

debris__flows__prop

Erik Rose

2020-02-22

Proposal to Estimate Sediment Flux in Headwater Creeks of the OCR using Charcoal Ages From Stream Bank Deposits

Summary

Insert summary here.

Project Description

The Project Goal

The goal of this project is to infer from the ages of charcoal stored in the sediment of stream banks something about how long sediments remain in stream bank deposits, and how much sediment is moving through the valley over time. From the research of Stephen Lancaster and colleagues we have a record of 370 charcoal samples along several creeks in the Oregon Coast Range (OCR), including lab estimates of the age distribution curve of each sample using radiocarbon dating (Frueh and Lancaster, 2014). The Lancaster dataset includes information from progressive surveys of these same few creeks, including elevation transects along the longitudinal profile of the stream. The field researchers recorded elevation transects perpendicular to the direction of streamflow at inflection points along the valley contour, recording the distance from outlet on a hipchain where the transect crosses the stream. This permits us to interpolate valley width, slope, and the volume of deposits along the length of the creek with greater accuracy than LiDAR-derived estimates.

Knowing something about the volume of stream deposits, and how long they persist in the valley, we should be able to say something about how much sediment is moving through the system. However, several challenges make the process of integrating these successive surveys less than straightforward. The locations of charcoal samples in the stream bank occur at randomly-selected points along the length of the stream, in order to represent the distribution of deposit ages currently evacuating into the stream. The locations of transects correspond to inflection points in valley contour, which is to say they occur at different places than the charcoal samples. Charcoal samples classify deposits as debris-flow deposits, fluvial fines or gravels, but they occur at random positions in the transect record, and while abundant, the sampling is not sufficient to infer the deposit composition of the majority of volumes on the valley floor delineated in the transect survey. We know from the distribution of charcoal ages that deposit types evacuate at different rates into the stream, and we know the volume of deposits along the stream, but we do not know the deposit type of most of the volumes.

Imagine the valley floor along the stream is a bucket that fills with dirt from the hillslopes and empties out of a hole in the bucket representing the outlet of the stream. The purpose for randomly selecting the location of charcoal samples was to represent the types of deposits currently evacuating into the stream, leaving out the hole in the bucket. If we assume the amount of sediment in the valley is currently in a steady state, then in the long term the mean amount of sediment going into the bucket is the same as the mean amount going out of the bucket. However, characterizing the mean amount going into the bucket is not a useful way to describe the system, which is punctuated by debris-flow deposits. Even if we could infer from bank stratigraphy the composition of deposits along the valley floor, and estimate from the volume of deposits the mean rate of flux in the valley, the mean rate of debris-flow inputs would not be useful for predicting activity on a given year, because the timing and volume of landslides is heterogenous and not likely to be the mean on any given year.

In order to characterize the dirt coming into the bucket, a CDF of debris-flow delivery probability over time would be preferable to an expected global mean.

Relating Modeled Delivery Probabilities to Observed Deposits

If the Miller-Burnett delivery probabilities are good at predicting debris-flow deposit locations, then we would expect to be more likely to observe deposits in places where delivery probability is high, and less likely to observe deposits where delivery probability is low. However, the record of valley deposits includes fluvial deposition, and we cannot easily distinguish debris-flow deposits from fluvial deposition. Low delivery probabilities may correspond to large cross-sectional areas in places where the valley accumulates substantial fluvial deposits.

Some portions of the stream are more likely to accumulate deposits over time, whereas others are more likely to experience scour. In a system characterized by riffles and pools, deposits are subject to more scouring force along the riffles than in the pools. Even with perfect knowledge of where debris-flow are depositing, we would not expect to see deposits accumulating where they are likely to be scoured out. We could observe substantial deposition in an area with low delivery probability if the potential for scour there is poor. The capacity of the stream to scour deposited material relates to slope and contributing area. As contributing area increases, the size of the stream increases and there is more water available to move debris, whereas the steeper the slope, the greater the force of gravity acting on the water, and the more material can move.

The amount of deposition accumulating in the valley is not subject only to how much volume is coming in from upstream and evacuating downstream. Valley topography dictates where deposits can accumulate. As the valley widens, more material can accumulate in the valley. Fluvial deposits can occur along the length of the valley floor, distributed by flooding events of the kind that may result from a debris-flow deposit jamming flow at the downstream neck of the valley. A wall-to-wall flood will deposit more material along an empty wide valley than an empty narrow valley, because the flood spreads across a greater area and more debris is able to settle out of the water.

When we relate the likelihood of observing a deposit to valley width, stream slope or contributing area, we are assuming the stream area falls within a range of attributes that characterize headwater streams of the Tyee formation. Larger contributing area increases debris-flow delivery likelihood when contributing area is less than 4 km^2 , because there are more potential source areas for debris-flow deliveries. But as contributing area increases beyond 4 km^2 , slopes decrease and valleys widen, changing the underlying dynamics, and then debris-flow delivery likelihood decreases as contributing area grows larger.

The relation between slope and contributing area in headwater streams follows a power law, with slope decreasing as contributing area increases. The relation breaks down on steep slopes, where ridged valleys initiate in bedrock chutes running up steep mountain slopes, and at steep transition points, such as a falls. Angle of repose plays a greater role in characterizing deposition as slope increases, and here we constrain our analysis to portions of the study area where slope is less than 12%. To constrain our analysis to portions of the study area where the valley walls are narrow enough to act as a bucket, and not so wide as to behave as an alluvial plain, we limit our sample to portions of the study area where the valley walls are less than 49 meters apart.

```
library(magrittr)
library(sp)
library(muddier)

# here i scrub NAs and bad values that need to be checked / redone
sub <- creeks[!is.na(creeks$slope) & !(creeks$NODE_ID %in% cedar$NODE_ID[105:156]),]

# subset by attribute ranges
sub <- sub[sub$slope < .2 &
          sub$contr_area < 6 &
          sub$slope > 0 &
```

```

sub$valley_width < 49 &
sub$valley_width > 0, ]

names(sub)
#> [1] "OBJECTID"      "NODE_ID"      "ToMouth_km"   "GRADIENT"
#> [5] "VALWIDTH_5"    "NodeNum"      "DebrisFlow"   "xsec_area"
#> [9] "contr_area"    "valley_width" "slope"        "creek_name"
nrow(sub)
#> [1] 714

```

While we expect that increases in delivery probability correspond to increases in deposit volume, we would not necessarily expect increases in deposit volume to correspond to increases in delivery probability, because not all deposits come from debris flows, and the stream scours some debris-flow deposits away before we can observe them. Indeed, delivery probabilities do not correspond closely to observed volumes.

```

dp <- sub$DebrisFlow # debris-flow delivery
xs <- sub$xsec_area  # cross-sectional area m2

setwd('/home/crumplecup/work')
png('cor_dp_xs.png',
    width = 7, height = 5, units = 'in', res = 300)
cor_creeks(xs, dp,
    creeks = sub$creek_name,
    creek_labs = levels(factor(sub$creek_name)),
    labs = c('cross-sectional area km2', 'delivery probability'),
    cols = c('rose', 'forest', 'ocean', 'violet'),
    leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

```

Fit to a linear model, the $R^2 = 0.00$, meaning delivery probability alone has poor explanatory power for cross-sectional area. Although the explanatory effect of delivery probability is statistically significant, a change across the full range of p values accounts for just over a 10 m^2 increase in cross-sectional area. This is not a meaningful increase in effect size when cross-sectional area can exceed 150 m^2 .

The Lancaster surveys include elevation transects perpendicular to the direction of stream flow. The field researchers chose transect positions that would reflect changes in valley topography, such that if you connected the dots between the lines of the valley walls in the transects, the result would be a reasonable representation of the valley contour.

The Miller-Burnett model represents the stream path as a series of nodes spaced 10-14 meters apart. We can interpolate the valley width and slope at stream nodes using the Lancaster survey, and the values will correspond to observed cross-sectional area and contributing area more closely than the LiDAR-derived variables from the Miller-Burnett model.

Delivery probability does not correlate with either valley width, slope or contributing area. Delivery probability does spike directly downstream of tributary junctions, and these spikes are evident on the scatterplot of delivery probability and contributing area below.

```

vw <- sub$valley_width # valley width
ca <- sub$contr_area   # contributing area
sl <- sub$slope        # slope

setwd('/home/crumplecup/work')
png('cor_dp_vw.png',

```

```

width = 7, height = 5, units = 'in', res = 300)
cor_creeks(vw, dp,
  creeks = sub$creek_name,
  creek_labs = levels(factor(sub$creek_name)),
  labs = c('valley width m', 'delivery probability'),
  cols = c('rose', 'forest', 'ocean', 'violet'),
  leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_dp_ca.png',
  width = 7, height = 5, units = 'in', res = 300)
cor_creeks(ca, dp,
  creeks = sub$creek_name,
  creek_labs = levels(factor(sub$creek_name)),
  labs = c('contributing area km2', 'delivery probability'),
  cols = c('rose', 'forest', 'ocean', 'violet'),
  leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_dp_sl.png',
  width = 7, height = 5, units = 'in', res = 300)
cor_creeks(sl, dp,
  creeks = sub$creek_name,
  creek_labs = levels(factor(sub$creek_name)),
  labs = c('slope', 'delivery probability'),
  cols = c('rose', 'forest', 'ocean', 'violet'),
  leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

```

Developing the idea that deposit volumes are not correlating with delivery probability because of mitigating factors, we can explore the relation between cross-sectional area and contributing area, slope and valley width in the plots below.

There is a tendency for cross-sectional area to increase as contributing area increases, but the overall fit is low, with an R^2 of 0.18. Cross-sectional area is not significantly related to slope. There is a close association between cross-sectional area and valley width, supporting the assertion that valley topography is a controlling factor of valley deposit volumes. Fit to a linear model, valley width explains 67% of the variation in cross-sectional area, with every meter increase of valley width associated with a 2.7 m^2 increase in cross-sectional area.

```

setwd('/home/crumplecup/work')
png('cor_xs_ca.png',
  width = 7, height = 5, units = 'in', res = 300)
cor_creeks(ca, xs,
  creeks = sub$creek_name,
  creek_labs = levels(factor(sub$creek_name)),
  labs = c('contributing area km2',

```

```

        'cross-sectional area m2'),
cols = c('rose', 'forest', 'ocean', 'violet'),
leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_xs_sl.png',
    width = 7, height = 5, units = 'in', res = 300)
cor_creeks(sl, xs,
    creeks = sub$creek_name,
    creek_labs = levels(factor(sub$creek_name)),
    labs = c('slope', 'cross-sectional area m2'),
    cols = c('rose', 'forest', 'ocean', 'violet'),
    leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_xs_vw.png',
    width = 7, height = 5, units = 'in', res = 300)
cor_creeks(vw, xs,
    creeks = sub$creek_name,
    creek_labs = levels(factor(sub$creek_name)),
    labs = c('valley width', 'cross-sectional area m2'),
    cols = c('rose', 'forest', 'ocean', 'violet'),
    leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

```

While stream gradient may decrease generally as contributing area increases, local variations produce areas of streamflow with higher or lower streampower than one might expect based upon position in the reach. We can characterize the expected streampower for a reach by relating contributing area A to slope S using a power law:

$$S = k_s A^{-\theta}$$

where θ represents the rate at which slope decreases as contributing area increases. We can relate logged slope to logged contributing area using a linear model, so that:

$$\log(S) = \log(k_s) - \theta \log(A)$$

where θ is the slope of a line with intercept $\log(k_s)$. Using the estimated θ for all surveyed creeks, we can back-calculate an individual k_s for a position in the channel by holding θ constant. The streampower coefficient k_s represents the degree to which local streampower is stronger or weaker than average streampower in the system.

Where streampower is stronger, we would expect deposits to have shorter residence times than where it is weaker. Exploring the possibility that streampower might explain some of the variation in deposit volumes, we examine the different relations of cross-sectional area to delivery probability and streampower, including the difference of delivery probability and streampower and their ratio.

Streampower coefficients are not associated with cross-sectional areas. Whether examined as a linear model, or as an accumulation difference or ratio, the R^2 of the fit is near zero.

```
lslope <- (sub$lslope + .026) %>% log # log slope with offset for negative grads
lcontr <- sub$lcontr_area %>% log # log contr area
as_mod <- lm(lslope ~ lcontr) # fit AS to power law
as_theta <- as_mod$coefficients[2] # pull theta value

# estimate individual ks values using pulled theta
log_ks <- lslope - as_theta * lcontr
ks <- exp(log_ks)

# accumulation index options
ndp <- dp / max(dp) # normalized p
nks <- ks / max(ks) # normalized k_s
acc_dif <- ndp - nks # p - k_s accumulation difference
acc_rat <- ndp / nks # p / k_s accumulation ratio

setwd('/home/crumplecup/work')
png('cor_xs_ks.png',
     width = 7, height = 5, units = 'in', res = 300)
cor_creeks(ks, xs,
            creeks = sub$creek_name,
            creek_labs = levels(factor(sub$creek_name)),
            labs = c('streampower coefficient',
                     'cross-sectional area m2'),
            cols = c('rose', 'forest', 'ocean', 'violet'),
            leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_xs_acc_dif.png',
     width = 7, height = 5, units = 'in', res = 300)
cor_creeks(acc_dif, xs,
            creeks = sub$creek_name,
            creek_labs = levels(factor(sub$creek_name)),
            labs = c('p - k_s',
                     'cross-sectional area m2'),
            cols = c('rose', 'forest', 'ocean', 'violet'),
            leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

setwd('/home/crumplecup/work')
png('cor_xs_acc_rat.png',
     width = 7, height = 5, units = 'in', res = 300)
cor_creeks(acc_rat, xs,
            creeks = sub$creek_name,
            creek_labs = levels(factor(sub$creek_name)),
            labs = c('p / k_s',
```

```

        'cross-sectional area m2'),
cols = c('rose', 'forest', 'ocean', 'violet'),
leg_pos = 'topright', a = .25)
dev.off()
#> pdf
#> 2

# gather variables into data.frame for linear modeling
dat <- data.frame(xsec_area = xs, # 1
                  log_xsec = log(xs), # 2
                  val_width = vw, # 3
                  log_val_width = log(vw), # 4
                  del_prob = dp, # 5
                  slope = sl, # 6
                  contr_area = ca, # 7
                  strm_pwr = ks, # 8
                  acc_dif = acc_dif, # 9
                  acc_rat = acc_rat) # 10

dat <- dat[xs > 0, ]

modlist <- list(dat[, c(1, 8)], dat[, c(1, 5, 8)],
               dat[, c(1, 9)], dat[, c(1, 10)])

(tab <- stat_tab(modlist))
#>
#> 1          forms          R2          AIC          BIC
#> 2 xsec_area ~ strm_pwr -0.0007750452 6539.468 6553.078
#> 3 xsec_area ~ del_prob + strm_pwr 0.0098725286 6533.084 6551.230
#> 4 xsec_area ~ acc_dif 0.0033266859 6536.634 6550.244
#> 5 xsec_area ~ acc_rat 0.0077074673 6533.594 6547.204

```

Valley width and contributing area have the strongest relation to cross-sectional areas, so let us examine some more complex linear models including valley width with contributing area, delivery probability and other variables. The following code snippet runs linear models of different variable combinations, and prints a table showing the formula for each model, the R^2 value, and the AIC and BIC values.

We consider a log-linear model of cross-sectional area and valley width, because both variables exhibit an exponential distribution upon visual inspection. This includes the notable exception that valley widths are unlikely to narrow beyond a minimum of 10 meters, as we are examining stream paths through the valley, so valley widths do not behave exponentially as they approach zero. The log-linear fit of cross-sectional area and valley width has an improved fit over the linear model, explaining 6% more variation in cross-sectional area.

For the linear fit of cross-sectional area to valley width, adding a variable to the model resulted in improved fit, with the exception of delivery probability. The greatest improvement of fit (3%) resulted from adding contributing area to the model, as we might expect based upon the greater degree of correlation between contributing area and cross-sectional area compared to slope, streampower coefficient, and the accumulation difference and ratio.

For the log-linear fit of cross-sectional area to valley width, adding a variable to the model resulted in improved fit, with the exceptions of the accumulation difference and ratio. The greatest improvement of fit (1.5%) resulted from adding contributing area to the model, as expected, followed by slope, streampower coefficient and delivery probability.

Note the AIC and BIC scores are lower, indicating less information loss, from the model using contributing area in the both the linear and log-linear model. Note also the R^2 value of the worst log-linear model exceeds

the best value of the linear models. Under both models, the effect of delivery probability on cross-sectional area is statistically significant, and in the expected positive direction, but the contribution of delivery probability to overall fit of the model is small.

```
modlist <- list(dat[ , c(1, 3)],
               dat[ , c(1, 3, 5)],
               dat[ , c(1, 3, 6)],
               dat[ , c(1, 3, 7)],
               dat[ , c(1, 3, 8)],
               dat[ , c(1, 3, 9)],
               dat[ , c(1, 3, 10)],
               dat[ , c(2, 4)],
               dat[ , c(2, 4, 5)],
               dat[ , c(2, 4, 6)],
               dat[ , c(2, 4, 7)],
               dat[ , c(2, 4, 8)],
               dat[ , c(2, 4, 9)],
               dat[ , c(2, 4, 10)])

(tab <- stat_tab(modlist))

#>
#> 1          xsec_area ~ val_width 0.6504556 5813.6571 5827.2672
#> 2      xsec_area ~ val_width + del_prob 0.6520017 5811.5947 5829.7414
#> 3      xsec_area ~ val_width + slope 0.6685461 5777.9856 5796.1323
#> 4      xsec_area ~ val_width + contr_area 0.7208356 5659.5209 5677.6677
#> 5      xsec_area ~ val_width + strm_pwr 0.6527280 5810.1532 5828.3000
#> 6      xsec_area ~ val_width + acc_dif 0.6499589 5815.6334 5833.7801
#> 7      xsec_area ~ val_width + acc_rat 0.6500580 5815.4379 5833.5847
#> 8          log_xsec ~ log_val_width 0.7520126 919.5248 933.1349
#> 9      log_xsec ~ log_val_width + del_prob 0.7566331 907.5437 925.6905
#> 10      log_xsec ~ log_val_width + slope 0.7633999 888.0866 906.2334
#> 11      log_xsec ~ log_val_width + contr_area 0.7736698 857.4670 875.6137
#> 12      log_xsec ~ log_val_width + strm_pwr 0.7543622 913.9525 932.0992
#> 13      log_xsec ~ log_val_width + acc_dif 0.7522051 919.9853 938.1321
#> 14      log_xsec ~ log_val_width + acc_rat 0.7525427 919.0445 937.1913
#tab[rev(order(tab$R2)),]
```

Building upon the log-linear model, and including contributing area, we can examine the effect of an additional variable on the fit of the model. We drop consideration of streampower coefficient, and the accumulation difference and ratio because they are partly derived from contributing area, a term already in the model. Delivery probability and slope both improve the fit of the model. Adding both delivery probability and slope to the model results in the highest R^2 value (0.734), and the lowest AIC and BIC values of all log-linear model we have examined, despite having the most independent terms. Contributing area remains an important term in the model, with fit decreasing when delivery probability and slope remain in the model but contributing area is removed.

Considering the model with the best fit, AIC and BIC values, delivery probability is statistically significant ($p = .007$) and the effect is in the expected positive direction, contributing area is statistically significant ($p = .028$) with an effect in the negative direction, which is unexpected, because increases in contributing area are weakly correlated with increases in cross-sectional area.

```
modlist <- list(dat[ , c(2, 4, 7)],
               dat[ , c(2, 4, 7, 5)],
               dat[ , c(2, 4, 7, 6)],
               dat[ , c(2, 4, 7, 5, 6)],
               dat[ , c(2, 4, 5, 6)])
```



```

(tab <- stat_tab(modlist))
#>
#> 1 log_xsec ~ log_val_width + contr_area 0.7736698
#> 2 log_xsec ~ log_val_width + contr_area + del_prob 0.7762642
#> 3 log_xsec ~ log_val_width + contr_area + slope 0.7761187
#> 4 log_xsec ~ log_val_width + contr_area + del_prob + slope 0.7793883
#> 5 log_xsec ~ log_val_width + del_prob + slope 0.7688338
#> AIC BIC
#> 1 857.4670 875.6137
#> 2 850.5066 873.1901
#> 3 850.9551 873.6386
#> 4 841.7976 869.0178
#> 5 873.0497 895.7332
mod_stat(dat[, c(2, 4, 7, 5, 6)])
#>
#> Call:
#> lm(formula = form, data = df)
#>
#> Residuals:
#> Min 1Q Median 3Q Max
#> -2.78780 -0.20517 0.01207 0.24325 1.57863
#>
#> Coefficients:
#> Estimate Std. Error t value Pr(>|t|)
#> (Intercept) -3.21902 0.16192 -19.880 < 2e-16 ***
#> log_val_width 2.07289 0.05355 38.707 < 2e-16 ***
#> contr_area -0.17974 0.03091 -5.815 9.28e-09 ***
#> del_prob 8.62293 2.58043 3.342 0.000878 ***
#> slope 1.85389 0.56637 3.273 0.001117 **
#> ---
#> Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 0.4431 on 685 degrees of freedom
#> Multiple R-squared: 0.7807, Adjusted R-squared: 0.7794
#> F-statistic: 609.5 on 4 and 685 DF, p-value: < 2.2e-16

```

The effect size of delivery probability in the model is not straightforward to interpret because we are relating the probabilities to logged cross-sectional areas. In the code snippet below, first we obtain predicted cross-sectional areas using observed data and MB delivery probabilities. Then we predict values using observed data, but changing the delivery probability to the minimum and maximum values in the dataset. In the plot below, the yellow line shows the predicted cross-sectional area from the normal data, with the minimum and maximum delivery probability versions above and below in blue. The spread of the blue lines around the yellow show the range of effect that variation in delivery probability can have upon cross-sectional area in the model.

The orange line on the plot represents the potential effect size of delivery probability on the model as the predictions from the maximum probability delivery model minus the predictions from the minimum. Potential effect size ranges from 0.9-35.9 m^2 , with 50% of observations falling between 5.3 and 15.8 m^2 , credible levels of deposition from debris-flow delivery. The gray line represents model error as the absolute values of observed minus predicted cross-sectional areas. Note that potential effect size exceeds model error in the majority of observations, suggesting the effect of delivery probability is large enough to consider meaningful in the model.

```

mod <- lm(log_xsec ~ log_val_width + contr_area + del_prob + slope, data = dat)

```

```

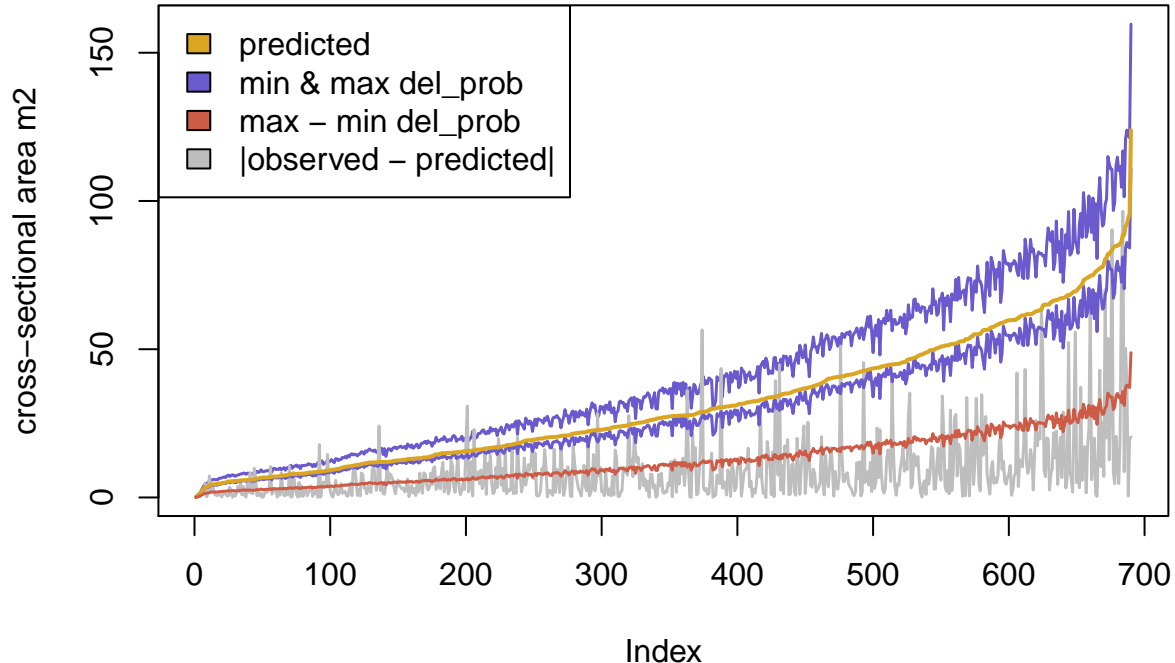
pred <- exp(predict(mod, dat))

min_dat <- dat
min_dat$del_prob <- min(dat$del_prob)
max_dat <- dat
max_dat$del_prob <- max(dat$del_prob)

min_pred <- exp(predict(mod, min_dat))
max_pred <- exp(predict(mod, max_dat))
difs <- max_pred - min_pred
quantile(difs)
#>      0%      25%      50%      75%     100%
#> 0.03788574 5.67443156 10.44410084 18.34175927 48.82941793

plot(max_pred[order(pred)],
     type = 'l', lwd = 1.5, col = 'slateblue',
     ylab = 'cross-sectional area m2')
lines(abs(dat$xsec_area[order(pred)] - pred[order(pred)]),
      lwd = 1.5, col = 'gray')
lines(min_pred[order(pred)], lwd = 1.5, col = 'slateblue')
lines(pred[order(pred)], col = 'goldenrod', lwd = 2)
lines(difs[order(pred)], lwd = 1.5, col = 'coral3')
legend('topleft',
      legend = c('predicted', 'min & max del_prob',
                  'max - min del_prob', '|observed - predicted|'),
      fill = c('goldenrod', 'slateblue', 'coral3', 'gray'))

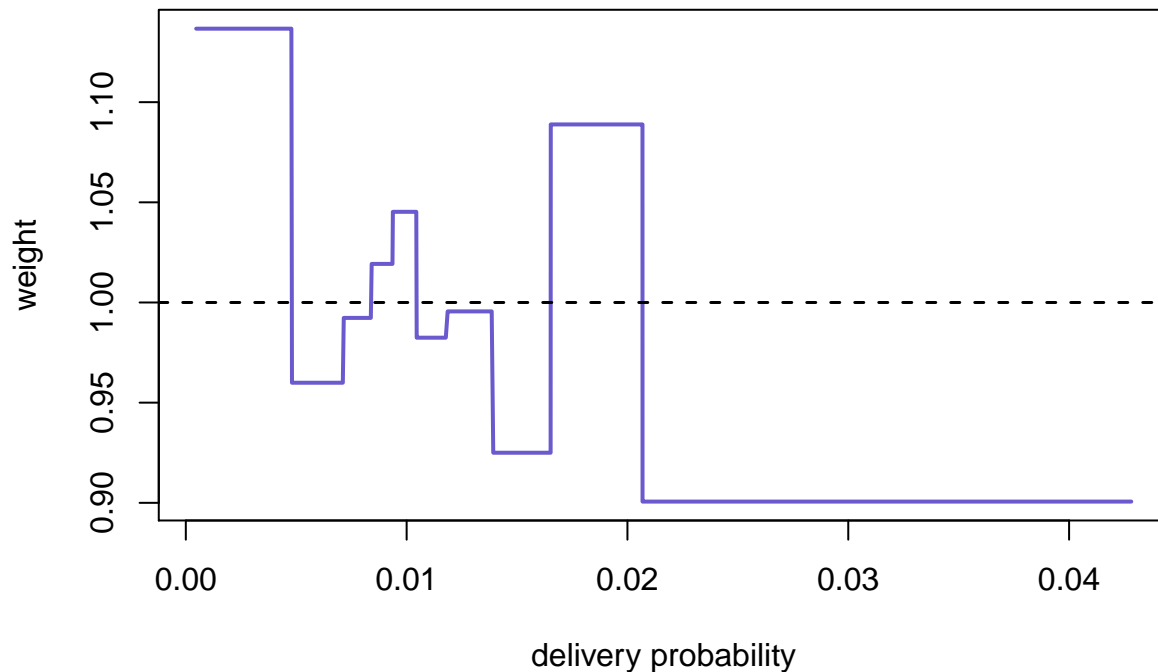
```



```

plot_weight(num = dat$xsec_area,
            den = pred,
            by = dat$del_prob,
            bin_no = 10,
            labs = c('delivery probability', 'weight'))

```



```

setwd('/home/crumplecup/work')
png('delivery_probs_adjusted.png',
    width = 7, height = 5, units = 'in', res = 300)
plot(ord$p_ks, ord$p0, pch = 20, log = 'x',
     col = rgb(red=.267, green=.671, blue=.471, alpha=.33),
     ylab = 'delivery probability', xlab = 'p / k_s',
     ylim = c(min(ord$p0 - ord$DebrisFlow/sum(ord$DebrisFlow)), max(ord$p0)))
points(ord$p_ks, ord$DebrisFlow/sum(ord$DebrisFlow),
       pch = 20, col = rgb(red=.306, green=.569, blue=.831, alpha=.33))
points(ord$p_ks, ord$p0 - ord$DebrisFlow/sum(ord$DebrisFlow),
       pch = 20, col = rgb(red=1, green=.592, blue=.478, alpha=.33))
legend('bottomright', legend = c('p', 'p0', 'p0-p'),
      fill = c(rgb(red=.306, green=.569, blue=.831, alpha=.33),
               rgb(red=.267, green=.671, blue=.471, alpha=.33),
               rgb(red=1, green=.592, blue=.478, alpha=.33)))
dev.off()

library(fitdistrplus)

#subset debris flow deposit ages
df <- as.matrix(char_pmfs)
df <- df[, charcoal$facies == 'DF']

#determine which sites have multiple samples
site <- charcoal$family[charcoal$facies == 'DF']
counts <- 0
for (i in seq_along(site)) {
  counts[i] <- length(site[site == site[i]])
}
mults <- site[counts > 1] %>% factor %>% levels

#select youngest age from each set of multiples

```

```

dfmn <- charcoal$mn[charcoal$facies == 'DF']
mins <- vector(length(site), mode = 'numeric')
for (i in seq_along(mults)) {
  mins[dfmn == min(dfmn[site == mults[i]])] <- 1
}

#screen out all but youngest from multiples
df <- df[ , counts == 1 | (counts > 1 & mins > 0)]

#order from youngest to oldest
dfmn <- dfmn[counts == 1 | (counts > 1 & mins > 0)]
df <- df[ ,order(dfmn)]
dfmns <- dfmn + 50

discrete_bin <- function(vals, bin_len) {
  n <- ceiling(max(vals))
  bins <- ceiling(n / bin_len)

  marks <- bin_len
  for (i in 2:(bins-1)) {
    marks[i] <- marks[i-1] + bin_len
  }
  marks <- c(marks, n)
  bin_vals <- list()
  for (i in 1:bins) {
    if (i == 1) {
      bin_vals[[i]] <- vals[vals <= marks[i]]
    }
    if (i > 1) {
      bin_vals[[i]] <- vals[vals > marks[i-1] &
                           vals <= marks[i]]
    }
  }
  return(bin_vals)
}

bin_mns <- discrete_bin(dfmn+50, 200)

fit_bins <- function(dat, ids, bins, index = as.numeric(rownames(char_pmfs))) {
  times <- list() # interarrival times
  fits <- list() # exponential fits
  coefs <- array(0, c(length(bins), 4)) # estimate, median and CIs
  for (i in 1:length(bins)) {
    mat <- dat[ , ids %in% bins[[i]]]
    n <- ncol(mat)
    mat_mns <- array(0, c(n-1, 1))
    mat_ar <- array(0, c(n-1, nrow(mat)))

    for(j in 1:(n-1)) {
      mat_ar[j,] <- convo(mat[,j+1], mat[,j], index) # convolved difference in age
      mat_mns[j] <- weighted.mean(sort(index), mat_ar[j,])
    }
  }
}

```

```

    }

    times[[i]] <- mat_mns
    fits[[i]] <- fitdist(as.numeric(mat_mns), 'exp')
    coefs[i,1] <- summary(fits[[i]])$estimate
    coefs[i,2:4] <- bootdist(fits[[i]])$CI
  }
  return(list(times, fits, coefs))
}

sm_bins <- list(bin_mns[[1]], bin_mns[[2]], bin_mns[[3]],
               bin_mns[[4]], bin_mns[[5]], bin_mns[[6]])
df_int <- fit_bins(df, dfmns, sm_bins)

df_est <- t(df_int[[3]])

xlabs <- 0
for (i in 1:length(sm_bins)) {
  xlabs[i] <- round(mean(sm_bins[[i]]))
}
xlab <- c('0-200', '201-400', '401-600', '601-800', '801-1000', '1001-1200')

setwd('/home/crumplecup/work')
png('interarrival_rates_df.png',
     width = 7, height = 5, units = 'in', res = 300)
plot(df_est[1,], type = 'l', lwd = 2, col = 'slateblue',
     xlab = 'age range of local sample window', ylab = 'interarrival rate',
     ylim = c(0, max(df_est[4,])), axes=F)
lines(df_est[2,], lwd = 2, col = 'forestgreen', lty = 2)
lines(df_est[3,], lwd = 2, col = 'coral3', lty = 2)
lines(df_est[4,], lwd = 2, col = 'coral3', lty = 2)
axis(1, at=1:length(xlab), labels=xlab)
axis(2)
legend('topleft', legend = c('Mean', 'Median', '95% CI'),
      fill = c('slateblue', 'forestgreen', 'coral3'))
dev.off()

# estimate interarrival times
index <- char_pmfs %>% rownames %>% as.numeric # age represented by row
df_n <- length(dfmns)
df_mns <- array(0, c(df_n-1, 1)) # mean interarrival time
df_ar <- array(0, c(df_n-1, nrow(char_pmfs))) # interarrival age pmf

for(i in 1:(df_n-1)) {
  df_ar[i,] <- convo(df[,i+1], df[,i], index) # convolved difference in age
  df_mns[i] <- weighted.mean(sort(index), df_ar[i,])
}

# fit exponential distribution to interarrival times
descdist(as.numeric(df_mns))
df_exp <- fitdist(as.numeric(df_mns), "exp")
df_gamma <- fitdist(as.numeric(df_mns), 'gamma')
df_wbl <- fitdist(as.numeric(df_mns), 'weibull')

```

```
labs <- c('exp', 'gamma', 'weibull')
denscomp(ft = list(df_exp, df_gam, df_wbl), legendtext = labs)
cdfcomp(ft = list(df_exp, df_gam, df_wbl), legendtext = labs)
qqcomp(ft = list(df_exp, df_gam, df_wbl), legendtext = labs)
ppcomp(ft = list(df_exp, df_gam, df_wbl), legendtext = labs)

gofstat(list(df_exp, df_gam, df_wbl))

boot_exp <- bootdist(df_exp, niter = 1001, silent = T)
summary(boot_exp)
```