STAT 231: Problem Set 2B

Matthew Perkins

due by 5 PM on Friday, March 5

Series B homework assignments are designed to help you further ingest and practice the material covered in class over the past week(s). You are encouraged to work with other students, but all code must be written by you and you must indicate below who you discussed the assignment with (if anyone).

Steps to proceed:

1. In RStudio, go to File > Open Project, navigate to the folder with the course-content repo, select the course-content project (course-content.Rproj), and click "Open"
2. Pull the course-content repo (e.g. using the blue-ish down arrow in the Git tab in upper right window)
3. Copy ps2B.Rmd from the course repo to your repo (see page 6 of the GitHub Classroom Guide for Stat231 if needed)
4. Close the course-content repo project in RStudio
5. Open YOUR repo project in RStudio
6. In the ps2B.Rmd file in YOUR repo, replace "YOUR NAME HERE" with your name
7. Add in your responses, committing and pushing to YOUR repo in appropriate places along the way
8. Run "Knit PDF"
9. Upload the pdf to Gradescope. Don't forget to select which of your pages are associated with each problem. *You will not get credit for work on unassigned pages (e.g., if you only selected the first page but your solution spans two pages, you would lose points for any part on the second page that the grader can't see).*

If you discussed this assignment with any of your peers, please list who here:

ANSWER:

MDSR Exercise 4.14 (modified)

Use the `Pitching` data frame from the `Lahman` package to identify every pitcher in baseball history who has accumulated at least 300 wins (`W`) and at least 3,000 strikeouts (`SO`).

a. How many pitchers meet this criteria?

   ANSWER: 10 pitchers meet this criteria.

```
PitchingTotals <- Pitching %>%
  select(playerID,stint,W,SO) %>%
  group_by(playerID) %>%
  summarize(
    total_W=sum(W),
    total_SO=sum(SO))%>%
    arrange(desc(total_SO))%>%
  filter(total_W>299 & total_SO > 3000)
PitchingTotals
```

```
## # A tibble: 10 x 3
##    playerID  total_W total_SO
##    <chr>       <int>    <int>
##  1 ryanno01      324     5714
##  2 johnsra05     303     4875
##  3 clemero02     354     4672
##  4 carltst01     329     4136
##  5 seaveto01     311     3640
##  6 suttodo01     324     3574
##  7 perryga01     314     3534
##  8 johnswa01     417     3509
##  9 maddugr01     355     3371
## 10 niekrph01     318     3342
```

b. Which of these pitchers had the most accumulated strikeouts? How many strikeouts had he accumulated? What is the most strikeouts he had in one season?

   ANSWER: Nolan Ryan accumulated the most strikeouts, with a total of 5714. In the 1973 season, he had 383 strikeouts.

```
Nol <- Pitching %>%
  filter(playerID == "ryanno01") %>%
    arrange(desc(SO))
Nol
```

```
##    playerID yearID stint teamID lgID  W  L  G GS CG SHO SV IPouts   H  ER HR
## 1  ryanno01   1973     1    CAL   AL 21 16 41 39 26   4  1    978 238 104 18
## 2  ryanno01   1974     1    CAL   AL 22 16 42 41 26   3  0    998 221 107 18
## 3  ryanno01   1977     1    CAL   AL 19 16 37 37 22   4  0    897 198  92 12
## 4  ryanno01   1972     1    CAL   AL 19 16 39 39 20   9  0    852 166  72 14
## 5  ryanno01   1976     1    CAL   AL 17 18 39 39 21   7  0    853 193 106 13
## 6  ryanno01   1989     1    TEX   AL 16 10 32 32  6   2  0    718 162  85 17
## 7  ryanno01   1987     1    HOU   NL  8 16 34 34  0   0  0    635 154  65 14
## 8  ryanno01   1978     1    CAL   AL 10 13 31 31 14   3  0    704 183  97 12
## 9  ryanno01   1982     1    HOU   NL 16 12 35 35 10   3  0    751 196  88 20
## 10 ryanno01   1990     1    TEX   AL 13  9 30 30  5   2  0    612 137  78 18
## 11 ryanno01   1988     1    HOU   NL 12 11 33 33  4   1  0    660 186  86 18
## 12 ryanno01   1979     1    CAL   AL 16 14 34 34 17   5  0    668 169  89 15
## 13 ryanno01   1985     1    HOU   NL 10 12 35 35  4   0  0    696 205  98 12
## 14 ryanno01   1991     1    TEX   AL 12  6 27 27  2   2  0    519 102  56 12
## 15 ryanno01   1980     1    HOU   NL 11 10 35 35  4   2  0    701 205  87 10
## 16 ryanno01   1984     1    HOU   NL 12 11 30 30  5   2  0    551 143  62 12
## 17 ryanno01   1986     1    HOU   NL 12  8 30 30  1   0  0    534 119  66 14
## 18 ryanno01   1975     1    CAL   AL 14 12 28 28 10   5  0    594 152  76 13
## 19 ryanno01   1983     1    HOU   NL 14  9 29 29  5   2  0    589 134  65  9
## 20 ryanno01   1992     1    TEX   AL  5  9 27 27  2   0  0    472 138  65  9
## 21 ryanno01   1981     1    HOU   NL 11  5 21 21  5   3  0    447  99  28  2
## 22 ryanno01   1971     1    NYN   NL 10 14 30 26  3   0  0    456 125  67  8
## 23 ryanno01   1968     1    NYN   NL  6  9 21 18  3   0  0    402  93  46 12
## 24 ryanno01   1970     1    NYN   NL  7 11 27 19  5   2  1    395  86  50 10
## 25 ryanno01   1969     1    NYN   NL  6  3 25 10  2   0  1    268  60  35  3
## 26 ryanno01   1993     1    TEX   AL  5  5 13 13  0   0  0    199  54  36  5
## 27 ryanno01   1966     1    NYN   NL  0  1  2  1  0   0  0      9   5   5  1
##     BB  SO BAOpp   ERA IBB WP HBP BK  BFP GF   R SH SF GIDP
## 1  162 383 0.203  2.87   2 15   7  0 1355  2 113  7  7   24
## 2  202 367 0.190  2.89   3  9   9  0 1392  1 127 12  4   24
## 3  204 341 0.193  2.77   7 21   9  3 1272  0 110 22 10   21
## 4  157 329 0.171  2.28   4 18  10  0 1154  0  80 11  3   NA
## 5  183 327 0.195  3.36   2  5   5  2 1196  0 117 13  4   12
## 6   98 301 0.187  3.20   3 19   9  1  988  0  96  9  5    4
## 7   87 270 0.200  2.76   2 10   4  2  873  0  75  9  1    6
## 8  148 260 0.220  3.72   7 13   3  2 1008  0 106 11 14   18
## 9  109 245 0.213  3.16   3 18   8  2 1050  0 100  9  3   12
## 10  74 232 0.188  3.44   2  9   7  1  818  0  86  3  5    5
## 11  87 228 0.227  3.52   6 10   7  7  930  0  98 10  8    7
## 12 114 223 0.212  3.60   3  9   6  0  937  0 104  8 10   14
## 13  95 209 0.239  3.80   8 14   9  2  983  0 108 11 12   16
## 14  72 203 0.172  2.91   0  8   5  0  683  0  58  3  9    7
## 15  98 200 0.236  3.35   1 10   3  1  982  0 100  7  7   17
## 16  69 197 0.212  3.04   2  6   4  3  760  0  78  4  6   10
## 17  82 194 0.188  3.34   5 15   4  0  729  0  72  5  4    9
## 18 132 186 0.213  3.45   0 12   7  0  864  0  90  6  7   19
## 19 101 183 0.195  2.98   3  5   4  1  804  0  74  7  5   20
## 20  69 157 0.238  3.72   0  9  12  0  675  0  75  6  7    5
## 21  68 140 0.188  1.69   1 16   1  2  605  0  34  5  3   10
## 22 116 137 0.219  3.97   4  6  15  1  705  1  78  3  0   NA
## 23  75 133 0.200  3.09   4  7   4  0  559  1  50 NA NA   NA
## 24  97 125 0.188  3.42   2  8   4  0  570  4  59  8  4   NA
## 25  53  92 0.180  3.53   3  1   1  3  375  4  38 NA NA   NA
## 26  40  46 0.220  4.88   0  3   1  0  291  0  47  2  2    3
## 27   3   6 0.350 15.00   1  1   0  0   17  0   5 NA NA   NA
```

MDSR Exercise 4.17 (modified)

a. The Violations data set in the `mdsr` package contains information regarding the outcome of health inspections in New York City. Use these data to calculate the median violation score by zipcode and dba for zipcodes in Manhattan. What pattern (if any) do you see between the number of inspections and the median score? Generate a visualization to support your response.

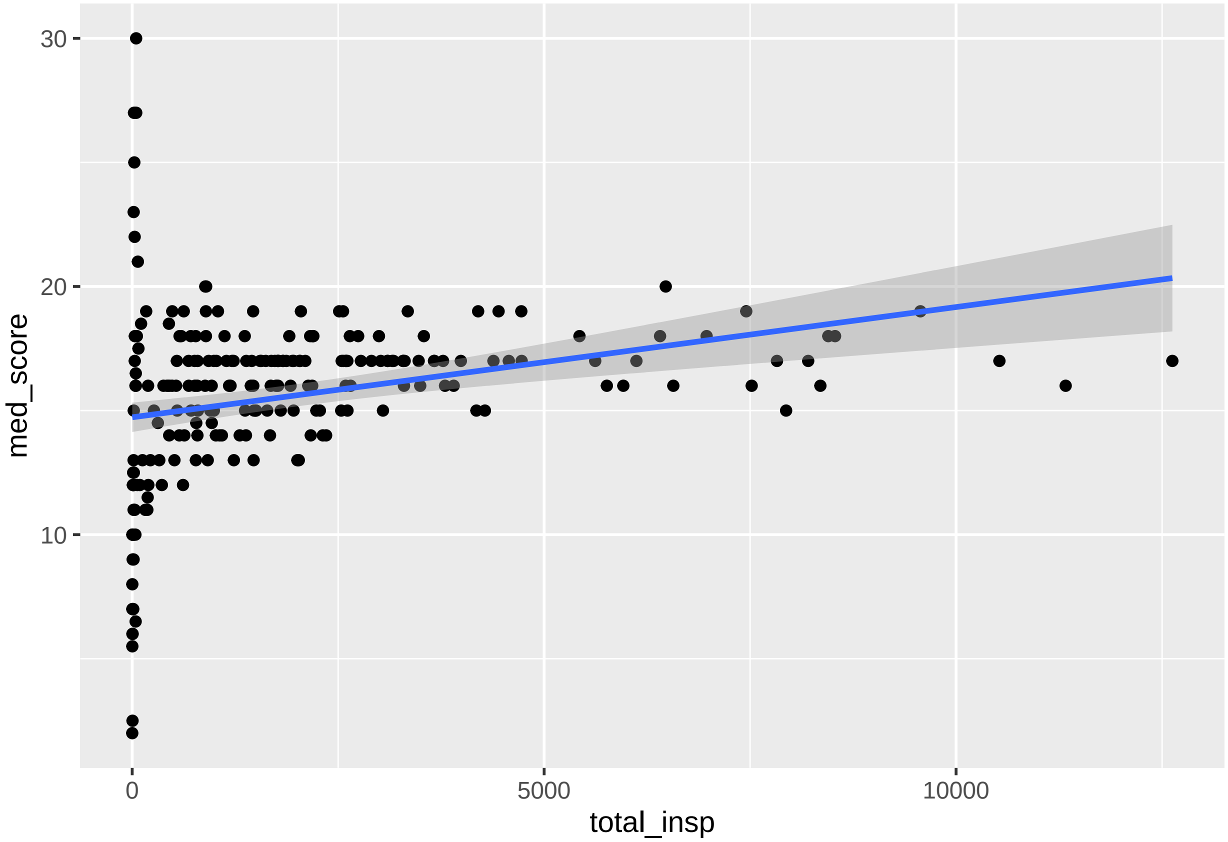
   ANSWER: While the pattern is not totally clear, it appears as though the median violation score by zipcode or by dba tends to increase as inspections increase.

```
ViolationsZip <- Violations %>%
  select(zipcode,score) %>%
  filter(is.na(score)==FALSE)%>%
  group_by(zipcode) %>%
  summarize(
    total_insp = length(zipcode),
    med_score = median(score)) %>%
    arrange(desc(med_score))
ViolationsZip
```

```
## # A tibble: 229 x 3
##    zipcode total_insp med_score
##      <int>      <int>     <dbl>
##  1   11001         48        30
##  2   11005         49        27
##  3   11352         22        27
##  4   10123         26        25
##  5   10311         18        23
##  6   11451         30        22
##  7   11697         69        21
##  8   10310        898        20
##  9   11220       6476        20
## 10   11428        887        20
## # ... with 219 more rows
```

```
ggplot(data = ViolationsZip) +
    geom_point(mapping = aes(
      x = total_insp,
      y = med_score))+
      geom_smooth(aes(x = total_insp,
      y = med_score), method = "lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



```
ViolationsDba <- Violations %>%
  select(dba,score,zipcode) %>%
  filter(is.na(score)==FALSE)%>%
  group_by(dba,zipcode) %>%
  summarize(
    total_insp = length(dba),
    med_score=median(score))%>%
    arrange(desc(med_score))
```

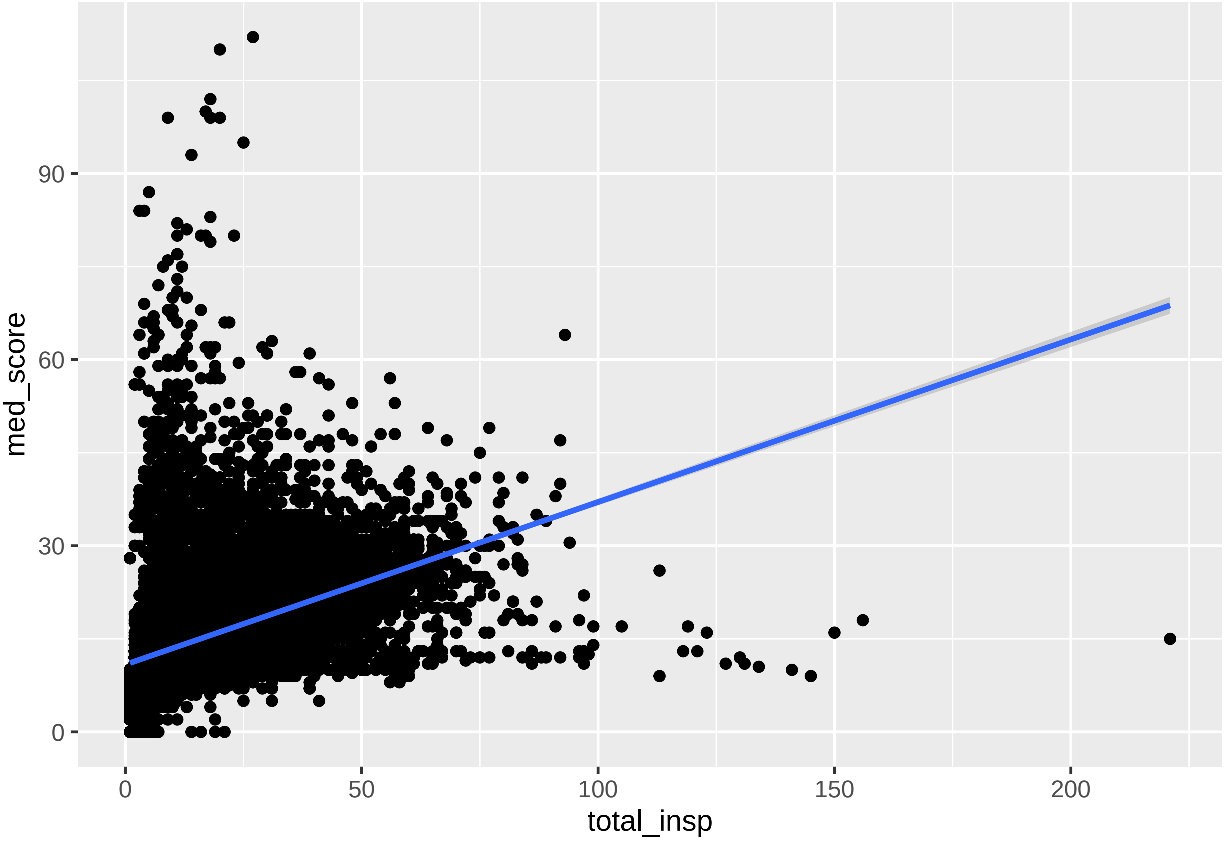
```
## `summarise()` has grouped output by 'dba'. You can override using the `.groups` argument.
```

```
ViolationsDba
```

```
## # A tibble: 23,353 x 4
## # Groups:   dba [19,758]
##    dba                             zipcode total_insp med_score
##    <chr>                             <int>      <int>     <dbl>
##  1 ROXY DINER                        10036         27       112
##  2 NEW BISMILLAH                     11216         20       110
##  3 FOOD CAVE                         11101         18       102
##  4 TEUTA QEBAPTORE                   10458         17       100
##  5 Gou Bang Zi Chicken               11354         18        99
##  6 RICHMOND COUNTY YACHT CLUB        10308          9        99
##  7 SANDWICH BAR                      11367         20        99
##  8 BONJOUR CREPES & WINE             10128         25        95
##  9 TEA MAGIC                    11354          14         93
## 10 BX PIZZA BAR RESTAURANT      10456           5         87
## # ... with 23,343 more rows
```

```
ggplot(data = ViolationsDba) +
    geom_point(mapping = aes(
      x = total_insp,
      y = med_score))+
      geom_smooth(aes(x = total_insp,
      y = med_score), method = "lm")
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



b. In your visualization in part (a), there should be at least a few points that stand out as outliers. For *one of the outliers*, add text to the outlier identifying what business it is and an arrow pointing from the text to the observation. First, you may want to `filter` to identify the name of the business (so you know what text to add to the plot).

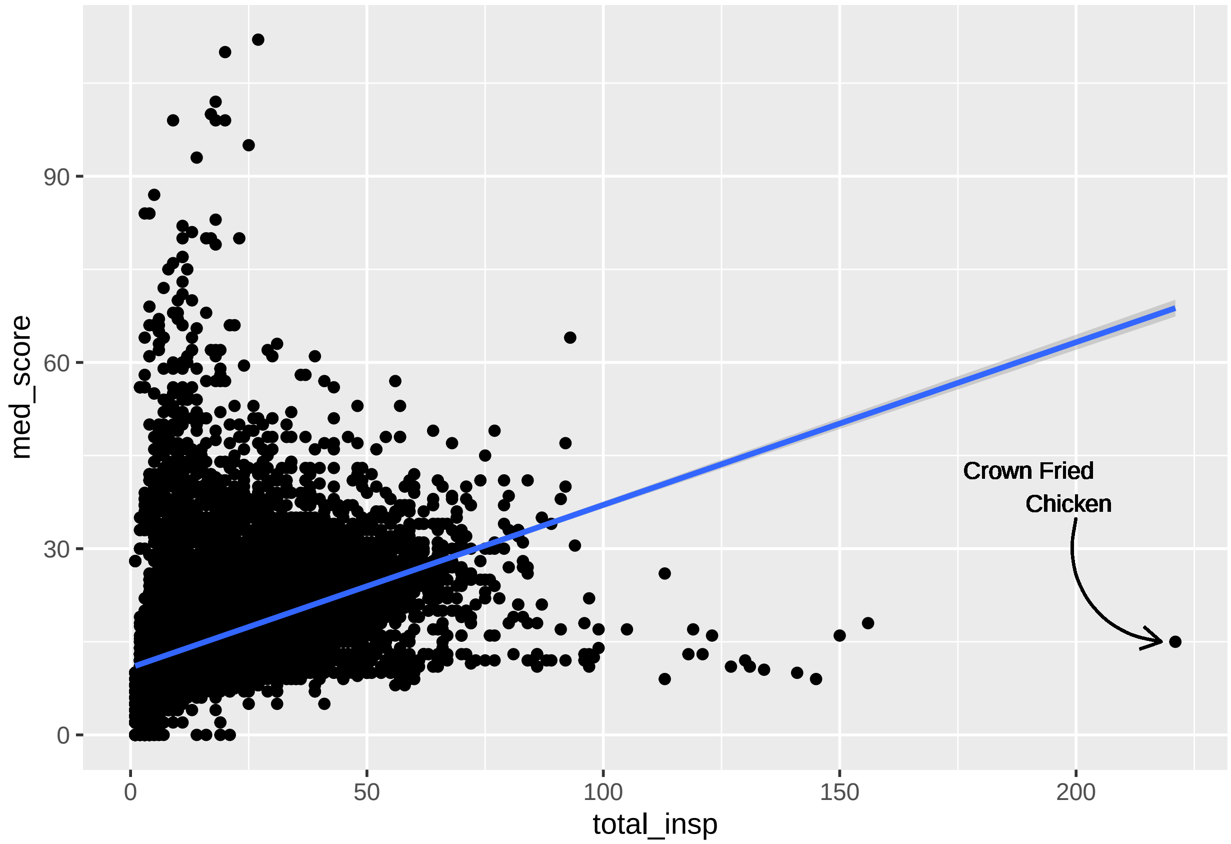
(Can't remember how to create a curved arrow in `ggplot`? The answers to this question on Stack Exchange may help. Can't remember how to add text to the plot in `ggplot`? Check out the text examples with `annotate` here, or answers to this question that use `geom_text`.)

```
filter(ViolationsDba, total_insp>200)
```

```
## # A tibble: 1 x 4
## # Groups:   dba [1]
##   dba                  zipcode total_insp med_score
##   <chr>                  <int>      <int>     <dbl>
## 1 CROWN FRIED CHICKEN    11212        221        15
```

```
ggplot(data = ViolationsDba) +
    geom_point(mapping = aes(
      x = total_insp,y=med_score))+
      geom_smooth(aes(x = total_insp,
      y = med_score), method = "lm")+
  geom_curve(aes(x=200,y=35,xend = 218,yend=15)
             ,arrow = arrow(length = unit(0.03,
              "npc")))+
  geom_text(x=190,y=40,label = 'Crown Fried
            Chicken',size=3)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



MDSR Exercise 5.7

Generate the code to convert the data frame shown with this problem in the textbook (on page 130, and shown below) to wide format (i.e., the result table). Hint: use `gather()` in conjunction with `spread()`; OR `pivot_longer()` in conjunction with `pivot_wider()`.

```
#Didn't use pivotlonger, but this got the job done!

FakeDataLong <- data.frame(grp = c("A","A","B", "B")
                           , sex = c("F", "M", "F", "M")
                           , meanL = c(0.22, 0.47, 0.33, 0.55)
                           , sdL = c(0.11, 0.33, 0.11, 0.31)
                           , meanR = c(0.34, 0.57, 0.40, 0.65)
                           , sdR = c(0.08, 0.33, 0.07, 0.27))

DataWide <- FakeDataLong %>%
  pivot_wider(
    names_from = sex,
    values_from = c(meanL,meanR,sdL,meanR,sdR),
    values_fill = 0) %>%
    select(grp, F.meanL = meanL_F,
           F.meanR = meanR_F,
           F.sdL = sdL_F,
           F.sdR = sdR_F,
           M.meanL = meanL_M,
           M.meanR = meanR_M,
           M.sdL = sdL_M, M.sdR = sdR_M)
DataWide
```

```
## # A tibble: 2 x 9
##   grp   F.meanL F.meanR F.sdL F.sdR M.meanL M.meanR M.sdL M.sdR
##   <chr>   <dbl>   <dbl> <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl>
## 1 A        0.22    0.34  0.11  0.08    0.47   0.570  0.33  0.33
## 2 B        0.33    0.4   0.11  0.07    0.55   0.65   0.31  0.27
```

PUG Brainstorming

What topics or questions are you interested in exploring related to your PUG theme? Dream big here. Don't worry about whether there is data out there that's available and accessible that you could use to address your questions/topics. Just brainstorm some ideas that get you excited. Then, email your PUG team with your ideas. Title the email "PS2B Brainstorming: PUG [#] [Topic]" and CC me (kcorreia@amherst.edu) on the email. If another PUG member already initiated the email, reply all to their email.

If you don't remember your PUG # and Topic, please see the file "PUGs" on the Moodle page under this week.

If you don't know your PUG members email address, go to the class's Google group conversations (e.g., by clicking the link "Link to Google group conversations" at the top of our Moodle course page). Then, on the navigation panel (left hand side), select "Members".

> ANSWER: Do not write anything here. Email your ideas to your PUG team and me in a message titled "PS2B Brainstorming: PUG [#] [Topic]".