Bigvis

BELFADIL Anas

10/22/2019

Setting up the environment

```
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
library(bigvis)
library(ggmap)
```

Importing the Data

```
Uber <- read_csv("/home/anas/Desktop/Courses/DVM/2 R and Big Data/Bigvis/uber-raw-data-sep14.csv")
options(pillar.sigfig = 5)
as_tibble(Uber)
```

```
## # A tibble: 1,028,136 x 4
##    `Date/Time`         Lat     Lon Base
##    <chr>             <dbl>   <dbl> <chr>
##  1 9/1/2014 0:01:00 40.220 -74.002 B02512
##  2 9/1/2014 0:01:00 40.75  -74.003 B02512
##  3 9/1/2014 0:03:00 40.756 -73.986 B02512
##  4 9/1/2014 0:06:00 40.745 -73.989 B02512
##  5 9/1/2014 0:11:00 40.814 -73.944 B02512
##  6 9/1/2014 0:12:00 40.674 -73.992 B02512
##  7 9/1/2014 0:15:00 40.747 -73.647 B02512
##  8 9/1/2014 0:16:00 40.661 -74.269 B02512
##  9 9/1/2014 0:32:00 40.374 -74.000 B02512
## 10 9/1/2014 0:33:00 40.763 -73.977 B02512
## # ... with 1,028,126 more rows
```

Binning and summarizing

First let's check a summary of our data to be able to choose a suitable binning width.

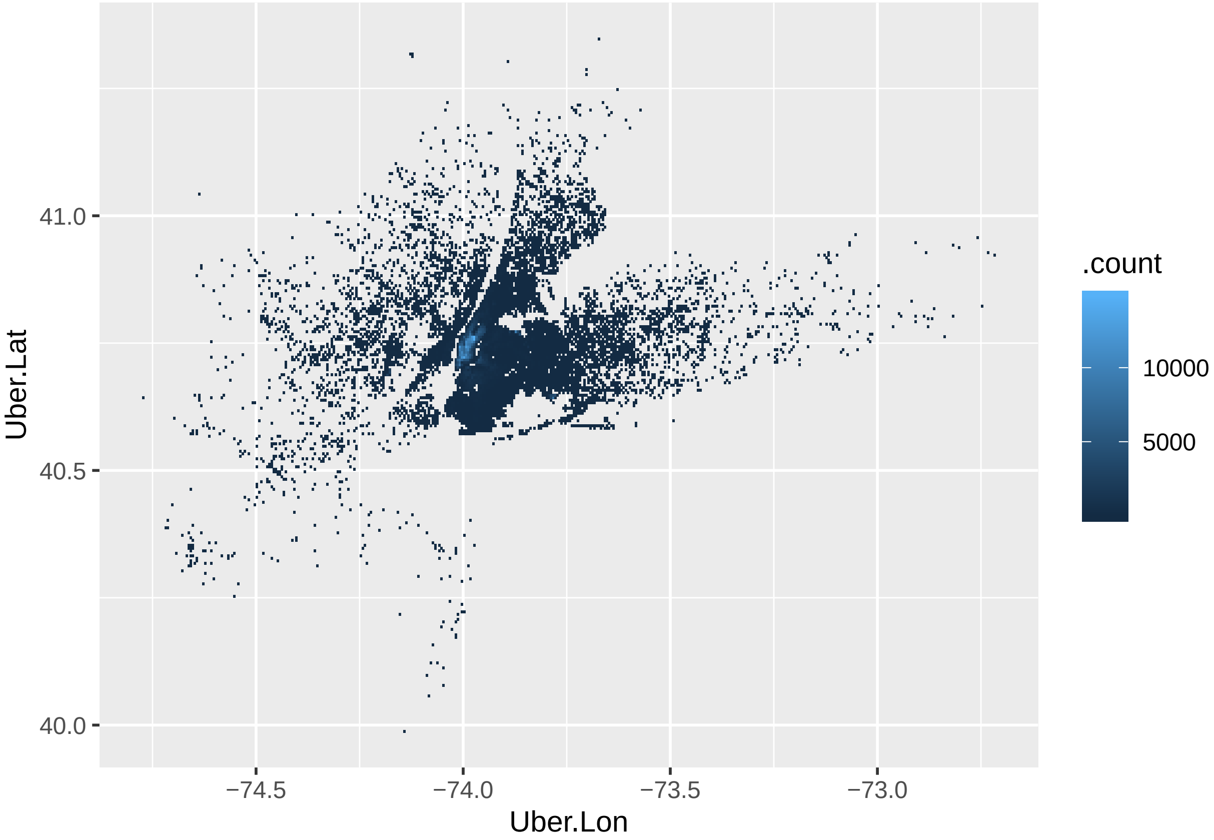
```
summary(as_tibble(Uber), digits = 6)
```

```
##   Date/Time               Lat                 Lon
##  Length:1028136     Min.   :39.9897   Min.   :-74.7736
##  Class :character   1st Qu.:40.7204   1st Qu.:-73.9962
##  Mode  :character   Median :40.7418   Median :-73.9831
##                     Mean   :40.7392   Mean   :-73.9718
##                     3rd Qu.:40.7612   3rd Qu.:-73.9628
##                     Max.   :41.3476   Max.   :-72.7163
##      Base
##  Length:1028136
##  Class :character
##  Mode  :character
##
##
##
```

We can see the range of Lat is 1.3579 and Lon is 2.0573, a bin width of 0.005 seems appropriate. For this 2d representation we are looking to find the localisation of departures trips, so a summary by mean is appropriate.

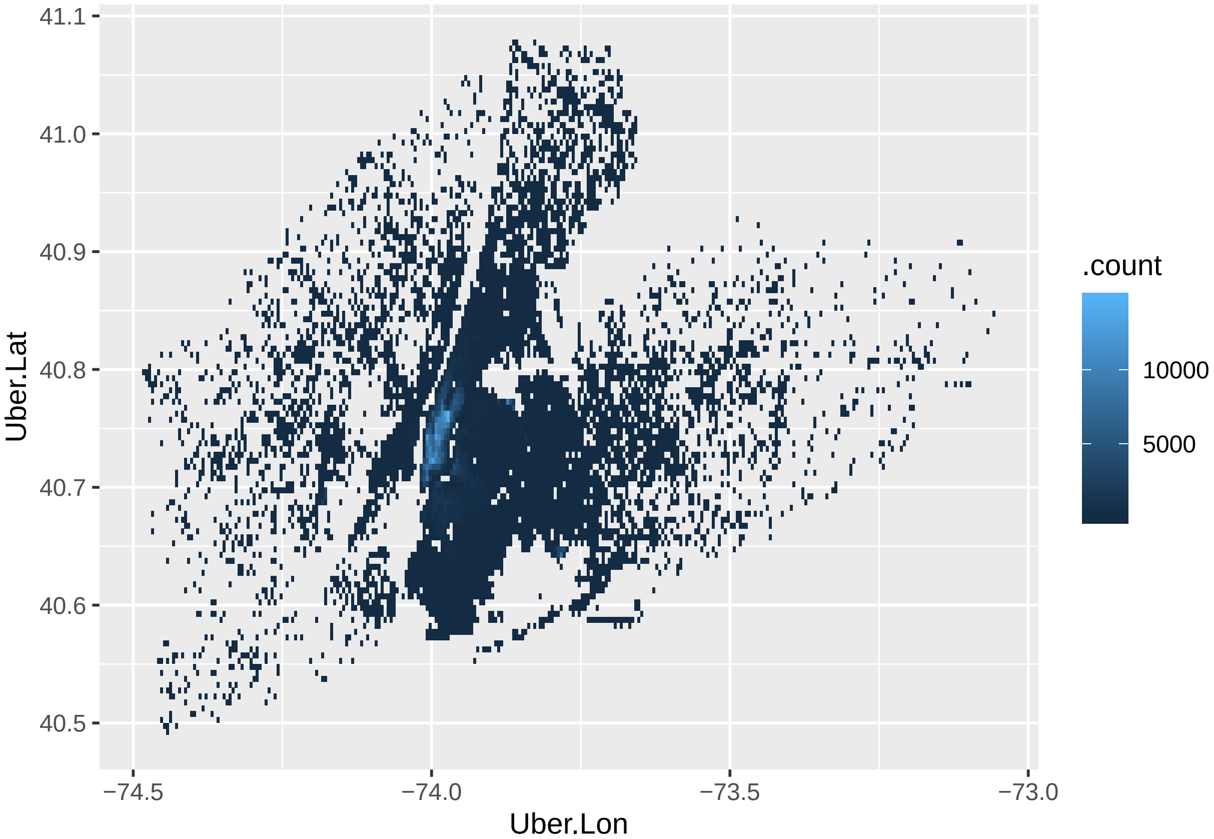
```
Uber_bin <- condense(bin(Uber$Lat, 0.005), bin(Uber$Lon, 0.005), summary = 'mean')
```

```
Uber_bin %>% ggplot(aes(Uber.Lon, Uber.Lat, fill = .count)) +
  geom_tile()
```



The function peel at bigvis package keeps specified proportion of data by removing the lowest density regions, either anywhere on the plot, or for 2d, just around the edges.

```
last_plot() %+% peel(Uber_bin)
```

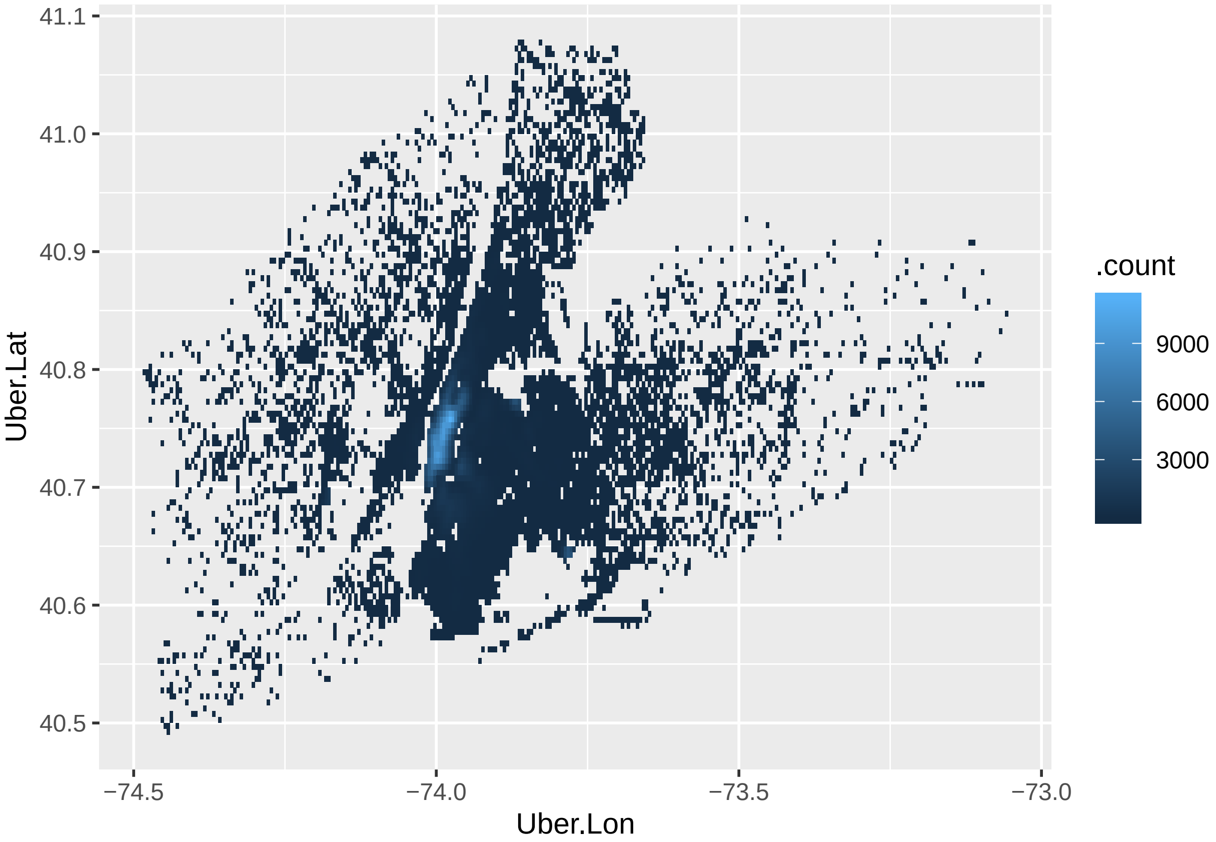


The peel function improved a lot our plot by getting ride off 1% of Data (the outliers), it's acceptable for our purpose in this problem, because the outliers have the least amount off departures anyway.

Smoothing

Smoothing allows us to resolve problems with excessive variability in the summaries and corrects for binning discontinuity.

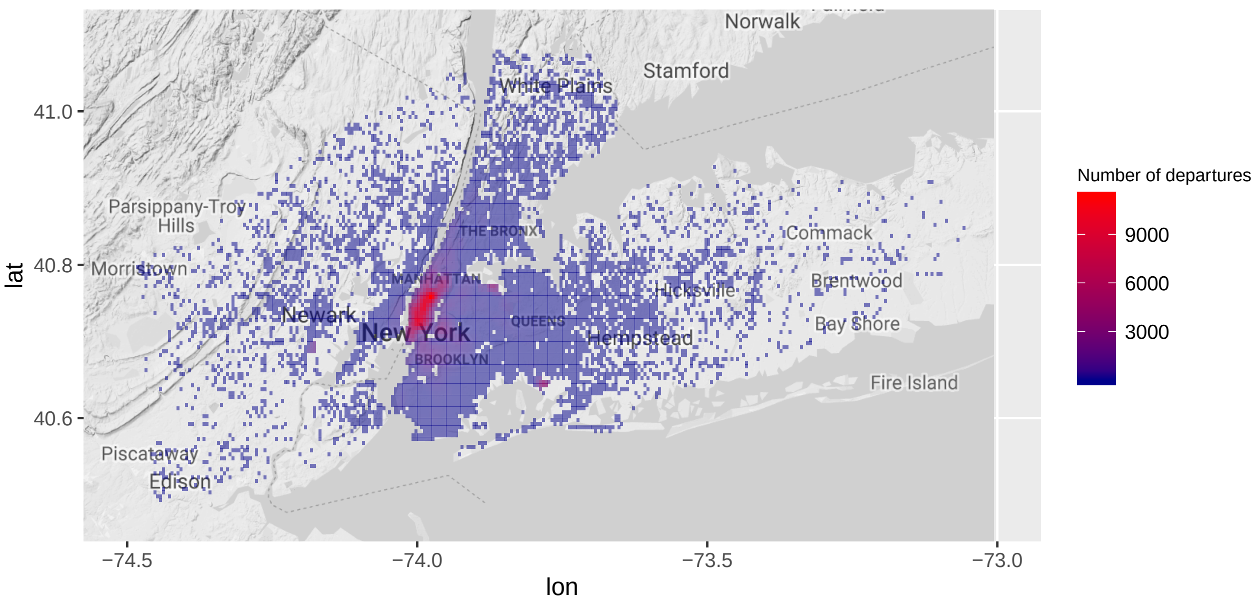
```
Uber_smooth <- smooth(peel(Uber_bin), h=c(0.01, 0.01))
Uber_smooth %>% ggplot(aes(Uber.Lon, Uber.Lat, fill = .count)) +
  geom_tile()
```



Refining the visualization

Let's add a map and tweak our plot for better visualization.

```
mid <- c(mean(Uber_smooth$Uber.Lon), mean(Uber_smooth$Uber.Lat))
map <- get_googlemap(center = mid, zoom = 9, color = "bw",
  style = "feature:road|visibility:off&style=element:labels|visibility:off&style=feature:administrative
ggmap(map) +
  geom_tile(data = Uber_smooth, aes(Uber.Lon, Uber.Lat, fill = .count, alpha = .count)) +
  scale_fill_continuous(name = 'Number of departures', low = "darkblue",high = "red") +
  scale_alpha_continuous(range = c (0.5, 1), guide = 'none') +
  xlim(-74.5, -73) +
  ylim(40.47, 41.1) +
  theme(legend.title = element_text( size = 8))
```



Session info

```
sessionInfo()
```

```
## R version 3.6.1 (2019-07-05)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.3 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_CA.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_CA.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_CA.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_CA.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] ggmap_3.0.0.901   bigvis_0.1.0.9000 Rcpp_1.0.2
##  [4] forcats_0.4.0     stringr_1.4.0     dplyr_0.8.3
##  [7] purrr_0.3.3       readr_1.3.1       tidyr_1.0.0
## [10] tibble_2.1.3      ggplot2_3.2.1     tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_0.2.5  xfun_0.10         haven_2.1.1
##  [4] lattice_0.20-38   colorspace_1.4-1  vctrs_0.2.0
##  [7] generics_0.0.2    htmltools_0.4.0   yaml_2.2.0
## [10] utf8_1.1.4         rlang_0.4.0        pillar_1.4.2
## [13] glue_1.3.1         withr_2.1.2        modelr_0.1.5
## [16] readxl_1.3.1       plyr_1.8.4         jpeg_0.1-8
## [19] lifecycle_0.1.0    munsell_0.5.0      gtable_0.3.0
## [22] cellranger_1.1.0   rvest_0.3.4        RgoogleMaps_1.4.4
## [25] codetools_0.2-16   evaluate_0.14      labeling_0.3
## [28] knitr_1.25         curl_4.2           fansi_0.4.0
## [31] broom_0.5.2        scales_1.0.0       backports_1.1.5
## [34] jsonlite_1.6       rjson_0.2.20       hms_0.5.1
## [37] png_0.1-7          digest_0.6.22      stringi_1.4.3
## [40] grid_3.6.1         bitops_1.0-6       cli_1.1.0
## [43] tools_3.6.1        magrittr_1.5       lazyeval_0.2.2
## [46] crayon_1.3.4       pkgconfig_2.0.3    zeallot_0.1.0
## [49] xml2_1.2.2         lubridate_1.7.4    assertthat_0.2.1
## [52] rmarkdown_1.16     httr_1.4.1         rstudioapi_0.10
## [55] R6_2.4.0           nlme_3.1-141       compiler_3.6.1
```