Future Protest Projections: Exploring Variables for 2025 Predictions

April 8, 2024

STAT 413: Adam Kashlak

Phong Ho [1617975], Randal (R.J.) Bilak [1584507], Tristin Schlauch [1672573]

In this project, we use many statistical methods to predict the number of protests that will happen in 2025 using the data provided. First, we begin by figuring out which factors are important for predicting protests by testing them via a method called parametric bootstrapping. Then, we will begin to use another method called Monte Carlo to make a 95% prediction interval about how many protests we might see. By doing this, we hope to make a good estimate of future protests and help us prepare for what might happen in 2025.

	X	year	month	prov	pop	protests
	<int>	<int>	<chr>	<chr>	<int>	<int>
1	1	2023	November	Alberta	4756408	20
2	2	2023	November	British Columbia	5581127	27
3	3	2023	November	Manitoba	1465440	10
4	4	2023	November	New Brunswick	842725	5
5	5	2023	November	Newfoundland and Labrador	540418	7
6	6	2023	November	Northwest Territories	44760	2

In the dataset, we have a total of 5 variables, one of which we aim to gain deeper insights into. The objective of this project is to create a prediction model, focusing on the variable "Protest," to forecast the number of protests that might occur in any given year with a certain level of confidence.

Firstly, we must undertake data cleaning and preprocessing. By using the head() function, it's evident that the "year" variable should ideally consist of categories representing different years rather than integers. Similarly, "month" and "provinces" are expected to be categorical rather than simple strings. To rectify this, we will proceed with the following procedures.

Pre-Processing

First of the id section is to be removed as it adds redundancy. For variables that are meant to be in a category we may use the function

as.factor()

and for the variables we wish to define as numeric instead of integers, we may use the function

as.numeric()

	year	month	prov	pop	protests
	<fct>	<fct>	<fct>	<dbl>	<dbl>
1	2023	November	Alberta	15.37500	20
2	2023	November	British Columbia	15.53490	27
3	2023	November	Manitoba	14.19767	10
4	2023	November	New Brunswick	13.64440	5
5	2023	November	Newfoundland and Labrador	13.20010	7
6	2023	November	Northwest Territories	10.70907	2

We may also mention that the population has values which are on a higher range of numbers compared to protest. So scaling down the population variable will make sure all the numeric in the data are on the same scale, which is important because it helps with comparing them easily and understanding the results better. It also makes the process of calculation more smooth. Another important reason to scale is that we are sure when population is 0, we expect the number of protests to be 0 which the function *log()* allows us to do.

We also want to look at how seasons affect protests compared to just looking at individual months. While focusing on seasons gives us a general idea of yearly trends, it means we lose some specific monthly details and might not be totally accurate because seasons can vary in length. When we leave out months from our model, we end up with a weird situation where the relationship between population and protests is inversly proportional, I.e., as population increases, our prediction of protests decrease. So, keeping months in our model is vital to preserve all necessary information.

```
glm(formula = protests ~ year + seasons + prov + pop, family = poisson(link = "log"),
    data = df)

Coefficients:
                               Estimate Std. Error z value Pr(>|z|)
(Intercept)                  108.36033   44.06429   2.459 0.013927 *
year2023                       0.22900    0.08222   2.785 0.005351 **
seasonsSpring                 -0.16426    0.06210  -2.645 0.008168 **
seasonsSummer                 -0.55482    0.05491 -10.103  < 2e-16 ***
seasonsWinter                 -0.30459    0.06443  -4.728 2.27e-06 ***
provBritish Columbia           1.74739    0.48382   3.612 0.000304 ***
provManitoba                  -8.39432    3.35008  -2.506 0.012221 *
provNew Brunswick            -12.69997    4.95171  -2.565 0.010325 *
provNewfoundland and Labrador -15.97063    6.17935  -2.585 0.009752 **
provNorthwest Territories    -35.10391   13.31026  -2.637 0.008355 **
provNova Scotia              -10.94779    4.27275  -2.562 0.010400 *
provNunavut                  -35.40430   13.59235  -2.605 0.009195 **
provOntario                    9.60874    3.46953   2.769 0.005615 **
provPrince Edward Island     -24.73423    9.47767  -2.610 0.009061 **
provQuebec                     5.02131    1.85971   2.700 0.006933 **
provSaskatchewan             -10.25272    3.87380  -2.647 0.008128 **
provYukon                    -33.96570   13.34008  -2.546 0.010892 *
pop                           -6.88343    2.87356  -2.395 0.016601 *
```

1 Model Selection

Upon completing the data preparation phase, we are poised to select a suitable model. Our choice falls upon *Poisson Regression*, aimed at forecasting protest occurrences. Our model is comprised of four predictors: year, month, province, and population size. Leveraging historical data, these factors will aid in forecasting the frequency of protests in the future. Subsequently, after training the model, we will assess the significance of each variable and interpret the results of factors influencing protests.

Fundamental of Poisson Regression:

Poisson Regression models the expected count of events Y as a function of predictor variables X using the Poisson distribution:

$$Y \sim Poisson(\lambda)$$

Where λ represents the expected count of events. The relationship between the predictor variables and λ is modeled using the logarithm link function:

$$log(\lambda) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k$$

where $\beta_0, \beta_1, ..., \beta_k$ are the coefficients estimated by the regression and $X_1, X_2, ..., X_k$ are the predictors. The interpretation of the coefficients is in terms of relative changes in the expected count of events, λ. For example, if β_1 is positive, it indicates that for a one-unit increase in X_1 , the expected count of events increases/decreases by a factor of e^{β_1} with differences up to a sign change of β_1 , after holding other variables constant.

Poisson Regression model:

```
glm(formula = protests ~ year + month + prov + pop, family = poisson(link = "log"),
    data = df)

Coefficients:
                               Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -177.86192   74.54008  -2.386 0.017027 *
year2023                       -0.29868    0.13766  -2.170 0.030034 *
monthAugust                    -0.75206    0.09666  -7.780 7.23e-15 ***
monthDecember                  -0.73557    0.13446  -5.471 4.48e-08 ***
monthFebruary                   0.09703    0.07590   1.278 0.201140
monthJanuary                   -0.30972    0.08397  -3.688 0.000226 ***
monthJuly                      -0.63486    0.09365  -6.779 1.21e-11 ***
monthJune                      -0.33059    0.08008  -4.128 3.65e-05 ***
monthMarch                     -0.05841    0.07870  -0.742 0.457968
monthMay                       -0.08094    0.07475  -1.083 0.278928
monthNovember                  -0.27110    0.11281  -2.403 0.016257 *
monthOctober                   -0.22354    0.11222  -1.992 0.046377 *
monthSeptember                 -0.03334    0.08178  -0.408 0.683506
provBritish Columbia           -1.36078    0.81139  -1.677 0.093524 .
provManitoba                   13.37793    5.67332   2.358 0.018372 *
provNew Brunswick              19.48336    8.38549   2.323 0.020154 *
provNewfoundland and Labrador  24.17999   10.45747   2.312 0.020766 *
provNorthwest Territories      51.35224   22.51260   2.281 0.022546 *
provNova Scotia                16.82176    7.23567   2.325 0.020081 *
provNunavut                    52.90702   22.99951   2.300 0.021428 *
provOntario                   -12.93414    5.87155  -2.203 0.027605 *
provPrince Edward Island       36.85816   16.04521   2.297 0.021611 *
provQuebec                     -7.05514    3.14611  -2.242 0.024929 *
provSaskatchewan               14.92069    6.55924   2.275 0.022920 *
provYukon                      52.74033   22.58829   2.335 0.019551 *
pop                            11.79415    4.86547   2.424 0.015348 *
```

From the summary function, it's evident that when selecting a significance level of $\alpha = 0.05$, the variables: provBritish Columbia, monthSeptember, monthMay, monthMarch, and monthFebruary are deemed non-significant, meaning that we fail to reject the hypothesis:

$$H_0 : \beta_k = 0 \text{ vs. } H_1 : \beta_k \neq 0$$

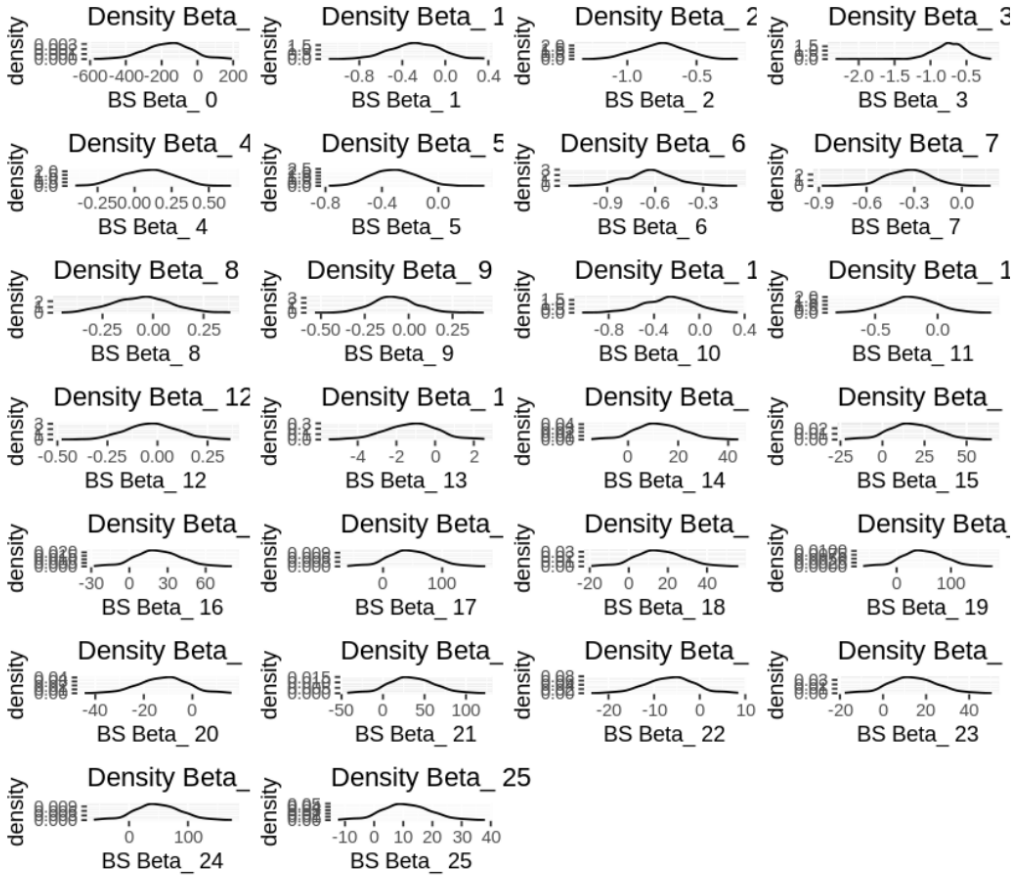
Conversely, the remaining variarexhibit signticance. Nonetheless, we intend to conduct another tedetermineertain the significance of each individual parameter through parametric bootstrapping. In the subsequent code snippet, a Poisson regression model is fitted to the data. This model estimates coefficients for the predictors (year, seasons, prov, pop). In this parametric bootstrap approach, rather than directly resampling from the dataframe, resampling is based on simulated data generated from the fitted Poisson regression model. By repeatedly fitting the model to these resampled datasets and calculating parameter coefficients, we can leverage the bootstrap parameters to estimate the sampling variability of the model parameters, construct confidence interandls, or assess uncertainty regarding the parameter estimates.

2 Bootstrapping Parameters

Parametric bootstrapping for a Poisson regression model involves several key steps to estimate the uncertainty associated with the model's parameters. Initially, the Poisson regression model is fitted to the original dataset, establishing the relationship between predictors and the outcome variable, such as protest counts. Then, new datasets are generated through simulation, using the estimated parameters from the original model. Each new dataset is created by sampling from a Poisson distribution, with mean values equal to the predicted counts from the original model. Subsequently, the Poisson regression model is refitted to each of these resampled datasets, resulting in parameter estimates for each iteration. By repeating this process numerous times, a distribution of parameter estimates is obtained. From this distribution, standard errors and confidence intervals for each parameter can be calculated, providing insights into the uncertainty surrounding the model's estimates.

We used code to conducts bootstrap resampling to estimate confidence intervals for the parameters. Initially, we utilize the boot.ci() function from the boot package in R to compute percentile-based bootstrap confidence intervals (type = "perc") for the first parameter (index = 1) of the model. Then, the results are stored in a list named boot_ci_list. We iterate this process over an index = i where i ranges from 1 to 26, and print the outcomes. This procedure provides us with a 95% confidence interval for each parameter, where Beta 0 represents the intercept.

```
Parameter: Beta_ 0
Bootstrap CI (95%): -424.5769 - 99.35159

Parameter: Beta_ 1
Bootstrap CI (95%): -0.7412705 - 0.1474053

Parameter: Beta_ 2
Bootstrap CI (95%): -1.100472 - -0.431664

Parameter: Beta_ 3
Bootstrap CI (95%): -1.198947 - -0.3446166

Parameter: Beta_ 4
Bootstrap CI (95%): -0.2281949 - 0.4021107

Parameter: Beta_ 5
Bootstrap CI (95%): -0.6010649 - -0.0034242

Parameter: Beta_ 6
Bootstrap CI (95%): -0.9348345 - -0.3167158

Parameter: Beta_ 7
Bootstrap CI (95%): -0.6447007 - -0.0672396

Parameter: Beta_ 8
Bootstrap CI (95%): -0.3248727 - 0.2291564

Parameter: Beta_ 9
Bootstrap CI (95%): -0.3228818 - 0.178325

Parameter: Beta_ 10
Bootstrap CI (95%): -0.7513795 - 0.1624772

Parameter: Beta_ 11
Bootstrap CI (95%): -0.6099544 - 0.1777668

Parameter: Beta_ 12
Bootstrap CI (95%): -0.269729 - 0.242004

Parameter: Beta_ 13
Bootstrap CI (95%): -4.072439 - 1.550148

Parameter: Beta_ 14
Bootstrap CI (95%): -7.644765 - 32.08168

Parameter: Beta_ 15
Bootstrap CI (95%): -11.62437 - 47.28539

Parameter: Beta_ 16
Bootstrap CI (95%): -14.83718 - 58.72715

Parameter: Beta_ 17
Bootstrap CI (95%): -32.69142 - 125.5443

Parameter: Beta_ 18
Bootstrap CI (95%): -10.06934 - 40.71432

Parameter: Beta_ 19
Bootstrap CI (95%): -32.55113 - 129.0511

Parameter: Beta_ 20
Bootstrap CI (95%): -32.28252 - 8.855624

Parameter: Beta_ 21
Bootstrap CI (95%): -22.76747 - 89.8804

Parameter: Beta_ 22
Bootstrap CI (95%): -17.37451 - 4.583404

Parameter: Beta_ 23
Bootstrap CI (95%): -9.431862 - 36.66875

Parameter: Beta_ 24
Bootstrap CI (95%): -31.48615 - 127.5534

Parameter: Beta_ 25
Bootstrap CI (95%): -6.298036 - 27.89207
```



We can see this matrix plot of all bootstrapped parameter and see where 0 falls within the density plot to determine if its significant or not.

So according to the bootstrapped confidence interval, the parameters: $\beta_2, \beta_3, \beta_5, \beta_6, \beta_7$ are the only significant parameters which corresponds to the monthAugust, monthDecember, monthJanurary, monthJuly, monthJune, monthMarch variable. This is very different compared to what the summary Z-test has given us. Why?

The answer is that the issue with using Z-scores in Poisson Regression is the assumption of equidispersion where the $Y \sim \text{Poisson}(\lambda)$ then $EY = \lambda$ and $\text{Var}(Y) = \lambda$ are all equally the same ie, The model assumes that the variance of the **protest** is equal to the mean. However, in most realistic data it often exhibit greater variability, an overdispersion, than what would be expected under the Poisson distribution.

When overdispersion occurs, the assumption of equidispersion is violated, and the standard errors of the coefficients estimated by the Poisson Regression may be underestimated. As a result, Z-scores calculated using these standard errors may be inaccurate, leading to potentially misleading conclusions about the significance of predictor variables. Which we see is evident for example in the Population variable or **Beta_25**.

In context to Poisson Regression, alternative methods such as quasi-Poisson (Overdispersed Poisson Regression) or Negative Binomial Regression are often employed. These methods explicitly account for overdispersion by allowing the variance to exceed the mean. In such models, the significance and interpretation of predictor variables, including the population variable, may differ from those in standard Poisson Regression due to the adjustments made to accommodate overdispersion. In the Overdispersed Poisson Regression case, we add a "dispersion" parameter $\phi > 0$. Then, our random variable Y will have $\mathrm{E}Z = \lambda$ and $\mathrm{Var}(Y) = \phi\lambda$. Just like the bootstrap, we can see the difference in parameter significance in the code below which almost lines up with bootstrapping significant parameters.

Overdispersed poisson regression model:

```
glm(formula = protests ~ year + month + prov + pop, family = quasipoisson(link = "log"),
    data = df)

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                 -177.86192  115.63769  -1.538 0.125183
year2023                      -0.29868    0.21356  -1.399 0.163083
monthAugust                   -0.75206    0.14996  -5.015 9.55e-07 ***
monthDecember                 -0.73557    0.20859  -3.526 0.000494 ***
monthFebruary                  0.09703    0.11775   0.824 0.410659
monthJanuary                  -0.30972    0.13027  -2.378 0.018118 *
monthJuly                     -0.63486    0.14528  -4.370 1.77e-05 ***
monthJune                     -0.33059    0.12423  -2.661 0.008248 **
monthMarch                    -0.05841    0.12209  -0.478 0.632734
monthMay                      -0.08094    0.11597  -0.698 0.485816
monthNovember                 -0.27110    0.17501  -1.549 0.122530
monthOctober                  -0.22354    0.17409  -1.284 0.200227
monthSeptember                -0.03334    0.12686  -0.263 0.792908
provBritish Columbia          -1.36078    1.25875  -1.081 0.280626
provManitoba                  13.37793    8.80130   1.520 0.129670
provNew Brunswick             19.48336   13.00882   1.498 0.135366
provNewfoundland and Labrador 24.17999   16.22319   1.490 0.137258
provNorthwest Territories     51.35224   34.92491   1.470 0.142615
provNova Scotia               16.82176   11.22506   1.499 0.135135
provNunavut                   52.90702   35.68028   1.483 0.139279
provOntario                  -12.93414    9.10882  -1.420 0.156761
provPrince Edward Island      36.85816   24.89173   1.481 0.139829
provQuebec                    -7.05514    4.88072  -1.446 0.149461
provSaskatchewan              14.92069   10.17567   1.466 0.143714
provYukon                     52.74033   35.04233   1.505 0.133468
pop                           11.79415    7.54804   1.563 0.119318
```

3 Monte Carlo Prediction Interval

In this section of the project, we are using a method called Monte Carlo simulation to create 95% prediction bands for each province in 2025 for a given month of the year. Monte Carlo simulation is a way to generate several guesses based on a model. By doing this for each province, we can predict not just one outcome, but a range of possible outcomes, considering how uncertain things might be. This may help us make better plans and decisions for the upcoming year. Combining Monte

Carlo simulation with prediction modeling lets us give more accurate predictions about what might happen in each province in 2025.

Before progressing to simulate from our model, we must make a change to year from *as.factor()* to *as.numeric()* because if we were to use categories like "2022" and "2023" for the "year" variable, the computer sees only those specific years and treats them as the only thing in the universe of that variable. This limited view might make it hard for the model to understand how things change over time, as it does not recognize "2025". But if we use numbers instead, the model can see the whole timeline from one year to the next. This helps it notice any trends or patterns happening over the years. So, by using numberical values, we make it easier for the model to predict what might happen in 2025 because it can understand the bigger picture of how things change over time.

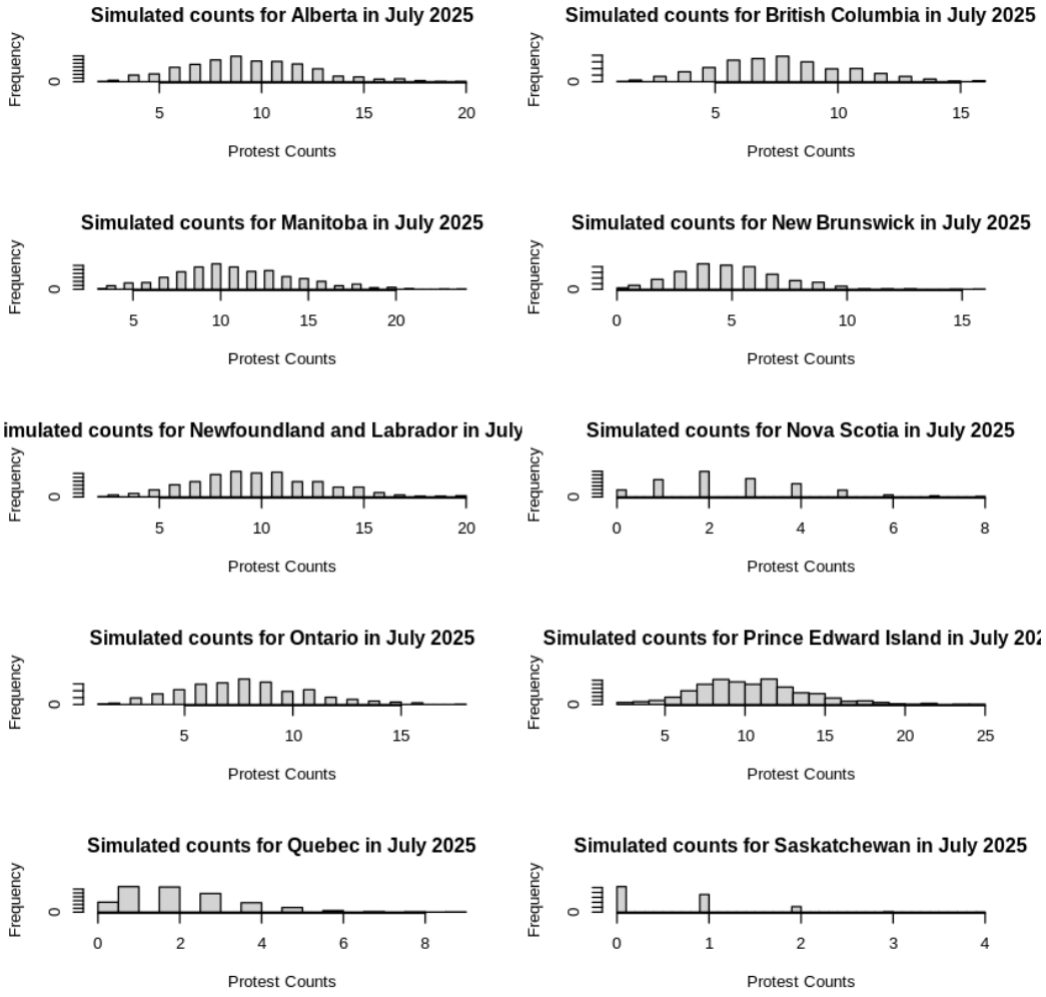
To conduct this we use the code which consists of two functions tailored for predicting and visualizing predicted protests using a Poisson regression model. The *predict.protests* function accepts parameters such as province, month, year, population, and the number of simulations. It proceeds by fitting a Poisson regression model, predicting the expected number of protests, generating simulated protest counts based on the predicted lambda, calculating 95% prediction bands, and presenting summary statistics. On the other hand, the plot.hist function serves a similar purpose but emphasizes plotting, generating a histogram of the simulated protest counts for a specified province, month, and year. These functions enable us to forecast future protest counts and visualize their distribution via histograms.

To predict protest numbers for 2025, we need to input additional variables into our model. These include the month, population, and province, though we're examining all provinces regardless. For consistency, we've chosen July, as it holds significance with Canada's birthdate. However, determining population figures required careful consideration. Large deviations from original data could skew our predictions significantly. To address this, we opted for the mean logarithm of each province's population, rounding up to the nearest whole number. This method assumes a moderate increase in population across all provinces, approximating a rise by a factor of about $\approx e^x, 0 \leq x \leq 1$. This ensures our predictions remain plausible, while allowing for variability in population growth.

prov	pop
Alberta	15.33744
British Columbia	15.50404
Manitoba	14.17198
New Brunswick	13.61455
Newfoundland and Labrador	13.18798
Northwest Territories	10.70852
Nova Scotia	13.85090
Nunavut	10.60930
Ontario	16.54453
Prince Edward Island	12.03984
Quebec	15.98423
Saskatchewan	13.98985
Yukon	10.69555

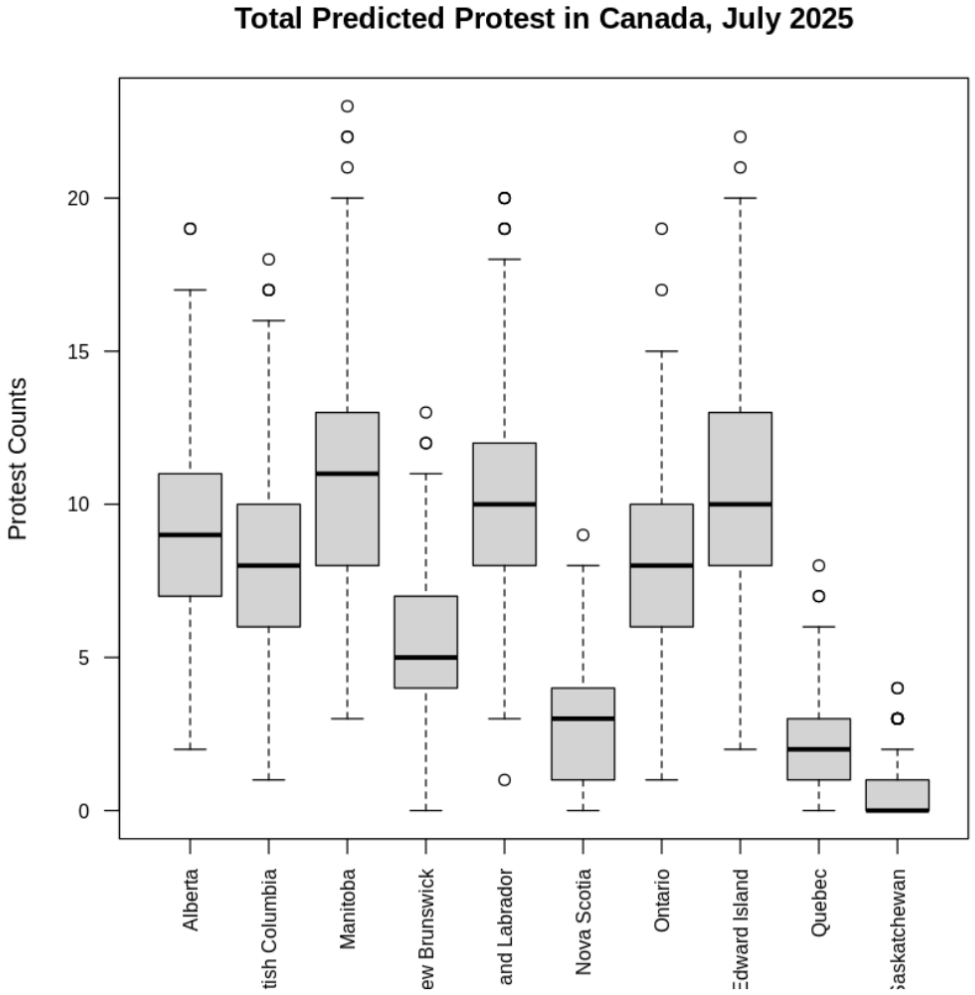
We then apply our code with a loop on all the pronvinces and their respecitive population in July 2025. The outcome is given as a plotted histogram and a summary of the simulation alongside with

- lower/upper bounds of the 95% prediction interval.



```
[1] "----- July 2025 , Alberta (Population: 8886110.52050787 ) -----"
Lower bound: 4
Upper bound: 16
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   7.000   9.000   9.338  11.000  21.000
[1] "----- July 2025 , British Columbia (Population: 8886110.52050787 ) -----"
Lower bound: 3
Upper bound: 14
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   6.000   8.000   8.004  10.000  15.000
[1] "----- July 2025 , Manitoba (Population: 3269017.37247211 ) -----"
Lower bound: 5
Upper bound: 18
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    8.00   11.00   10.71   13.00   23.00
[1] "----- July 2025 , New Brunswick (Population: 1202604.28416478 ) -----"
Lower bound: 2
Upper bound: 10
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   4.000   5.000   5.212   6.000  15.000
[1] "----- July 2025 , Newfoundland and Labrador (Population: 1202604.28416478 )
-----"
Lower bound: 4.475
Upper bound: 16.525
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    8.00   10.00    9.94   12.00   22.00
[1] "----- July 2025 , Nova Scotia (Population: 1202604.28416478 ) -----"
Lower bound: 0
Upper bound: 6
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.000   2.000   2.576   3.250   9.000
[1] "----- July 2025 , Ontario (Population: 24154952.7535753 ) -----"
Lower bound: 3
Upper bound: 14
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    6.00    8.00    8.04   10.00   18.00
[1] "----- July 2025 , Prince Edward Island (Population: 442413.39200892 )
-----"
Lower bound: 5
Upper bound: 17
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   3.00    8.00   11.00   10.79   13.00   23.00
[1] "----- July 2025 , Quebec (Population: 8886110.52050787 ) -----"
Lower bound: 0
Upper bound: 6
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.000   2.000   2.322   3.000  10.000
[1] "----- July 2025 , Saskatchewan (Population: 1202604.28416478 ) -----"
Lower bound: 0
Upper bound: 3
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   0.000   0.000   0.604   1.000   4.000
```

A boxplot is made to summarize our findings on one graph.



Based on the summary of findings for the predicted protests in July 2025, each province seems to have varying levels of predicted protest activity. Here's a breakdown of the observations:

- Alberta: Predicted protests range from 4 to 16, with a mean of approximately 9.3.
- British Columbia: Predicted protests range from 3 to 14, with a mean of around 8.
- Manitoba: Predicted protests range from 5 to 18, with a mean close to 10.7.
- New Brunswick: Predicted protests range from 2 to 10, with a mean of about 5.2.
- Newfoundland and Labrador: Predicted protests range from approximately 4.5 to 16.5, with a mean close to 9.9.
- Nova Scotia: Predicted protests range from 0 to 6, with a mean of about 2.6.
- Ontario: Predicted protests range from 3 to 14, with a mean close to 8.
- Prince Edward Island: Predicted protests range from 5 to 17, with a mean close to 10.8.
- Quebec: Predicted protests range from 0 to 6, with a mean of approximately 2.3.
- Saskatchewan: Predicted protests range from 0 to 3, with a mean around 0.6.

From these results, it's evident that certain provinces, such as Manitoba and Prince Edward Island, exhibit higher predicted protest counts compared to others like Nova Scotia and Saskatchewan. The predicted protest counts vary based on factors such as population size, historical data trends, and possibly other unobserved variables specific to each province. These insights could be valuable for understanding and potentially addressing social or political tensions across different regions of Canada.

As a final point, we see that Monte Carlo sampling helps us understand how certain or uncertain our predictions are by trying out lots of different scenarios. This helps us see how our data and our model might vary. We use it to figure out a range where we think the actual values might fall, like the 95% bounds we calculated. But there are some things to watch out for. First, our predictions are only as good as the data we used to make them. If our data isn't very good, our predictions might not be either. Second, the assumptions we make when using our model might not always be true in real life. For example, we might assume that certain things are related in a certain way, but that might not be the case. Finally, Monte Carlo sampling assumes that we know all the details of our model perfectly, which might not be true in practice. So, while it's a useful tool, we need to be careful when interpreting the results and remember the limitations of both our model and our data. One of these assumptions is that the variation in our data is consistent across all levels of our predictors, which is called equidispersion. However, this might not always hold true in real-world situations. Additionally, Incorporating cross-validation techniques can enhance the reliability and generalizability of the predictions in this project. Cross-validation involves splitting the dataset into multiple subsets, training the model on a portion of the data, and then evaluating its performance on the remaining unseen data. This process helps assess how well the model performs on new data and can provide insights into its robustness and potential for overfitting.

This research process has been both rewarding and insightful. Through the application of statistical techniques and data analysis, we have gained valuable insights into the dynamics of protest activity in Canada, laying the groundwork for further exploration and refinement of predictive models in this domain. Moving forward, continued efforts to improve model accuracy and robustness will be essential for enhancing the utility of such predictions in informing decision-making and policy development. Overall, our study underscores the complexity of modeling social phenomena like protests, while also highlighting the potential benefits of employing rigorous statistical methods to better understand and anticipate these dynamics.