

# Rake II/ Strain-Rate

Erdbebengruppe

23.1.2021

## Rake modifiziert

Hier ist nun der Rake anders modifiziert und auch inhaltlich nun stimmig.

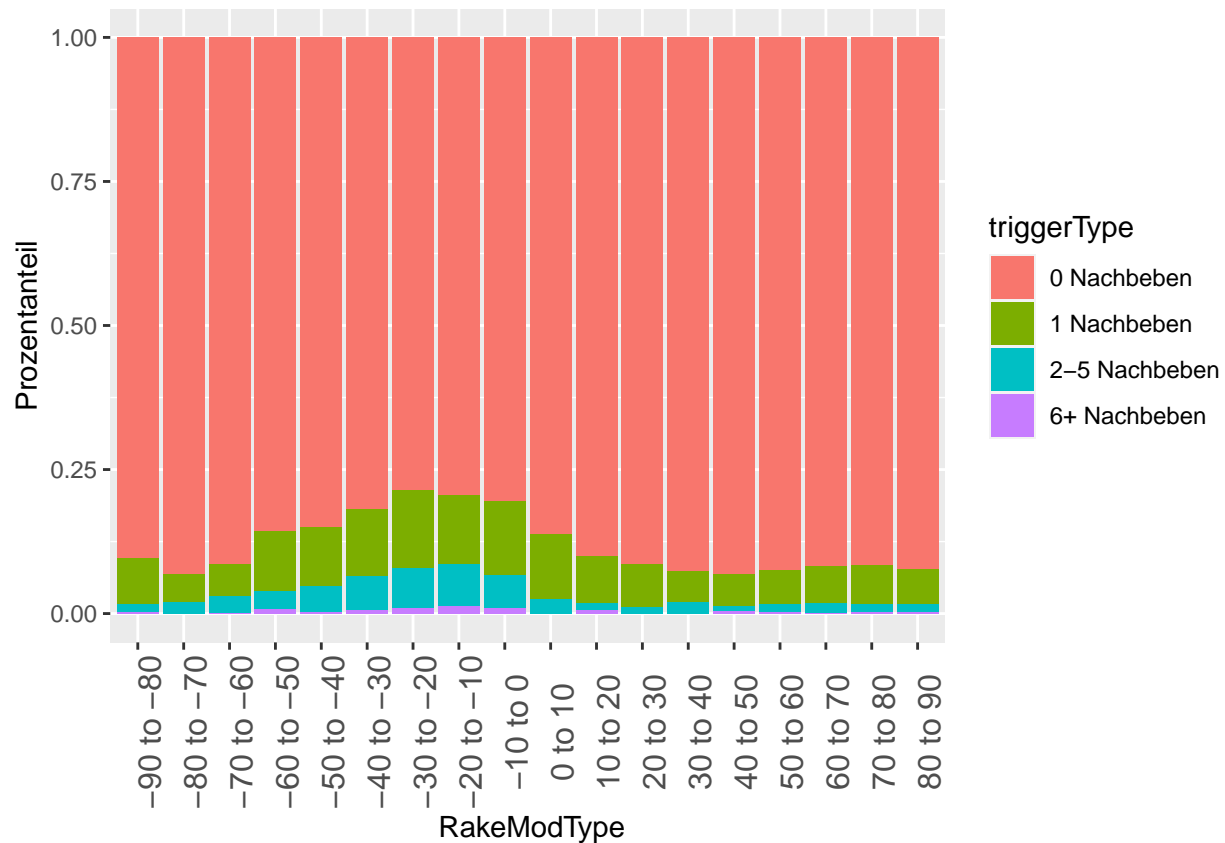
```
# -90 "perfekte" Abschiebung bis zu 90° "perfekte" Aufschiebung
rake_mod <- full_data$rake
for (i in seq_len(length(rake_mod))) {
  if (rake_mod[[i]] < 0) {
    rake_mod[[i]] <- (-90) + abs(rake_mod[[i]] + 90)
  }
  else {
    rake_mod[[i]] <- 90 - abs(rake_mod[[i]] - 90)
  }
}
full_data$rake_mod <- rake_mod
head(full_data[, c("rake", "rake_mod")], n = 10)
```

```
##      rake rake_mod
## 1      89      89
## 2      96      84
## 3      55      55
## 4     -130     -50
## 5     -123     -57
## 6     -145     -35
## 7      110      70
## 8     -142     -38
## 9       87      87
## 10     -71     -71
```

Auch graphisch lässt sich nun hier ein leichter Zusammenhang vermuten.

```
# Kategorisiere die rake_mod Variable
labels_rake_mod <- character(18)
for (i in seq_len(length(labels_rake_mod))) {
  labels_rake_mod[[i]] <- paste0(seq(-90, 90, 10)[[i]], " to ", c(seq(-90, 90, 10))[[i + 1]])
}
full_data$rakeModType <- cut(full_data$rake_mod, breaks = c(-90.1, seq(-80, 90, 10)),
                             labels = labels_rake_mod)

# Graphisch erneut über einen zu 100% gestackten Barplot
full_data$count <- rep(1, nrow(full_data))
ggplot(data = full_data, aes(x = rakeModType, y = count, fill = triggerType)) +
  geom_bar(position = "fill", stat = "identity") + xlab("RakeModType") + ylab("Prozentanteil") +
  theme(axis.text.x = element_text(angle = 90, size = 12))
```



Diesen Zusammenhang haben wir versucht mit Regressionsplines in ein Modell nur mit der Magnitude ( $aexp(bmag)$ , wie zuletzt) aufzunehmen, jedoch werden die geschätzten Koeffizienten größtenteils nicht signifikant. Also vermutlich bieten sie sobald die Magnitude mit drin ist nur noch relativ wenig Erklärungswert der Streuung. Die R-interne gam-Funktion die selbst über “Penalisierte Splines” die Regressionsplines bestimmt, schafft es zwar die Splines signifikant ins Modell einzubringen, jedoch wird auch hier das Modell, dadurch nur geringfügig besser. Hier einmal beide Modelloutputs und die zugehörigen AIC.

```
# Modell ohne rake_mod
test_model1 <- gam(formula = triggerCountTh ~ I(8.971e-08 * exp(2.581 * mag)), family = nb(), data = fu
summary(test_model1)

##
## Family: Negative Binomial(0.183)
## Link function: log
##
## Formula:
## triggerCountTh ~ I(8.971e-08 * exp(2.581 * mag))
##
## Parametric coefficients:
##
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.92226    0.02990  -64.30  <2e-16 ***
## I(8.971e-08 * exp(2.581 * mag))  0.58540    0.03126   18.73  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) = -3.58e+250    Deviance explained = 99.6%
```

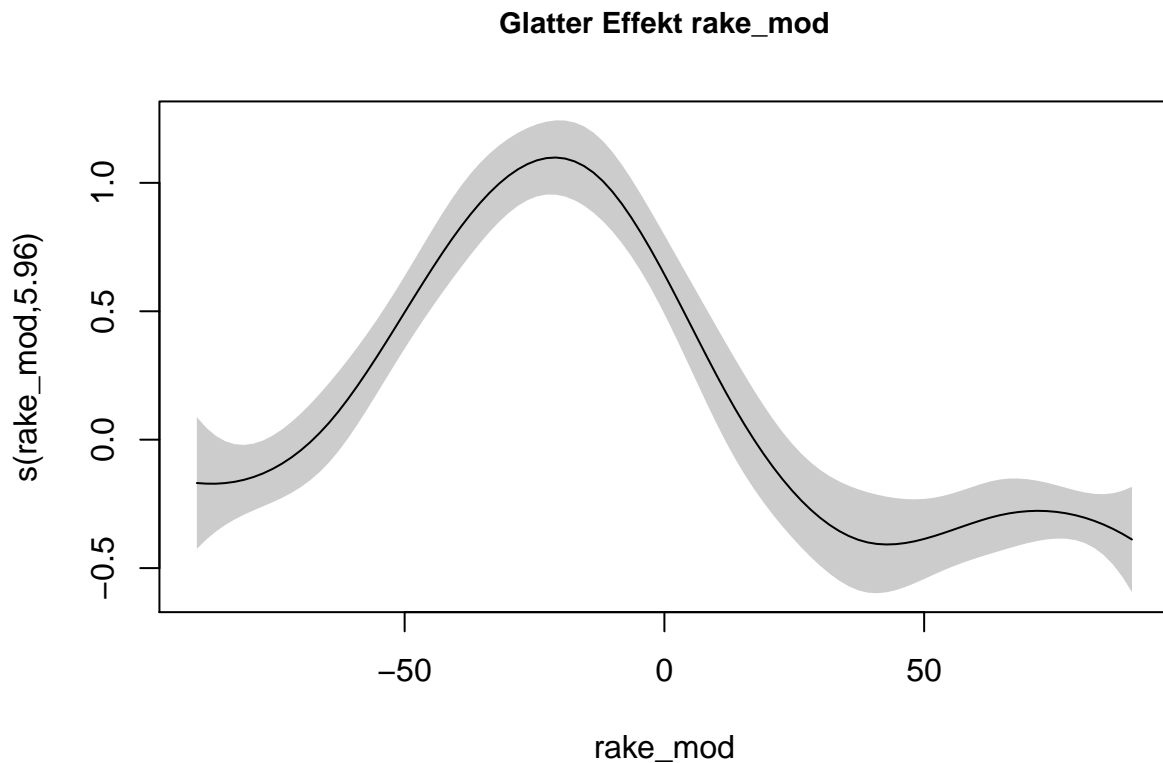
```
## -REML = 5934.2  Scale est. = 1          n = 13709
AIC(test_model1)

## [1] 11865.33
# Modell mit rake_mod, geschätzter Thetaparameter aus test_model1
test_model2 <- gam(formula = triggerCountTh ~ I(8.971e-08 * exp(2.581 * mag)) + s(rake_mod, bs = "ps"),
summary(test_model2)

##
## Family: Negative Binomial(0.183)
## Link function: log
##
## Formula:
## triggerCountTh ~ I(8.971e-08 * exp(2.581 * mag)) + s(rake_mod,
##      bs = "ps")
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.059812   0.031754  -64.87  <2e-16 ***
## I(8.971e-08 * exp(2.581 * mag))  0.654784   0.004533  144.44  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df Chi.sq p-value
## s(rake_mod) 5.964  6.761    294  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  -4.42e+280  Deviance explained = 26.2%
## UBRE = -0.67249  Scale est. = 1          n = 13709
AIC(test_model2)

## [1] 11553.92
```

Ansich würde der von gam gewählte Regressionspline den Zusammenhang eigentlich wie wir ihn visuell gefunden haben genauso einbringen. Hier ein Graph wie das gam-Modell rake\_mod einfließen lässt.



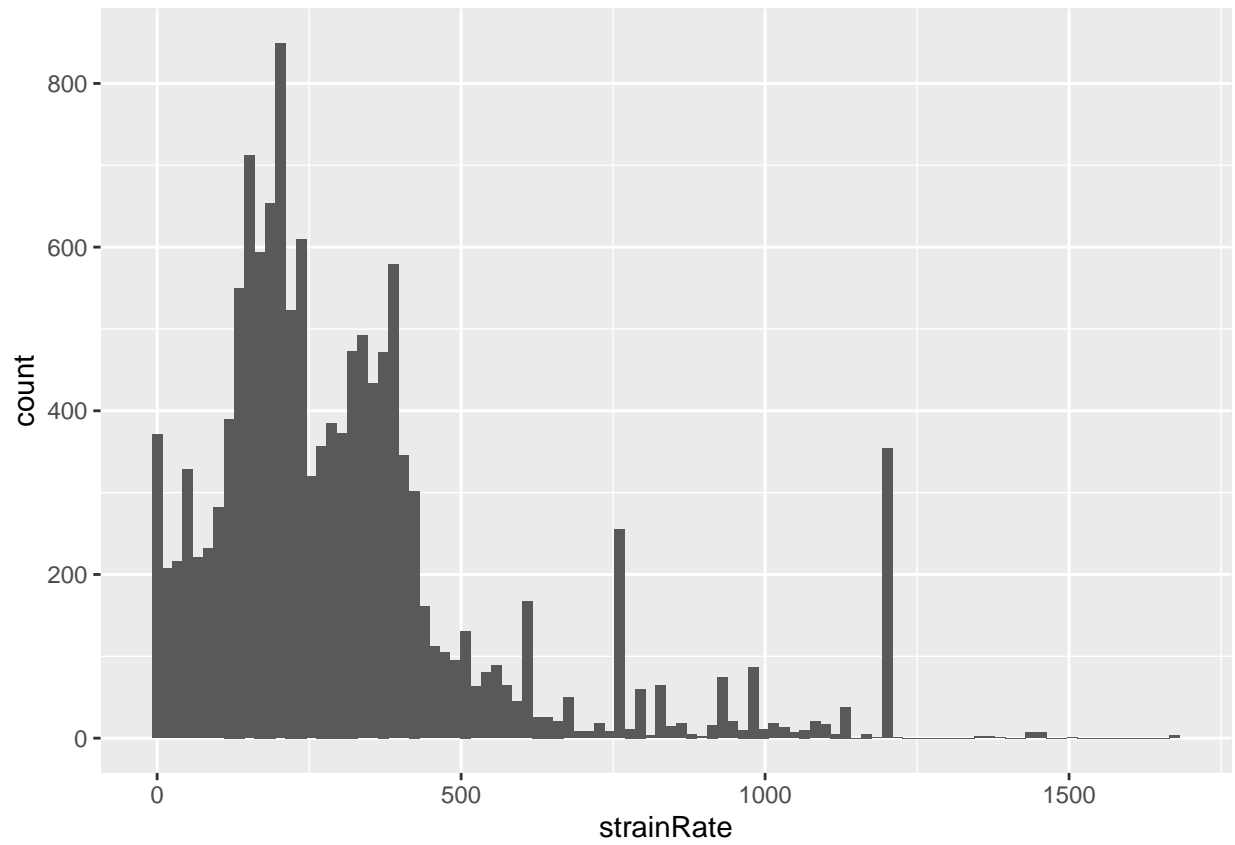
Wir werden demnächst noch einen Termin mit Herrn Küchenhoff machen und unter anderem dieses Problem besprechen.

## Strain-Rate

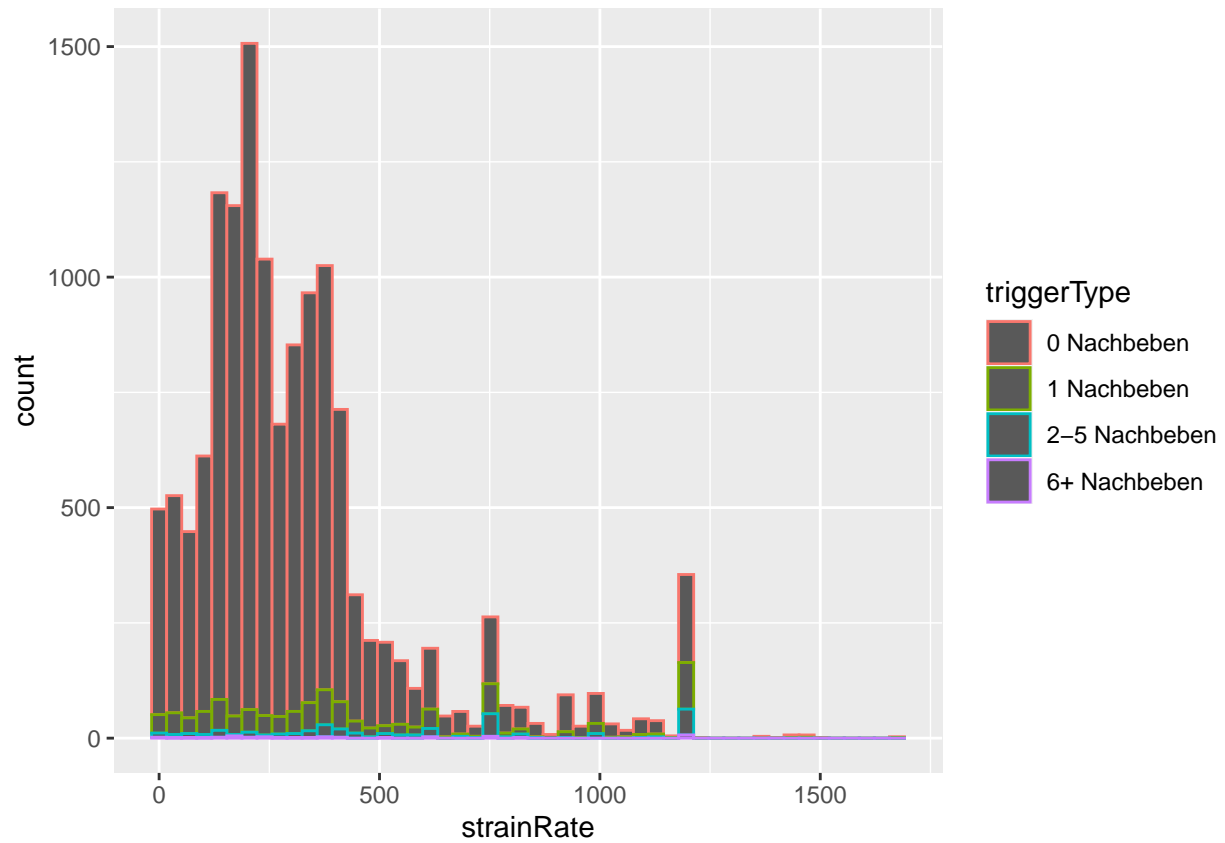
Die Strain-Rate lässt grafisch einen leicht positiven linearen Zusammenhang zur Nachbebenanzahl vermuten. Im Folgenden ein paar Outputs und Plots zu unserer bisherigen Strain-Rate-Analyse.

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.1  157.8   251.5   317.7   387.5   1674.1
```

```
# Histogramme
ggplot(full_data, aes(strainRate)) + geom_histogram(bins = 100)
```

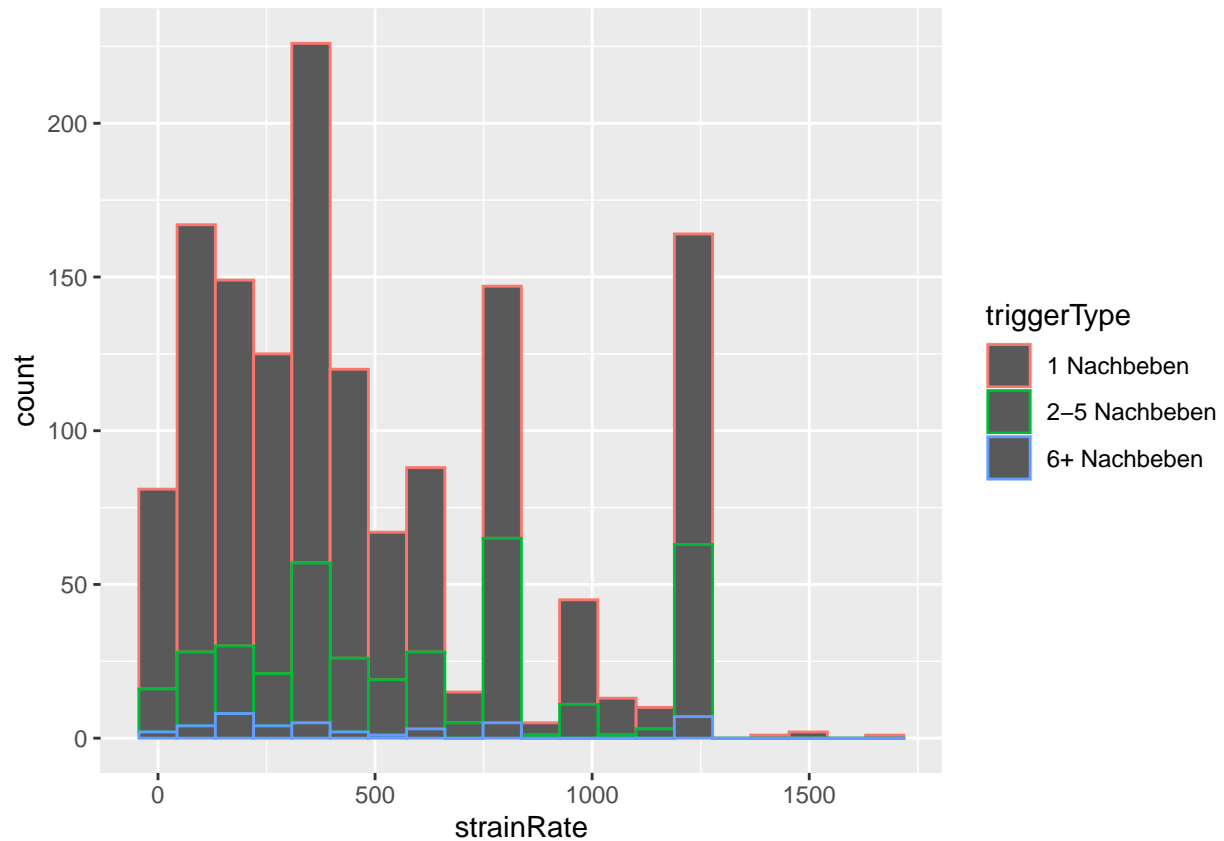


```
ggplot(full_data, aes(strainRate, colour = triggerType)) + geom_histogram(bins = 50)
```



```
# Nur Beben, die Nachbeben triggern
```

```
ggplot(full_data[full_data$triggerType != "0 Nachbeben", ], aes(strainRate, colour = triggerType)) + ge
```



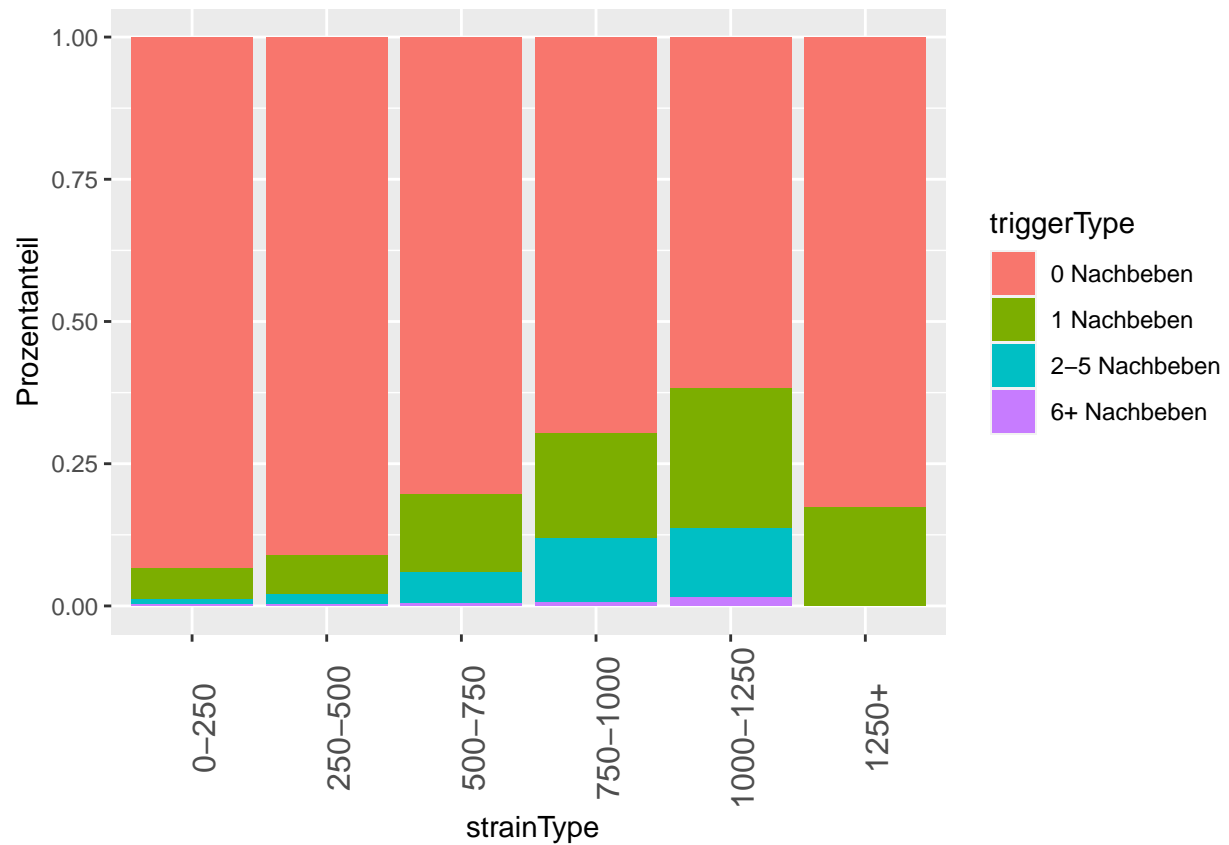
```
# Kategoriale Variable fuer die grafische Analyse
```

```
full_data$strainType <- cut(full_data$strainRate, breaks = c(-0.1, 250, 500, 750, 1000, 1250, max(full_data$strainRate)), right = FALSE)
summary(full_data$strainType)
```

```
##      0-250   250-500   500-750   750-1000  1000-1250   1250+
##      6826      4919      802        650        489        23
```

```
# Barplot
```

```
ggplot(data = full_data, aes(x = strainType, y = count, fill = triggerType)) +
  geom_bar(position = "fill", stat = "identity") + xlab(" strainType") + ylab("Prozentanteil") +
  theme(axis.text.x = element_text(angle = 90, size = 12))
```



Der Wert von 1205.640431472 für die Strain-Rate kommt 351 mal in unseren Daten vor, aber das ist vermutlich ein Artefakt vom ursprünglichen Interpolieren dieser Variable?