

360 Final Project

Group: Flora Shi, Belle Xu

20 April, 2021

```
stop <- read.table("stop-and-frisk.dat", header = TRUE)
```

Exploratory Data Analysis

Exploratory data analysis should support project goals and help guide specification of model.

```
#make categorical variables factor
stop <- stop %>%
  mutate(precinct = factor(precinct)) %>%
  mutate(eth = factor(eth, levels = c("3", "1", "2"))) %>%
  mutate(crime = factor(crime))

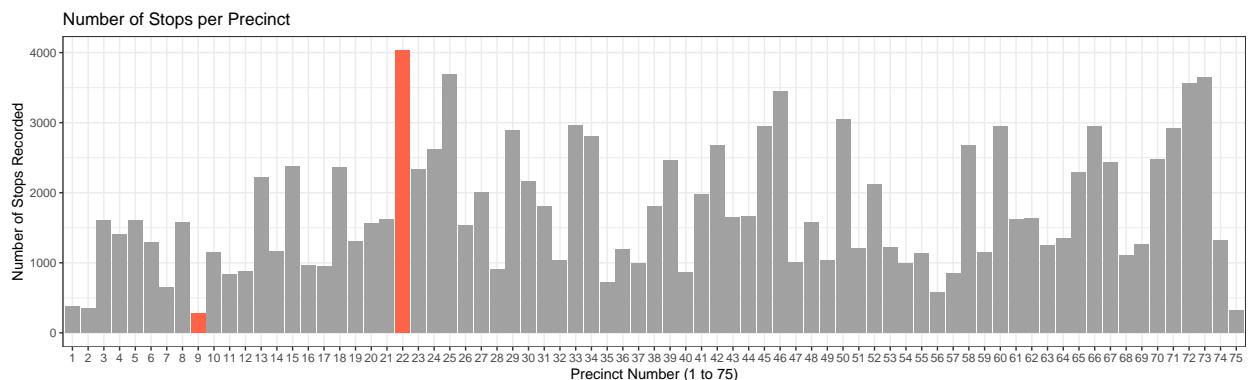
#crime rate for a certain crime for each eth in a certain precinct
stop <- stop %>% mutate(crime_rate = past.arrests/pop)
#compute population proportion with in each precinct
stop <- stop %>% group_by(precinct) %>%
  mutate(total_pop = sum(pop/4))
stop <- stop %>% mutate(pop_prop = pop/total_pop)

#make a by-precinct dataframe
stop <- stop %>%
  group_by(precinct) %>%
  mutate(stops_in_precinct = sum(stops))

stops_by_precinct <- stop %>% distinct(stops_in_precinct) %>%
  pull(stops_in_precinct)
precinct <- seq(from = 1, to = 75, by = 1)
pop_by_precinct<- stop %>% group_by(precinct) %>%
  mutate(pop_in_precinct = sum(pop)) %>%
  distinct(pop_in_precinct) %>%
  pull(pop_in_precinct)
by_precinct_df <- data.frame(pop_by_precinct = pop_by_precinct,
                               stops_by_precinct = stops_by_precinct,
                               precinct = as.factor(precinct))

by_precinct_df <- by_precinct_df %>%
  mutate(toHighlight = if_else(stops_by_precinct == min(stops_by_precinct),
                             | stops_by_precinct == max(stops_by_precinct),
                             "yes", "no"))
```

```
#number of stops by precinct
ggplot(data = by_precinct_df,
       mapping = aes(x = precinct, y = stops_by_precinct, fill = toHighlight)) +
  geom_col() +
  scale_fill_manual(values= c("yes"="tomato", "no"="#a1a1a1"), guide = FALSE) +
  labs(title = "Number of Stops per Precinct", x = "Precinct Number (1 to 75)",
       y = "Number of Stops Recorded")
```



```
#Need interpretation
# get the max and min stops with function and just type it
min(by_precinct_df$stops_by_precinct)
```

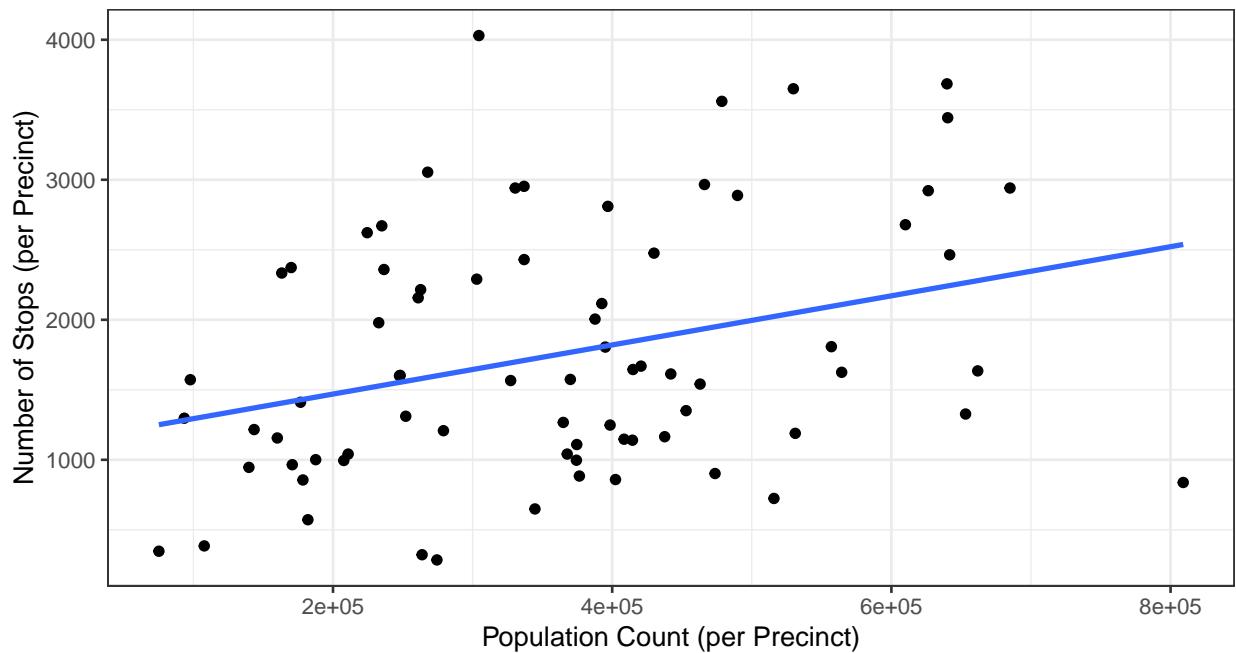
```
## [1] 285
```

```
max(by_precinct_df$stops_by_precinct)
```

```
## [1] 4030
```

```
#number of stops vs. precinct pop
ggplot(data = by_precinct_df,
       mapping = aes(x = pop_by_precinct, y = stops_by_precinct)) +
  geom_point() +
  labs(title = "Number of Stops vs. Population per Precinct",
       x = "Population Count (per Precinct)",
       y = "Number of Stops (per Precinct)") +
  geom_smooth(method = "lm", formula = y~x, se = FALSE)
```

Number of Stops vs. Population per Precinct

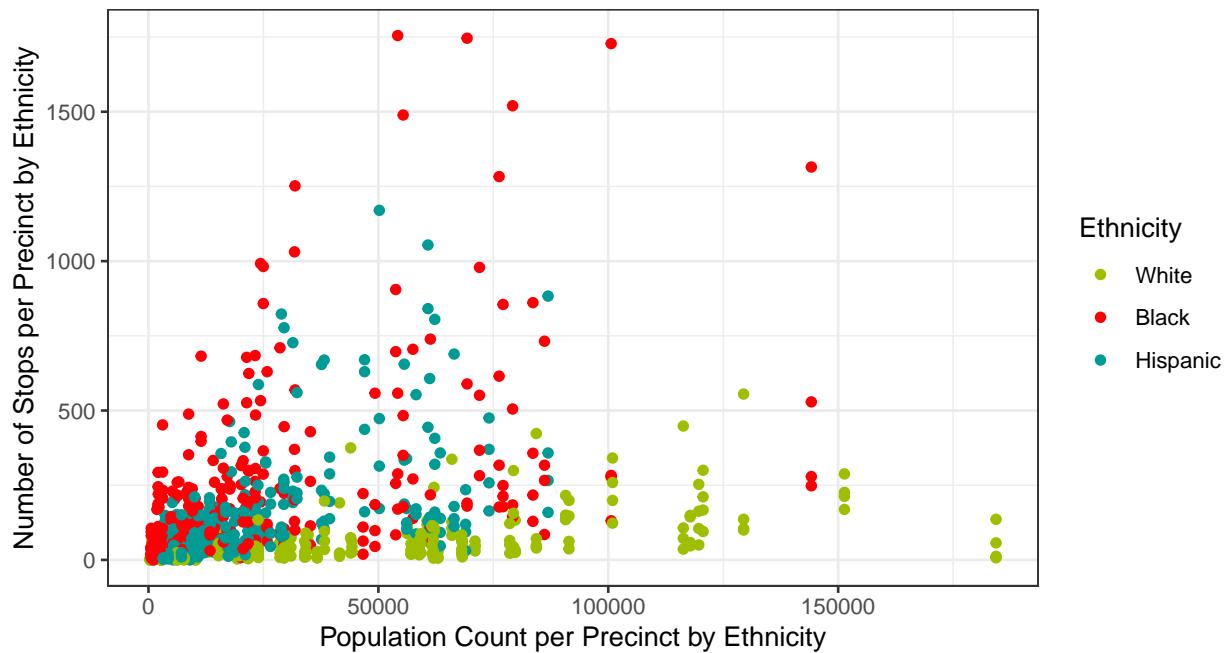


```
#no obvious trend between population in precinct vs. stops in a precinct
#maybe weakly positive
```

```
#number of stops for each ethnicity vs.
# population for each ethnicity per precinct

ggplot(data = stop, mapping = aes(x = pop, y = stops, colour = eth)) +
  geom_point() +
  labs(title = "Number of Stops vs. Population per Precinct by Ethnicity",
       x = "Population Count per Precinct by Ethnicity",
       y = "Number of Stops per Precinct by Ethnicity",
       colour = "Ethnicity") +
  scale_color_manual(values = c("#9EBE00", "#FD0006", "#009B95"),
                     labels = c("White", "Black", "Hispanic"))
```

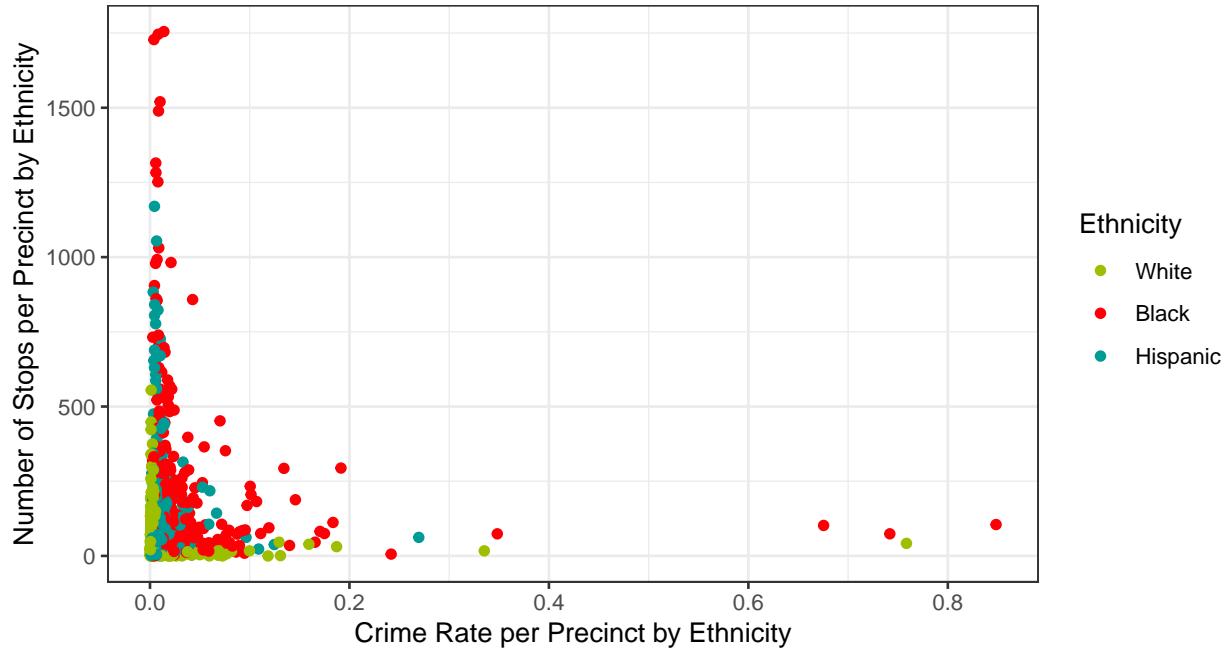
Number of Stops vs. Population per Precinct by Ethnicity



```
#There does seem to be some trend because we can see that none of the
# points for the white population is in high number of stops
# despite having very high population
```

```
#num of stops vs crime rate
ggplot(data = stop, mapping = aes(x = crime_rate, y = stops, colour = eth)) +
  geom_point() +
  labs(title = "Number of Stops vs. Crime Rate in the Past Year per Precinct by Ethnicity",
       x = "Crime Rate per Precinct by Ethnicity",
       y = "Number of Stops per Precinct by Ethnicity",
       colour = "Ethnicity") +
  scale_color_manual(values = c("#9EBE00", "#FD0006", "#009B95"),
                     labels = c("White", "Black", "Hispanic"))
```

Number of Stops vs. Crime Rate in the Past Year per Precinct by Ethnicity



```
#crime rate is generally low with a few exceptions; however black
# and hispanic stop count are still
# a lot higher in low crime rate regions
```

```
#get datasets for different crime types
stop.1 <- stop %>% filter(crime == 1)
stop.2 <- stop %>% filter(crime == 2)
stop.3 <- stop %>% filter(crime == 3)
stop.4 <- stop %>% filter(crime == 4)
#investigate mean and var of each stop
Violence <- c(mean(stop.1$stops), var(stop.1$stops))
Weapon <- c(mean(stop.2$stops), var(stop.2$stops))
Property <- c(mean(stop.3$stops), var(stop.3$stops))
Drug <- c(mean(stop.4$stops), var(stop.4$stops))
df <- data.frame(Violence, Weapon, Property, Drug)
row.names(df) <- c("Mean", "Variance")
df
```

	Violence	Weapon	Property	Drug
## Mean	142.5689	256.8356	117.9822	66.70222
## Variance	22489.0946	123810.0309	18919.3479	4978.23683

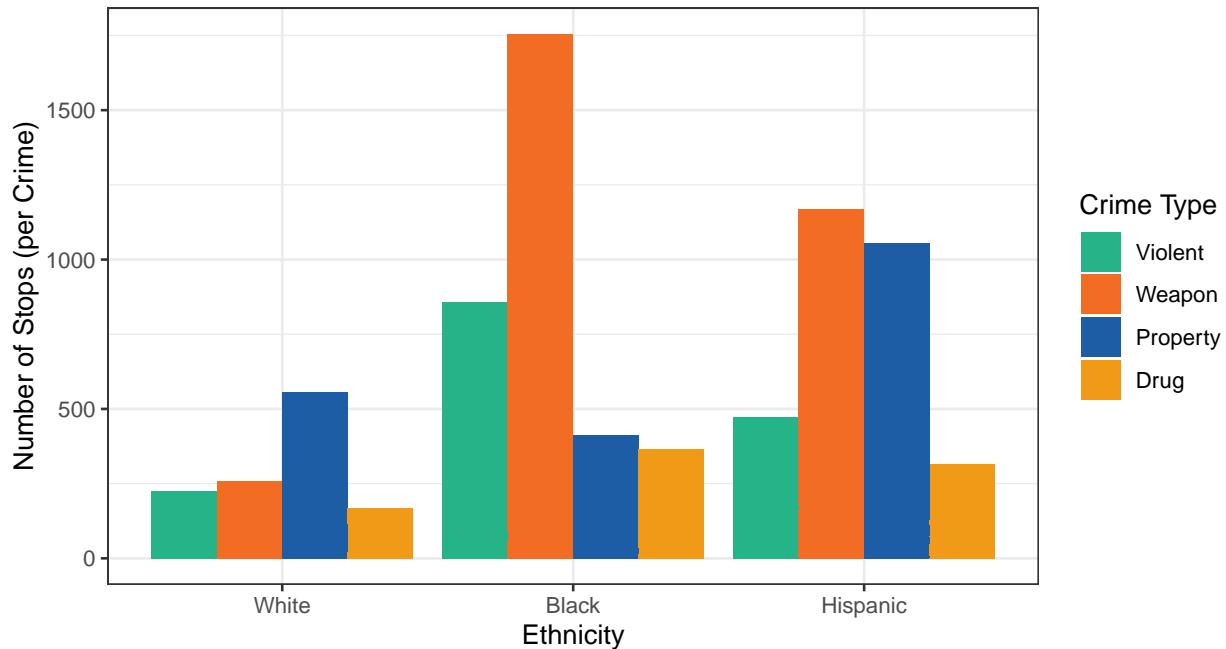
```
#number of stops per crime for each ethnicity
ggplot(data = stop, mapping = aes(x = eth, y = stops, fill = crime)) +
  geom_col(position = "dodge") +
  labs(title = "Number of Stops per Crime for each Ethnicity",
       x = "Ethnicity",
       y = "Number of Stops (per Crime)",
       fill = "Crime Type") +
  scale_x_discrete(labels=c("3" = "White", "1" = "Black",
```

```

    "2" = "Hispanic")) +
scale_fill_manual(values = c("#25B388", "#F26C24", "#1C5DA6", "#F09A18"),
                 labels = c("Violent", "Weapon",
                           "Property", "Drug"))

```

Number of Stops per Crime for each Ethnicity

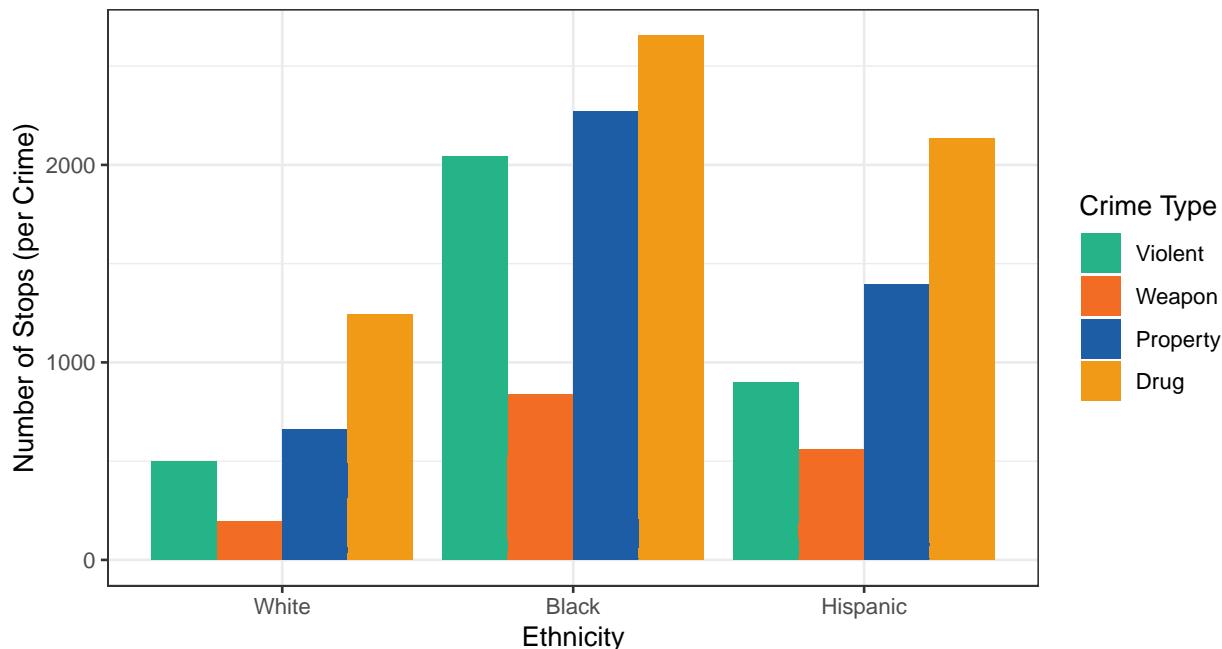


```

#number of past arrests per crime for each ethnicity
ggplot(data = stop, mapping = aes(x = eth, y = past.arrests, fill = crime)) +
  geom_col(position = "dodge") +
  labs(title = "Number of Past Arrests per Crime for each Ethnicity",
       x = "Ethnicity",
       y = "Number of Stops (per Crime)",
       fill = "Crime Type") +
  scale_x_discrete(labels=c("3" = "White", "1" = "Black",
                           "2" = "Hispanic")) +
  scale_fill_manual(values = c("#25B388", "#F26C24", "#1C5DA6", "#F09A18"),
                 labels = c("Violent", "Weapon",
                           "Property", "Drug"))

```

Number of Past Arrests per Crime for each Ethnicity



```
#from the two plots above, the stop reasons and past arrests seem to have no relation
```

Modeling

Rstan GLM

```
#glm for crime 1; done for testing only. don't repeat the code for other crimes yet.
#assuming negative binom sampling model

stan.glm.1 <- stan_glm(data = stop.1,
                        formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:eth + crime_ra
```

Violent Crimes

```
## Warning: There were 3 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```

#no r-hat warning = converged. no ess warning = good enough ess. apparently divergent transitions can b

#checking if omitting stuff will make the model better
stan.glm.2 <- update(stan.glm.1, formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth+ pop_prop:

## Warning: There were 3 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

## Warning: Examine the pairs() plot to diagnose sampling problems

stan.glm.3 <- update(stan.glm.2, formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth)

## Warning: There were 32 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

## Warning: Examine the pairs() plot to diagnose sampling problems

stan.glm.4 <- update(stan.glm.3, formula = stops ~ crime_rate + pop_prop + eth)
stan.glm.5 <- update(stan.glm.4, formula = stops ~ crime_rate + pop_prop)

#checking if adding/changing stuff will make the model better
stan.glm.6 <- stan_glm(data = stop.1,
                        formula = stops ~ past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop:eth + past
                        family = neg_binomial_2, #how do I justify not having a prior?
                        seed = 360,
                        refresh = 0)
stan.glm.7 <- update(stan.glm.6, formula = stops ~ crime_rate+ pop+ eth+ crime_rate:eth+ pop:eth +crime_
stan.glm.8 <- update(stan.glm.6, formula = stops ~ past.arrests+ pop+ eth+ past.arrests:eth+ pop:eth + p
stan.glm.9 <- update(stan.glm.6, formula = stops~past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop

#look at coefficients
summary(stan.glm.1)

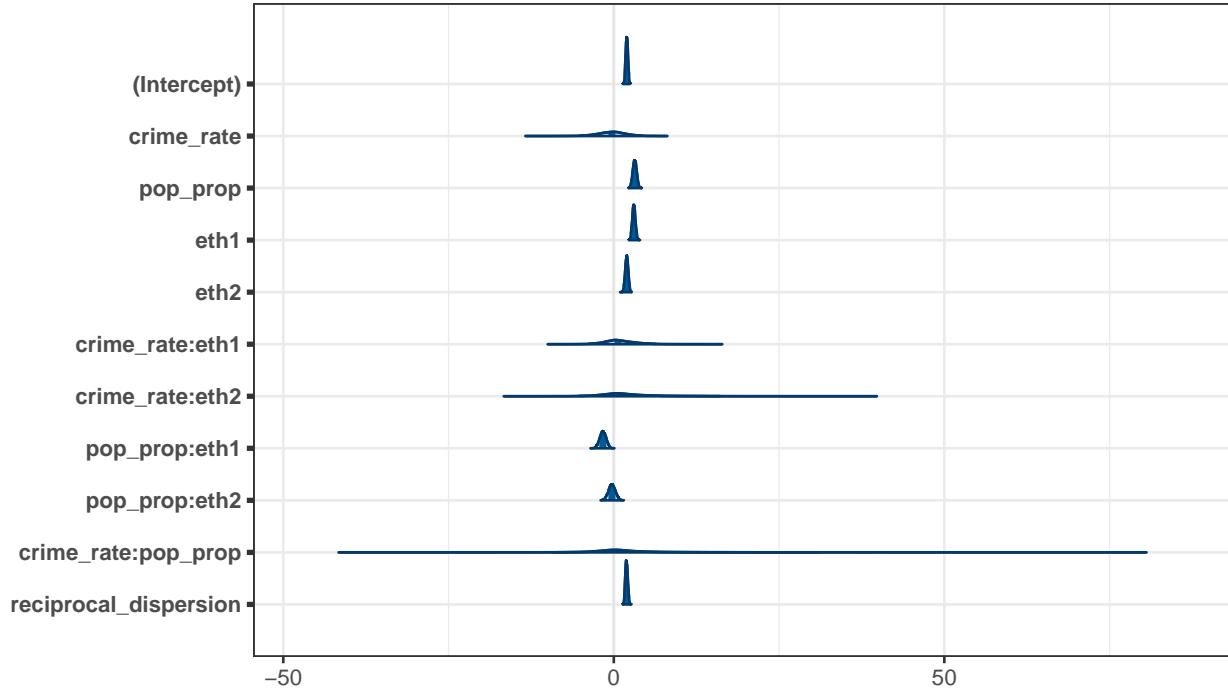
## 
## Model Info:
##   function:      stan_glm
##   family:       neg_binomial_2 [log]
##   formula:      stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:eth +
##   crime_rate:pop_prop
##   algorithm:    sampling
##   sample:       4000 (posterior sample size)
##   priors:       see help('prior_summary')
##   observations: 225
##   predictors:   10
##
## Estimates:
##                               mean     sd    10%    50%    90%
## (Intercept)            2.0     0.2   1.7    2.0    2.2

```

```

## crime_rate      -0.4    2.3 -3.1 -0.3    2.4
## pop_prop       3.2     0.3  2.8   3.2    3.5
## eth1           3.0     0.2  2.7   3.0    3.3
## eth2           2.0     0.2  1.7   2.0    2.2
## crime_rate:eth1 0.9     2.5 -1.9   0.7    4.0
## crime_rate:eth2 2.3     5.2 -2.5   1.2    8.8
## pop_prop:eth1 -1.7     0.5 -2.3  -1.7   -1.0
## pop_prop:eth2 -0.3     0.5 -1.0  -0.3    0.4
## crime_rate:pop_prop 3.0    9.6 -3.8   0.8   12.3
## reciprocal_dispersion 1.9     0.2  1.7   1.9    2.2
##
## Fit Diagnostics:
##          mean    sd   10%   50%   90%
## mean_PPD 150.2 14.2 132.7 149.0 168.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (for de
##
## MCMC diagnostics
##          mcse Rhat n_eff
## (Intercept) 0.0  1.0 2409
## crime_rate  0.0  1.0 2476
## pop_prop   0.0  1.0 2528
## eth1        0.0  1.0 2123
## eth2        0.0  1.0 2138
## crime_rate:eth1 0.0  1.0 2562
## crime_rate:eth2 0.1  1.0 2914
## pop_prop:eth1  0.0  1.0 2178
## pop_prop:eth2  0.0  1.0 2621
## crime_rate:pop_prop 0.2  1.0 2895
## reciprocal_dispersion 0.0  1.0 3951
## mean_PPD    0.2  1.0 4252
## log-posterior 0.0  1.0 1885
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective sample
#
#posterior distributions
mcmc_areas(as.matrix(stan.glm.1), prob = 0.95, prob_outer = 1)

```



```
round(coef(stan.glm.1), 3)
```

```
##          (Intercept)      crime_rate      pop_prop       eth1
##            1.967        -0.287        3.161      3.028
##             eth2      crime_rate:eth1      crime_rate:eth2      pop_prop:eth1
##            1.957         0.650         1.183     -1.651
##      pop_prop:eth2  crime_rate:pop_prop
##            -0.283         0.837
```

```
round(posterior_interval(stan.glm.1, prob = 0.95), 3)
```

	2.5%	97.5%
## (Intercept)	1.641	2.293
## crime_rate	-5.258	4.099
## pop_prop	2.592	3.739
## eth1	2.586	3.494
## eth2	1.509	2.397
## crime_rate:eth1	-3.796	6.271
## crime_rate:eth2	-5.287	16.060
## pop_prop:eth1	-2.618	-0.725
## pop_prop:eth2	-1.296	0.743
## crime_rate:pop_prop	-9.121	30.896
## reciprocal_dispersion	1.591	2.300

```
#posterior predictive check
loo1 <- loo(stan.glm.1, save_psis = TRUE)
loo2 <- loo(stan.glm.2, save_psis = TRUE)
loo3 <- loo(stan.glm.3, save_psis = TRUE)
loo4 <- loo(stan.glm.4, save_psis = TRUE)
```

```

loo5 <- loo(stan.glm.5, save_psis = TRUE)
loo6 <- loo(stan.glm.6, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo7 <- loo(stan.glm.7, save_psis = TRUE)
loo8 <- loo(stan.glm.8, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo9 <- loo(stan.glm.9, save_psis = TRUE)

rstanarm::loo_compare(loo1, loo2, loo3, loo4, loo5, loo6, loo7, loo8, loo9) #loo6 is the best.

##          elpd_diff se_diff
## stan.glm.6    0.0     0.0
## stan.glm.8   -13.2     5.0
## stan.glm.9   -15.1     3.9
## stan.glm.7   -25.0     6.2
## stan.glm.1   -31.7     7.4
## stan.glm.2   -31.9     7.4
## stan.glm.3   -36.9     7.9
## stan.glm.4   -37.0     7.9
## stan.glm.5  -129.5    11.2

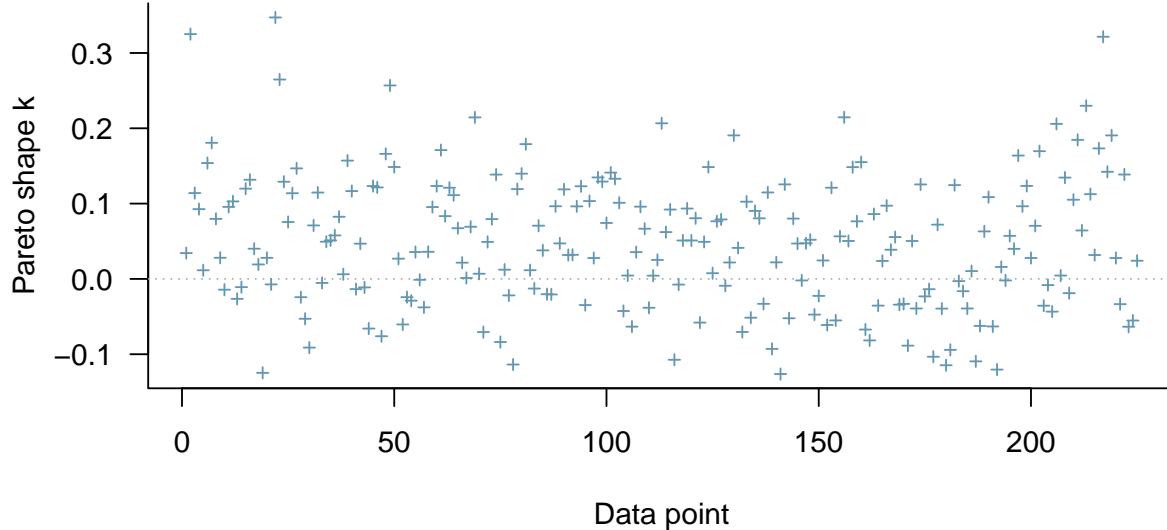
#making sure that bayesian ver of AIC (pareto k) is good
loo1

## 
## Computed from 4000 by 225 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo  -1229.3 19.5
## p_loo      7.4  0.8
## looic     2458.5 39.0
## -----
## Monte Carlo SE of elpd_loo is 0.1.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.

#check for outliers (if there are outliers, then post pred will be sensitive to 1 observation)
plot(loo1, label_points = TRUE)

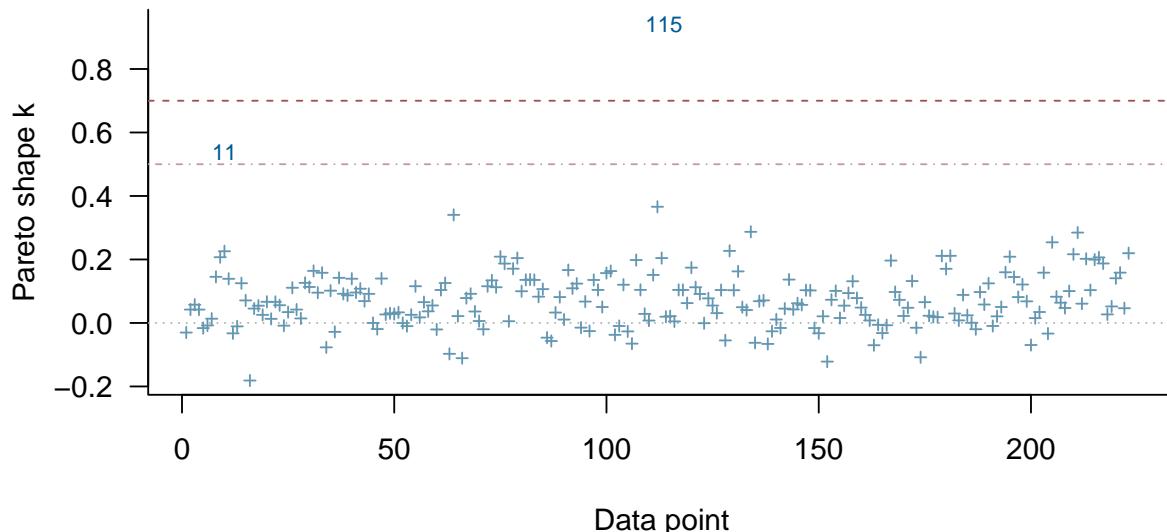
```

PSIS diagnostic plot



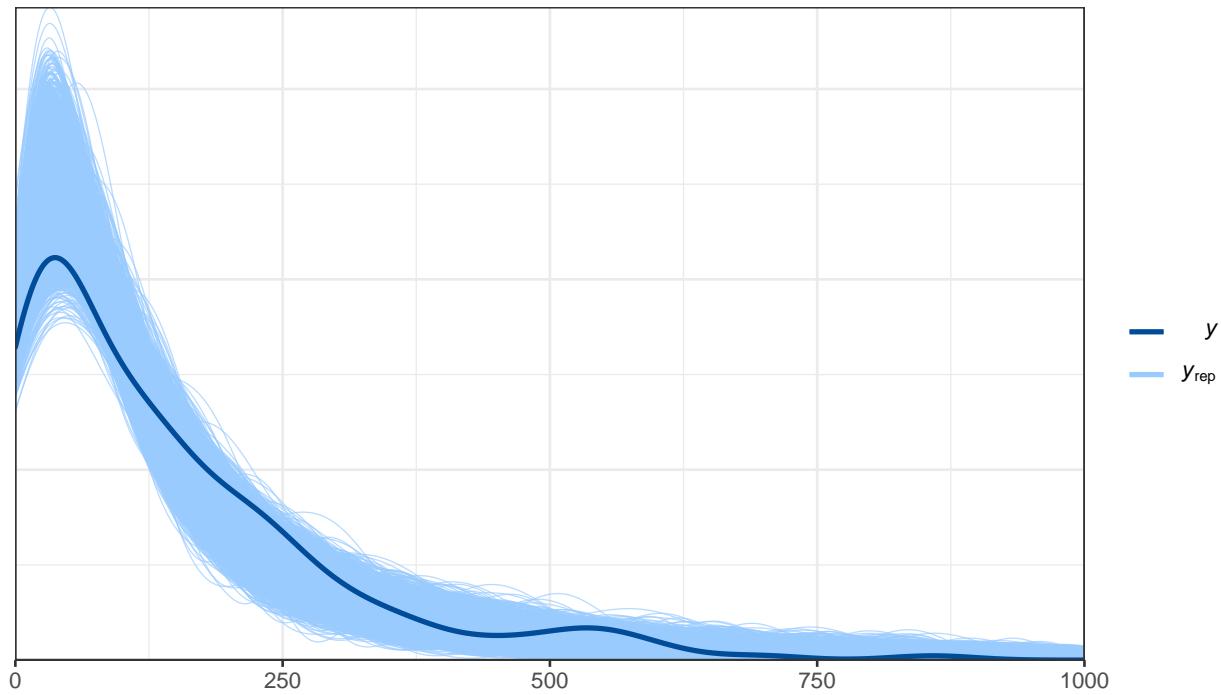
```
plot(loo6, label_points = TRUE)
```

PSIS diagnostic plot

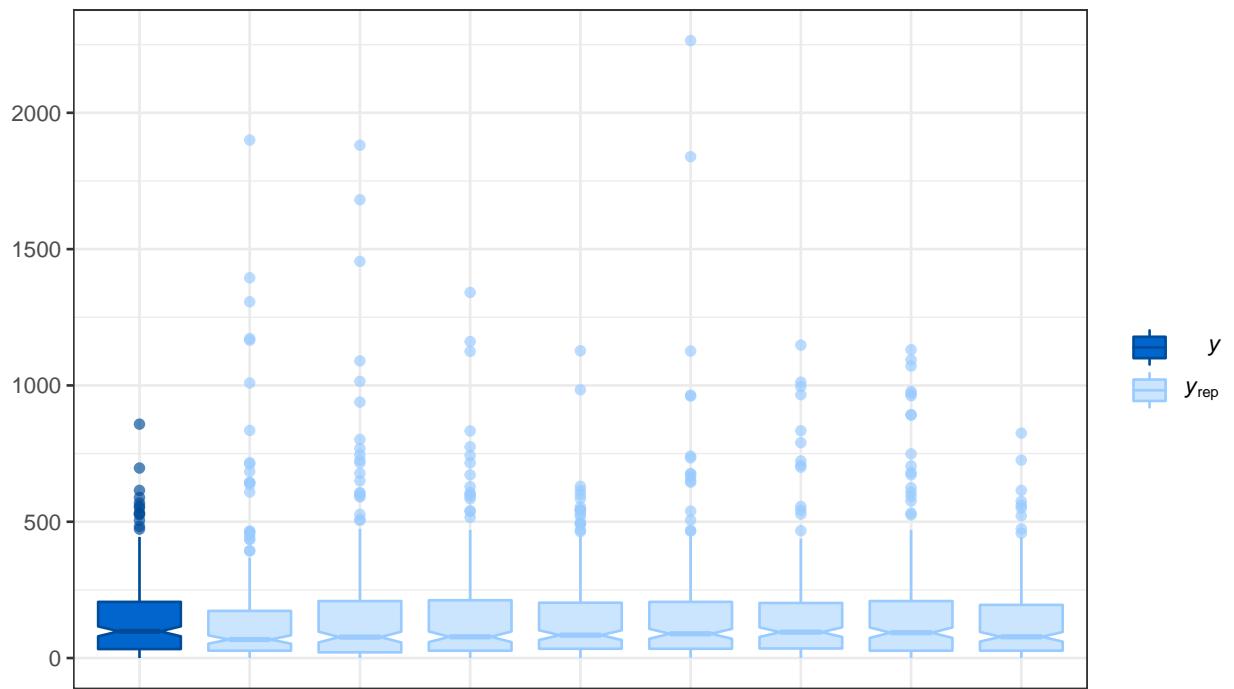


```
#individual model diagnostics - stan.glm.1
y.postpred <- posterior_predict(stan.glm.1)
color_scheme_set("brightblue")
ppc_dens_overlay(stop.1$stops, y.postpred) + xlim(0, 1000) #xlim() truncates so that we focus on the pa
```

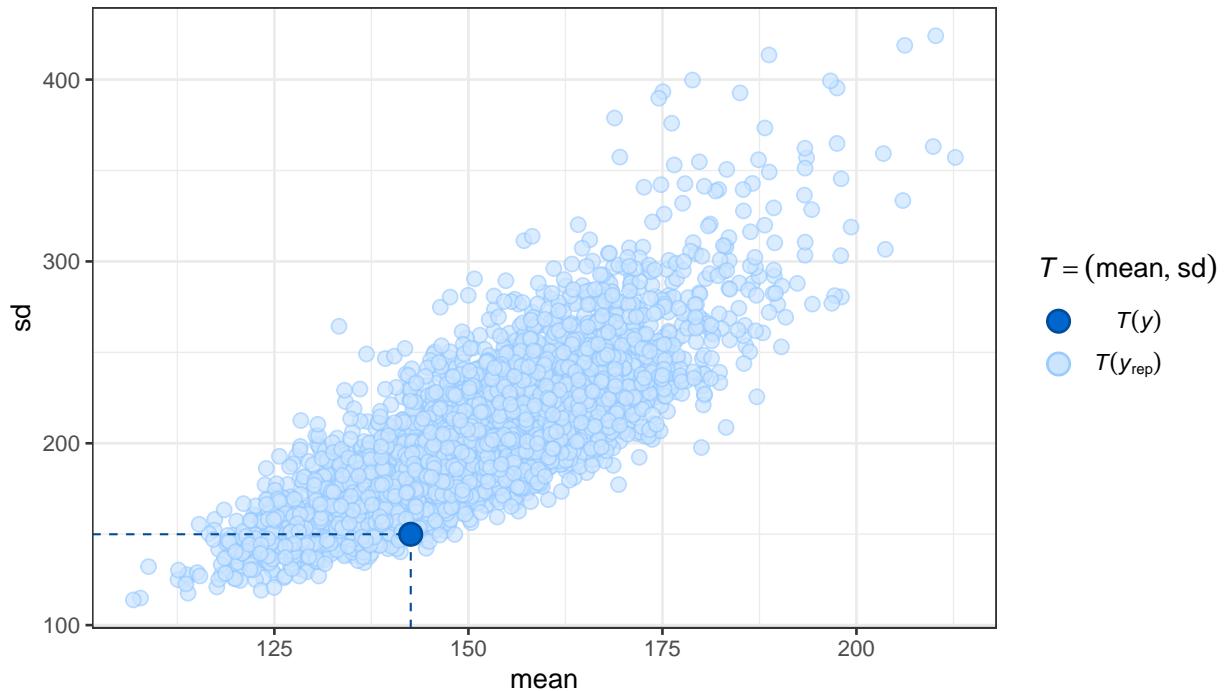
```
## Warning: Removed 8634 rows containing non-finite values (stat_density).
```



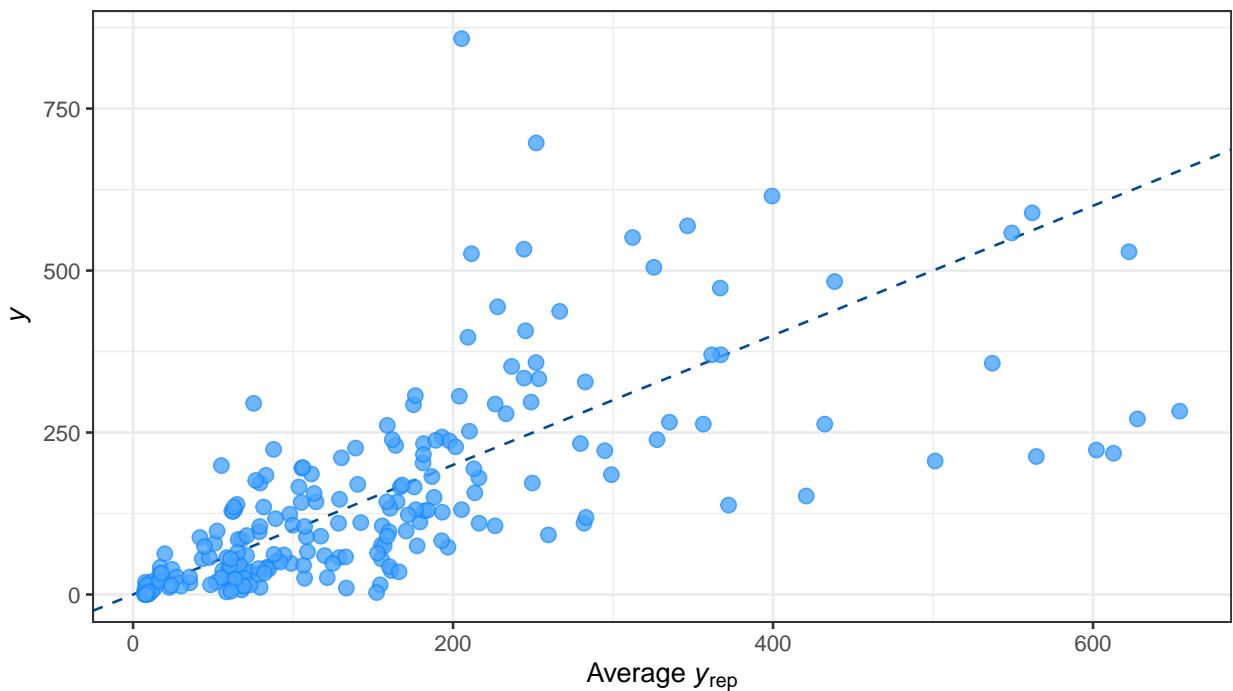
```
pp_check(stan.glm.1, plotfun = "boxplot") #check median/range
```



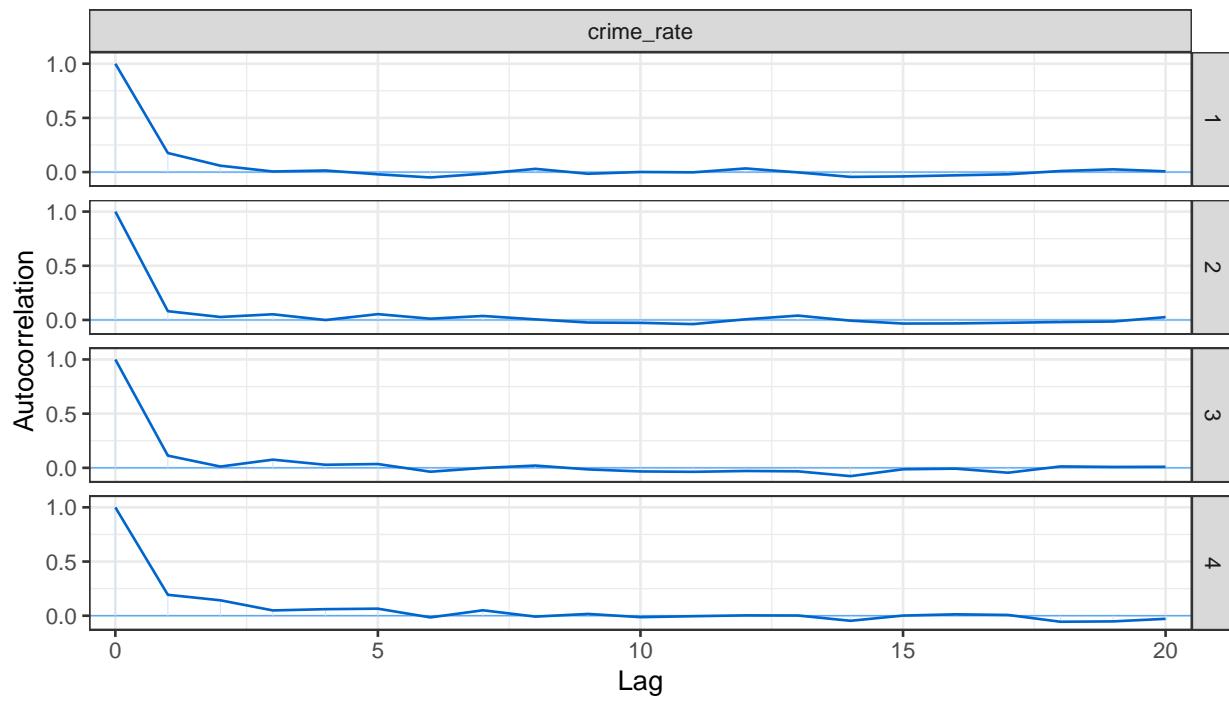
```
pp_check(stan.glm.1, plotfun = "stat_2d", stat = c("mean", "sd"))
```



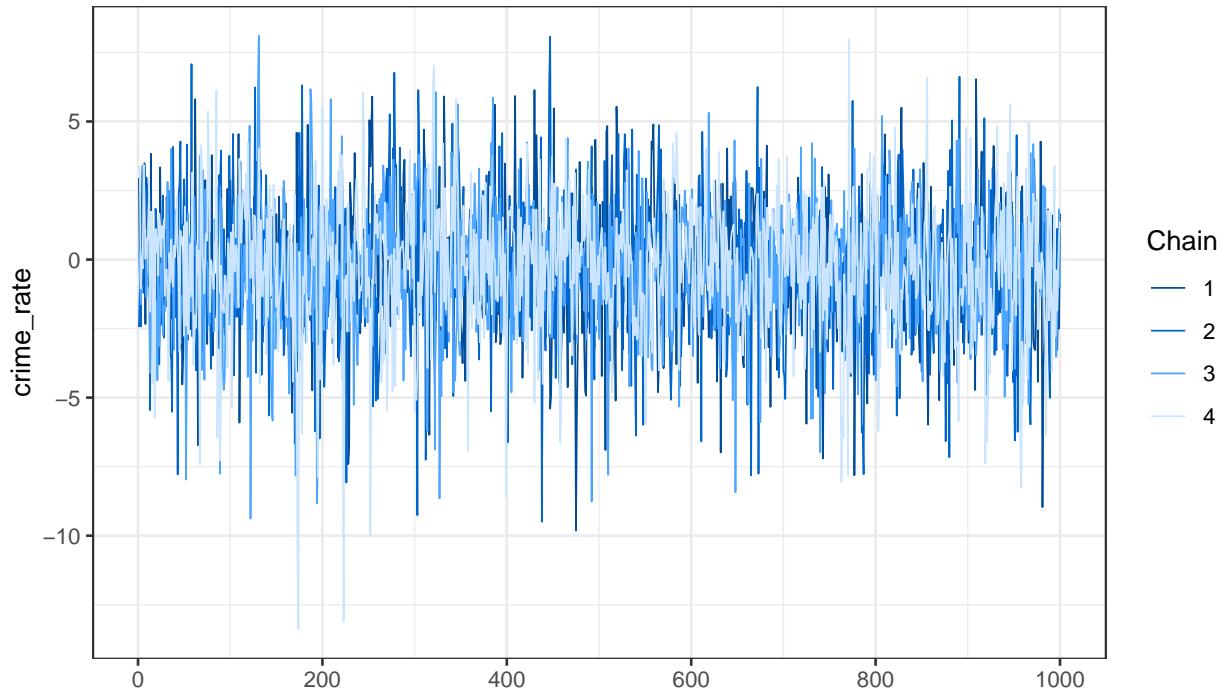
```
# Scatterplot of two test statistics (capture the mean somewhat but
# sd is kind of bad. at least it's not high sd)
pp_check(stan.glm.1, plotfun = "scatter_avg") # Scatterplot of y vs. average yrep (our model is good for
```



```
plot(stan.glm.1, "acf", pars = "crime_rate")# autocorrelation by chain
```

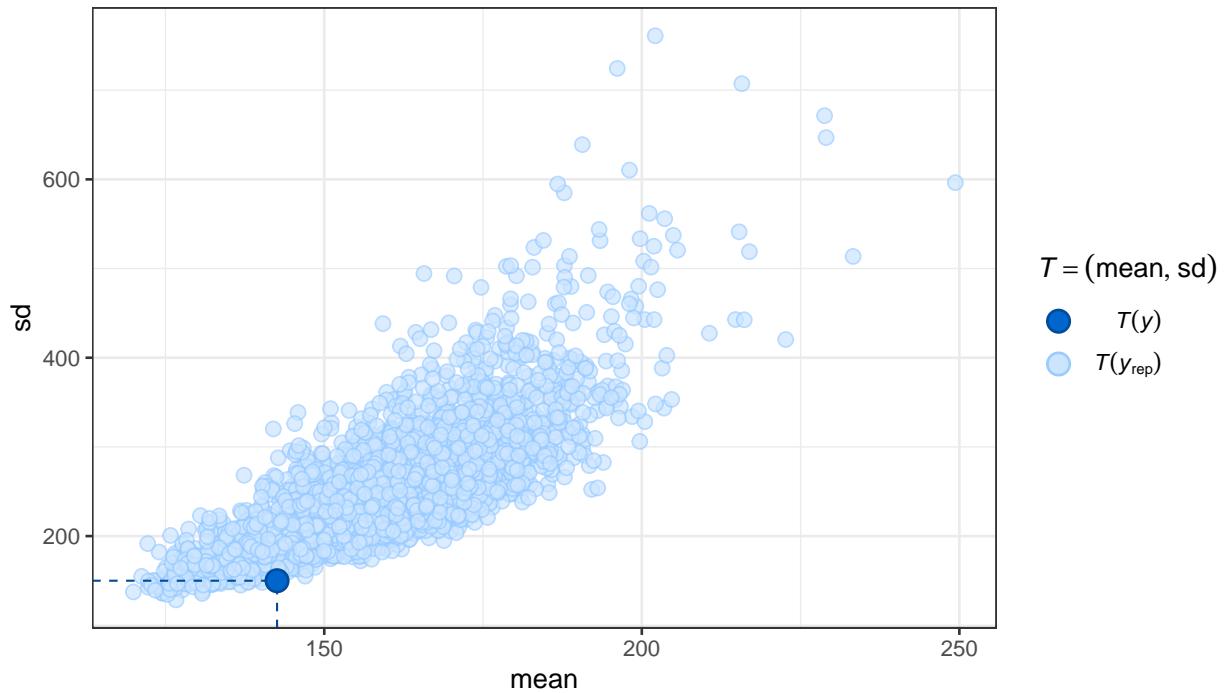


```
plot(stan.glm.1, "trace", pars = "crime_rate") #traceplot. how to separate by chain?
```

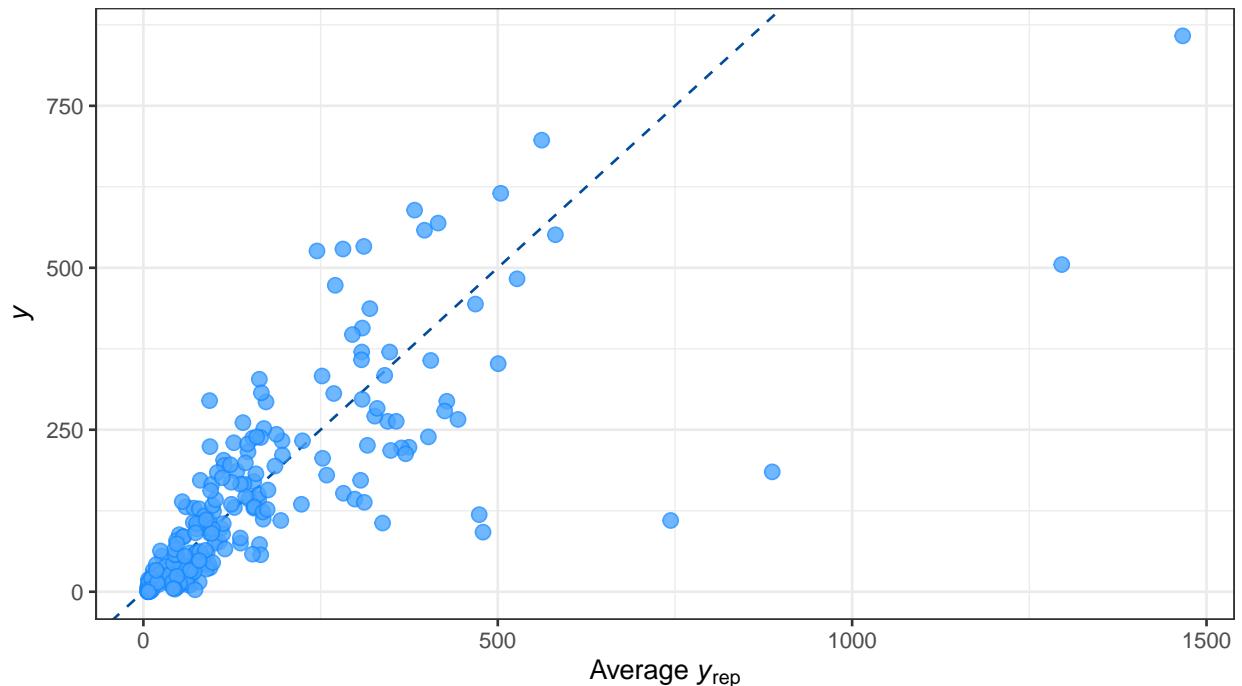


```
glm.matrix <- as.matrix(stan.glm.1)
```

```
pp_check(stan.glm.6, plotfun = "stat_2d", stat = c("mean", "sd"))
```

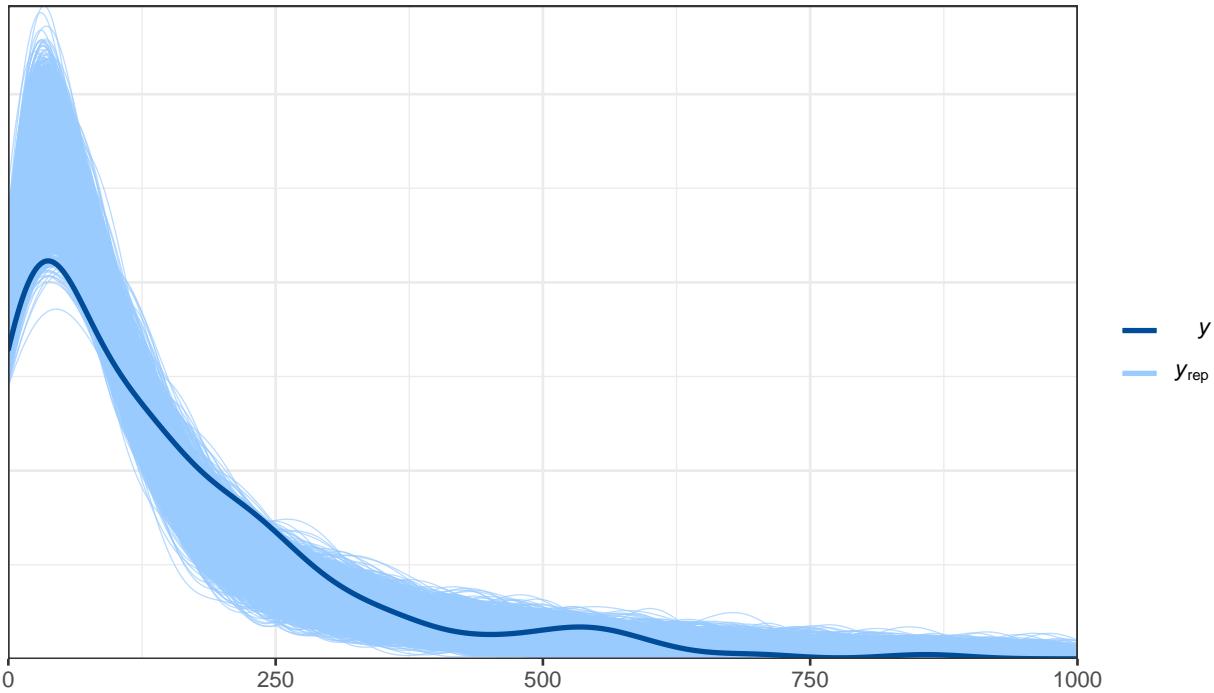


```
pp_check(stan.glm.6, plotfun = "scatter_avg")
```



```
#individual model diagnostics - stan.glm.6
y.postpred <- posterior_predict(stan.glm.6)
color_scheme_set("brightblue")
ppc_dens_overlay(stop.1$stops, y.postpred) + xlim(0, 1000) #xlim() truncates so that we focus on the pa
```

```
## Warning: Removed 12212 rows containing non-finite values (stat_density).
```



```
stan.glm.1 <- stan_glm(data = stop.2,
                         formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:eth + crime_ra
                         family = neg_binomial_2,
                         seed = 360,
                         prior = cauchy(0, 2.5),
                         prior_intercept = cauchy(0, 2.5),
                         refresh = 0)
```

Weapon Offense

```
## Warning: There were 5 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```
#checking if omitting stuff will make the model better
```

```
stan.glm.2 <- update(stan.glm.1, formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:
```

```
## Warning: There were 2 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```

stan.glm.3 <- update(stan.glm.2, formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth)

## Warning: There were 25 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

## Warning: Examine the pairs() plot to diagnose sampling problems

stan.glm.4 <- update(stan.glm.3, formula = stops ~ crime_rate + pop_prop + eth)

## Warning: There were 2 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

## Warning: Examine the pairs() plot to diagnose sampling problems

stan.glm.5 <- update(stan.glm.4, formula = stops ~ crime_rate + pop_prop)

#checking if adding/changing stuff will make the model better
stan.glm.6 <- stan_glm(data = stop.2,
                        formula = stops ~ past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop:eth + past
                        family = neg_binomial_2, #how do I justify not having a prior?
                        seed = 360,
                        refresh = 0)
stan.glm.7 <- update(stan.glm.6, formula = stops ~ crime_rate+ pop+ eth+ crime_rate:eth+ pop:eth +crime)
stan.glm.8 <- update(stan.glm.6, formula = stops ~ past.arrests+ pop+ eth+ past.arrests:eth+ pop:eth + )
stan.glm.9 <- update(stan.glm.6, formula = stops~past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop)

#posterior predictive check
loo1 <- loo(stan.glm.1, save_psis = TRUE)
loo2 <- loo(stan.glm.2, save_psis = TRUE)
loo3 <- loo(stan.glm.3, save_psis = TRUE)
loo4 <- loo(stan.glm.4, save_psis = TRUE)
loo5 <- loo(stan.glm.5, save_psis = TRUE)
loo6 <- loo(stan.glm.6, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument
##          save_psis = TRUE to get a more accurate estimate of the elpd.

loo7 <- loo(stan.glm.7, save_psis = TRUE)
loo8 <- loo(stan.glm.8, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument
##          save_psis = TRUE to get a more accurate estimate of the elpd.

loo9 <- loo(stan.glm.9, save_psis = TRUE)

rstanarm::loo_compare(loo1, loo2, loo3, loo4, loo5, loo6, loo7, loo8, loo9)

##           elpd_diff se_diff
## stan.glm.6      0.0      0.0

```

```

## stan.glm.8 -18.1      7.3
## stan.glm.9 -29.1      6.2
## stan.glm.7 -47.0      8.6
## stan.glm.1 -50.3      7.4
## stan.glm.2 -50.4      7.4
## stan.glm.4 -69.1      9.0
## stan.glm.3 -69.3      9.0
## stan.glm.5 -159.6     12.3

```

```

stan.glm.1 <- stan_glm(data = stop.3,
                        formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth+ pop_prop:eth + crime_ra
                        family = neg_binomial_2,
                        seed = 360,
                        prior = cauchy(0, 2.5),
                        prior_intercept = cauchy(0, 2.5),
                        refresh = 0)

```

Property Crimes

```

## Warning: There were 5 divergent transitions after warmup. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.

```

```

## Warning: Examine the pairs() plot to diagnose sampling problems

```

```

#checking if omitting stuff will make the model better
stan.glm.2 <- update(stan.glm.1, formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth+ pop_prop:
stan.glm.3 <- update(stan.glm.2, formula = stops ~ crime_rate+ pop_prop+ eth+ crime_rate:eth)
stan.glm.4 <- update(stan.glm.3, formula = stops ~ crime_rate + pop_prop + eth)
stan.glm.5 <- update(stan.glm.4, formula = stops ~ crime_rate + pop_prop)

```

```

#checking if adding/changing stuff will make the model better
stan.glm.6 <- stan_glm(data = stop.3,
                        formula = stops ~ past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop:eth + past
                        family = neg_binomial_2, #how do I justify not having a prior?
                        seed = 360,
                        refresh = 0)
stan.glm.7 <- update(stan.glm.6, formula = stops ~ crime_rate+ pop+ eth+ crime_rate:eth+ pop:eth +crime_
stan.glm.8 <- update(stan.glm.6, formula = stops ~ past.arrests+ pop+ eth+ past.arrests:eth+ pop:eth + p
stan.glm.9 <- update(stan.glm.6, formula = stops~past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop)

```

```

#posterior predictive check
loo1 <- loo(stan.glm.1, save_psis = TRUE)

```

```

## Warning: Found 3 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

```

```

loo2 <- loo(stan.glm.2, save_psis = TRUE)

```

```

## Warning: Found 2 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

```

```

loo3 <- loo(stan.glm.3, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo4 <- loo(stan.glm.4, save_psis = TRUE)

## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo5 <- loo(stan.glm.5, save_psis = TRUE)
loo6 <- loo(stan.glm.6, save_psis = TRUE)
loo7 <- loo(stan.glm.7, save_psis = TRUE)

## Warning: Found 2 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo8 <- loo(stan.glm.8, save_psis = TRUE)

## Warning: Found 2 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument

loo9 <- loo(stan.glm.9, save_psis = TRUE)

rstanarm::loo_compare(loo1, loo2, loo3, loo4, loo5, loo6, loo7, loo8, loo9)

##          elpd_diff se_diff
## stan.glm.8    0.0     0.0
## stan.glm.7   -4.9     2.7
## stan.glm.2  -13.9     6.7
## stan.glm.1  -14.3     6.7
## stan.glm.6  -15.5     6.5
## stan.glm.9  -15.7     6.4
## stan.glm.4  -20.5     6.9
## stan.glm.3  -20.8     6.8
## stan.glm.5 -40.6     9.3

```

```

stan.glm.1 <- stan_glm(data = stop.4,
                        formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:eth + crime_ra
                        family = neg_binomial_2,
                        seed = 360,
                        prior = cauchy(0, 2.5),
                        prior_intercept = cauchy(0, 2.5),
                        refresh = 0)

```

```

#checking if omitting stuff will make the model better
stan.glm.2 <- update(stan.glm.1, formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth + pop_prop:
stan.glm.3 <- update(stan.glm.2, formula = stops ~ crime_rate + pop_prop + eth + crime_rate:eth)
stan.glm.4 <- update(stan.glm.3, formula = stops ~ crime_rate + pop_prop + eth)
stan.glm.5 <- update(stan.glm.4, formula = stops ~ crime_rate + pop_prop)

```

```
#checking if adding/changing stuff will make the model better
stan.glm.6 <- stan_glm(data = stop.4,
                        formula = stops ~ past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop:eth + past
                        family = neg_binomial_2, #how do I justify not having a prior?
                        seed = 360,
                        refresh = 0)
stan.glm.7 <- update(stan.glm.6, formula = stops ~ crime_rate+ pop+ eth+ crime_rate:eth+ pop:eth +crime_
stan.glm.8 <- update(stan.glm.6, formula = stops ~ past.arrests+ pop+ eth+ past.arrests:eth+ pop:eth + )
stan.glm.9 <- update(stan.glm.6, formula = stops~past.arrests+ pop_prop+ eth+ past.arrests:eth+ pop_prop
```

```
#posterior predictive check
loo1 <- loo(stan.glm.1, save_psis = TRUE)
loo2 <- loo(stan.glm.2, save_psis = TRUE)
```

Drug Drimes

```
## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument
```

```
loo3 <- loo(stan.glm.3, save_psis = TRUE)
loo4 <- loo(stan.glm.4, save_psis = TRUE)
loo5 <- loo(stan.glm.5, save_psis = TRUE)
loo6 <- loo(stan.glm.6, save_psis = TRUE)
loo7 <- loo(stan.glm.7, save_psis = TRUE)
loo8 <- loo(stan.glm.8, save_psis = TRUE)
```

```
## Warning: Found 1 observation(s) with a pareto_k > 0.7. We recommend calling 'loo' again with argument
```

```
loo9 <- loo(stan.glm.9, save_psis = TRUE)
```

```
rstanarm::loo_compare(loo1, loo2, loo3, loo4, loo5, loo6, loo7, loo8, loo9)
```

	elpd_diff	se_diff
## stan.glm.6	0.0	0.0
## stan.glm.8	-4.7	7.7
## stan.glm.9	-7.3	3.8
## stan.glm.7	-14.3	7.4
## stan.glm.2	-15.4	5.7
## stan.glm.1	-15.7	5.6
## stan.glm.4	-16.3	5.6
## stan.glm.3	-16.5	5.7
## stan.glm.5	-56.3	8.8