BEG15-008 Mark Newman

bone surrogate stiffness testing

*Omnia analysis in triem partis divisis*  
— Caesar, *De Bello Analytica*

# background

Data analysis is generally divided into three parts:

1. munge the data, possibly from different sources, into a consistent format
2. analyse the data to generate “useful” results
3. present the findings informatively, either as text or graphs or …

# munge

First load the vital packages we will use later…

library(bluer) # functions to work with Bluehill data files  
library(data.table) # need here as bluer uses data.tables to hold all data  
library(stringr) # simplified regex tools, to pull specimen & study ID apart  
library(ggplot2) # awesome plotting

## get sample info from files

define the study (i.e. where to start looking for samples)

# use a relative path to the extdata folder  
study\_root <- "../inst/extdata"

Then find all RawData files under this starting point. Identify the studies that might be there by looking at parts of the full pathname and extract any headers that there might be in the RawData files.

samples <- bh\_find\_specimens(study\_root)  
headers <- bh\_get\_headers(samples)

Augment the samples table by adding in important bits of the headers (i.e. the blank row number and the specimen label), which are extracted from the header and joined to the specimen data by filename. The full pathname is guaranteed to be unique per file, unlike anything we might generate.

samples <- samples[bh\_get\_labels(headers), on = "filename"]

Doing it this way makes a copy of the data.table, but it is unlikely that the samples table will ever be big enough for this to be a problem.

## identify the individual specimens

Parse the label, sample, specimen and filename fields to identify & group the RawData according to what is important for this study. This includes things like labelling repeats, distinct physical samples et.

The general process is common, but the details will vary considerably by study.  
In this case we are identifying tests by study, material, bone (geometry) and repeat. Start by classifying into studies according to (mainly) filename…

# there were two different synbone studies, identified as: initial and redo  
samples[filename %like% "Initial synbone", study := "synbone.1"]  
samples[filename %like% "redo\_synbone", study := "synbone.2"]  
  
# there were some pilot tests  
samples[filename %like% "pilot", study := "pilot"]  
samples[filename %like% "radius", study := "pilot"]  
  
# some strainrate tests, on a synbone but not having a study yet  
samples[filename %like% "SLT-SL-200" & is.na(study), study := "strainrate"]  
  
# and everything that's left should be the multi-material study  
samples[is.na(study), study := "multi"]

Make any required fix-ups for typos & other errors.  
As it happens, these have all been fixed in the source files now.

# Wrong labels entered  
# # clean up the ID for the Initial synbone study  
# samples[ID %like% "^synbone", ID := str\_replace(ID, "synbone ", "")]

Parse the filenames & labels to get material, bone and rep

# synbone.1 has material & rep in ID, which may start with "synbone "  
syn\_regex <- "^(?:synbone ?)?(.+)[\\.\\\_](\\d+)\\.\\d+$"  
samples[study == "synbone.1",  
 c("material", "rep") := data.table(str\_match(ID, syn\_regex)[, -1])]  
# synbone.2 has material.rep in the label  
samples[study == "synbone.2",  
 c("material", "rep") := tstrsplit(label, "\\.")]  
  
# make a material & rep for the radius pilot study (no rep encoded in names)  
# coerce rep to character as column type was set when used regex above  
samples[filename %like% "radius",  
 c("material", "rep") := .("bone",as.character(.I))]  
  
# for the multi study the material is the sample name  
samples[study == "multi", material := sample]  
# for multi study label is bone.repeat, so use strsplit  
samples[study == "multi", c("bone", "rep") := tstrsplit(label, "\\.")]  
  
# change the "cadaver.\*" material to bone (shorter & more correct)  
samples[material %like% "cadaver", material := "bone"]  
  
# all unallocated tests (synbone, pilot & strainrate) were based on bone 17L  
samples[is.na(bone), bone :="17L"]  
  
# alocate material and rep for the miscellaneous studies  
# used a synbone for strain rate testing  
samples[study == "strainrate",  
 c("material", "rep") := .("SLT-SL-200", as.character(.I))]  
# the ABS pilot used FDM\_ABS, not the more robust Digital\_ABS  
samples[study == "pilot" & ID %like% "ABS",  
 c("material", "rep") := .("FDM\_ABS", as.character(.I))]

Most of early studies were single ramps, but the later studies were multi-cycle tests. Label the studies accordantly so we know which ones to chop into cycles later.

# Most of the pilot studies were single ramps, as was the synbone.1 study.  
samples[c("pilot", "synbone.1"), loading := "ramp", on = "study"]  
# Others were repeated (cyclic) tests  
samples[c("synbone.2", "multi", "strainrate"), loading := "cyclic", on = "study"]  
# pilot bone 2 was also cyclic  
samples[study == "pilot" & rep == 2, loading := "cyclic"]

Now we have all samples identified with study, material, bone and rep, check that the combinations are unique

all\_samples\_N <- samples[, .N]  
unique\_ID\_N <- samples[, 1, by="study,material,bone,rep"][, .N]  
stopifnot(all\_samples\_N == unique\_ID\_N)  
# make a UID for later use  
samples[, UID := paste(study, material, bone, rep, sep = ".")]

## load the actual test data

Read all the raw data files listed in samples into a single data.table Note that these all need to have the same number of columns. Samples includes everything found under study\_root so will need to make sure to only use RawData files that are comparable, or use bh\_min\_cols to restrict to a subset that should always be present (Time, Extension, Load).

sample\_data <- samples[, bh\_read\_data(filename, blank\_row, bh\_min\_cols),  
 by = filename]  
  
# Combine sample info with data points just read in.  
# Add the identifiers etc. collected in samples, keyed by the filename.  
sample\_data <- samples[sample\_data, on = "filename"]  
  
# These were all compression tests so invert the  
# Load & Extension to plot the "right" way up.  
bh\_make\_compressive(sample\_data)

## seperate into cycles

For multi-cycle tests, segment into cycles and find the third cycle. If there is no third cycle, choose the biggest (second, or first).

For each cyclic test, label the cycles and segments based on Extension. Using , but could use , as long as it is unique to each individual test.

sample\_data[loading == "cyclic",  
 c("cycle", "seg", "peaks") := label\_cycles(Extension),  
 by = UID]  
  
# for ramp tests there is one cycle and it is a load segment  
sample\_data[loading == "ramp",  
 c("cycle", "seg") := .(1L, "load"),  
 by = UID]

We only want the loading part of cycle 3, but if there is no cycle 3 use the biggest available. This also works with the studies, which only have one “cycle”.

third\_cycle <- sample\_data[, .(cycle=min(max(cycle), 3)), by=UID]  
third\_cycle[, seg := "load"] # only want the load parts  
  
# copy just those bits out of sample\_data   
useful\_data <- sample\_data[third\_cycle, on=c("UID", "cycle", "seg")]

The pilot, strainrate and synbone.1 studies were not the same as the rest, so only want to keep the multi & synbone.2 studies

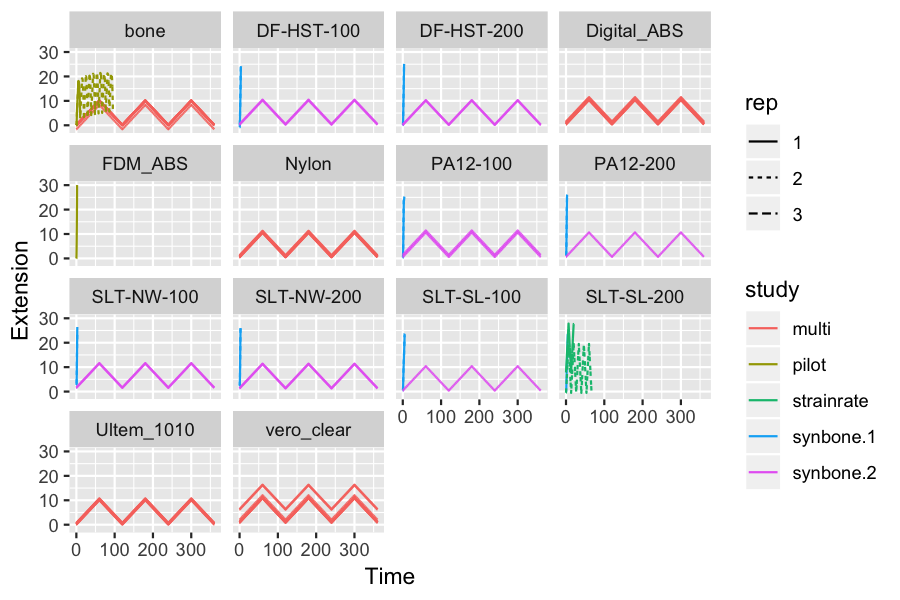
useful\_data <- useful\_data[study %in% c("multi", "synbone.2"), ]

# Analyse

## check inputs are OK

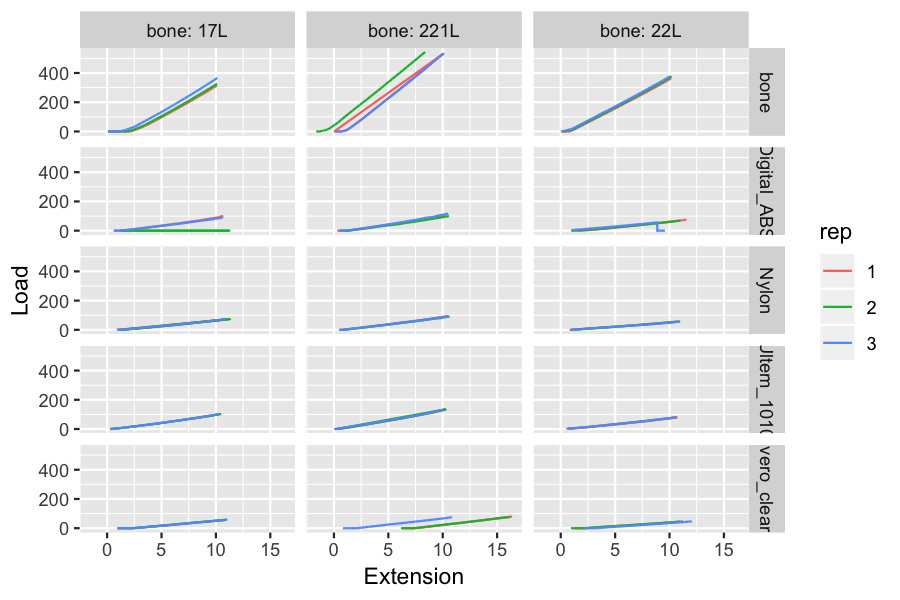
Start with a quick sanity plot of everything, to make sure we are reading the data correctly.

p0 <- ggplot(sample\_data) +  
 aes(x = Time, y = Extension, group = filename, colour = study, linetype = rep) +  
 geom\_path() +  
 facet\_wrap(material~.)  
print(p0)



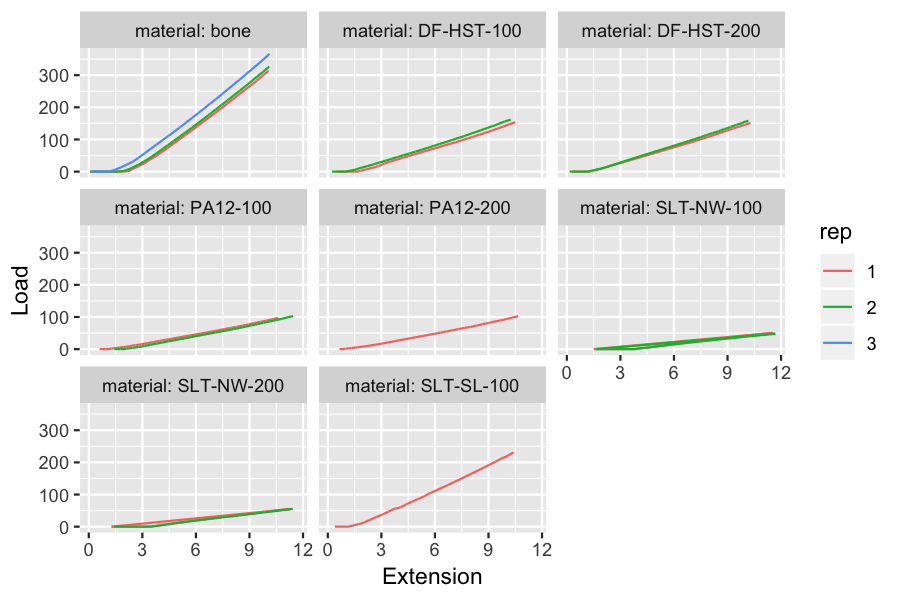
Check the multi study with a plot to check that sample repeats are similar…

p.multi <- ggplot(useful\_data[study == "multi", ]) +  
 aes(x = Extension, y = Load, group = filename, colour = rep) +  
 geom\_path() +  
 facet\_grid(cols = vars(bone), rows=vars(material),  
 labeller = labeller(.cols = label\_both))  
print(p.multi)



Compare the cyclic synbone tests with the multi tests. Note that all the synbone specimens were bone 17L, so only get matching samples from the multi study.

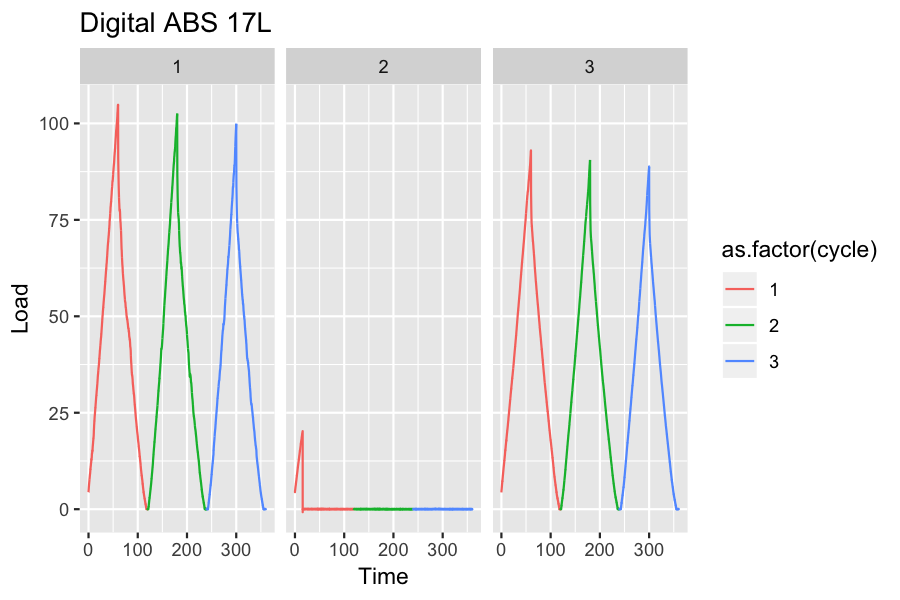
syn\_and\_bone <- useful\_data[(study == "synbone.2") |  
 (study == "multi" & material == "bone" & bone == "17L"), ]  
  
p.synbone <- ggplot(syn\_and\_bone) +  
 aes(x = Extension, y = Load, group = filename, colour = rep) +  
 geom\_path() +  
 facet\_wrap(~material, ncol = 3,  
 labeller = labeller(.cols = label\_both))  
print(p.synbone)



## exclude a bad test

These plots reveal a problem in Digital ABS bone 17L, rep 2, where the bone fractured before the test was complete.

ggplot(sample\_data[UID %like% "multi.Digital\_ABS.17L", ])+aes(x=Time, y=Load, colour=as.factor(cycle))+geom\_line()+facet\_wrap(rep~.)+ggtitle("Digital ABS 17L")



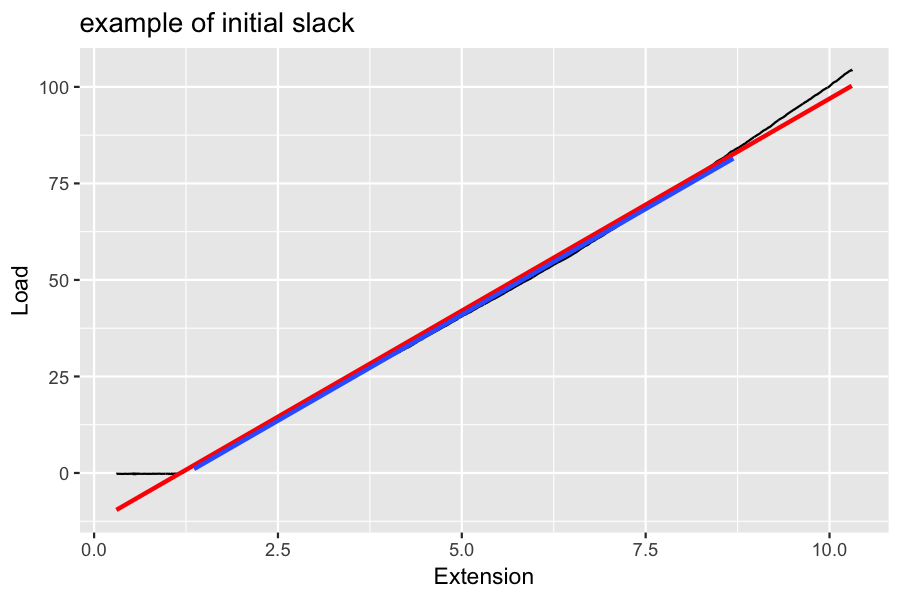
This result will be removed from further analysis.

## some real analysis

### trim the data

Remove any initial slack, where the testing machine has moved before the load starts. This can be due to slack in the test rig, the sample moving in the grips, or just starting the test in the wrong position, as seen with Digital ABS bone 221L, rep 1. Fitting the data including the initial slack can give the wrong slope (red) compared to fitting just the actual test (blue), although extra points near zero have less effect than point after a break.

DT <- useful\_data[UID=="multi.Digital\_ABS.221L.1"]  
ggplot(DT) + aes(x = Extension, y = Load) + geom\_line(colour = "black", alpha=1.0) +  
 geom\_smooth(method="lm", formula = y ~ x, colour = "red", alpha = 0.01) +  
 geom\_smooth(data=DT[Load %between% c(1.8, 83.5), ],   
 method="lm", formula = y ~ x, alpha = 0.01) +  
 ggtitle("example of initial slack")



Also remove any data after the peak, in case that wasn’t trimmed by the cycle selection earlier. Plot the resulting trimmed third (or earlier if no third) loading cycle.

# calculate stiffness  
trimmed <- useful\_data[, trim\_slack(.SD, Load ~ Extension), by=UID]  
stiff <- trimmed[, lm\_simple(Load ~ Extension, .SD), by="study,material,bone,rep"]

Drop the results from the previously identified bad test.

# the results of the linear fits  
lm\_cols <- c("int","slope","p","rsq")  
# erase the results for this one test  
stiff[.("multi","Digital\_ABS","17L","2"), (lm\_cols) :=NA,  
 on=c("study","material","bone","rep")]

And list the stiffness for each test.

print(stiff)

## study material bone rep int slope p rsq  
## 1: multi Digital\_ABS 221L 1 -15.188073 11.306636 0 0.9983091  
## 2: multi Digital\_ABS 221L 2 -16.453973 10.652886 0 0.9976584  
## 3: multi Digital\_ABS 221L 3 -17.813348 12.413968 0 0.9977247  
## 4: multi Digital\_ABS 22L 1 -18.109426 7.883939 0 0.9971281  
## 5: multi Digital\_ABS 22L 2 -12.333614 7.109324 0 0.9969886  
## 6: multi Digital\_ABS 17L 1 -15.207426 10.196448 0 0.9962499  
## 7: multi Digital\_ABS 17L 3 -13.320981 9.348933 0 0.9976172  
## 8: multi Nylon 221L 1 -8.898282 9.568168 0 0.9988494  
## 9: multi Nylon 221L 2 -9.678829 9.230478 0 0.9989408  
## 10: multi Nylon 221L 3 -9.153846 9.361145 0 0.9984059  
## 11: multi Nylon 22L 1 -5.672707 5.694733 0 0.9985622  
## 12: multi Nylon 22L 2 -6.487930 5.635617 0 0.9983877  
## 13: multi Nylon 22L 3 -6.233090 5.458121 0 0.9980250  
## 14: multi Nylon 17L 1 -10.575533 7.324298 0 0.9962356  
## 15: multi Nylon 17L 2 -12.300102 7.436616 0 0.9972608  
## 16: multi Nylon 17L 3 -9.249356 7.290375 0 0.9970552  
## 17: multi Ultem\_1010 221L 1 -8.863532 13.316413 0 0.9982859  
## 18: multi Ultem\_1010 221L 2 -7.708396 13.322524 0 0.9976559  
## 19: multi Ultem\_1010 221L 3 -4.118977 12.988325 0 0.9982235  
## 20: multi Ultem\_1010 22L 2 -5.757338 7.849201 0 0.9981521  
## 21: multi Ultem\_1010 22L 1 -6.804669 7.763616 0 0.9970236  
## 22: multi Ultem\_1010 22L 3 -4.958955 7.857800 0 0.9977571  
## 23: multi Ultem\_1010 17L 1 -7.743359 10.211403 0 0.9977542  
## 24: multi Ultem\_1010 17L 2 -7.889742 10.181881 0 0.9970962  
## 25: multi Ultem\_1010 17L 3 -7.473719 10.264758 0 0.9971367  
## 26: multi bone 17L 1 -101.146966 40.437038 0 0.9978034  
## 27: multi bone 22L 1 -35.838932 38.857665 0 0.9997450  
## 28: multi bone 221L 1 -67.608281 59.472918 0 0.9998049  
## 29: multi bone 22L 2 -32.028119 39.475806 0 0.9995720  
## 30: multi bone 221L 2 42.072098 59.378550 0 0.9996005  
## 31: multi bone 17L 2 -95.811452 41.006438 0 0.9976253  
## 32: multi bone 221L 3 -66.293531 58.989908 0 0.9996569  
## 33: multi bone 17L 3 -72.661278 42.322857 0 0.9970886  
## 34: multi bone 22L 3 -22.564851 39.197101 0 0.9992560  
## 35: synbone.2 SLT-SL-100 17L 1 -40.350580 25.566204 0 0.9985036  
## 36: synbone.2 PA12-100 17L 1 -14.326186 10.206266 0 0.9980218  
## 37: synbone.2 PA12-100 17L 2 -24.040518 10.820120 0 0.9987216  
## 38: synbone.2 PA12-200 17L 1 -13.688388 10.519848 0 0.9971151  
## 39: synbone.2 DF-HST-100 17L 1 -30.373920 17.217164 0 0.9993451  
## 40: synbone.2 DF-HST-100 17L 2 -23.057659 17.682139 0 0.9986644  
## 41: synbone.2 DF-HST-200 17L 1 -23.727406 16.750889 0 0.9991856  
## 42: synbone.2 DF-HST-200 17L 2 -24.859273 17.756167 0 0.9994414  
## 43: synbone.2 SLT-NW-100 17L 1 -23.456185 6.487227 0 0.9994734  
## 44: synbone.2 SLT-NW-100 17L 2 -22.375397 6.121965 0 0.9990684  
## 45: synbone.2 SLT-NW-200 17L 1 -22.700695 6.951618 0 0.9994245  
## 46: synbone.2 SLT-NW-200 17L 2 -22.651274 6.850871 0 0.9991723  
## 47: multi vero\_clear 221L 1 -68.337539 9.009183 0 0.9998248  
## 48: multi vero\_clear 221L 2 -64.236619 8.707505 0 0.9998341  
## 49: multi vero\_clear 221L 3 -17.789787 8.424243 0 0.9997301  
## 50: multi vero\_clear 22L 1 -12.093259 5.128544 0 0.9994191  
## 51: multi vero\_clear 22L 2 -9.728184 4.910406 0 0.9996291  
## 52: multi vero\_clear 22L 3 -16.497775 5.162847 0 0.9995244  
## 53: multi vero\_clear 17L 1 -16.337543 6.647689 0 0.9997313  
## 54: multi vero\_clear 17L 2 -16.322972 6.823845 0 0.9998739  
## 55: multi vero\_clear 17L 3 -16.861011 6.712954 0 0.9997434  
## study material bone rep int slope p rsq

The repeats for each material are very similar, so average them to compare each material to the results for bone.

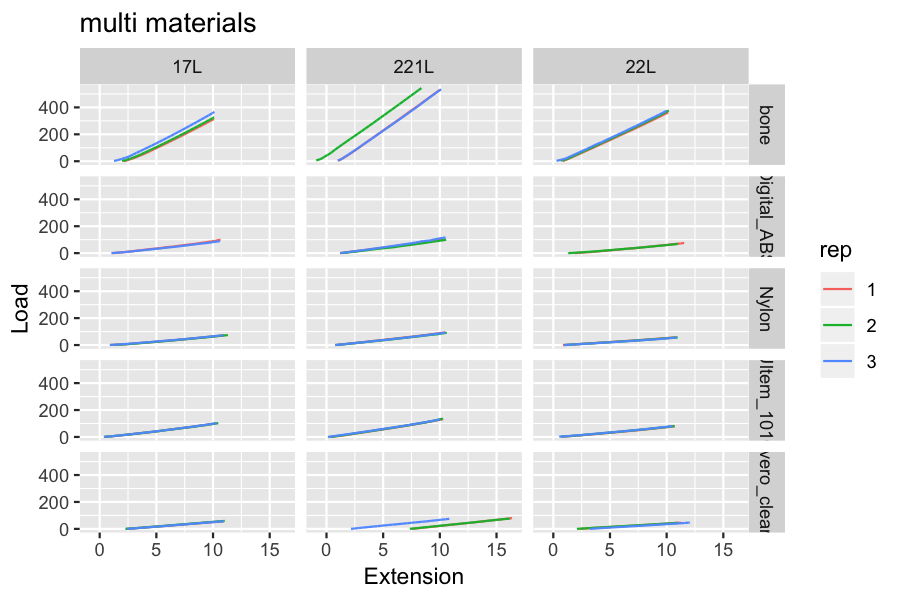
# using na.rm will skip the bad test identified above.  
mean\_stiff <- stiff[, lapply(.SD, mean, na.rm = TRUE), by="study,material,bone", .SDcols=lm\_cols]

# Present

## multi-material

Compare by bone shape and material, showing that the bone appears to be substantially stiffer than any of the other materials

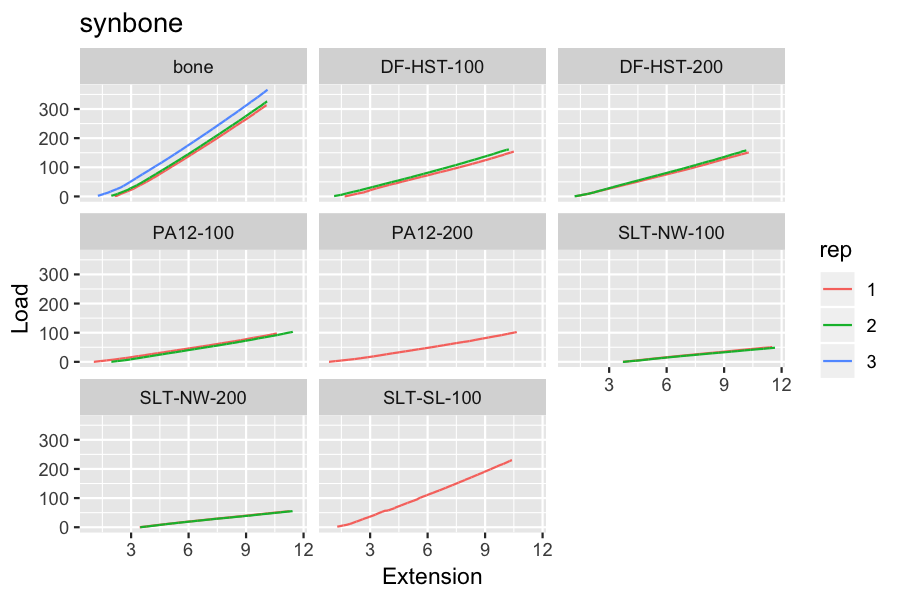
ggplot(trimmed[study == "multi", ]) +  
 aes(x = Extension, y = Load, colour = rep) +   
 geom\_line() +  
 facet\_grid(material ~ bone) +  
 ggtitle("multi materials")



## synbone

All the synbone samples were modelled after bone 17L, so include that in the results for comparison.

ggplot(trimmed[study == "synbone.2" | (material == "bone" & bone =="17L"), ]) +   
 aes(x = Extension, y = Load, colour = rep) +  
 geom\_line() +   
 facet\_wrap(~material) +   
 ggtitle("synbone")



## numerical results

The bending stiffness for each material are tabulated. As the stiffness was very similar for repeats of the same material they were averaged.

print(mean\_stiff[order(study,material)])

## study material bone int slope p rsq  
## 1: multi Digital\_ABS 221L -16.485131 11.457830 0 0.9978974  
## 2: multi Digital\_ABS 22L -15.221520 7.496632 0 0.9970583  
## 3: multi Digital\_ABS 17L -14.264203 9.772691 0 0.9969336  
## 4: multi Nylon 221L -9.243652 9.386597 0 0.9987320  
## 5: multi Nylon 22L -6.131242 5.596157 0 0.9983250  
## 6: multi Nylon 17L -10.708330 7.350430 0 0.9968505  
## 7: multi Ultem\_1010 221L -6.896968 13.209087 0 0.9980551  
## 8: multi Ultem\_1010 22L -5.840321 7.823539 0 0.9976442  
## 9: multi Ultem\_1010 17L -7.702273 10.219347 0 0.9973290  
## 10: multi bone 17L -89.873232 41.255444 0 0.9975058  
## 11: multi bone 22L -30.143968 39.176858 0 0.9995243  
## 12: multi bone 221L -30.609905 59.280459 0 0.9996874  
## 13: multi vero\_clear 221L -50.121315 8.713644 0 0.9997963  
## 14: multi vero\_clear 22L -12.773073 5.067266 0 0.9995242  
## 15: multi vero\_clear 17L -16.507175 6.728163 0 0.9997829  
## 16: synbone.2 DF-HST-100 17L -26.715789 17.449651 0 0.9990048  
## 17: synbone.2 DF-HST-200 17L -24.293339 17.253528 0 0.9993135  
## 18: synbone.2 PA12-100 17L -19.183352 10.513193 0 0.9983717  
## 19: synbone.2 PA12-200 17L -13.688388 10.519848 0 0.9971151  
## 20: synbone.2 SLT-NW-100 17L -22.915791 6.304596 0 0.9992709  
## 21: synbone.2 SLT-NW-200 17L -22.675984 6.901245 0 0.9992984  
## 22: synbone.2 SLT-SL-100 17L -40.350580 25.566204 0 0.9985036  
## study material bone int slope p rsq

The bending stiffness of each bone will vary due to differences in geometry. This effect was removed by computing the relative stiffness for each material to that of bone stiffness(material)/stiffness(bone).

# ratio of stiffness to bone material stiffness, for each bone shape  
ratios <- mean\_stiff[, .(material= .SD[, material],   
 ratio=.SD[, slope]/.SD[material=="bone", slope]), by="bone"]  
# and sort in descending order  
ratios <- ratios[order(-ratio), .SD, by="bone"]  
print(ratios)

## bone material ratio  
## 1: 221L bone 1.0000000  
## 2: 221L Ultem\_1010 0.2228236  
## 3: 221L Digital\_ABS 0.1932817  
## 4: 221L Nylon 0.1583422  
## 5: 221L vero\_clear 0.1469902  
## 6: 22L bone 1.0000000  
## 7: 22L Ultem\_1010 0.1996980  
## 8: 22L Digital\_ABS 0.1913536  
## 9: 22L Nylon 0.1428434  
## 10: 22L vero\_clear 0.1293433  
## 11: 17L bone 1.0000000  
## 12: 17L SLT-SL-100 0.6197050  
## 13: 17L DF-HST-100 0.4229660  
## 14: 17L DF-HST-200 0.4182121  
## 15: 17L PA12-200 0.2549930  
## 16: 17L PA12-100 0.2548316  
## 17: 17L Ultem\_1010 0.2477091  
## 18: 17L Digital\_ABS 0.2368824  
## 19: 17L Nylon 0.1781687  
## 20: 17L SLT-NW-200 0.1672808  
## 21: 17L vero\_clear 0.1630854  
## 22: 17L SLT-NW-100 0.1528185  
## bone material ratio

It can be seen that even the best materials are substantially less stiff than bone.

Most of the synbones were stiffer than the other materials. This is probably in part due to the foam filling in the sysnbones.

The cortex material has a significant effect on stiffness, but the different foam densities do not have any independent effect on stiffness.

# just the synbones have foam  
# split the cortex & foam parts out of the material identifier & add slope  
foam\_effect <- stiff[study %like% "synbone",   
 data.table(  
 str\_match(material, "(.\*)\\-(\\d{3})$")[,-1],  
 slope)  
 ]  
setnames(foam\_effect, c("cortex", "foam", "stiffness"))  
  
# now do a two way ANOVA  
model<-aov(stiffness ~ cortex + foam, data=foam\_effect)  
print(summary(model))

## Df Sum Sq Mean Sq F value Pr(>F)   
## cortex 3 415.1 138.36 802.705 3.05e-09 \*\*\*  
## foam 1 0.1 0.06 0.357 0.569   
## Residuals 7 1.2 0.17   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1