Notes

JC

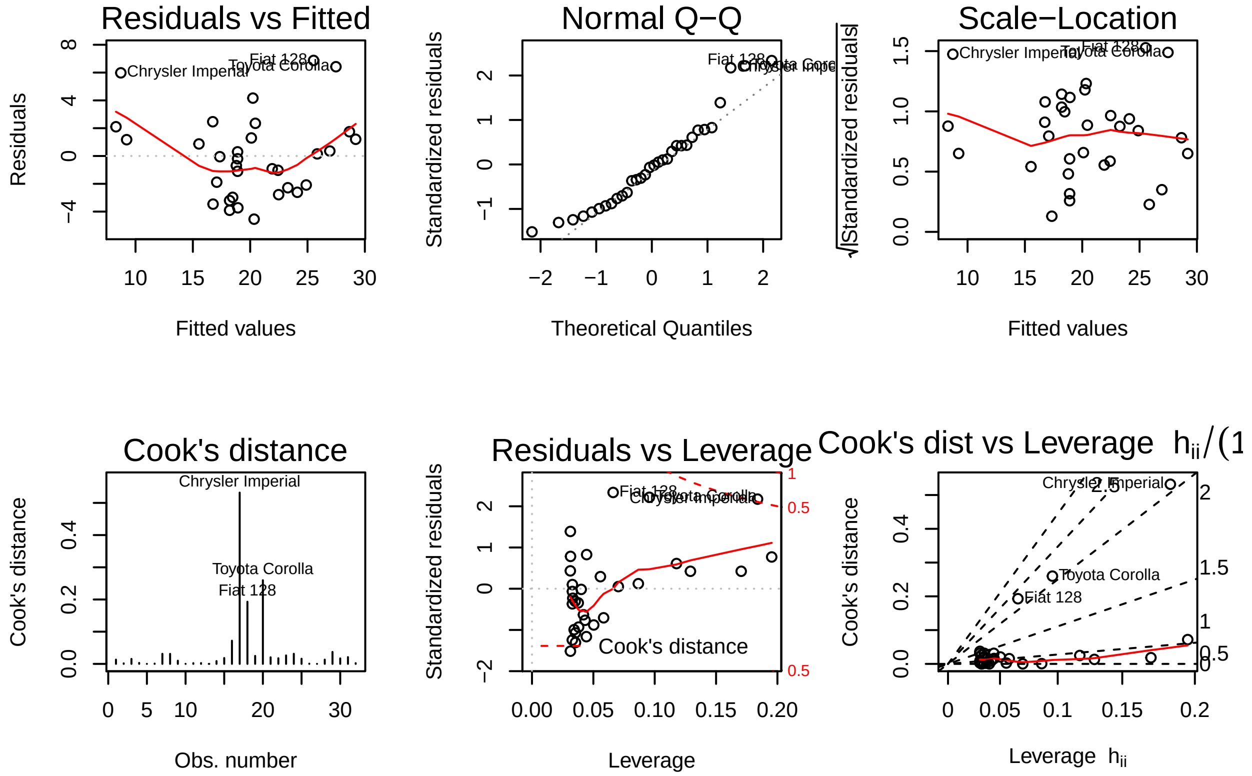
Why?

Save some time. Quick & Dirty model diagnostic plots in ggplot.

Plan

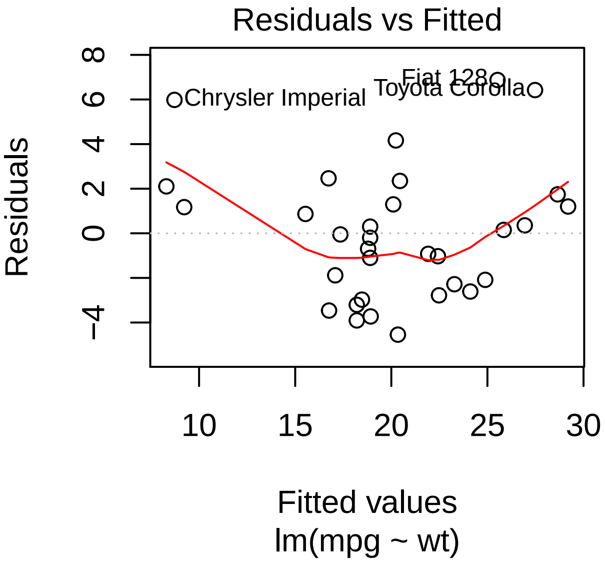
1. Redo base plots in ggplot

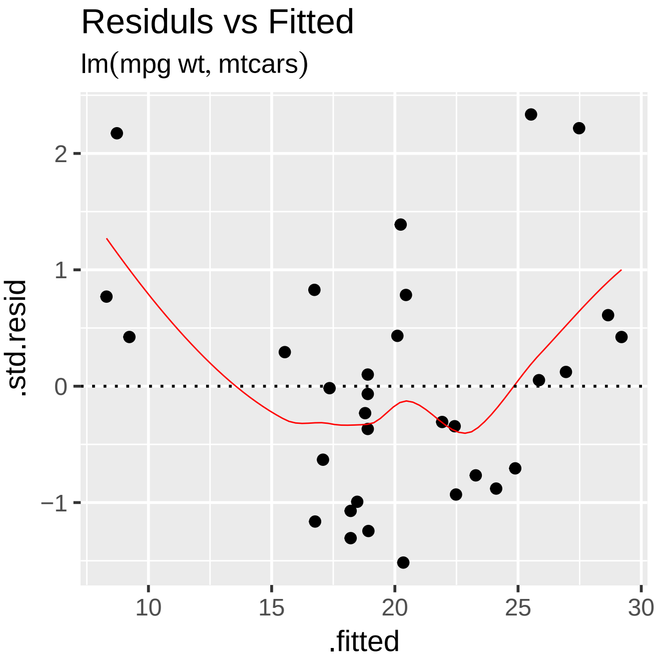
```r
par(mfrow=c(2,3))
lm.1 <- lm(mpg ~ wt,data=mtcars)
plot(lm.1,which=1:6)
```



2. Create ggplot versions ~ make tidy

```r
library(tidyverse)
library(broom)

My.Mod <- lm.1

#Base
plot(My.Mod,which=1)

#GGplot
D1  <- augment(My.Mod) %>% ggplot(aes(x=.fitted,y=.std.resid)) +
  geom_point() +
  geom_smooth(se=FALSE,colour="red",size=.25) +
  geom_hline(yintercept=0,linetype=3) +
  labs(title="Residuls vs Fitted",subtitle=My.Mod$call)

D1
```





3. Functionalise

How to determine models?
What sort of arguments, functionality?
PAss through
Design ?

```
## What should it look like?

# Pehaps???

MyModel %>% ggdiag()

Data %>% MyModel %>% ggdiag()

# plot index no or rather name?
Data %>% MyModel %>% ggdiag(Plot=1:2)
Data %>% MyModel %>% ggdiag(Plot=c("RVFit","qq"))
```

4. Ensure plays nice with existing

x. Other Thoughts
Practice with RProj, Git.
Don't re-invent the wheel - just make quicker to change the flat.

Questions ???

Logic for text annotation? How determine cutoff. Starting assumptiom? Start from a) model (eg lm) or b) tidy output of model

```
# How do you see how current plot is coded?
# This doesnt help
plot
```

```
## function (x, y, ...)
## UseMethod("plot")
## <bytecode: 0x7f93042ab180>
## <environment: namespace:graphics>
```