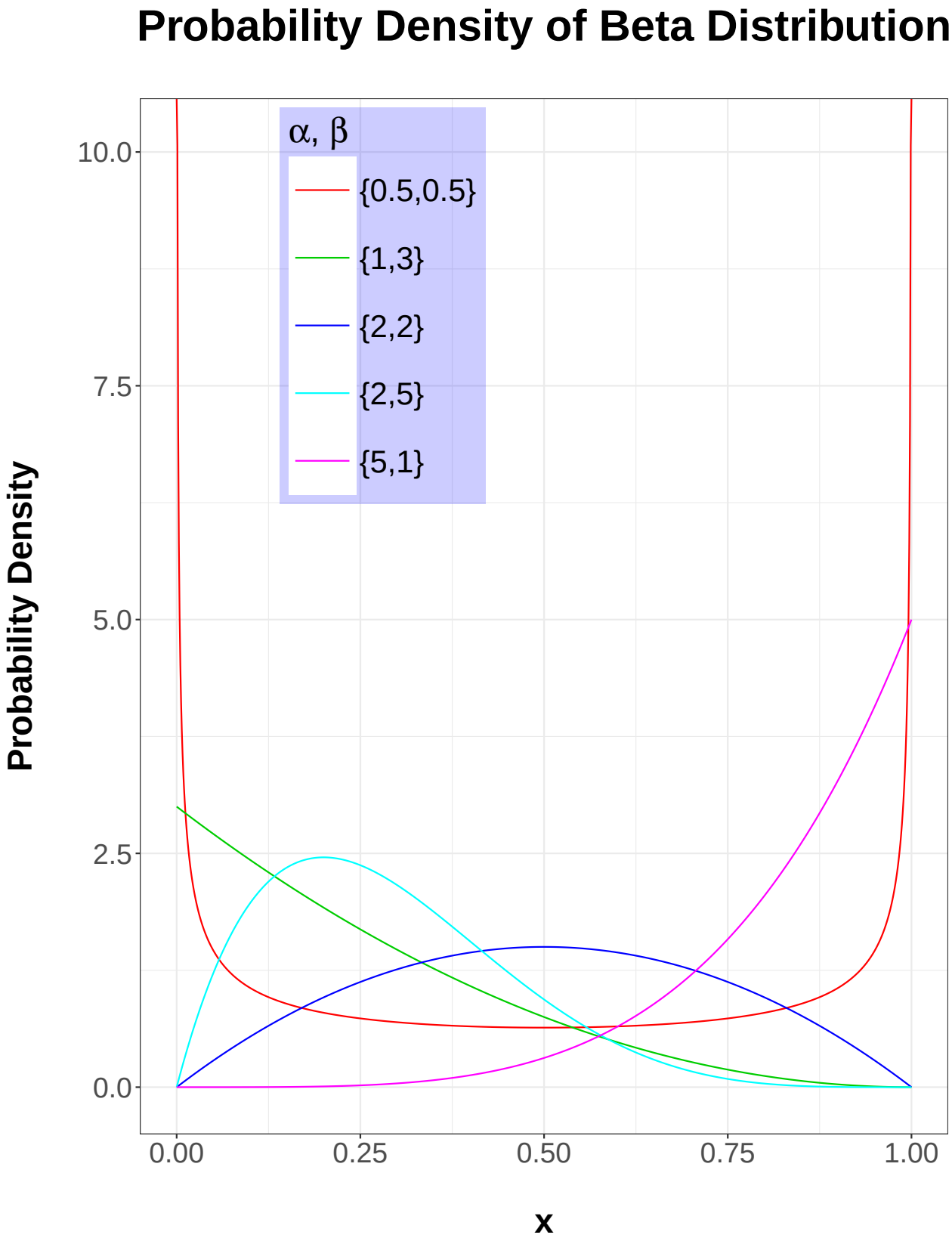
Case Study on Office Traffic Stops

Chris Dong

Question 1

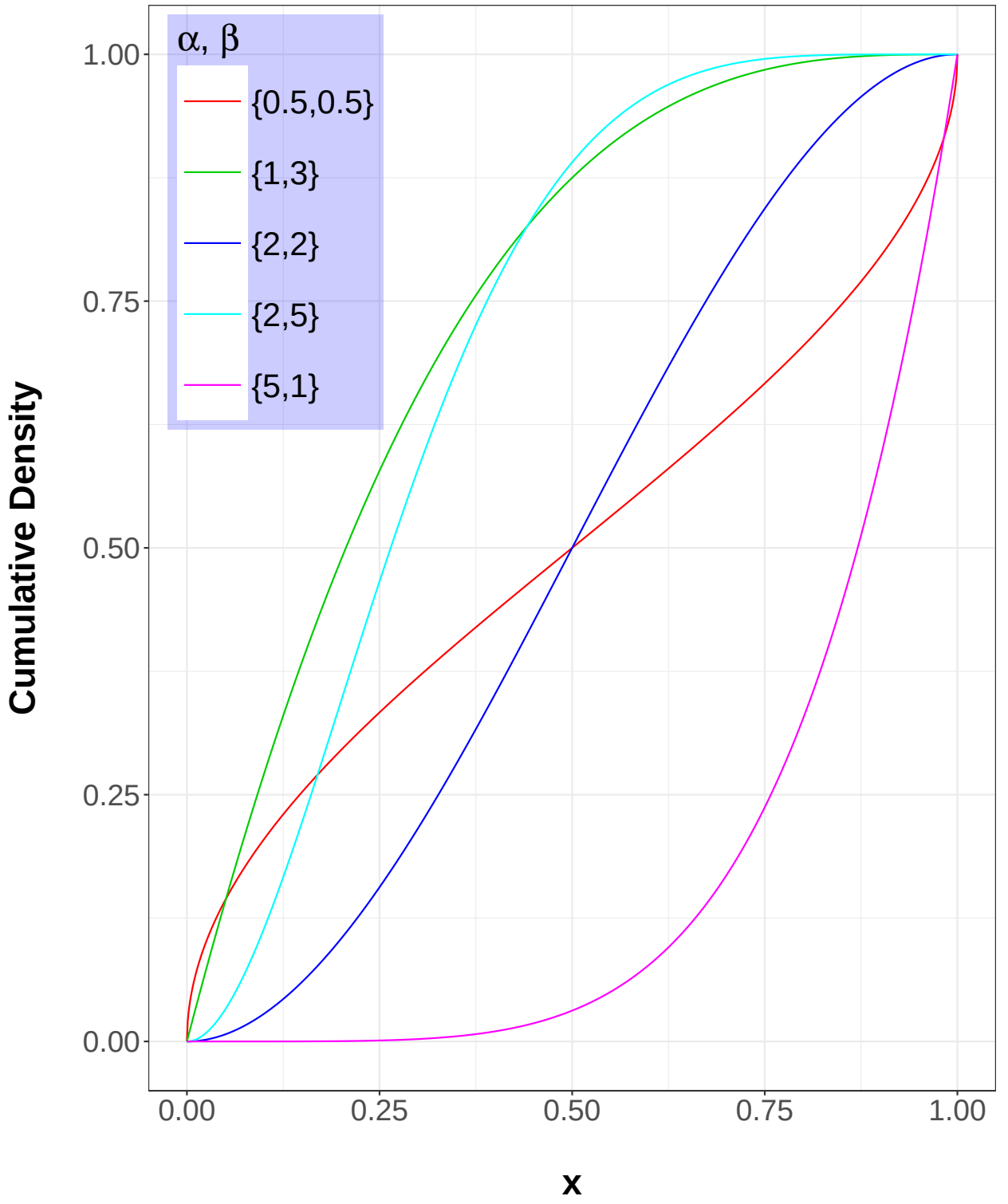
1.1 Using a single line graph, generate superimposed probability density functions of the beta distribution where{α,β}$\in${0.5,0.5},{5,1},{1,3},{2,2},{2,5}}. The pdf of each each {α,β} tuple should be differentiated using color, and a legend should be included.

```
x<-seq(0,1,.001) # creating a sequence of x-values

dbetaDF <- tibble( # data frame containing the beta probability densities
  tuple0.5_0.5 = dbeta(x = x, shape1 = 0.5, shape2 = 0.5),
  tuple5_1 = dbeta(x = x, shape1 = 5, shape2 = 1),
  tuple1_3 = dbeta(x = x, shape1 = 1, shape2 = 3),
  tuple2_2 = dbeta(x = x, shape1 = 2, shape2 = 2),
  tuple2_5 = dbeta(x = x, shape1 = 2, shape2 = 5)
)

dbetaDF2 <- dbetaDF %>% # aggregating the columns into just two
  gather("beta_alpha","value",
         tuple0.5_0.5, tuple5_1, tuple1_3, tuple2_2, tuple2_5, 1:5) %>%
  mutate(x = rep(seq(0,1,.001),5),
         alpha = c(rep(0.5,1001), rep(5,1001), rep(1,1001), #alpha values
                   rep(2, 1001), rep(2,1001)),
         beta = c(rep(0.5,1001), rep(1,1001), rep(3,1001), #beta values
                   rep(2, 1001), rep(5,1001)),
         type = rep("pdf", 5005),
         beta_alpha = paste('{',alpha,',',beta,'}',sep='')) #for labeling

dbetaDF2 %>% ggplot(aes(x = x, y = value, color = beta_alpha)) +
    geom_line() + xlab("\nx") + ylab("Probability Density\n") +
    labs(col = TeX('$\\alpha$, $\\beta$'),
         title = "Probability Density of Beta Distribution\n") +
    theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          legend.text = element_text(size = 20),
          legend.title = element_text(size = 22, face = "bold"),
          legend.key.height=unit(3,"line"),
          legend.key.width =unit(3, "line"),
          legend.position = c(.3, .80),
          legend.background = element_rect(fill = alpha ('blue', 0.2)))+
    scale_color_manual(values = 2:6)
```


Probability Density of Beta Distribution
α, β
{0.5,0.5}
{1,3}
{2,2}
{2,5}
{5,1}
Probability Density
10.0
7.5
5.0
2.5
0.0
0.00
0.25
0.50
0.75
1.00
x

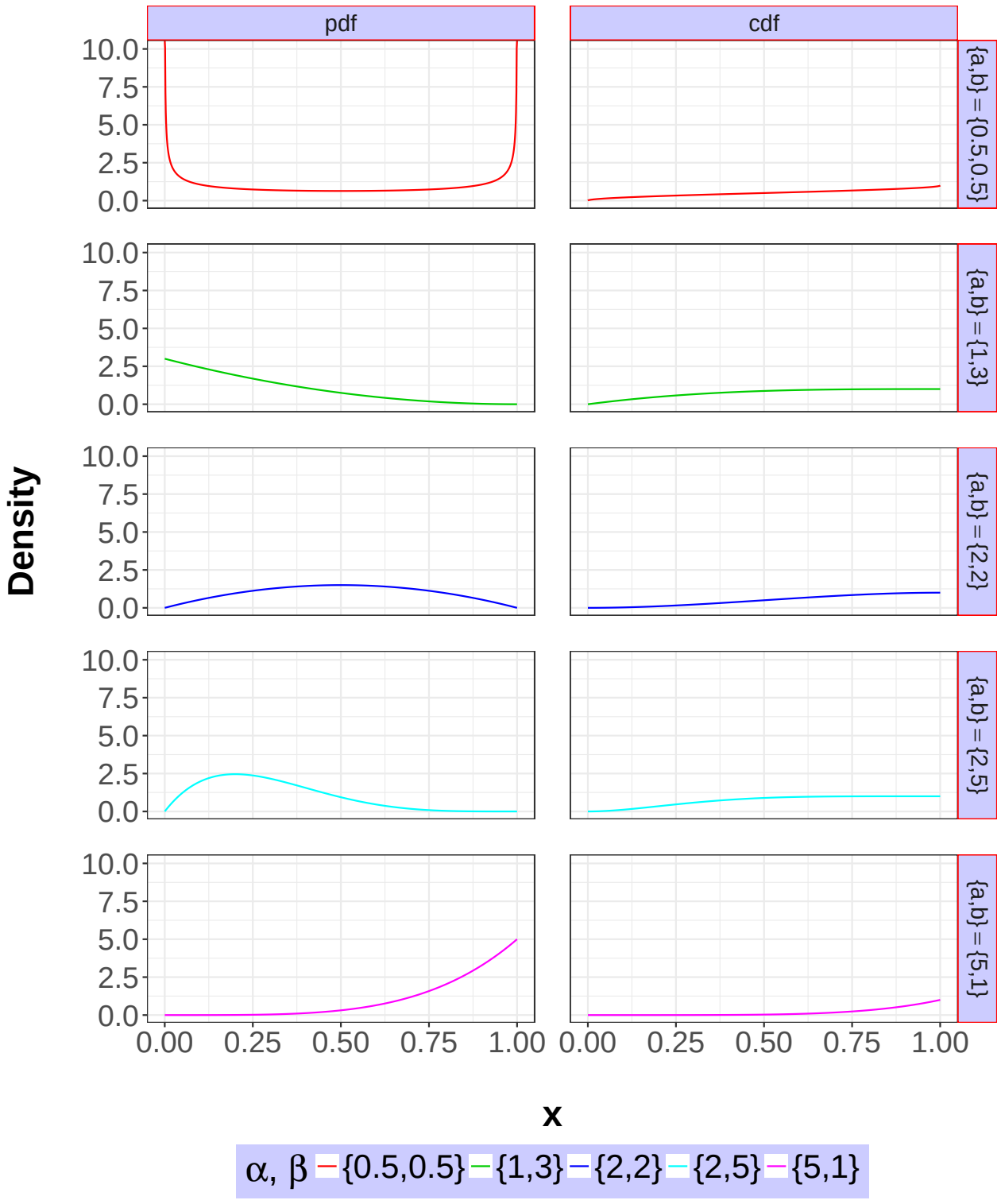
1.2 Using a single line graph, generate superimposed cumulative density functions of the beta distribution where$\{\alpha,\beta\}\in\{0.5,0.5\},\{5,1\},\{1,3\},\{2,2\},\{2,5\}\}$. The pdf of each each $\{\alpha,\beta\}$ tuple should be differentiated using color, and a legend should be included.

```
pbetaDF <- tibble( #same as question 1 but with pbeta or cumulative density
  tuple0.5_0.5 = pbeta(q = x, shape1 = 0.5, shape2 = 0.5),
  tuple5_1 = pbeta(q = x, shape1 = 5, shape2 = 1),
  tuple1_3 = pbeta(q = x, shape1 = 1, shape2 = 3),
  tuple2_2 = pbeta(q = x, shape1 = 2, shape2 = 2),
  tuple2_5 = pbeta(q = x, shape1 = 2, shape2 = 5)
)

pbetaDF2 <- pbetaDF %>%
  gather("beta_alpha","value",
        tuple0.5_0.5, tuple5_1, tuple1_3, tuple2_2, tuple2_5, 1:5) %>%
  mutate(x = rep(seq(0,1,.001),5),
         alpha = c(rep(0.5,1001), rep(5,1001), rep(1,1001),
                 rep(2, 1001), rep(2,1001)),
         beta = c(rep(0.5,1001), rep(1,1001), rep(3,1001),
                 rep(2, 1001), rep(5,1001)),
         type = rep("cdf", 5005),
         beta_alpha = paste('{',alpha,',',beta,'}',sep=''))

pbetaDF2 %>% ggplot(aes(x = x, y = value, color = beta_alpha)) +
    geom_line() + xlab('\nx') + ylab("Cumulative Density\n") +
    labs(col = TeX('$\\alpha$, $\\beta$'),
         title = "Cumulative Density of Beta Distribution\n") +
    theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          legend.text = element_text(size = 20),
          legend.title = element_text(size = 22, face = "bold"),
          legend.key.height=unit(3,"line"),
          legend.key.width =unit(3, "line"),
          legend.position = c(.15, .80),
          legend.background = element_rect(fill = alpha ('blue', 0.2)))+
    scale_color_manual(values = 2:6)
```


Cumulative Density of Beta Distribution
α, β
{0.5,0.5}
{1,3}
{2,2}
{2,5}
{5,1}
1.00
0.75
0.50
0.25
0.00
Cumulative Density
0.00
0.25
0.50
0.75
1.00
x

1.3 Combining output from 1.1 and 1.2, create a single graphical output with all *pdfs* and *cdfs*. They will all be line graphs and will be faceted using two columns and five rows. The title of each facet column will bepdfandcdf. The titles of the row facets, located on the right hand side of the graph, will have the {α,β} tuple values, i.e., {α,β} ={x, y}, where x and y are the tuple values. The titles must have the correct greek symbols α β in LaTeX (not some weird, ugly, pixelated character), and the tuple values must begenerated dynamically based on their values (not hard coded). The output will be a single graphical objectwith 10 sub-graphs in two columns and five rows.

```
combinebetaDF <- rbind(dbetaDF2,pbetaDF2) #combine question 1 and 2
combinebetaDF %<>% mutate(
  type = factor(type, levels=c("pdf","cdf")) #so pdf is left of cdf
  )

combinebetaDF %>% ggplot(aes(x))+
  geom_line(aes(y = value, color = beta_alpha)) +
  facet_grid(alpha + beta ~ type, # 5 by 2
             labeller = label_bquote(paste('{a,b}' ==  '{', ... =
             .(alpha),',',.(beta),'}'))) +
    xlab('\nx') + ylab("Density\n") +
    labs(col = TeX('$\\alpha$, $\\beta$'),
         title = "PDF and CDF of Beta Distribution\n") +
    theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          strip.background = element_rect(color = "red", fill = '#CCCCFF'),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          panel.spacing = unit(1.5, "lines"),
          legend.text = element_text(size = 20),
          legend.title = element_text(size = 22, face = "bold"),
          legend.key.height=unit(1,"line"),
          legend.key.width =unit(1, "line"),
          legend.position = "bottom",
          legend.background = element_rect(fill = alpha ('blue', 0.2))) +
    guides(fill = guide_legend(keywidth = 3)) +
    scale_color_manual(values = 2:6)
```


PDF and CDF of Beta Distribution
pdf
cdf
{a,b} = {0.5,0.5}
{a,b} = {1,3}
{a,b} = {2,2}
{a,b} = {2,5}
{a,b} = {5,1}
Density
10.0
7.5
5.0
2.5
0.0
0.00
0.25
0.50
0.75
1.00
x
α, β {0.5,0.5} {1,3} {2,2} {2,5} {5,1}

Question 2

First, I will load the data and make all of the variables lowercase.

```
officers <- readr::read_csv("Officer_Traffic_Stops.csv", col_types = cols())
names(officers) <- tolower(names(officers))
```

```
glimpse(officers)
```

```
## Observations: 79,884
## Variables: 17
## $ month_of_stop            <chr> "2016/01", "2016/01", "2016/01", "201...
## $ reason_for_stop          <chr> "Speeding", "Stop Light/Sign", "Speed...
## $ officer_race             <chr> "White", "White", "White", "Black/Afr...
## $ officer_gender           <chr> "Male", "Male", "Male", "Male", "Male...
## $ officer_years_of_service <int> 6, 6, 6, 2, 6, 6, 6, 2, 7, 9, 8, 5, 9...
## $ driver_race              <chr> "White", "Black", "Black", "Black", "...
## $ driver_ethnicity         <chr> "Non-Hispanic", "Non-Hispanic", "Non-...
## $ driver_gender            <chr> "Male", "Male", "Male", "Female", "Ma...
## $ driver_age               <int> 63, 35, 30, 29, 45, 65, 40, 28, 57, 2...
## $ was_a_search_conducted   <chr> "No", "No", "No", "No", "No", "No", "...
## $ result_of_stop           <chr> "Citation Issued", "Verbal Warning", ...
## $ cmpd_division            <chr> "Eastway Division", "Eastway Division...
## $ objectid                 <int> 1001, 1002, 1003, 1004, 1005, 1006, 1...
## $ creationdate             <dttm> 2016-12-20 23:49:30, 2016-12-20 23:4...
## $ creator                  <chr> "charlottedata", "charlottedata", "ch...
## $ editdate                 <dttm> 2016-12-20 23:49:30, 2016-12-20 23:4...
## $ editor                   <chr> "charlottedata", "charlottedata", "ch...
```

Using `glimpse()`, I can easily see that `creationdate`, `creator`, `editdate`, `editor`, `objectid` do not contain useful information so I will remove them.

```
officers %<>% select(-c(objectid:editor))
```

Data Cleaning - I will go through all the variables to see if anything should be changed.

First, `month_of_stop` can be separated into only months. I will remove the `year` variable since it's all 2016.

```
officers %<>% separate(month_of_stop, into = c("year", "month")) %>% select(-year)
```

```
officers %>% group_by(reason_for_stop) %>% count() %>% arrange(desc(n)) %>%
  knitr::kable(align=c('c','c'), col.names = c("Reason for Stop", "Frequency"))
```

Reason for Stop	Frequency
Vehicle Regulatory	32405
Speeding	22222
Stop Light/Sign	7946
Vehicle Movement	7535
Safe Movement	4827
Investigation	1992
Other	1926
SeatBelt	631
CheckPoint	286
Driving While Impaired	114

For `reason_for_stop` , since Speeding may be interesting to look at it –it is also the second most common

in this data (and often occurs in real life). I can make a new variable indicating the boolean of whether or not the ticket was due to speeding.

For `result_for_stop`, I will change into an ordered factor with the levels from least severe to the most severe.

```
officers %<>% mutate(
  is_speeding = ifelse(reason_for_stop == "Speeding", T, F),
  result_of_stop = ordered(officers$result_of_stop,
          levels = c("No Action Taken", "Verbal Warning",
                     "Written Warning", "Citation Issued","Arrest")))
```

```
officers %>% select(driver_race) %>% table() %>% as_data_frame() %>%
  arrange(desc(n)) %>%
  mutate(perc = round(n/sum(n),2)) %>% knitr::kable(align=c('c','c','c'),
         col.names = c("Driver Race", "Frequency", "Percentage"))
```

Driver Race	Frequency	Percentage
Black	42970	0.54
White	32969	0.41
Other/Unknown	2422	0.03
Asian	1461	0.02
Native American	62	0.00

Because other `driver_race` other than White or Black make up a very tiny percentage of our data, I will filter them out from our analysis.

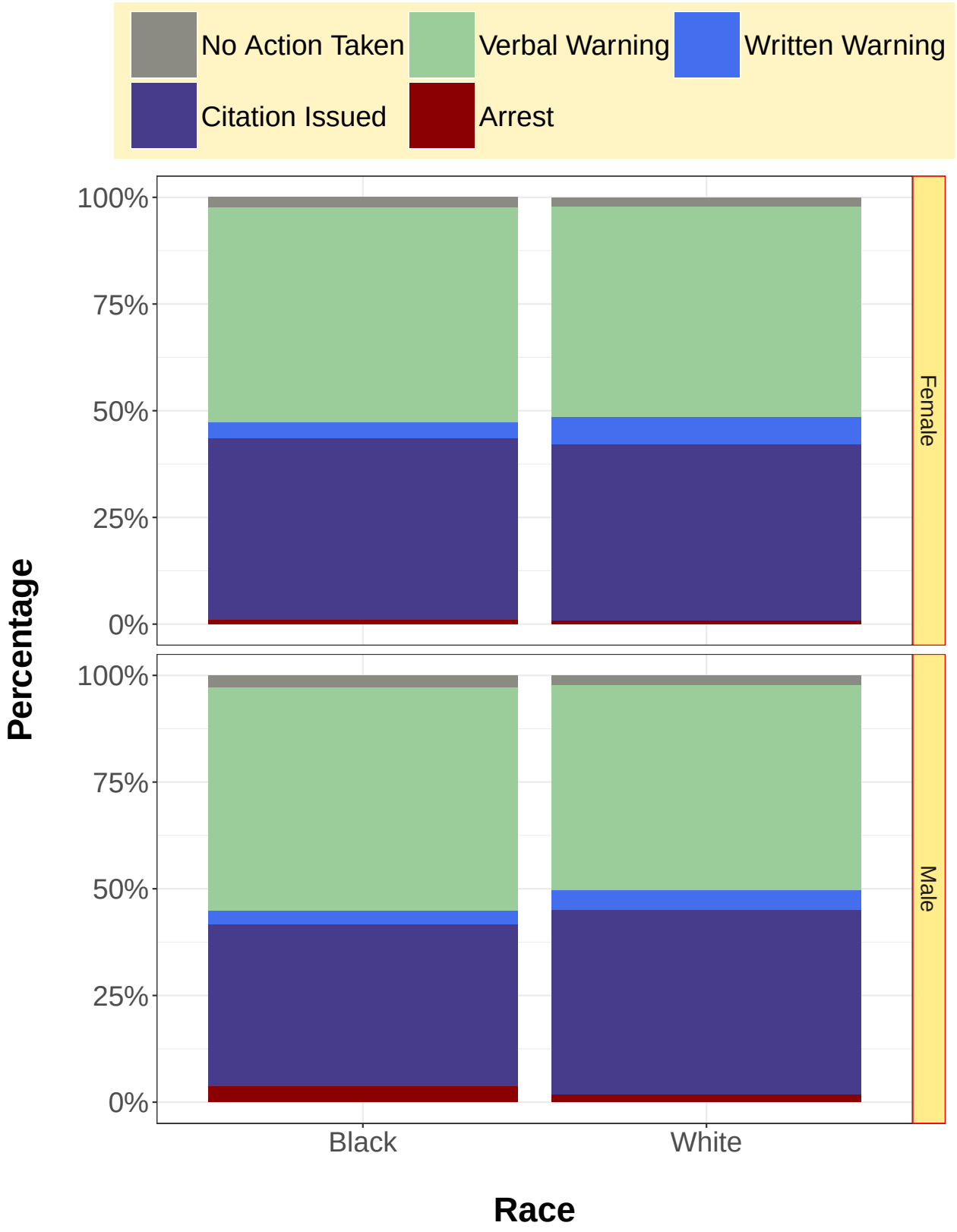
```
officers %<>% filter(driver_race=="Black" | driver_race=="White")
```

I will remove the "Division" part of `cmpd_division` for display purposes.

```
officers %<>% mutate(cmpd_division = gsub("\\s*\\w*$", "",  officers$cmpd_division))
```

I will begin my analysis by looking at the impact on `driver_race` and `driver_gender` with the `result_of_stop` variable. I am using proportions instead of raw counts to scale the data. Looking at the plot on the next page, the bars look more or less the same. There does not seem to be a difference in the final result of the stop when considering *only* race and gender. I will account for age to see if it changes anything.

```
officers %>% group_by(driver_race, result_of_stop, driver_gender) %>% count() %>%
  ggplot(aes(x=driver_race, y = n, fill = result_of_stop))+
    geom_bar(position = "fill",stat = "identity")+
      scale_y_continuous(labels = percent_format()) +
      facet_grid(driver_gender~.)+
      scale_fill_manual(values = c("ivory4","darkseagreen3",
                                   "royalblue2", "slateblue4","red4"))+
    xlab("\nRace") + ylab("Percentage\n") +theme_bw() + theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          strip.background = element_rect(color = "red", fill = 'lightgoldenrod1'),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          legend.text = element_text(size = 18), legend.key.height=unit(3,"line"),
          legend.key.width =unit(3, "line"), legend.position = "top",
          legend.background = element_rect(fill = alpha ('lightgoldenrod1', 0.5))) +
    guides(fill = guide_legend(nrow=2, byrow = T)) + labs(fill="")
```


No Action Taken
Verbal Warning
Written Warning
Citation Issued
Arrest
Percentage
100%
75%
50%
25%
0%
Female
Male
Black
White
Race

Here, I turn age into categorical because logically, there is not much of a difference between, say age 25 or age 26. I will calculate the breakpoints using quantiles. I will also remove the relatively few cases where the driver is under 18 years old to remove any edge cases or outliers.

```
officers %>% filter(driver_age>=18) %>% select(driver_age) %>% unlist() %>% quantile()
```

```
##   0%  25%  50%  75% 100%
##   18   26   34   45   99
```

```
officers %<>% filter(driver_age>=18) %>%
  mutate(age_group = cut(driver_age,
    breaks = c(17,26,34,45,Inf), labels = c("18-26","27-34","35-45","Above 45")))
```

Is there any relationship between the race of the officer and the race of the driver? This table shows that we may get misleading results by including too few variables into the equation. Looking at strictly this table, there does not seem to be any discrimination going on. However, if we consider age, it is a completely different story.

```
officers %<>% mutate(officer_race = gsub("Black/African American", "Black", officers$officer_race))
officers %>% filter(officer_race == "Black" | officer_race == "White") %>% group_by(officer_race, drive
  group_by(is_speeding, officer_race) %>% mutate(percent = n * 100/sum(n)) %>% arrange(desc(n)) %>%
  filter(is_speeding==T) %>% ungroup() %>% select(-is_speeding)%>%
  knitr::kable(digits = 1, align=rep('c',4),
               col.names = c("Officer Race", "Driver Race", "Count","Percentage"))
```

Officer Race	Driver Race	Count	Percentage
White	White	7723	53.7
White	Black	6659	46.3
Black	White	2101	51.7
Black	Black	1961	48.3

Is there any difference in behaviors after considering age?

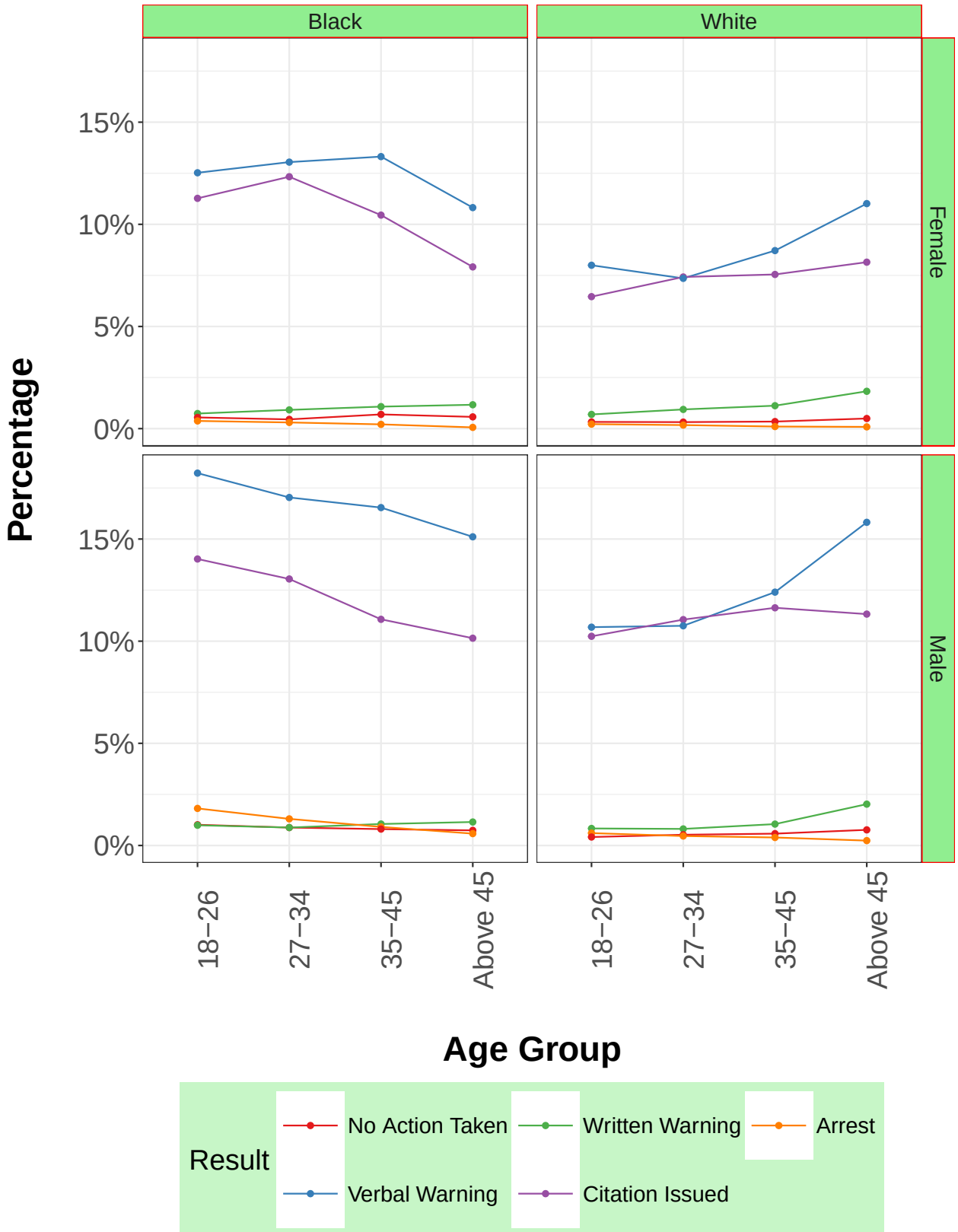
```
officers %>% group_by(driver_race, age_group) %>% count(result_of_stop) %>%
  group_by(age_group) %>% mutate(percent = n * 100/sum(n)) %>% arrange(desc(n)) %>%
  head(10) %>% knitr::kable(digits = 1, align=rep('c',5),
               col.names = c("Driver Race", "Age", "Result","Count","Percentage"))
```

Driver Race	Age	Result	Count	Percentage
Black	18-26	Verbal Warning	6460	30.8
Black	35-45	Verbal Warning	5879	29.9
Black	27-34	Verbal Warning	5382	30.1
Black	18-26	Citation Issued	5314	25.3
Black	27-34	Citation Issued	4540	25.4
White	Above 45	Verbal Warning	4390	26.8
Black	Above 45	Verbal Warning	4242	25.9
Black	35-45	Citation Issued	4238	21.5
White	35-45	Verbal Warning	4158	21.1
White	18-26	Verbal Warning	3925	18.7

From this table, we see a much clearer picture of what is happening. This table is weighted by the four age groups that each comprise of 25 % of the data. In the top 10 occurrences, we see Blacks at the top 5, which consists of 3 Verbal Warnings and 2 Citation Issued. Overall, there are 3 Citation Issued for Blacks but *none*

for Whites. In addition, Whites have fewer Verbal Warnings as well. Here is a plot that shows this. In the plot (on the next page), we see that Blacks tend to get more Verbal Warnings or Citations Issued compared to Whites. It also seems higher for Males than Females.

```
officers %>% group_by(driver_race, age_group, driver_gender) %>% count(result_of_stop) %>%
  group_by(age_group) %>% mutate(percent = n /sum(n)) %>% arrange(desc(n)) %>%
  ggplot(aes(x = age_group, y = percent, group = result_of_stop, color = result_of_stop)) +
    geom_line() + geom_point() + facet_grid(driver_gender~driver_race) +
    scale_color_brewer(palette = "Set1", name = "Result")+
  xlab("\nAge Group") + ylab("Percentage\n") +theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          strip.background = element_rect(color = "red", fill = 'lightgreen'),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          axis.text.x = element_text(angle = 90),
          legend.text = element_text(size = 14),
          legend.key.height=unit(3,"line"), legend.key.width =unit(3, "line"),
          legend.position = "bottom", legend.direction = "horizontal",
          legend.background = element_rect(fill = alpha ('lightgreen', 0.5)))+
  guides(colour = guide_legend(nrow =2, title.position = "left" )) +
  scale_y_continuous(labels = percent_format())
```

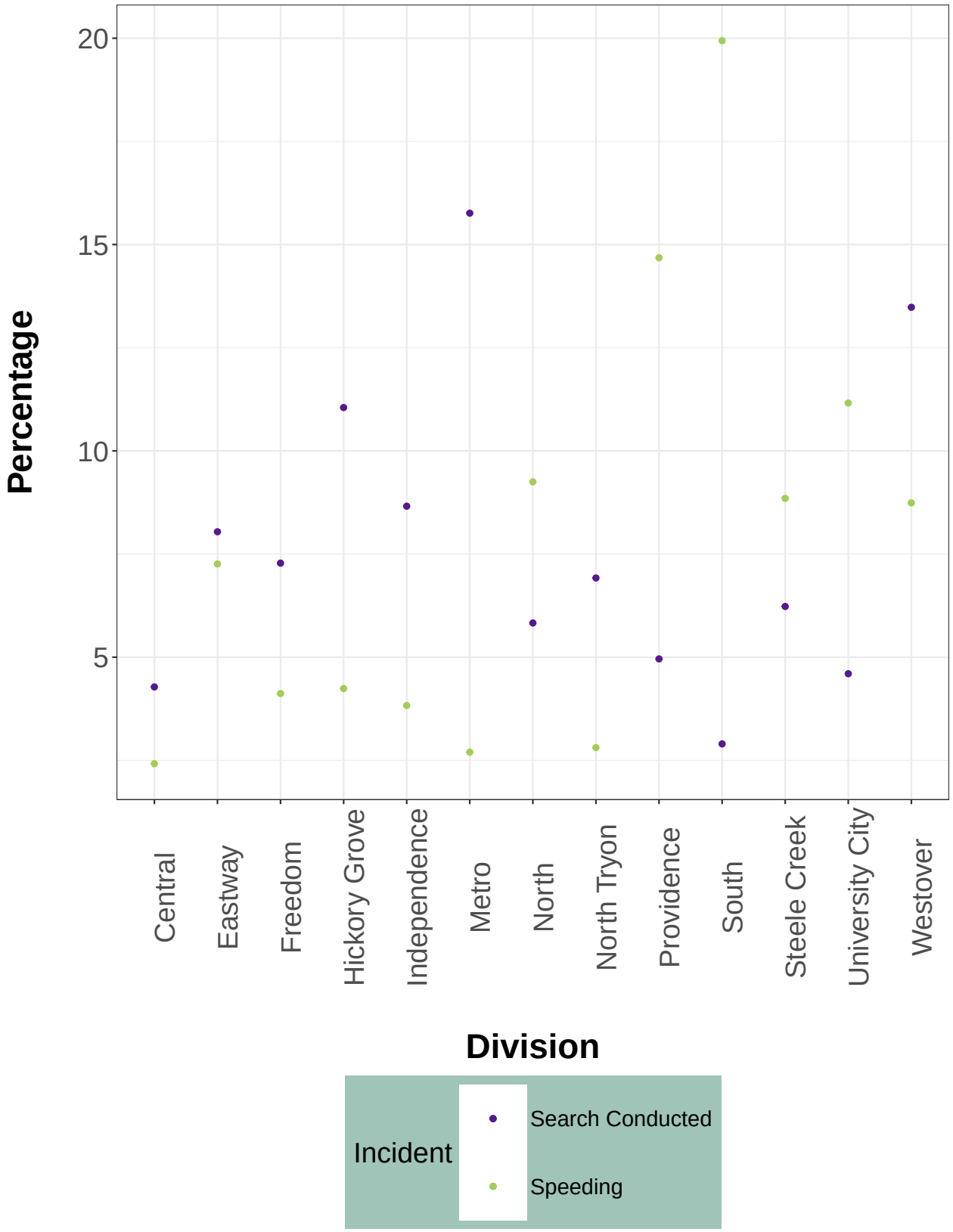

Black
White
Female
Male
Percentage
15%
10%
5%
0%
18−26
27−34
35−45
Above 45
Age Group
Result
No Action Taken
Written Warning
Arrest
Verbal Warning
Citation Issued

Hypothetically, if we were to go to a specific division, which one should we be wary about to avoid getting a speeding ticket? I will make a table of frequency counts and the corresponding percentages of both Speeding and Searching and combine the two tables together. The numbers mean that for instance, in the South Division, there is about a 20% chance of getting caught for speeding and in the Westover Division, there is about a 13.5% chance of having to get your car pulled over and searched.

```
speed <- officers %>% filter(!is.na(cmpd_division)) %>%
   group_by(cmpd_division, is_speeding) %>% summarise(n=n()) %>%
  filter(is_speeding==TRUE) %>% ungroup() %>%
  mutate(percent = round(n * 100 /sum(n), 2)) %>%
   arrange(desc(percent)) %>% select(-is_speeding)
search <- officers %>% filter(!is.na(cmpd_division)) %>%
   group_by(cmpd_division, was_a_search_conducted) %>% summarise(n=n()) %>%
   filter(was_a_search_conducted =="Yes") %>% ungroup() %>%
   mutate(percent = round(n * 100 /sum(n), 2)) %>%
   arrange(desc(percent)) %>% select(-was_a_search_conducted)
full_join(speed, search, by = "cmpd_division") %>%
  knitr::kable(digits = 1, align=rep('c',5),
               col.names = c("Division", "Speed", "% Speed","Search","% Search"))
```

Division	Speed	% Speed	Search	% Search
South	3498	19.9	80	2.9
Providence	2575	14.7	137	5.0
University City	1958	11.2	127	4.6
North	1623	9.2	161	5.8
Steele Creek	1552	8.8	172	6.2
Westover	1533	8.7	372	13.5
Eastway	1273	7.3	222	8.0
Hickory Grove	744	4.2	305	11.1
Freedom	723	4.1	201	7.3
Independence	671	3.8	239	8.7
North Tryon	493	2.8	191	6.9
Metro	473	2.7	435	15.8
Central	424	2.4	118	4.3

```
full_join(speed, search, by = "cmpd_division") %>% ggplot(aes(x=cmpd_division))+
  geom_point(aes(y=percent.x, color = "red")) +
  geom_point(aes(y=percent.y, color = "blue"))+
  xlab("\nDivision") + ylab("Percentage\n") +theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          axis.text.x = element_text(angle = 90),
          legend.text = element_text(size = 14),
          legend.key.height=unit(3,"line"), legend.key.width =unit(3, "line"),
          legend.position = "bottom", legend.direction = "horizontal",
          legend.background = element_rect(fill = alpha ('aquamarine4', 0.5)))+
  guides(colour = guide_legend("Incident",nrow =2, title.position = "left" )) +
  scale_colour_manual(labels = c("Search Conducted", "Speeding"),
                      values = c("purple4", "darkolivegreen3") )
```


Percentage
20
15
10
5
Central
Eastway
Freedom
Hickory Grove
Independence
Metro
North
North Tryon
Providence
South
Steele Creek
University City
Westover
Division
Incident
Search Conducted
Speeding

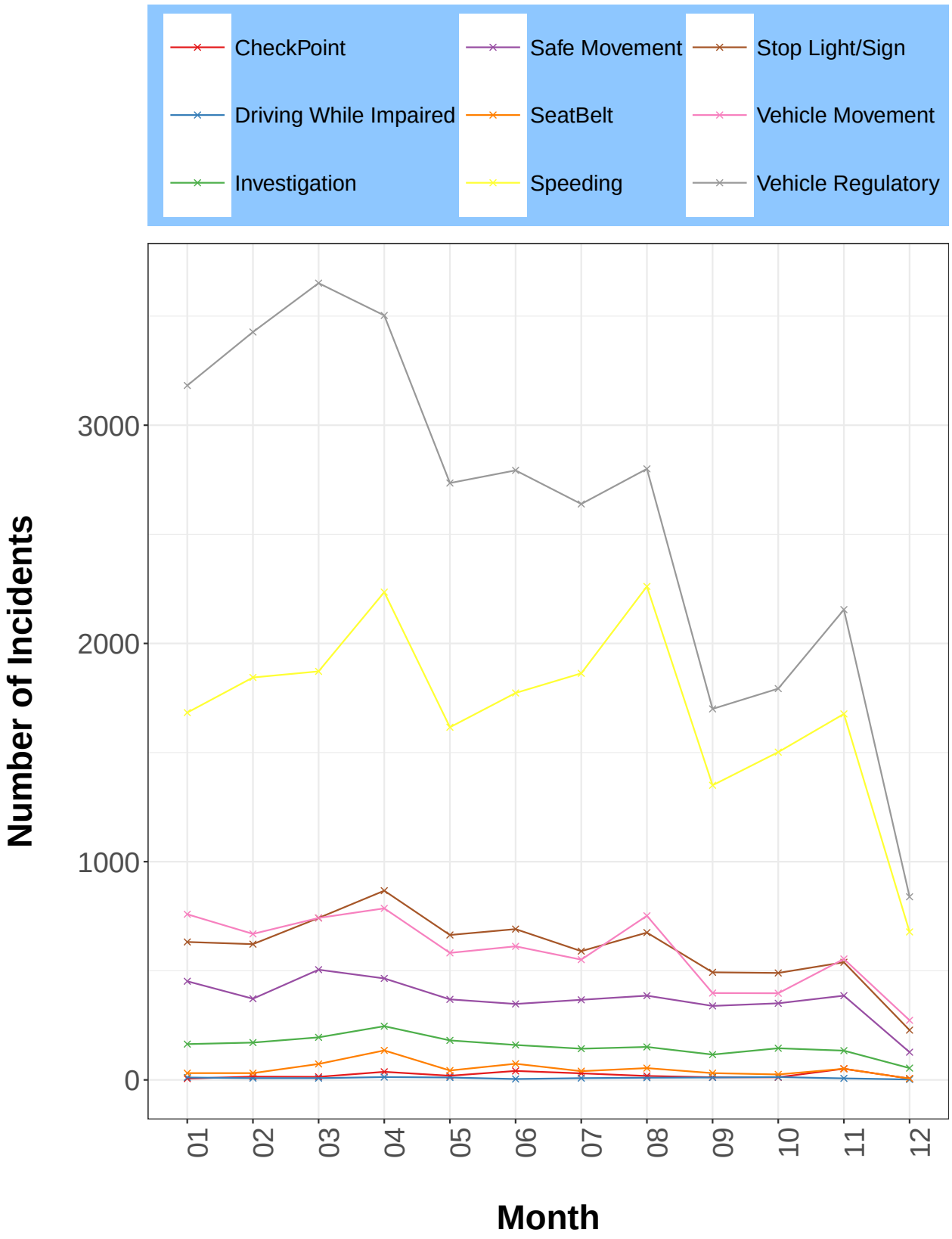
So, those driving in an area controlled by the South or Providence Division should be particularly cautious about going over the speed limit. Those in the Metro or Westover should be careful because there may be a search.

Next, I will look at speeding incidents by month in 2016 in Charlotte, North Carolina.

For Vehicle Regulatory and Speeding related incidents, on average, there seems to be an upward trend up to March and then it decreases as it reaches December. For Speeding, there is a lot of peaks – this may indicate some seasonality in the data. One reasoning for this trend may be that officers may be evaluated quarterly. So, near the end of a specific quarter, they may be more inclined, on average, to stop civilians for Vehicle Regulatory or Speeding violations.

Interestingly, Vehicle Regulatory and Speeding, for the most part, increase and decrease during the same months. Perhaps, these two variables are related in some way.

```
officers %>% group_by(month) %>% count(reason_for_stop) %>%
  filter(reason_for_stop!="Other") %>%
  ggplot(aes(x=month, y=n, group = reason_for_stop,
             color = reason_for_stop))+
  geom_point(shape = 4) + geom_line() +
  xlab("\nMonth") + ylab("Number of Incidents\n") +theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          axis.text.x = element_text(angle = 90),
          legend.text = element_text(size = 14),
          legend.key.height=unit(3,"line"), legend.key.width =unit(3, "line"),
          legend.position = "top", legend.direction = "horizontal",
          legend.background = element_rect(fill = alpha ('dodgerblue1', 0.5)))+
  guides(colour = guide_legend("",nrow =3, title.position = "left" )) +
  scale_color_brewer(palette = "Set1")
```


CheckPoint
Driving While Impaired
Investigation
Safe Movement
SeatBelt
Speeding
Stop Light/Sign
Vehicle Movement
Vehicle Regulatory
Number of Incidents
3000
2000
1000
0
01
02
03
04
05
06
07
08
09
10
11
12
Month

I will create an ordered factor from the variable `officer_years_of_service` with the four quantiles as the break points.

```
officers %>% select(officer_years_of_service) %>% unlist() %>%  quantile()
```

```
##   0%  25%  50%  75% 100%
##    1    4    9   17   35
```

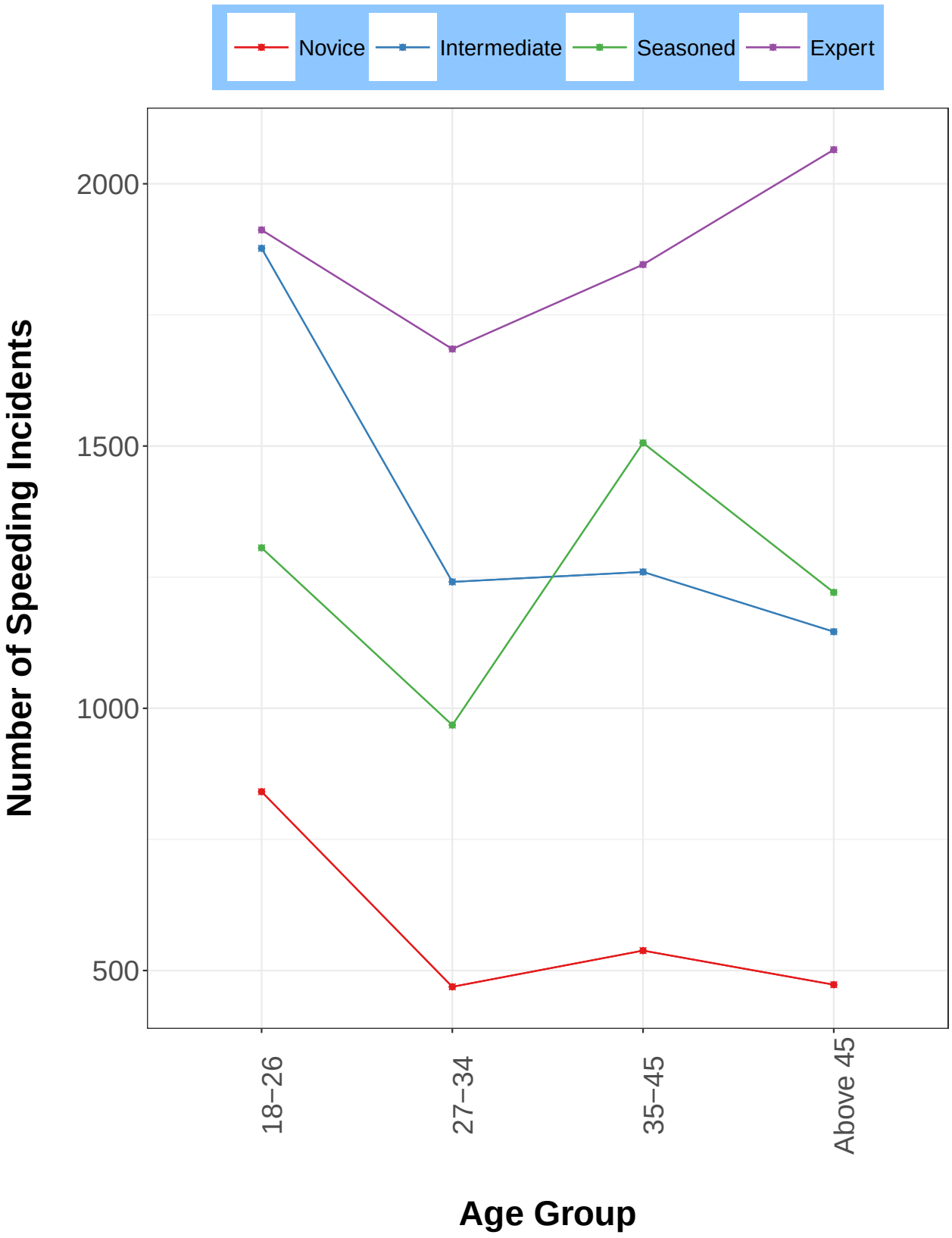
```
officers %<>%
  mutate(officer_exp = cut(officers$officer_years_of_service,
    breaks = c(-Inf,4,9,17, Inf), labels = c("Novice","Intermediate","Seasoned","Expert")))
```

We see that `Expert` Officers seem to stop people for speeding much higher than less-experienced officers. While `Novice`, `Intermediate`, and `Seasoned` seem to go easier on drivers older than 45, `Expert` Officers seem to do the complete opposite. Officers who just started working give stop people for speeding much less, which makes sense because they are not familiar with their job yet.

Conclusion

A main takeaway from looking at this dataset is that we need to consider many different variables simultaneously or you may get misleading results. For example, officer discrimination towards Blacks over Whites is not so clear until we consider Age and Gender along with it. Officers behave differently depending on the division. It is also possible that the people themselves who are getting stopped is the actual cause (but can't really tell from the data available). The seasonal peaks when plotting through time is interesting because I didn't expect a trend to occur. Finally, officer job experience seems to be correlated with age, with officers with high experience having different results than newcomers.

```
officers %>% group_by(age_group, officer_exp) %>% count(is_speeding) %>%
  filter(is_speeding==T) %>% ggplot(aes(x=age_group, y=n, group=officer_exp,
                                        color=officer_exp))+
  geom_point()+geom_line()+
  geom_point(shape = 4) + geom_line() +
  xlab("\nAge Group") + ylab("Number of Speeding Incidents\n") +theme_bw() +
    theme(text = element_text(size=18),
          plot.title = element_text(hjust = 0.5, size = 28, face = "bold"),
          axis.text = element_text(size = 18),
          axis.title = element_text(size = 22, face = "bold"),
          axis.text.x = element_text(angle = 90),
          legend.text = element_text(size = 14),
          legend.key.height=unit(3,"line"), legend.key.width =unit(3, "line"),
          legend.position = "top", legend.direction = "horizontal",
          legend.background = element_rect(fill = alpha ('dodgerblue1', 0.5)))+
  guides(colour = guide_legend("",nrow =1, title.position = "left" )) +
  scale_color_brewer(palette = "Set1")
```


Novice
Intermediate
Seasoned
Expert
Number of Speeding Incidents
2000
1500
1000
500
18–26
27–34
35–45
Above 45
Age Group