Hypothesis 1

Mofi Islam and Dennis Wollersheim

2018-04-30

## Loading required package: wrapr

## Loading required package: data.table

## data.table 1.10.4.3

## The fastest way to learn (by data.table authors): https://www.datacamp.com/courses/data-analysis-the-data-table-way

## Documentation: ?data.table, example(data.table) and browseVignettes("data.table")

## Release notes, videos and slides: http://r-datatable.com

##   
## Attaching package: 'data.table'

## The following object is masked from 'package:wrapr':  
##   
## :=

## Loading required package: seas

## Loading required package: magrittr

## Loading required package: tidyverse

## ── Attaching packages ─────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 2.2.1 ✔ purrr 0.2.4  
## ✔ tibble 1.4.2 ✔ dplyr 0.7.4  
## ✔ tidyr 0.8.0 ✔ stringr 1.3.0  
## ✔ readr 1.1.1 ✔ forcats 0.3.0

## ── Conflicts ────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::between() masks data.table::between()  
## ✖ tidyr::extract() masks magrittr::extract()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::first() masks data.table::first()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::last() masks data.table::last()  
## ✖ purrr::set\_names() masks magrittr::set\_names()  
## ✖ purrr::transpose() masks data.table::transpose()

## Loading required package: broom

## Loading required package: knitr

## Loading required package: kableExtra

## Loading required package: pander

## Loading required package: lubridate

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:data.table':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday, week, yday, year

## The following object is masked from 'package:base':  
##   
## date

## Loading required package: RPostgreSQL

## Loading required package: DBI

## Loading required package: tmap

## Loading required package: tmaptools

## Loading required package: grid

## Loading required package: readstata13

## Loading required package: ordinal

##   
## Attaching package: 'ordinal'

## The following object is masked from 'package:dplyr':  
##   
## slice

# Methods

# Methods

Patients who were dispensed at least in 15 quarters were identified as regular users. Patients who were dispensed at least in 9 quarters were identified as long-term users. Patients who were dispensed 1-8 quarters were identified as short-term users. Patients who were dispensed just one occasion only were identified as one off users.

# Distribution of types of users in two states, stratified by age-group and sex

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| age | NSW\_F | NSW\_M | VIC\_F | VIC\_M |
| 0-19 | 4443 | 4388 | 4264 | 4128 |
| 20-44 | 36805 | 32432 | 37479 | 30441 |
| 45-64 | 31273 | 30311 | 27934 | 25883 |
| 65+ | 32026 | 25843 | 26651 | 21063 |

# Distribution of types of users in two states, group proportion for the state and sex

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| age | NSW\_F | NSW\_M | VIC\_F | VIC\_M |
| 0-19 | 4.25 | 4.72 | 4.43 | 5.06 |
| 20-44 | 35.20 | 34.88 | 38.91 | 37.34 |
| 45-64 | 29.91 | 32.60 | 29.00 | 31.75 |
| 65+ | 30.63 | 27.80 | 27.67 | 25.84 |

# Distribution of types of users in two states, group proportion for the age group

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| age | NSW\_F | NSW\_M | VIC\_F | VIC\_M |
| 0-19 | 25.80 | 25.48 | 24.76 | 23.97 |
| 20-44 | 26.83 | 23.65 | 27.33 | 22.19 |
| 45-64 | 27.10 | 26.27 | 24.21 | 22.43 |
| 65+ | 30.33 | 24.48 | 25.24 | 19.95 |

# What proportion of prescriptions and DDDs are consumed by the regular, long-term, short-term and one-off users? - combined and state-wise

## Combined NSW and VIC total scripts first first

|  |  |
| --- | --- |
| usage\_category | n\_script |
| one-off | 259363 |
| short-term | 885568 |
| long-term | 646049 |
| regular | 1233276 |

## State level # of scripts total

|  |  |  |
| --- | --- | --- |
| usage\_category | NSW | VIC |
| one-off | 135651 | 123712 |
| short-term | 462987 | 422581 |
| long-term | 352101 | 293948 |
| regular | 662652 | 570624 |

## State level total scripts percentage of total

|  |  |  |
| --- | --- | --- |
| usage\_category | NSW | VIC |
| one-off | 8.41 | 8.77 |
| short-term | 28.70 | 29.95 |
| long-term | 21.82 | 20.83 |
| regular | 41.07 | 40.44 |

## State level DDD actual standardised

|  |  |  |
| --- | --- | --- |
| usage\_category | NSW | VIC |
| one-off | 2.348262 | 2.34254 |
| short-term | 10.403463 | 10.38926 |
| long-term | 11.776854 | 11.29333 |
| regular | 35.027676 | 35.55887 |

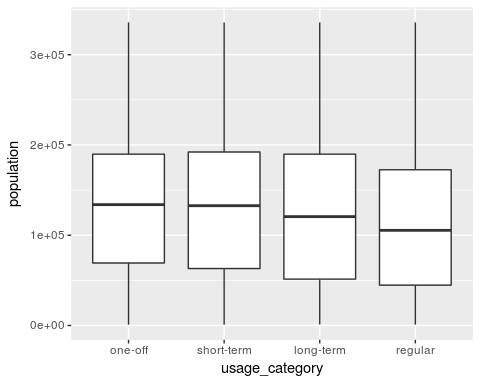
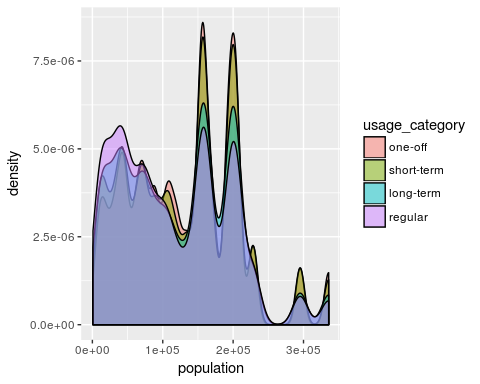
## State level DDD proportion % of state total

|  |  |  |
| --- | --- | --- |
| usage\_category | NSW | VIC |
| one-off | 3.94 | 3.93 |
| short-term | 17.47 | 17.44 |
| long-term | 19.77 | 18.95 |
| regular | 58.81 | 59.68 |

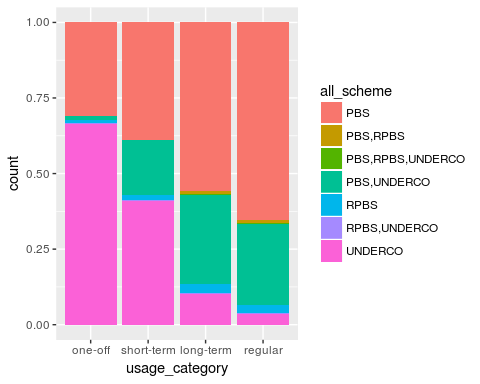
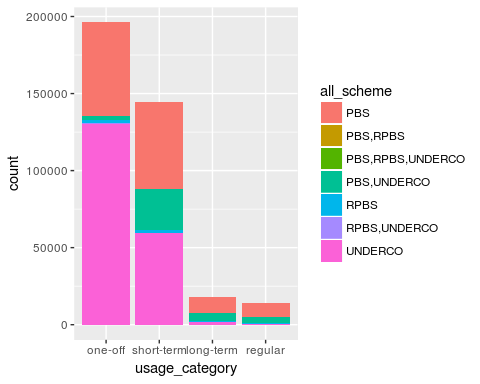
# Is there any trend in type of users on type of opioids? E.g. regular users use xxx more than others etc.

## formula: usage\_category ~ all\_scheme + sex + +age + population + seifa + urbanization  
## data: .  
##   
## link threshold nobs logLik AIC niter max.grad cond.H   
## logit flexible 372208 -325387.95 650811.89 12(1) 4.33e-07 4.5e+13  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## all\_schemePBS,RPBS 2.059e+00 6.753e-02 30.493 < 2e-16 \*\*\*  
## all\_schemePBS,RPBS,UNDERCO 2.637e+00 1.508e-01 17.490 < 2e-16 \*\*\*  
## all\_schemePBS,UNDERCO 1.515e+00 1.163e-02 130.279 < 2e-16 \*\*\*  
## all\_schemeRPBS 2.196e-01 2.834e-02 7.749 9.27e-15 \*\*\*  
## all\_schemeRPBS,UNDERCO 1.226e+00 1.051e-01 11.671 < 2e-16 \*\*\*  
## all\_schemeUNDERCO -8.947e-01 8.575e-03 -104.340 < 2e-16 \*\*\*  
## sexM -1.502e-01 6.759e-03 -22.224 < 2e-16 \*\*\*  
## age.L 1.153e+00 1.473e-02 78.263 < 2e-16 \*\*\*  
## age.Q -4.836e-01 1.176e-02 -41.110 < 2e-16 \*\*\*  
## age.C 7.496e-02 7.317e-03 10.244 < 2e-16 \*\*\*  
## population -3.801e-07 4.899e-08 -7.758 8.63e-15 \*\*\*  
## seifa.L -1.249e-01 7.457e-03 -16.744 < 2e-16 \*\*\*  
## seifa.Q -1.248e-01 6.991e-03 -17.845 < 2e-16 \*\*\*  
## seifa.C -2.951e-02 6.593e-03 -4.477 7.59e-06 \*\*\*  
## urbanizationUrban -9.514e-02 1.450e-02 -6.561 5.34e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Threshold coefficients:  
## Estimate Std. Error z value  
## one-off|short-term -0.16386 0.01447 -11.32  
## short-term|long-term 2.49248 0.01547 161.07  
## long-term|regular 3.41879 0.01675 204.14

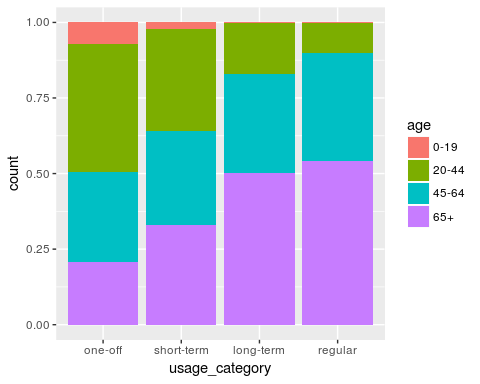
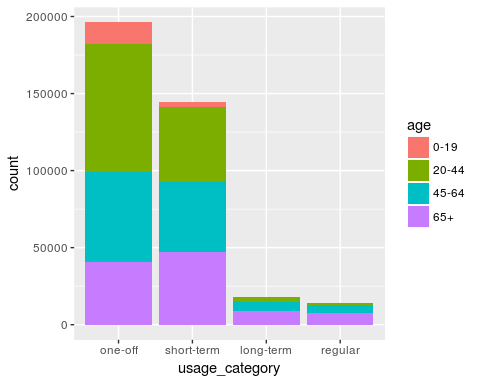
# plotting usage\_category and population, raw and proportional



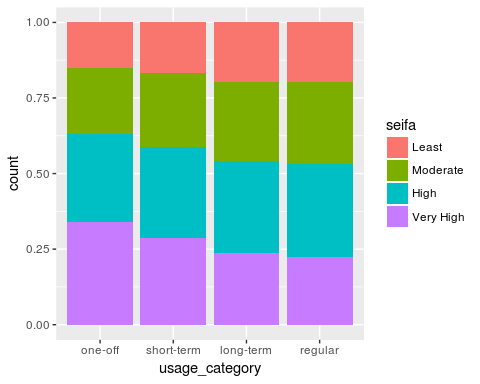
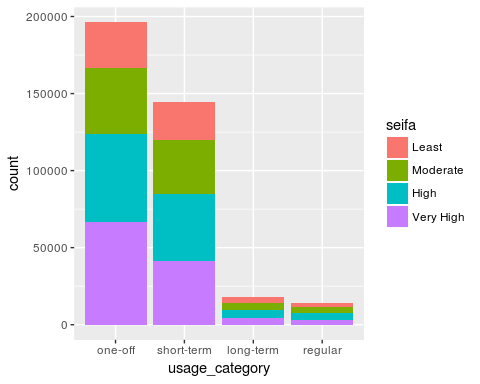
# plotting usage\_category and scheme, both raw and proportional



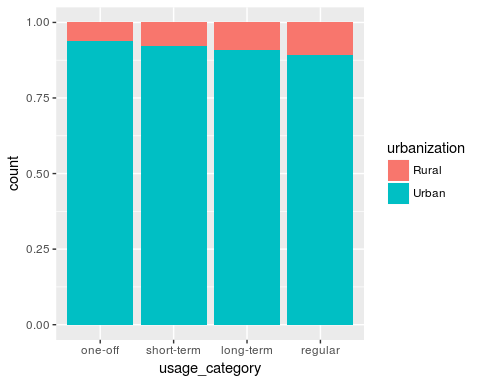
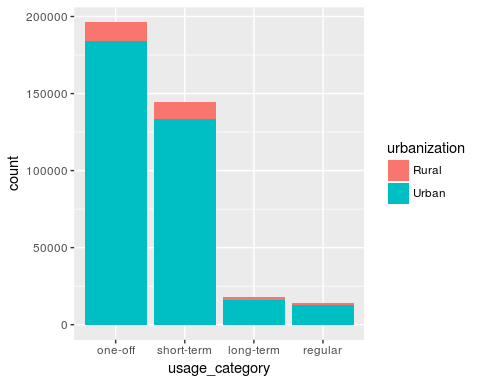
# plotting usage\_category and age, both raw and proportional



# plotting usage\_category and seifa, both raw and proportional



# plotting usage\_category and urbanisation, raw and proportional

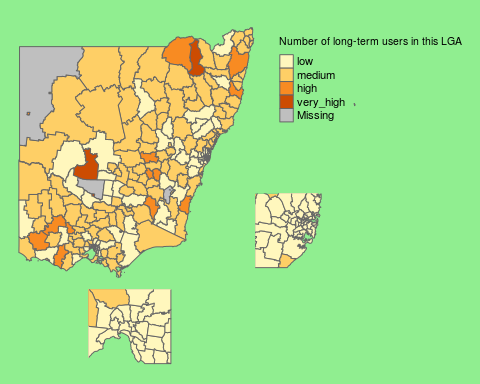


# Identify the LGAs that have higher RATE of regular and long-term users (proportional to population)

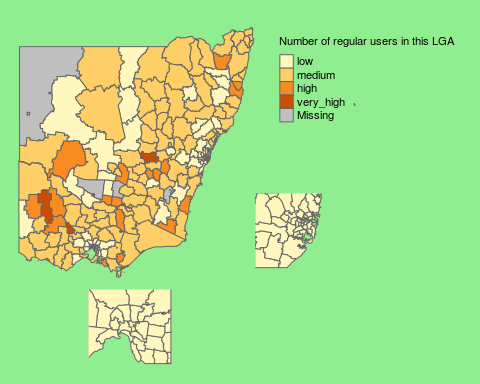
# after standardization for age and sex.

## Warning: Column `age` has different attributes on LHS and RHS of join  
  
## Warning: Column `age` has different attributes on LHS and RHS of join

## Under coverage: 4 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.

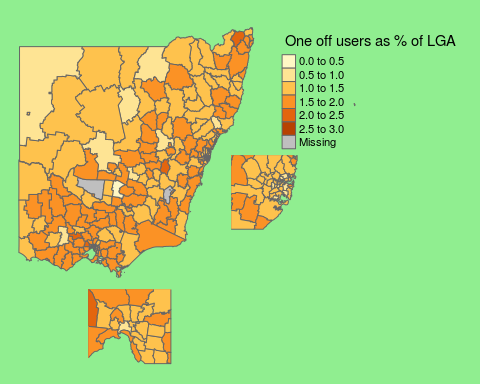


## Under coverage: 5 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.



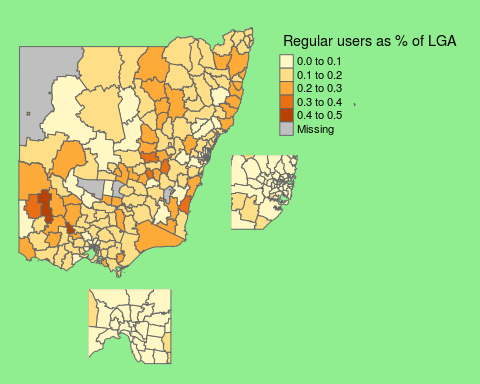
df\_patient %>%  
 inner\_join( df\_patient\_usage, by=c("pin")) %>%  
 filter(is\_geographic\_LGA(lga)) %>%   
 group\_by( lga, usage\_category ) %>%  
 summarise( n\_user = n() \* multiplier ) %>%  
 join\_population\_grouped( qw( "lga usage\_category" )) %>%  
 mutate( proportion = n\_user / population \* 100 ) %>%  
 group\_by( usage\_category ) %>%  
 mutate( usage\_level = cut( proportion, 4,   
 qw("low medium high very\_high"),   
 ordered=TRUE )) %>%  
 ungroup() %>%  
 {.} -> df\_lga\_summary  
  
#  
print\_map\_proportion = function( target\_category, name ) {  
  
 df\_lga\_summary %>%   
 filter( as.integer(usage\_category)==target\_category ) %>%  
 append\_data( base\_map,   
 .,   
 key.shp="LGA\_CODE11",   
 key.data="lga"   
 ) %>%  
 {.} -> df  
  
 df %>%  
 tm\_shape( ) +   
 tm\_polygons( "proportion",   
 title = paste("", name, "users as % of LGA")  
 ) +  
 tm\_layout(frame=FALSE,  
 legend.position = c("right", "top"), bg.color="lightgreen",  
 inner.margins = c(.25,.02,.02,.25)) %>%  
 {.} -> map  
  
 df %>%  
 tm\_shape( xlim=get\_state\_capital\_xlim(2),  
 ylim=get\_state\_capital\_ylim(2) ) +  
 tm\_polygons( "proportion", legend.show=FALSE) +  
 tm\_layout(frame=FALSE, bg.color="lightgreen") %>%  
 {.} -> m\_melbourne  
  
 df %>%   
 tm\_shape( xlim=get\_state\_capital\_xlim(1),  
 ylim=get\_state\_capital\_ylim(1) ) +  
 tm\_polygons( "proportion", legend.show=FALSE) +  
 tm\_layout(frame=FALSE, bg.color="lightgreen") %>%  
 {.} -> m\_sydney  
  
 print(map)   
 # print insets  
 print(m\_sydney, vp=viewport(x= .55, y= .5, width= 0.2, height= 0.2))  
 print(m\_melbourne, vp=viewport(x= 0.27, y= 0.15, width= 0.2, height= 0.2))  
}  
print\_map\_proportion ( 1, 'One off')

## Under coverage: 2 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.



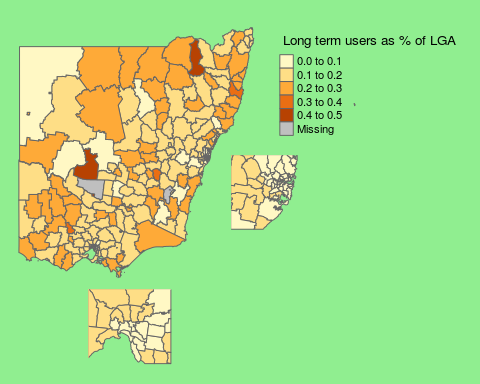
print\_map\_proportion ( 4, 'Regular')

## Under coverage: 4 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.



print\_map\_proportion ( 3, 'Long term')

## Under coverage: 3 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.



print\_map\_proportion ( 2, 'Short term')

## Under coverage: 2 out of 234 shape features did not get appended data. Run under\_coverage() to get the corresponding feature id numbers and key values.

