STAT 447: Project

Kevin Liu (94200474)

April 19, 2025

In this report, I will be approaching the Vancouver Police Department crime database. The database contains crime data from 2013 until present, with columns for type, time, and location. The database can be found on the VPD crime statistics page. I chose this data set as it is the same one I used in STAT 201, analyzing how COVID-19 affected the proportion of Commercial Break and Enter crimes.

I will be applying a simple and improved model to the annual Commercial Break and Enter crimes. Finding a model that can accurately predict this data would be useful, as it would allow people to see a trend (hopefully downward) in crime statistics and identify anomalies.

Further information on Vancouver crime statistics can be found here, with live updates and information on how crimes are identified.

```
set.seed(1)
crime_data = read.csv("crime_data/crime_data.csv")
head(crime_data)
```

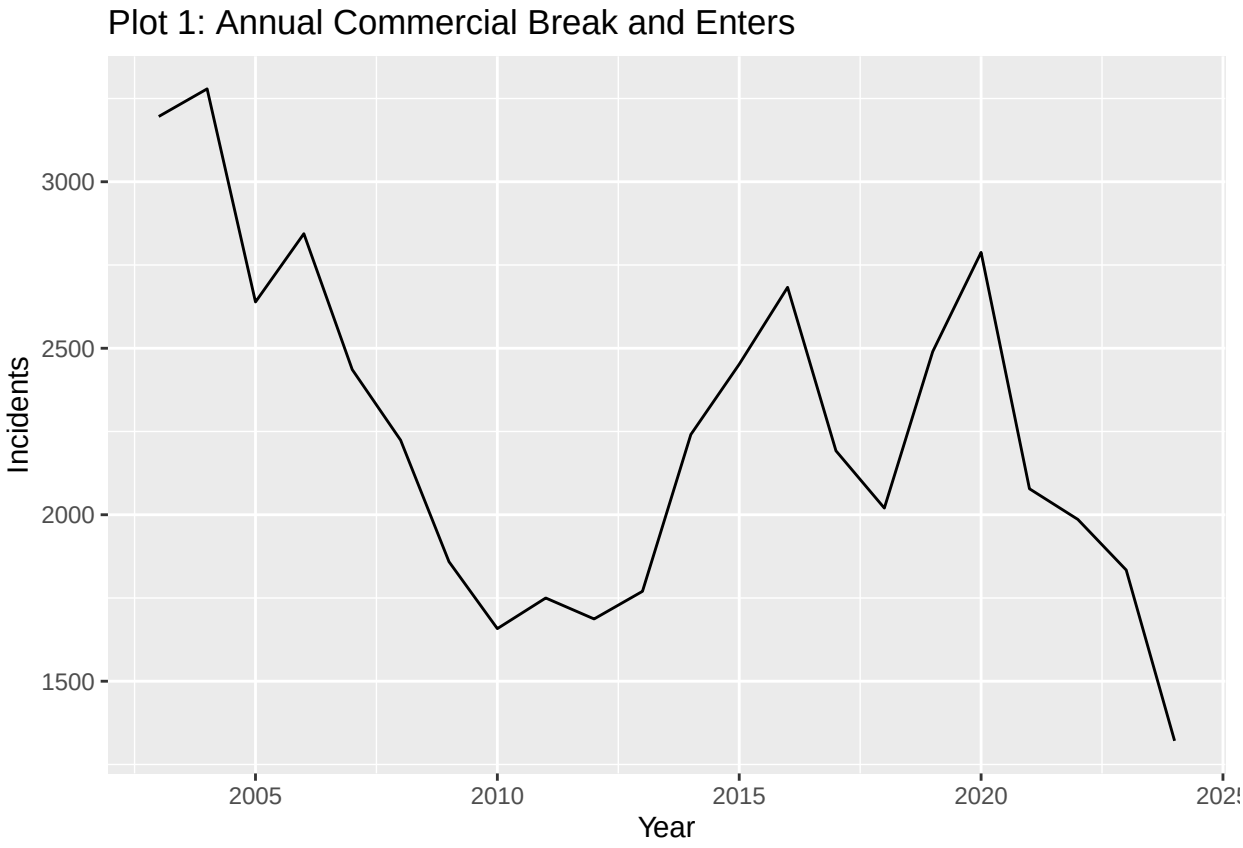
```
##                          TYPE YEAR MONTH DAY HOUR MINUTE   HUNDRED_BLOCK
## 1 Break and Enter Commercial 2012    12  14    8     52
## 2 Break and Enter Commercial 2019     3   7    2      6   10XX SITKA SQ
## 3 Break and Enter Commercial 2019     8  27    4     12 10XX ALBERNI ST
## 4 Break and Enter Commercial 2021     4  26    4     44 10XX ALBERNI ST
## 5 Break and Enter Commercial 2014     8   8    5     13 10XX ALBERNI ST
## 6 Break and Enter Commercial 2020     7  28   19     12 10XX ALBERNI ST
##   NEIGHBOURHOOD        X       Y
## 1      Oakridge 491285.0 5453433
## 2      Fairview 490613.0 5457110
## 3      West End 491004.8 5459177
## 4      West End 491007.8 5459174
## 5      West End 491015.9 5459166
## 6      West End 491015.9 5459166
```

First, I will clean and filter the data. I will remove 2025 from the table as the year is incomplete.

```
crime_data = crime_data %>%
  filter(!is.na(YEAR)) %>%
  filter(YEAR != 2025) %>%
  filter(TYPE == "Break and Enter Commercial")

crime_data %>%
  reframe(YEAR = as.integer(YEAR)) %>%
  group_by(YEAR) %>%
  reframe(count = n()) %>%
```

```
  ggplot() +
  geom_line(aes(x = YEAR, y = count)) +
  labs(x = "Year", y = "Incidents", title = "Plot 1: Annual Commercial Break and Enters")
```



From here, I will apply a simple model to the data set using stan. The simple model I will be using is $Count_{BEC} \sim Poisson(\lambda_t)$

$log(\lambda_t) = \alpha + \beta * Year_t$

With $Count_{BEC}$ being the annual number of Commercial Break and Enter crimes. We willcenter YEAR to reduce multicollinearity. The priors we chose for this model are $\alpha = Norm(0, 10)$ and $\beta = Norm(0, 1)$. This is to allow a large range of of values for α, while having a reasonable range for the slope.

```
simple = crime_data %>%
  reframe(YEAR = as.integer(YEAR),
            YEARC = YEAR - mean(YEAR)) %>%
  group_by(YEARC, YEAR) %>%
  reframe(count = n())

simple_data = list(
  N = nrow(simple),
  y = simple$count,
  x = simple$YEARC
)

fit = stan(
```

```
  file = "simple.stan",
  data = simple_data,
  seed = 1,
  refresh = 0
)
```

```
## Warning in readLines(file, warn = TRUE): incomplete final line found on
## 'C:\Users\kevin\OneDrive\Desktop\STAT 447\Project\simple.stan'
```

```
print(fit, pars = c("alpha", "beta"), probs = c(0.025, 0.5, 0.975))
```

```
## Inference for Stan model: anon_model.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##        mean se_mean sd  2.5%   50% 97.5% n_eff Rhat
## alpha  7.72       0  0  7.71  7.72  7.73   930 1.01
## beta  -0.02       0  0 -0.02 -0.02 -0.02  4372 1.00
##
## Samples were drawn using NUTS(diag_e) at Sat Apr 19 23:31:12 2025.
## 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).
```

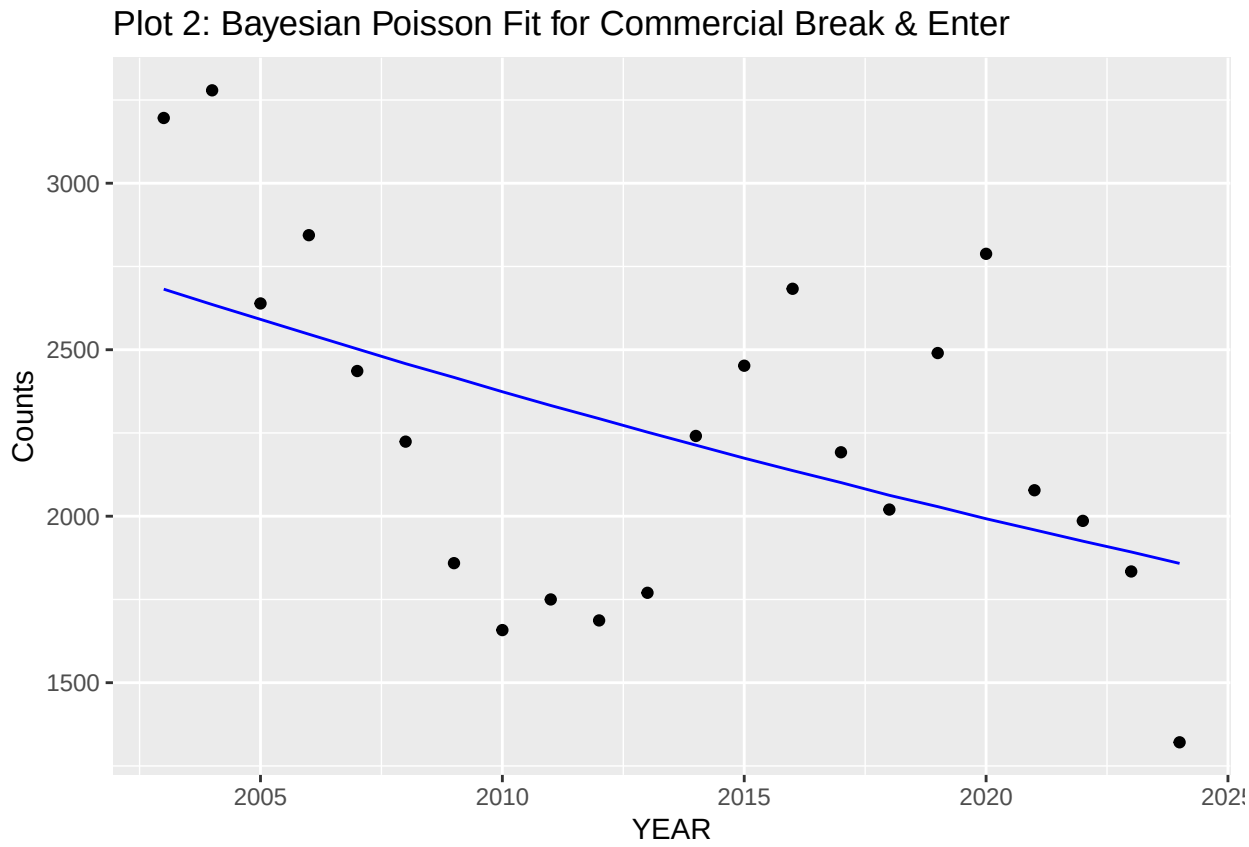
From this we can see that there is an average log crime count of 7.72 with a negative slope of log(-0.02), which is 0.98. We can plot out the predictions to see how well it fits with the actual data.

From the table, we can see that we have small se_mean, large n_eff, and an Rhat close to 1. This means that the model is a good fit.

```
posterior = extract(fit)

simple$pred = colMeans(posterior$y_pred)

ggplot(simple, aes(x = YEAR)) +
  geom_point(aes(y = count), color = "black") +
  geom_line(aes(y = pred), color = "blue") +
  labs(title = "Plot 2: Bayesian Poisson Fit for Commercial Break & Enter",
       y = "Counts")
```



We can attempt to apply an improved model to the data, as linear model may not provide the best predictions for something like crime statistics. We can attempt to use a quadratic model, based on the pattern of the real data points. (The stan model for this section using mu instead of y_pred because in testing the code, errors appeared with the log rate parameter being too high).

$Count_{BEC} \sim Poisson(\lambda_t)$

$log(\lambda_t) = \alpha + \beta 1 * Year_t + \beta 2 * Year_t^2$

```
improved = crime_data %>%
  reframe(YEAR = as.integer(YEAR),
            YEARC = YEAR - mean(YEAR)) %>%
  group_by(YEARC, YEAR) %>%
  reframe(count = n())

improved$YEARC2 = improved$YEARC^2

improved_data = list(
  N = nrow(improved),
  y = improved$count,
  x = improved$YEARC,
  x2 = improved$YEARC2
)

fit = stan(
  file = "improved.stan",
  data = improved_data,
```

```
  seed = 1,
  refresh = 0
)
```

```
## Warning in readLines(file, warn = TRUE): incomplete final line found on
## 'C:\Users\kevin\OneDrive\Desktop\STAT 447\Project\improved.stan'
```

```
## Warning: There were 1 transitions after warmup that exceeded the maximum treedepth. Increase max_tre
## https://mc-stan.org/misc/warnings.html#maximum-treedepth-exceeded
```

```
## Warning: There were 3 chains where the estimated Bayesian Fraction of Missing Information was low. S
## https://mc-stan.org/misc/warnings.html#bfmi-low
```

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

```
## Warning: The largest R-hat is 1.76, indicating chains have not mixed.
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#r-hat
```

```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```
## Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#tail-ess
```

```
print(fit, pars = c("alpha", "beta1", "beta2"), probs = c(0.025, 0.5, 0.975))
```

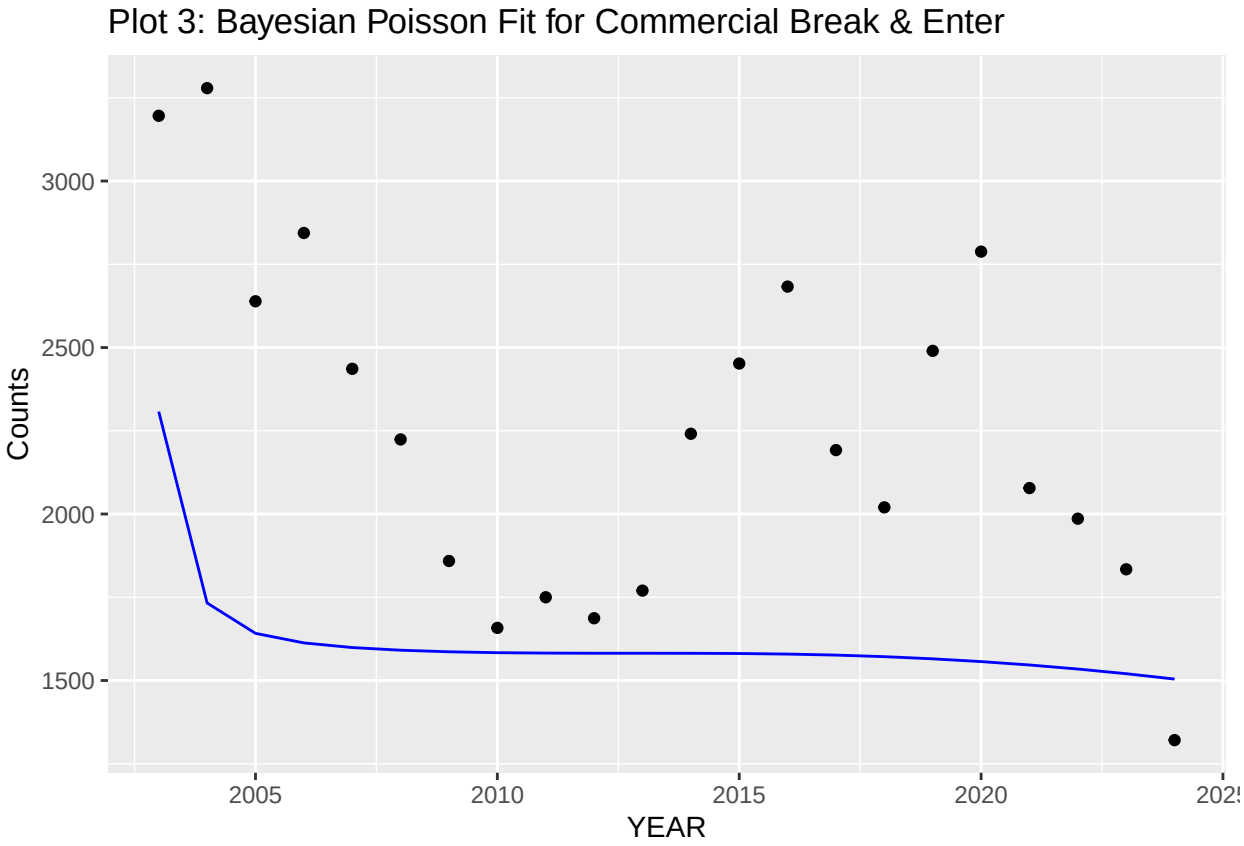
```
## Inference for Stan model: anon_model.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##        mean se_mean   sd  2.5%   50% 97.5% n_eff Rhat
## alpha  6.38    1.50 2.35  0.77  7.66  7.68     2 6.19
## beta1 -0.04    0.02 0.03 -0.11 -0.02 -0.02     2 6.11
## beta2  0.02    0.02 0.03  0.00  0.00  0.07     2 6.13
##
## Samples were drawn using NUTS(diag_e) at Sat Apr 19 23:31:52 2025.
## 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).
```

From the table, we can see an se_mean that is not close to 0 for alpha, a small n_eff, and a large Rhat. This means that the quadratic model is a poor fit for the crime statistics. This means that the quadratic term was unnecessary and the linear model was sufficient. We can plot this out to see how poor of a fit it is.

```r
posterior = extract(fit)

improved$pred = colMeans(posterior$mu)

ggplot(improved, aes(x = YEAR)) +
  geom_point(aes(y = count), color = "black") +
  geom_line(aes(y = pred), color = "blue") +
  labs(title = "Plot 3: Bayesian Poisson Fit for Commercial Break & Enter",
       y = "Counts")
```



In conclusion, we can see that the Commercial Break and Enter crime statistics can be represented with a simple Poisson model of $Crime_{BEC} \sim Poisson(\lambda_t), log(\lambda_t) = 7.72 - 0.02 * Year_t$. The model generally captures the trend of crime, however it does not cover potential seasonality/abnormalities.

Some key limitations would be the number of observations; using the annual crime statistics gives us only 22 observations, which may be a limiting factor. Further investigation may want to use monthly crime statistics, with the model potentially including a variable for the month. Additionally, for something like crime statistics, it would be difficult to incorporate all factors into prediction, as it is affected by a large number of things. For example, we can see a dip in crime in 2020, which contributed to the shape that suggested a quadratic model. However, this dip was not likely not due to year, as this was when COVID-19 first popped up.