In [7]:
```
library(tidyverse)
library(repr)
library(digest)
library(gridExtra)
```

In [58]:
```
data<-read_csv("data/crimedata_2022.csv")

data <- data %>%
    filter(NEIGHBOURHOOD != "NA")%>%
    mutate(type = as.factor(TYPE),
           year = as.factor(YEAR),
           month = as.factor(MONTH),
           day = as.factor(DAY),
           neighbourhood = as.factor(NEIGHBOURHOOD)) %>%
    select(type,year,month,day,neighbourhood)
#head(data)

#a.summation by neighbouhood, so we have daily number of crimes for
#each day of each month of 2022
daily_crime <- data %>%
    group_by(neighbourhood,day,month) %>%
    summarize(crimes_per_day = n())
#head(daily_crime)

#a.1 Visualization
options(repr.plot.width = 20, repr.plot.height = 20)
daily_crime_by_neighbourhoods <- daily_crime %>%
    ggplot(aes(x=crimes_per_day)) +
    geom_histogram() +
    facet_wrap(~neighbourhood) +
    theme_bw() +
    theme(text = element_text(size=25)) +
    labs(x = "Number of crime incident per day")+
    ggtitle("Daily crime incidents rate by neighbourhood")

daily_crime_by_neighbourhoods

#b.calculate the average of crimes_per_day, so we summed up crimes_
# in 2022 and compute a average, called it the yearly_average_crime
average_daily_crime <- daily_crime %>%
    group_by(neighbourhood) %>%
    summarize(yearly_average_crime_per_day = mean(crimes_per_day))
head(average_daily_crime)

#c.calculate the SD of crimes_per_day for each neighbourhood
daily_crime_sd <- daily_crime %>%
    group_by(neighbourhood) %>%
    summarize(crime_per_day_sd = sd(crimes_per_day))
head(daily_crime_sd)

#Option A
#d.find the two neighbourhoods with least yearly average and the mo

#e.run hypothesis test on the two neighbourhood
```

```
#Option B
#f.aggregate data from 2017~2022, repeat the above steps except thi
# of interest is year

#g.find the two years with the least and most daily average crime

#h.run hypothesis test on the two neighbourhood

#i.construct 95CI for step d or g
```

```
Rows: 4688 Columns: 10
```

```
── Column specification ────────────────────────────────────────────────────────
```

```
Delimiter: ","
chr (3): TYPE, HUNDRED_BLOCK, NEIGHBOURHOOD
dbl (7): YEAR, MONTH, DAY, HOUR, MINUTE, X, Y
```

```
ℹ Use `spec()` to retrieve the full column specification for this
data.
ℹ Specify the column types or set `show_col_types = FALSE` to quie
t this message.

`summarise()` has grouped output by 'neighbourhood', 'day'. You ca
n override using the `.groups` argument.

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

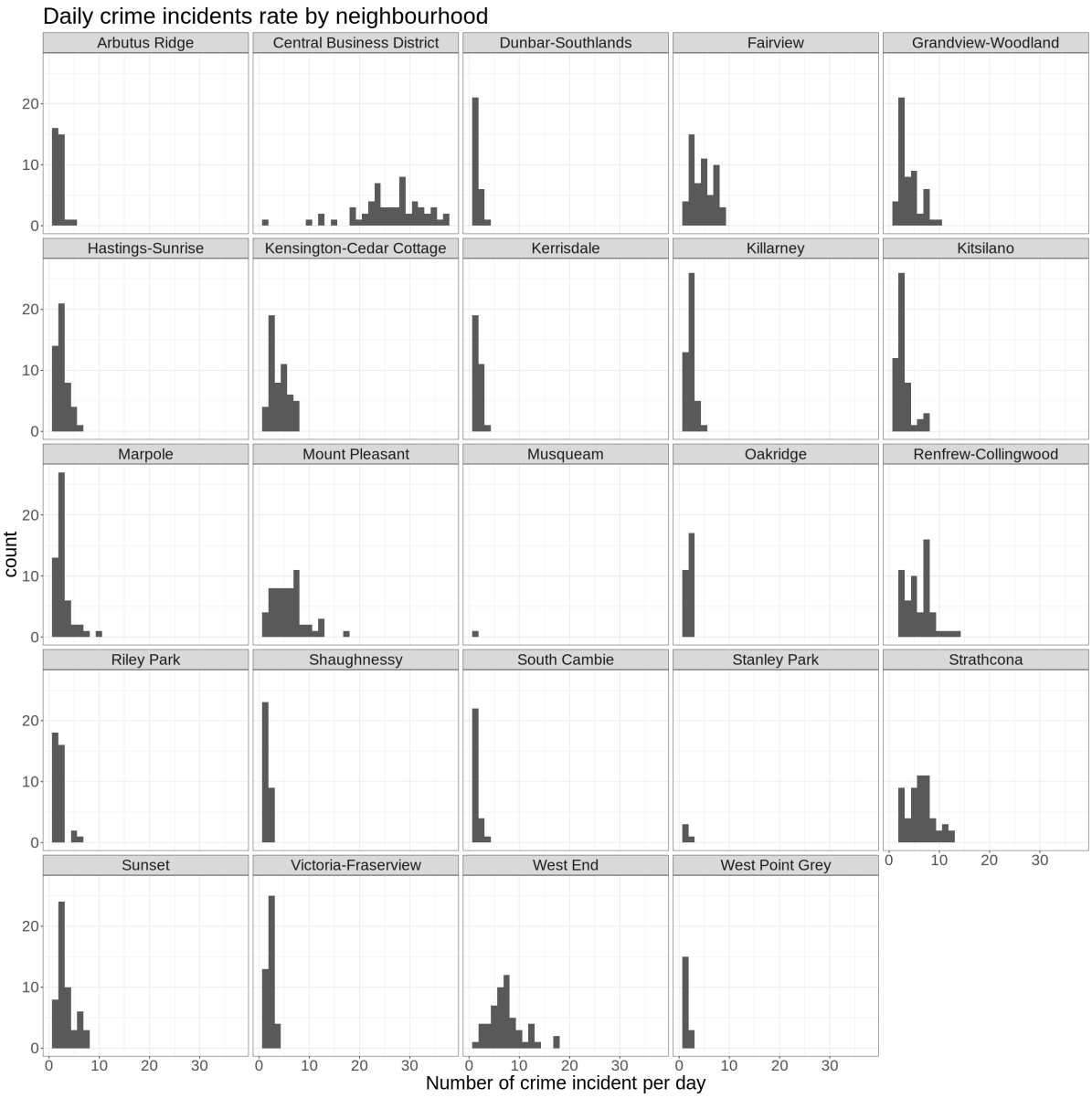
A tibble: 6 × 2

neighbourhood <fct>	yearly_average_crime_per_day <dbl>
Arbutus Ridge	1.757576
Central Business District	25.875000
Dunbar-Southlands	1.357143
Fairview	4.654545
Grandview-Woodland	4.057692
Hastings-Sunrise	2.520833

A tibble: 6 × 2

neighbourhood <fct>	crime_per_day_sd <dbl>
Arbutus Ridge	0.9692234
Central Business District	6.9545603
Dunbar-Southlands	0.7310209
Fairview	2.2045255
Grandview-Woodland	2.1549165

Hastings-Sunrise 1.3989294



Bootstraping

Assumption 1:

we are interested in the true crime rate(daily,monthly,yearly you name it) of Vancouver municipality <- this forms the populaiton of interest

Assumption 2:

we think the data on hand is not the full population, because some crimes may not be reported <- in this case we will bootstrap from the entire sample

Alternative Assumption 2:

We can think the data on hand IS THE POPULATION, then we can take a sample from this data set, then run bootstrap on the sample to obtain a bootstraping distribution and infer a estimate for the standard error of the true sampling distribution. Since we assumed our data is the population, we can verify our bootstrap estimate at the end by comparing it to a sampling distribution we generated from the data

What we can obtain from bootstrapping:

> With Assumption 2, we obtain an estimate of SE of the true sampling distribution formed by sampling from all crimes that ever happened in Vancouver, reported or not. We can then use this to construct a 95CI around our "sample" mean(I'm referring to the entire dataset as our sample), this will be a CI for the true underlying crime rate

> With Alternative Assumption 2, we obtain an estimate of SE of the sampling distribution formed from all crimes that got reported to police in Vancouver. We can then use this to construct a 95CI around our aritificially generated "sample" mean(a sample we randomly taken from the dataset of a sample size we like such as n = 2000), this will be a CI for the average crime rate for the dataset on hand.