Lambert$W \times F_X$ transforms in Stan

In this blog post, I discuss and show how to use Lambert$W \times F_X$ [2] transforms for modeling skewed and asymmetric heavy tailed data. One advantage of using Lambert$W \times F_X$ transforms over skewed and heavy tailed distributions is that we can continue to use methods that are based on simpler distributions while still accounting for skew and kurtosis. For example, the distributions for individuals in a group may appear mostly normal while some appear to be have heavy tails. Instead of using a student-t for all groups they could all be modeled using a Lambert$W$ $timesNormal$ with a kurtosis parameters for each.

Over the past two months as Stan's Google Summer of Code (GSOC) intern I've studied Lambert$W \times F_X$ random variables [2]. My goal is to implement them as distributions in Stan to benefit workflows where the underlying distribution are distorted with excess-skew and heavy-tails. As a first step, we wrote Stan functions that carry out the Lambert$W \times F_X$ transform. We used these Stan-language functions to verify parameter recovery and to prototype skewed and heavy-tailed model workflows.

This blog-post illustrates both aspects: it highlights the parameter recovery of Lambert$W \times F_X$ random variables and also demonstrates how to add them to your workflow. At the end of our summer project, we aim to provide these transforms as Stan built-in functions.

Contents

1 Lambert$W$ transforms reintroduced

There are three-types of Lambert$W$ transforms. Start with a random variable X (taken to be Gaussian in this blog post). Center and rescale it, to obtain $U = \frac{X-\mu}{\sigma}$. Then the three categories of Lambert$W$ transform are -

Skewed	$Y = U \exp\{\gamma U\}$	$\gamma \in \mathbb{R}$
Symmetric Heavy-Tailed	$Y = U \exp\{\frac{\delta}{2} U^2\}$	$\delta \geq 0$
Asymmetric Heavy-Tailed	$Y = \begin{cases} U \exp\{\frac{\delta_l}{2} U^2\} & U \leq 0 \\ U \exp\{\frac{\delta_r}{2} U^2\} & U > 0 \end{cases}$	$\delta_l, \delta_r \geq 0$

Each random variable Y is Lambert$W \times F_X$ distributed [2], also known as Lambert$W \times \mathcal{N}(\mu, \sigma)$.

Where does the Lambert$W$ or product-logarithm function factor in? Just as the logarithm $\log(x)$ inverts the exponential $\exp(y)$, as $x = \exp(\log(x))$, the product-logarithm $W(z)$ inverts the product-exponential $u \exp(u)$, as $z = W(z) \exp(W(z))$. With this, we can express Y's density in terms of X's.

2 Simulation study

Lambert$W$ transforms can be used in a flexible variety of situations, which we highlight with a regression model. In this section, we fit the Asymmetric Heavy-Tailed transform to Normal-distributed data with either one-sided or two-sided heavy-tails.

For ease of exposition, we simulate data for a pair of variables (X,Y) where X is the feature and Y is the response. We define the true relationship between X and Y as

$$Y \sim \alpha + \beta X + \epsilon$$

In our setup $X \sim \mathcal{N}(\mu_X = 1, 1)$, $\epsilon \sim \mathcal{N}(0, \sigma = 3/2)$ and thus $\mathbb{E}[Y] = \alpha + \beta X$.

- For the data-generating process, we have unknown parameters $(\alpha, \beta, \sigma) = (1, 1, 3/2)$ for intercept, slope and error standard-deviation respectively.
- And, for the Asymmetric Heavy-Tailed Lambert$W \times F_X$ transform, we have two unknown parameters (δ_l, δ_r), whose values depend on the two cases we consider
    1. Normal-distributed data with one-sided heavy-tails $(\delta_l, \delta_r) = (0, 1/3)$
    2. Normal-distributed data with two-sided heavy-tails $(\delta_l, \delta_r) = (2/3, 1/3)$

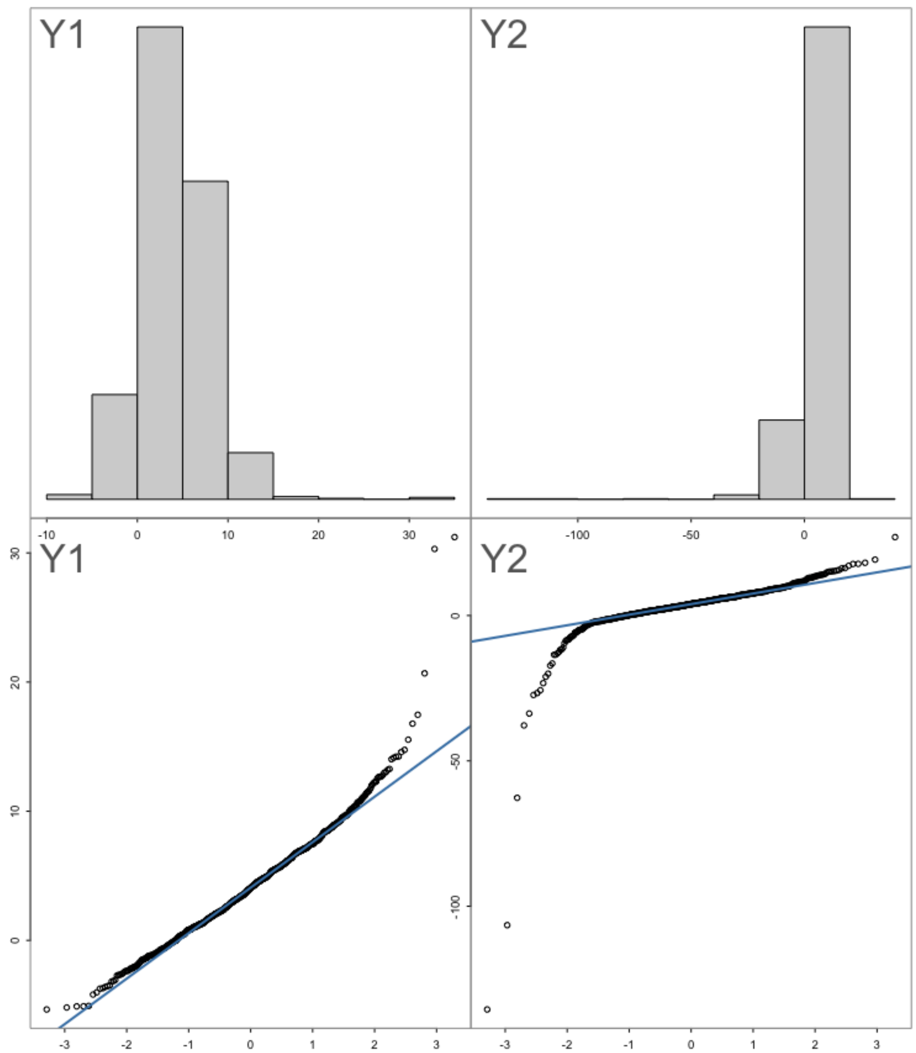
To summarize, we want to recover $(\alpha, \beta, \sigma, \delta_l, \delta_r)$ with data generated as follows.

```
epsilon <- function(N, sigma, delta_left, delta_right) {
    u <- rnorm(N)
    ifelse(u <= 0, u*exp(delta_left/2*u^2)*sigma,
                   u*exp(delta_right/2*u^2)*sigma)
}

N <- 1000; mu_x <- 1; alpha <- 1; beta <- 3; sigma <- 3/2

x <- rnorm(N, mu_x, 1)

y1 <- alpha + beta*x + epsilon(N, sigma, 0, 1/3)
y2 <- alpha + beta*x + epsilon(N, sigma, 2/3, 1/3)
```



(Histogram) Y1 The mean $\mathbb{E}[y_i] = 1 + 3[X] = 4$ with excess right kurtosis with Y2 showing asymmetric kurtosis. (QQ plot) Only the left tail of Y1 is normal, but the right tail is heavier. Y2 shows much greater kurtosis on the left tail vs the right tail.

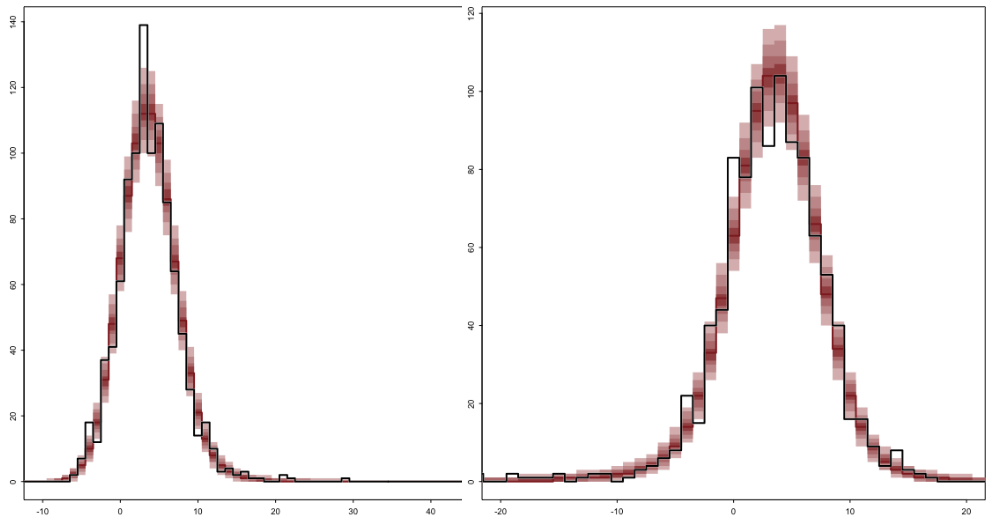
Given a sample from the data generating process (X, Y), we now show that we can recover the unknown parameters $(\alpha, \beta, \sigma, \delta_l, \delta_r)$ using Asymmetric Heavy-Tailed Lambert$W \times \mathcal{N}(\mu = \alpha + \beta X, \sigma = 3/2)$ transform.

y_1	var(true val)	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
1	$\alpha(1)$	0.97	0.97	0.07	0.07	0.86	1.08	1.00	2435.13	2623.44
2	$\beta(3)$	2.97	2.97	0.05	0.05	2.89	3.04	1.00	2535.51	2711.36
3	$\sigma(1.5)$	1.40	1.40	0.04	0.04	1.33	1.47	1.00	3238.89	2345.71
4	$\delta_l(0)$	0.01	0.01	0.01	0.01	0.00	0.04	1.00	2566.76	1873.71
5	$\delta_r(.3)$	0.32	0.32	0.04	0.04	0.26	0.39	1.00	3455.14	2586.87

y_2	var(true val)	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
1	$\alpha(1)$	1.03	1.03	0.08	0.08	0.90	1.16	1.00	3303.34	2885.34
2	$\beta(3)$	2.97	2.97	0.06	0.06	2.88	3.07	1.00	3032.88	2892.40
3	$\sigma(1.5)$	1.49	1.49	0.06	0.06	1.39	1.60	1.00	3804.82	3063.84
4	$\delta_l(.6)$	0.62	0.61	0.06	0.06	0.52	0.73	1.00	4435.63	2863.12
5	$\delta_r(.3)$	0.34	0.34	0.04	0.04	0.27	0.42	1.00	4114.28	3162.88

We can simulate samples from the posterior (Stan code below) and run a posterior retrodictive check [1]. By comparing quantiles of generated posterior samples against original data, we can be sure that the model fits the data in distribution.

```
generated quantities {
    real x_new = normal_rng(1, 1);
    real mu_new = alpha + beta * x_new;
    real u = normal_rng(0, 1);
    real delta = u <= 0 ? delta_left : delta_right;
    real y_post_pred = u * exp(delta/2 * square(u))*sigma + mu_new;
}
```



Quantized generated samples (red) of y_1 (left) and y_2 (right) vs. actual data (black). Since there are no systematic deviations (red vs. black), particularly in the tails, the data and model are consistent with each other.

In the next section, we derive the Lambert$W \times \mathcal{N}(\mu, \sigma)$ likelihood, which illustrates that the Normal seed in the Lambert$W$ transform can be substituted by another distribution with continuous support. For example, with this theory, we can readily define Lambert$W \times \text{Exp}(\lambda)$ random variables.

3 How to go the LambertW way

As described in the papers by Georg M. Goerg [2, 3], start with a random variable X that has density F_X with finite mean and variance, center and rescale it as $U = \frac{X-\mu}{\sigma}$. Then, if we transform U by $U \exp(\frac{\delta}{2}U^2)$, the corresponding random variable has density involving a derivative of the product logarithm.

Take $X \sim \mathcal{N}(\mu, \sigma)$ and $\delta \geq 0$. Let $Y \sim U \exp\left(\frac{\delta}{2}U^2\right)$ be the random variable of interest. Define $\beta = (\mu, \sigma)$, $z = \frac{y-\mu}{\sigma}$ and $W_\delta(z) = \text{sgn}(z)\sqrt{\frac{W(\delta z^2)}{\delta}}$ where $W_\delta(z)$ is the inverse of $U \exp(\frac{\delta}{2}U^2)$.

The theory states

$$
\begin{aligned}
g_Y(y \mid \beta, \delta) &= f_X(W_\delta(z)\sigma + \mu \mid \beta) \cdot \left|\frac{d}{dz}W_\delta(z)\right| \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{1}{2}W_\delta(z)^2\} \cdot \frac{W_\delta(z)}{z[1 + W(\delta z^2)]}
\end{aligned}
$$

For $Y_1, \ldots, Y_N$ i.i.d, the log-likelihood is

$$\prod_{i=1}^{N} g_y(y_i \mid \beta, \delta) = \left(-N \log \sigma - \frac{1}{2} z^2 e^{-W(\delta z^2)}\right) + \left(\frac{1}{2} \log W(\delta z^2) - \frac{1}{2} \log \delta - \log |z| - \log(1 + W(\delta z^2))\right)$$

where the first parenthetical is the likelihood of $\sigma W(\delta z^2) + \mu \sim f_X(\cdot \mid \beta)$ and the second parenthetical is a reformulation of the log-likelihood of $\left|\frac{d}{dz} W_\delta(z)\right|$. Since z is squared, we're dealing with the non-negative input into $W(z)$ and can always use the principal branch of the LambertW function. The finite mean and variance restriction can be lifted, but is not shown here.

To disambiguate the different LambertW transforms, we call the above transform the LambertW Normal h transform, it models Gaussian RVs whose tails are symmetrically heavier. A natural extension is what we call the LambertW Normal hh transform, Gaussian RVs whose tails are asymmetrically (left or right of the mean) heavier. Again, start with $X \sim \mathcal{N}(\mu, \sigma)$, with U as before, then for parameter $\delta_l, \delta_r \geq 0$ define

$$Y = \begin{cases} U \exp\left(\frac{\delta_l}{2} U^2\right) & U \leq 0 \\ U \exp\left(\frac{\delta_r}{2} U^2\right) & U > 0 \end{cases}$$

The likelihood function remains the same, except it now branches on $delta_l$ and $delta_r$ based on the input value U. Since Lambert$W \times F_X$ transforms are continuous-bijective and map 0 to 0, choosing the correct parameter δ_l or δ_r is unambigious, regardless of whether we're given values of X or values of Y.

In Stan, the math translates directly to likelihood increments

```
z = (y[i] - mu[i]) / sigma;
if (z <= 0) {
    w_delta_z_sq = lambert_w0(delta_left * square(z));
    sigma * sqrt(w_delta_z_sq / delta_left) ~ normal(0, sigma);
    target += (0.5 * log(w_delta_z_sq) - 0.5 * log(delta_left)
                - log(fabs(z)) - log1p(w_delta_z_sq));
} else {
    w_delta_z_sq = lambert_w0(delta_right * square(z));
    sigma * sqrt(w_delta_z_sq / delta_right) ~ normal(0, sigma);
    target += (0.5 * log(w_delta_z_sq) - 0.5 * log(delta_right)
                - log(fabs(z)) - log1p(w_delta_z_sq));
}
```

Conclusion

In a real-world regression setting, we seldom know the true distribution of the data-generating process. But, if we can identify the relationship as approximately Normal except for the tails, then we can apply LambertW x Normal hh transforms to model the feature-response relationship. The main advantage of the transformation approach over directly-fitting a heavy-tailed probability distribution is the ability to back-transform heavy-tailed observations into a Gaussian kernel. Being able to separate the normal and non-normal components helps you interpret the inference and resolve computational frustrations.

The flexibility of LambertW transforms does not stop here. If you have approximately Exponential data except for a heavy-right tail, you can apply a LambertW x Exponential h transform. If you have approximate-but-skewed Normal data, then you can apply a LambertW x Exponential s transform. Neither of these are described here, but will be part of the inbuilt functionality we plan to deliver as part of this GSOC project.

Full Stan Code For Lambert$W \times Normal$ hh

```
data {
  int N;
  vector[N] y;
  vector[N] x;
}
parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
  real<lower=0> delta_left;
  real<lower=0> delta_right;
}
transformed parameters {
  vector[N] mu = alpha + beta * x;
}
model {
  alpha ~ normal(0.5, 1);
  beta ~ normal(3, 1);
  sigma ~ normal(0, 1.5*sqrt(pi()/2));
  delta_left ~ exponential(3);
  delta_right ~ exponential(3);

  real z, w_delta_z_sq;
  for (i in 1:N) {
    z = (y[i] - mu[i]) / sigma;
    if (z <= 0) {
      w_delta_z_sq = lambert_w0(delta_left * square(z));
      sigma * sqrt(w_delta_z_sq / delta_left) ~ normal(0, sigma);
      target += 0.5 * log(w_delta_z_sq) - 0.5 * log(delta_left)
       - log(fabs(z)) - log1p(w_delta_z_sq);
    } else {
      w_delta_z_sq = lambert_w0(delta_right * square(z));
      sigma * sqrt(w_delta_z_sq / delta_right) ~ normal(0, sigma);
      target += 0.5 * log(w_delta_z_sq) - 0.5 * log(delta_right)
       - log(fabs(z)) - log1p(w_delta_z_sq);
    }
  }
}
generated quantities {
  vector[N] y_new;
  for (i in 1:N)
  {
    real x_new = normal_rng(1, 1);
    real mu_new = alpha + beta * x_new;
    real u = normal_rng(0, 1);
    real delta = u <= 0 ? delta_left : delta_right;
    y_new[i] = u * exp(delta/2 * square(u))*sigma + mu_new;
  }
}
```

References

[1] Michael Betancourt. Towards a principled bayesian workflow, 2020.

[2] Georg M Goerg. Lambert w random variables—a new family of generalized skewed distributions with applications to risk estimation. *The Annals of Applied Statistics*, 5(3):2197–2230, 2011.

[3] Georg M Goerg. The lambert way to gaussianize heavy-tailed data with the inverse of tukey'sh transformation as a special case. *The Scientific World Journal*, 2015, 2015.