in ‘ArmyMemoTemplate.docx’. Please modify it to have the appropriate information on the first page header and the second page office symbol and subject header. I currently have not figured out how to modify this directly from a .rmd file.

### Centering. Currently tables and figures render left justified. If you desire to center these, this must be done post processing. You can center text using the ##### marker.

### Referencing. Table and Figure numbers do automatically populate in the captions, but, unlike latex or html, it does not seem easily possible to reference figures or tables in the body of your text.

# Formatting the text.

## Paragraphs. AR 25-50 identifies four levels of paragraphs for a memorandum with the following styles. All are aligned left with 12 pt Times New Roman font.

|  |  |  |  |
| --- | --- | --- | --- |
| **Level** | **Numbering** | **Indent** | **Markdown** |
| First level | 1. | None | # |
| Second level | a. | 0.25“ | ## |
| Third level | (1) | 0.5“ | ### |
| Fourth level | (a) | 0.5“ | #### |
| Non-numbered paragraphs. | No number | None | No code; leave a new blank line. |

## Special Formatting.

|  |  |
| --- | --- |
| Formatted Text | Markdown Code |
| **Bold Text** | \*\*Bold Text\*\* |
| *Italicized Text* | \*Italicized Text\* |
| ~~Strike Through Text~~ | ~~Strike Through Text~~ |
| Superscript1 | Superscript^1^ |
| Subscript1 | Subscript~1~ |
| [Link](https://www.google.com) | [Link](myURL) |

## Bulleted lists. You may bullet lists using dashes and spaces. For example, the following code renders the following list:

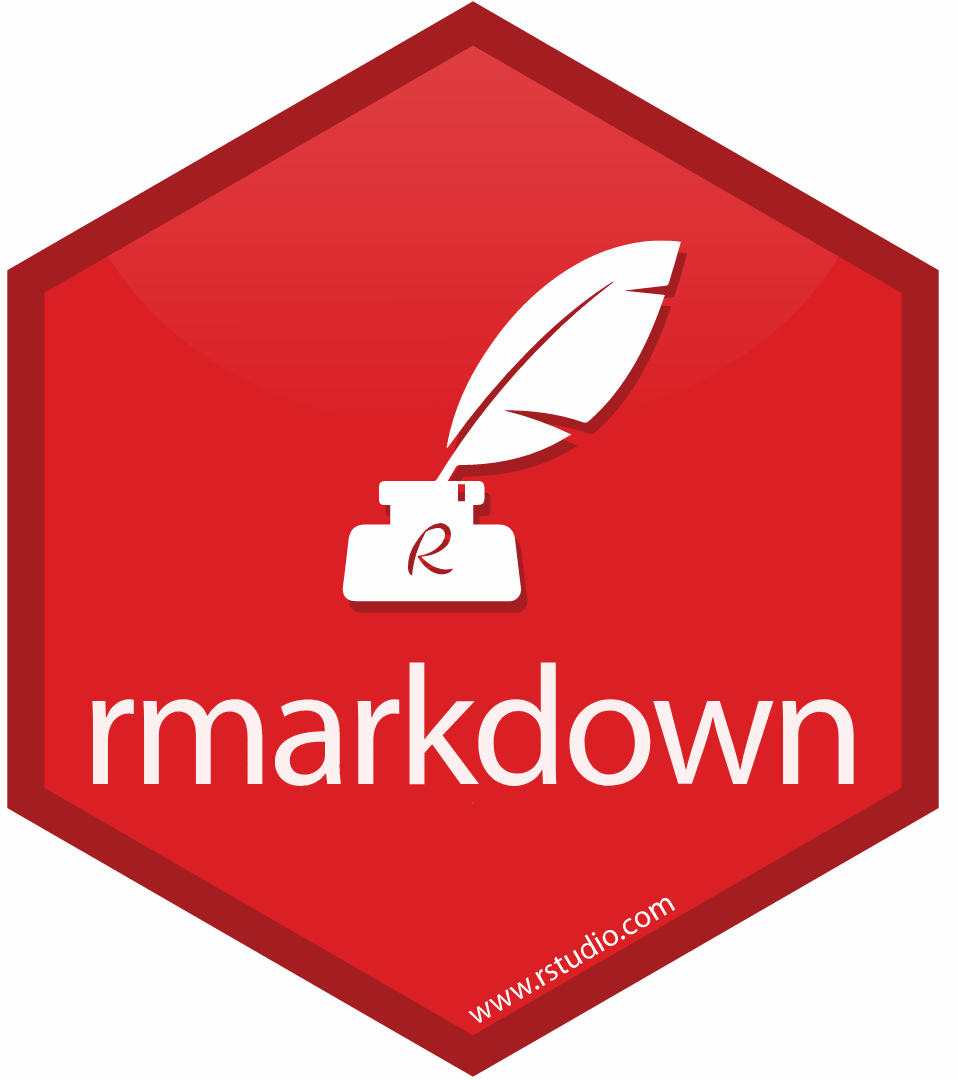
\*\* Code \*\*

- First bullet  
- Second bullet  
    - Two tabs indents to the next level bullet.  
    - Second sub-bullet.

\*\* Output \*\*

* First bullet
* Second bullet
  + Two tabs indents to the next level bullet.
  + Second sub-bullet.

## Figures. If you simply have a figure, you may use the standard markdown format to render the figure. The code is: ![Caption](Figure\_Path`){width=desiredwidth}.



1. RMarkdown

## Math. You can show mathematical equations using the dollar sign ($).

$ Equation $ produces an in-line equation like this: .

$$ Equation $$ produces an equation block like this:

# R Output. The real utility of this is that you have some need to analyze data and produce reports and/or produce repeated reports from some data source. You do this by calling R.

## R Code Chunks. You can call r by indicating n R code chunk as follows:

```{r sectionName, …}  
*R code here*  
```

For example, you can manipulate data and get useful infromation as follows (note, this requires dplyr):

library(dplyr)  
NumCarsGoodGasMileage <- mtcars %>% filter(mpg > 20) %>% count  
NumCarsGoodGasMileage <- as.character(NumCarsGoodGasMileage)

Note how this shows your code and the resulting information. If you have “echo = FALSE”, your code will be suppressed.

If you want to call this in line you can do this as follows:

* **Code**: ` r NumCarsGoodGasMileage` cars have good gas mileage.
* **Yield**: 14 cars have good gas mileage.

## R Tables. Often (typically), you want to produce tables from R data sources. This is usefully done with the pander package. Note that pander has a wealth of options to style your tables. See <https://www.r-project.org/nosvn/pandoc/pander.html> for more information.

library(pander)  
myTable <- mtcars %>% select(mpg, cyl, hp) %>% head(10)  
pander::pander(myTable, 'First 10 Rows of Selected Columns of mtcars')

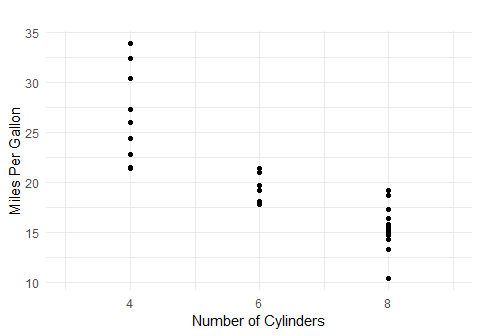
1. First 10 Rows of Selected Columns of mtcars

|  |  |  |  |
| --- | --- | --- | --- |
|  | mpg | cyl | hp |
| **Mazda RX4** | 21 | 6 | 110 |
| **Mazda RX4 Wag** | 21 | 6 | 110 |
| **Datsun 710** | 22.8 | 4 | 93 |
| **Hornet 4 Drive** | 21.4 | 6 | 110 |
| **Hornet Sportabout** | 18.7 | 8 | 175 |
| **Valiant** | 18.1 | 6 | 105 |
| **Duster 360** | 14.3 | 8 | 245 |
| **Merc 240D** | 24.4 | 4 | 62 |
| **Merc 230** | 22.8 | 4 | 95 |
| **Merc 280** | 19.2 | 6 | 123 |

# https://www.rdocumentation.org/packages/pander/versions/0.6.3/topics/panderOptions

## R Plots. Similarly, one may wish to plot different aspects of data sources. This is done in a similar fashion.

library(ggplot2)  
myDF <- mtcars %>% select(mpg, cyl, hp)  
p1 <- ggplot(myDF) + geom\_point(aes(x = cyl, y = mpg)) +   
 theme\_minimal() + xlim(c(3, 9)) +   
 xlab('Number of Cylinders') + ylab('Miles Per Gallon') + ggtitle('')  
p1



1. mtcars Cylinders vs. Gas Mileage

## R Analysis. You may also want to show outputs of your analysis. This example uses pander to show a linear regression that models miles per gallon as a function of number of cylinders and horsepower.

# We're going to consider the number of cylinders to be categorical  
myDF$cyl <- as.factor(myDF$cyl)   
  
# Build the model using lm  
myLM <- lm(myDF$mpg ~ myDF$cyl + myDF$hp)   
  
# Rename the coefficients for clarity  
names(myLM$coefficients) <- c('Intercept', '6 Cylinder', '8 Cylinder', 'Horsepower')  
  
# Round the output to at most 3 decimal places  
pander::panderOptions('round', 3)  
  
#Produce the table using pander  
pander::pander(myLM, 'mtcars model: mpg ~ cyl + hp')

1. mtcars model: mpg ~ cyl + hp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Estimate | Std. Error | t value | Pr(>|t|) |
| **Intercept** | 28.65 | 1.588 | 18.04 | 0 |
| **6 Cylinder** | -5.968 | 1.639 | -3.64 | 0.001 |
| **8 Cylinder** | -8.521 | 2.326 | -3.663 | 0.001 |
| **Horsepower** | -0.024 | 0.015 | -1.56 | 0.13 |

## Other uses of R. Thus far we have shown the code from R. Generally, you won’t want to do that. To suppress code set echo=FALSE (this is the default). The following example pulls the headline off of the top story from Reuters and stores it as a variable you can reference later. This requires rvest and httr.

We can then call R in line like this. The Reuters headline as of 2019-06-02 15:16:15 is:

##### **Tariffs on Mexico won’t derail trade deal: U.S. officials**.

# \*Signature block and enclosures.\*\* The signature block and enclosures are listed as a table to keep the spacing correct. The &nbsp; ugliness is there to keep the spacing clean (on word). You can add additional rows as necessary for more enclosures or more lines in the signature block.

|  |  |
| --- | --- |
| X Encls | FIRST MI LAST |
| 1. Enclosure 1 | RANK, BRANCH |
| 2. Enclosure 2 | Job |
| 3. Enclosure N |  |
|  |  |