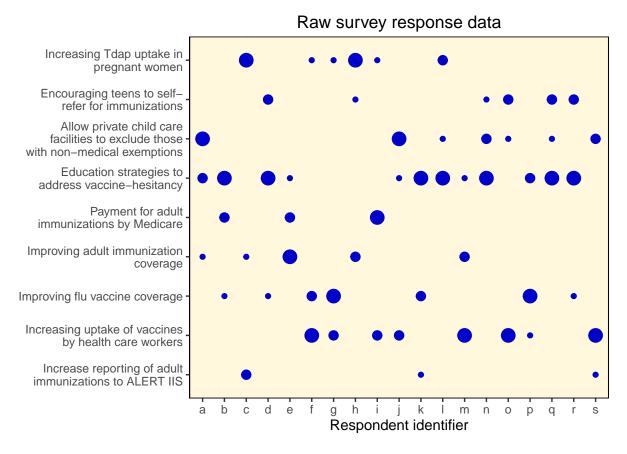# Analysing survey data in Public Health

17 May 2016

Exploring the data

A survey was conducted amongst 19 public health professionals about which topics they would like discussed at the IPAT meeting.

Each respondent selected 3 from 9 topics in order of preference - 1 being the first priority. The plot below shows how each respondent voted for each topic.

Raw survey response data

Each respondent has been assigned a letter so they can be identified during the analysis. The size of dot indicates the priority for that respondent. Reading up and down each column will reveal the choice each respondent made. Scanning across each row will indicate the overall feeling for that topic.

In this raw state, we can glean some limited information from the data. For example, we can see that *Education strategies to address vaccine-hesitancy* has a lot of priority 1 votes, while *Increase reporting of adult immunizations to ALERT IIS* was not all that popular. Otherwise, it is hard to tell which topics should be selected, and the impact of that decision.

Further insight can be gained by re-arranging the plot data as follows:

- order the topics from top to bottom according to overall popularity, with the most popular at the top and the least at the bottom.
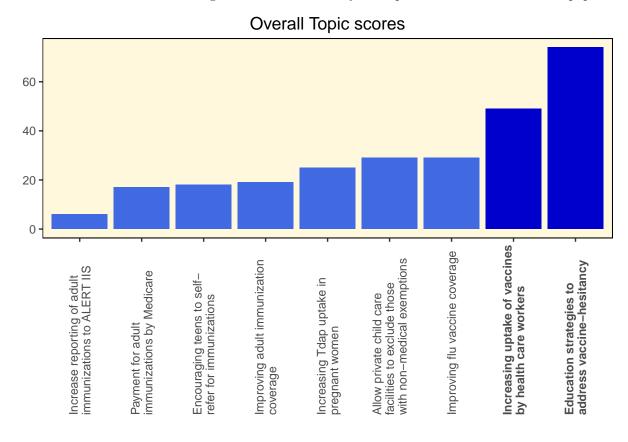
- draw a cut-off line depending on how many topics can be realistically prepared for in advance of the IPAT meeting. Topics above the line will be taken forward, topics below the line will not.
- calculate how satisified respondents are about the selected topics. Re-arrange the respondents on the horizontal axis in order of satisfaction, with the least satisfied on the left hand side and the most satisifed on the right.

Following this approach will allow two conclusions to be drawn. Firstly, how well the selected topics match the overall preferences and secondly, those respondents who are not satisfied with the outcome.

Most popular topics

The first job is to calculate a score for each topic based on the aggregate priorities of those who voted for that topic. The plot below shows the overall scores for each survey topic. The score has been calculated to reflect accurately the first, second and third preferences of the survey respondents.

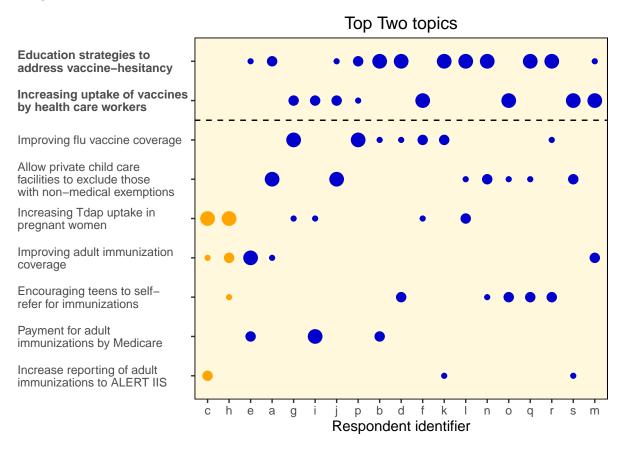There are two clear winners at the right hand side and clearly the topics on the far left are not so popular.

Overall Topic scores

However it is not at all clear what the differences are between the next three most popular topics - the scores are very close.

Implications of selecting just two topics

In this section we draw the cut-off line after the second most popular topic. We also re-order the respondents according to their satisfaction with this. This results in the distribution below.

Those votes marked in orange show those respondents who didn't get any of their preferences selected.
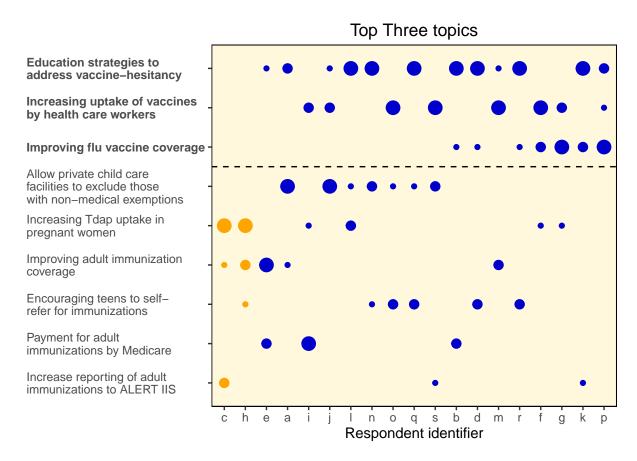
We can see that 17 respondents had at least one of their topics selected in the top two topics. Only two respondents had none of their topics chosen.

Furthermore, 11 of the respondents have had their first choice included. Respondents e, a, g, i, j and p didn't get their first choice, however they have some of their other choices selected. Respondents c and h are the least satisfied.

Options for including a third topic
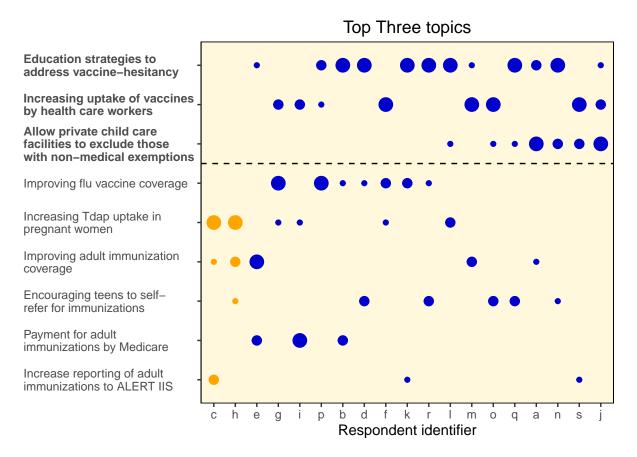
Because of the close nature of the next three popular topics, we need to look more closely at the effect of including each one as the third choice. These three options are presented below.

Option (a)
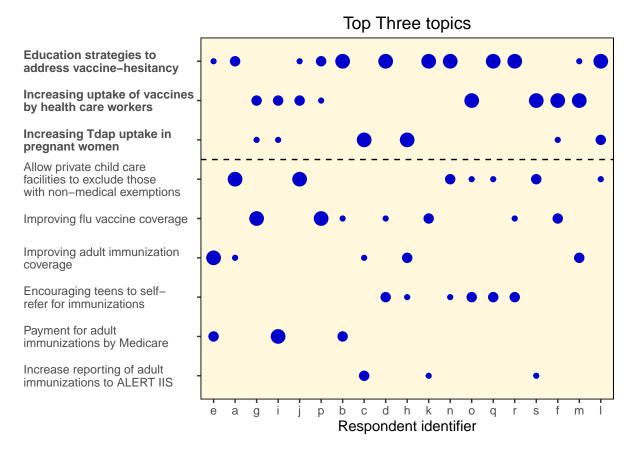
Although two respondents still don't have any of their topics selected, there are now only 6 respondents who didn't get their first choice (and of those, 4 had at least one of their other choice in the cut.)

Option (b)

Top Three topics

This has a similar impact to option (a) - six priority 1 items didn't make the cut and a couple of people still have nothing.

Option (c)

Top Three topics
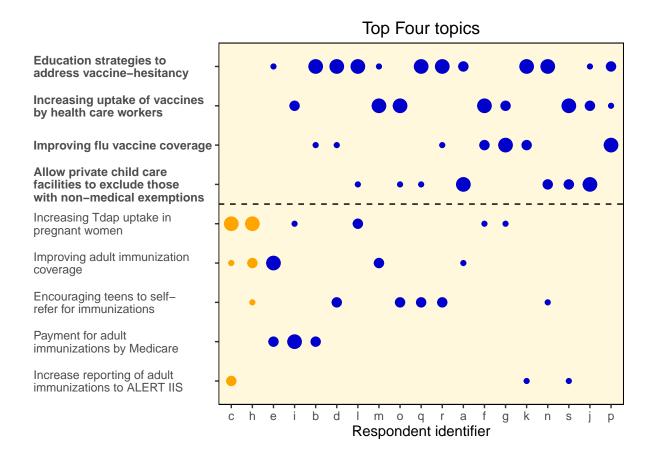
This is very interesting - no orange dots! This means that every single respondent has at least one of their choices included if these three topics are chosen.

This looks a good option to choose if it is possible to take three topics forward to the IPAT meeting.
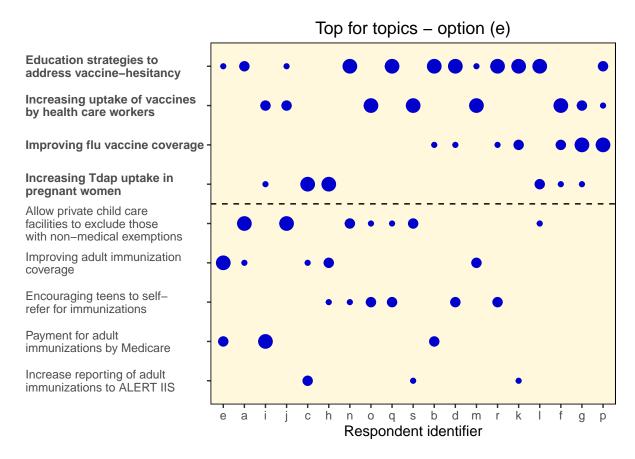
Top Four topics

In the event that four topics can be presented, there are a couple of options for topic number 4

Option (d)

Option (e)

Top for topics – option (e)

Again, with this option we can see all respondents would be satisfied with these choices.

Appendix 1 - Raw Data

The raw data is summarised in the following table. Each respondent provided a priority 1 to 3 for each topic. The subject of each topic is defined in a separate table.

| RespondentID	Topic1	Topic2	Topic3	Topic4	Topic5	Topic6	Topic7	Topic8	Topic9
a			1	2		3			
b				1	2		3		
c	1					3			2
d		2		1			3		
e				3	2	1			
f	3						2	1	
g	3						1	2	
h	1	3				2			
i	3				1			2	
j			1	3				2	
k				1			2		3
l	2		3	1					
m				3		2		1	
n		3	2	1					
o		2	3					1	
p				2			1	3	
q		2	3	1					
r		2		1			3		
s			2					1	3

topic	subject
Topic1	Increasing Tdap uptake in pregnant women
Topic2	Encouraging teens to self-refer for immunizations
Topic3	Allow private child care facilities to exclude those with non-medical exemptions
Topic4	Education strategies to address vaccine-hesitancy
Topic5	Payment for adult immunizations by Medicare
Topic6	Improving adult immunization coverage
Topic7	Improving flu vaccine coverage
Topic8	Increasing uptake of vaccines by health care workers
Topic9	Increase reporting of adult immunizations to ALERT IIS

This data is stored in an excel file with two sheets. The data can be edited in the excel sheet and the analysis re-run.

Appendix 2 - Analysis code

In the code below, the variable *number_to_select* refers to how many topics will be chosen to be taken forward for policy analysis. For the bulk of the analysis, this is set at 3, however some effects are shown for a value of 4.

The following code provides the main calculations and analysis:

```r
calculate_scores <- function(number_to_select=4, force_swap=NULL) {
        # Get the data from excel and flatten into a data set that can
        # be manipulated more easily
        response_data <<- read.xlsx(xlsfile,1)
        topic_data <<- read.xlsx(xlsfile,2)

        respondent <- names(response_data)[1]
        survey<-melt(response_data, id.vars = respondent, variable.name="topic",
                     na.rm=TRUE, value.name="priority")
        survey$RespondentID <- as.factor(survey$RespondentID)
        number_respondents=nrow(response_data)

        # Calculate the score for each response
        lowest_priority=max(survey$priority)
        survey <- survey %>% mutate(score=(lowest_priority + 1 - priority)^2)

        # Calculate the total score per topic
        topic_scores <- survey %>% group_by(topic) %>%
                        summarise(topic_score=sum(score)) %>%
                        arrange(topic_score)

        # Put a 1 in the weight columm for the last number of topics determined by
        # number_to_select variable.  Other topics have weight zero.  This selects
        # the topics which are the most popular
        topic_scores$weight <- c(rep(0, nrow(topic_scores)-number_to_select),
                                 rep(1,number_to_select))

        if(!is.null(force_swap)){
                        topic_scores$weight[force_swap[1]] <- 0
                        topic_scores$weight[force_swap[2]] <- 1
                        tmp <- topic_scores$topic_score[force_swap[1]]
                        topic_scores$topic_score[force_swap[1]] <- topic_scores$topic_score[force_swap[2]]
                        topic_scores$topic_score[force_swap[2]] <- tmp
        }

        # Add the topic descriptions
        topic_scores <- merge(topic_scores, topic_data, by="topic")
        topic_scores <- arrange(topic_scores, weight, topic_score)

        # Mix the weights into the survey data
        survey <- merge(survey, topic_scores, by="topic")

        # Calculate the satisfaction for each respondent against the selected most
        # popular topics
        survey <- survey %>% mutate(topic_satisfaction=score^2*weight)
```

```
        resp <- survey %>% group_by(RespondentID) %>%
                summarise(satisfaction=sum(topic_satisfaction)) %>%
                arrange(satisfaction)

        survey <- merge(survey, resp, by="RespondentID")

        # Create the topic labels with the subject of each topic suitably word wrapped
        ylabels <- str_wrap(as.vector(topic_scores$subject), width = 30)
        names(ylabels)<-topic_scores$topic

        resp<<-resp
        topic_scores<<-topic_scores
        survey<<-survey
        ylabels <<- ylabels
}
```

The bar plot showing the overall topic scores is produced with the following:

```
plot_topic_score <- function(){
        ggplot(topic_scores, aes(x=topic, y=topic_score)) +
        geom_bar(stat="identity",
                fill=ifelse(topic_scores$weight == 1,'mediumblue', 'royalblue')) +
        scale_x_discrete(limits=topic_scores$topic,
                        labels=ylabels) +
        ylab("") + xlab("") +
        theme(panel.grid.minor=element_blank(),
                panel.grid.major=element_blank(),
                panel.background = element_rect(fill = "cornsilk"),
                panel.border = element_rect(fill="transparent", colour="black"),
                axis.text.x = element_text(angle = 90,  hjust=0, vjust=0.5,
                                face=ifelse(topic_scores$weight == 0,'plain','bold')))
}
```

The plot of each respondent's preference, ordered by satisfaction is:

```
plot_responses <- function (number_to_select) {

ggplot(survey, aes(x=RespondentID, y=topic)) +
        geom_point(size=sqrt(survey$score)*1.5,
                colour=ifelse(survey$satisfaction == 0,'orange',
                                ifelse(survey$weight == 1,'mediumblue', 'mediumblue'))) +
        scale_y_discrete(limits=topic_scores$topic,
                        labels=ylabels) +
        ylab("") + xlab("Respondent identifier") +
        scale_x_discrete(limits=resp$RespondentID) +
        theme(panel.grid.minor=element_blank(),
                panel.grid.major=element_blank(),
                panel.background = element_rect(fill = "cornsilk"),
                panel.border = element_rect(fill="transparent", colour="black"),
                axis.text.y = element_text(hjust=0,
                                face=ifelse(topic_scores$weight == 0,'plain','bold'))) +
        geom_hline(yintercept = nrow(topic_scores) - number_to_select + 0.5,
                linetype="dashed")
}
```