A Bayesian Analysis of the Variate Strength of PLA

Introduction

Material selection is a key component to the design of any product. Engineers often select materials based on a few key characteristics. These characteristics vary by product, but often include items such as cost, ease of manufacture, weight, strength, and sometimes even aesthetics. In this project we describe a process for characterizing the strength of a couple of samples of polylactic acid (PLA). PLA is a widely popular material for cheap and rapid prototyping when used as filament for 3D printing.

As a polymer, PLA does not have a single characteristic strength. Instead, the strength of PLA follows a distribution of values based on the polymer chain orientations, which are randomly determined. This leads to a conundrum for engineers, who have many equations and formulas for part design that rely on material strength as a key known factor. To solve this conundrum, it is standard to use an “expected” or “probable” strength for the material in question. We present a Bayesian model to predict such a strength for Overture PLA and provide several estimates that could be used depending upon the specific context of the design problem.

Methods

Data Collection

The breaking strength (kN) of two groups of 3D-printed PLA links was measured using a 50kN Instron tensile tester. We followed standard operating procedure for a tensile test - slow rate motion (5mm/min) that provided a continual application of force until the link broke. The maximum force applied just prior to failure was recorded for a total of 21 links from two groups. The groups visually vary in color, one white the other gray, and potential mechanical variation between the two groups could be due to being different batches of PLA. The link design was typical of tensile test specimens, with wide ends and a narrow center to ensure consistent fracture in the neck.

Model Development

We are willing to assume that the strength of the links follows a Weibull distribution with shape α and scale β. We chose this because the Weibull is a flexible, skewed distribution. To estimate these parameters, we use our knowledge of Overture PLA and our desired certainty for its strength. Our prior knowledge and the design of our specific links led us to be 99% sure that the strength of each link would be between 1.62 and 3.00 kN. Our beliefs about the distribution of α are derived from the published material properties of this Overture PLA, namely that the tensile strength is 46.6 MPa. Using this strength, we calculate an upper and lower bound for the strength using the range of possible cross-sectional area values of the link. These values range from 0.1 in.^{2} for 100% infill to 0.054 in.^{2} for 0% infill. The entire range of infill values is present throughout the link due to the specific infill pattern. Using these values as the upper and lower bounds of a 99% credible interval, we calculate the hyperparameters of the prior on α. For the scale parameter β, we choose a Gamma prior that is relatively flat, as this distribution highly favors large values. This results in the following model:

$$f(y \mid \alpha, \beta) \sim \text{Weibull}(\alpha, \beta)$$
$$\pi(\alpha) \sim \text{Weibull}(2.72, 11.35)$$
$$\pi(\beta) \sim \text{Gamma}(0.001, 1)$$

Since our data appears to fall into two groups (white links and gray links) that indicate the product batch in which they were manufactured, we first determine if the strength differs by group. Therefore, we developed two models to test whether either of the parameters of the distribution of strength differs between the white and the gray links.

H_0: The distributions of link strengths share a common value of α and β regardless of filament color

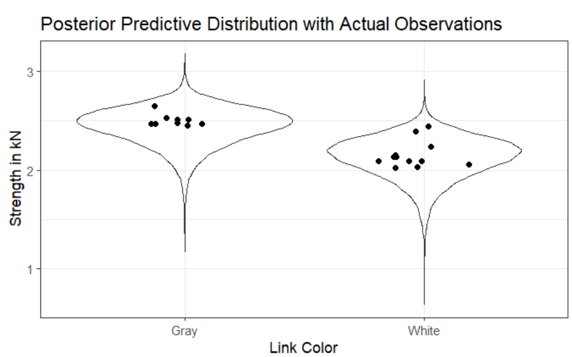
H_1: The distributions of link strengths have unique values of α and β based on the filament color

Due to the computational complexity of these models, we implemented an MCMC algorithm using Stan to generate a sample from the posterior distribution of the parameters under each model. Under the initial assumption that each model was equally likely, they were compared using bridge sampling to determine the probability of each model given the data.

Results & Discussion

Model Comparison

We are 99.97% sure that the link strengths follow different distributions based on the filament color. Using this model, we now compute useful values for estimating the strengths of both groups, though our final report would only include the stronger group because that is the batch an engineer would pursue for use in a design.



Application

Different product design intentions will lead to various levels of requisite certainty. We report a lower bound on the predicted strength for several different levels of surety.

Confidence	White Link Strength (kN)	Gray Link Strength (kN)
99.99%	0.9512979	1.38
99%	1.5029528	1.86
95%	1.7387802	2.09
90%	1.8461761	2.192314

Interpretation

Now that we have a series of predicted values, we must choose which one to use in the design equations. The safe road is to always go with more surety; however, the question then becomes how much more surety is desired? The greater the certainty, the smaller the predicted value as it is always possible that a particular specimen will fail at functionally zero force due to internal inconsistencies and flaws. There are a couple of places to look for guidance in this decision: industry-specific standards and similar cases within fatigue design.

First, some industries have set standards for the certainty used in their designs. An example of this is the aerospace industry, where all designs must have a 99.99% certainty. However, very few industries have such standards.

Second, an engineer may look towards a similar problem faced with designing around fatigue life. The fatigue life of metals is also stochastic, with some metals having much more variance than others. While there are no industry-specific standards for this, there are generally accepted guidelines and a great deal of engineering intuition built up over the decades and centuries of experience. A commonly accepted value is 95% certainty for non-critical parts and 99% certainty for critical components.

Therefore, the most likely values that we would report to an engineer inquiring on the strength of PLA are the values for which we are 95% and 99% certain, which are 2.09 kN and 1.86 kN, respectively.

Limitations & Future Work

The largest limitation to this problem is the amount of data collected. We had 21 total samples, split between two groups such that the smaller had only 9 samples. This is enough data to give us a general idea of the strength of each batch, but having more initial data would provide us with more precise estimates of the underlying parameters. In addition, we could have data in many more groups. This would allow us to test if there is a relationship between the strength and the batch using a hierarchical model to incorporate the batch effect.

A final limitation worth noting, and thus a potential improvement, would be to gather data on strength through more varied methods. Tensile tests are the go-to standard, but 3-point bending and compressive testing would also yield informative values for the strength of the PLA. The varied methods of testing data could offset any noise due to link design and printer error.

Conclusion

This project demonstrates that FDM-formed PLA parts suffer from a distribution of strengths that varies from batch to batch and provides a guide on how to address this with a Bayesian model. We determined that each batch of PLA has a unique distribution of strengths. Using the strongest batch, we calculated useful strength values at the 90%, 95%, 99%, and 99.99% certainty levels. These values were 2.19 kN, 2.09 kN, 1.86 kN, and 1.38 kN, respectively.

Appendix – Code

```r
# to find a, b hyperparameters for the prior distribution on alpha
f <- function(params) {
  alpha <- params[1]
  beta <- params[2]

(pweibull(3, shape = alpha, scale = beta) - pweibull(1.62, shape = alpha, scale
  = beta) - .99)^2
   + (pweibull(1.62, shape = alpha, scale = beta) - .005)^2

}

# optim(c(1,1),f)
# optim(c(2.740281,11.194368),f)
optim(c(2.720831,11.345502),f)[1]

$par
[1]  2.720831 11.345502
```

Stan Model and Analysis

Link Model under the Null Hypothesis

```stan
data {
  int<lower = 0> N;
  real<lower = 0> y[N];
}

parameters {
  real alpha;
  real beta;
}

model {
  alpha ~ weibull(2.720831,11.345502);
  beta ~ gamma(.001, 1);

  for (i in 1:N) {
    y[i] ~ weibull(alpha, beta);
  }
}

generated quantities {
  real linkStrength;

  linkStrength = weibull_rng(alpha, beta);
}
```

```
saveRDS(LinkMod0, "LinkMod0.rds")

LinkMod0 <- readRDS('LinkMod0.rds')

dataList.LinkMod0 <- list(
  N = nrow(tensileData),
  y = tensileData$Strength.kN
)

fit.LinkMod0 <- stan(model_code = LinkMod0@model_code,
                  data = dataList.LinkMod0,
                  chains = 5, iter = 6000, warmup = 2000,
                  control = list(adapt_delta = 0.98))



fit.LinkMod0
LinkMod0.df <- stan_to_df(fit.LinkMod0)

# save df for future reference
write.csv(LinkMod0.df,
"C:/Users/genericStudent/OneDrive - genericCollege/genericFolderSpace /LinkMod0.csv",
        row.names=TRUE)
```

```
Inference for Stan model: anon_model.
5 chains, each with iter=6000; warmup=2000; thin=1;
post-warmup draws per chain=4000, total post-warmup draws=20000.

              mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
alpha        12.43    0.02 1.97  8.77 11.06 12.37 13.70 16.53  8465    1
beta          2.40    0.00 0.05  2.31  2.37  2.40  2.43  2.49  8609    1
linkStrength  2.30    0.00 0.24  1.75  2.16  2.32  2.46  2.68 19541    1
lp__          2.51    0.01 1.02 -0.16  2.12  2.82  3.24  3.50  6453    1

Samples were drawn using NUTS(diag_e) at Wed Feb 14 16:37:18 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

Summary

Link Model under the Alternate Hypothesis

```
data {
  int<lower = 0> N;
  int<lower = 1, upper = 2> color[N]; // 1 is white, 2 is gray
  real<lower = 0> y[N];
}

parameters {
  real alpha[2];
  real beta[2];
}
```

```
model {
  alpha ~ weibull(2.720831,11.345502);
  beta ~ gamma(.001, 1);

  for (i in 1:N) {
    y[i] ~ weibull(alpha[color[i]], beta[color[i]]);
  }
}

generated quantities {
  real grayLinkStrength;
  real whiteLinkStrength;

  whiteLinkStrength = weibull_rng(alpha[1], beta[1]);
  grayLinkStrength = weibull_rng(alpha[2], beta[2]);
}

saveRDS(LinkMod1, "LinkMod1.rds")

LinkMod1 <- readRDS('LinkMod1.rds')

dataList.LinkMod1 <- list(
  N = nrow(tensileData),
  color = tensileData$ColorCode,
  y = tensileData$Strength.kN
)

fit.LinkMod1 <- stan(model_code = LinkMod1@model_code,
                  data = dataList.LinkMod1,
                  chains = 5, iter = 6000, warmup = 2000,
                  control = list(adapt_delta = 0.98))



fit.LinkMod1
LinkMod1.df <- stan_to_df(fit.LinkMod1)

# save df for future reference
write.csv(LinkMod1.df,
"C:/Users/genericStudent/OneDrive - genericCollege/genericFileLocation/LinkMod1.csv",
          row.names=TRUE)
```

```
Inference for Stan model: anon_model.
5 chains, each with iter=6000; warmup=2000; thin=1;
post-warmup draws per chain=4000, total post-warmup draws=20000.

                   mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
alpha[1]          13.08    0.02 2.40  8.62 11.41 13.01 14.67 17.98 13602    1
alpha[2]          17.01    0.03 3.29 10.72 14.74 16.98 19.24 23.62 13163    1
beta[1]            2.22    0.00 0.05  2.11  2.18  2.22  2.25  2.33 11170    1
beta[2]            2.52    0.00 0.05  2.43  2.49  2.52  2.56  2.64 10810    1
grayLinkStrength   2.44    0.00 0.20  1.99  2.34  2.47  2.58  2.76 18143    1
whiteLinkStrength  2.13    0.00 0.21  1.63  2.01  2.15  2.27  2.48 18676    1
lp__               9.17    0.02 1.46  5.47  8.47  9.50 10.23 10.98  7288    1

Samples were drawn using NUTS(diag_e) at Wed Feb 14 16:46:03 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

Summary

Model Comparison

```r
evidence0 <- bridge_sampler(fit.LinkMod0)$logml
evidence1 <- bridge_sampler(fit.LinkMod1)$logml

logsumexp <- function(x1, x2) {
  x1 + log(1 + exp(x2 - x1))
}

lpr0 <- evidence0 - logsumexp(evidence0, evidence1)
lpr1 <- evidence1 - logsumexp(evidence0, evidence1)

c(exp(lpr0), exp(lpr1))
```

Output: [1] 0.0003159277 0.9996840723

```r
lowerBoundWhite <- quantile(LinkMod1$whiteLinkStrength, probs = c(0.0001, 0.01,
 0.05, 0.10, 0.5))
lowerBoundGray <- quantile(LinkMod1$grayLinkStrength, probs = c(0.0001, 0.01,
 0.05, 0.10, 0.5))

lowerBoundWhite

    0.01%        1%        5%       10%       50%
0.9512979 1.5029528 1.7387802 1.8461761 2.1520872

lowerBoundGray

   0.01%       1%       5%      10%      50%
1.384119 1.864236 2.088411 2.192314 2.468109
```