Traffic Mortality

Shreya Rao

While the rate of fatal road accidents has been decreasing steadily since the 80's, the past ten years have seen a stagnation in this reduction. Coupled with the increase in number of miles driven in the nation, the total number of traffic related-fatalities has now reached a ten year high and is rapidly increasing.

[This particular dataset was compiled and released as a CSV-file by FiveThirtyEight under the CC-BY4.0 license.]

```
# Check the name of the current folder
current_dir <- getwd()
print(current_dir)
```

```
## [1] "C:/Users/Shreya/Documents/Projects"
```

```
# List all files in this folder
file_list <- list.files()
print(file_list)
```

```
## [1] "miles-driven.csv"      "road-accidents.csv"    "traffic-mortality.Rmd"
```

```
# List files inside the datasets folder
file_list_ds <- list.files("C:/Users/Shreya/Documents/Projects/")
print(file_list_ds)
```

```
## [1] "miles-driven.csv"      "road-accidents.csv"    "traffic-mortality.Rmd"
```

```
# View the first 20 lines of road-accidents.csv in the datasets folder
accidents_head <- readLines("C:/Users/Shreya/Documents/Projects/road-accidents.csv", n=20)
print(accidents_head)
```

```
##  [1] "##### LICENSE #####"
##  [2] "# This data set is modified from the original at fivethirtyeight (https://github.com/fivethirty
##  [3] "# and it is released under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/)"
##  [4] "##### COLUMN ABBREVIATIONS #####"
##  [5] "# drvr_fatl_col_bmiles = Number of drivers involved in fatal collisions per billion miles (201
##  [6] "# perc_fatl_speed = Percentage Of Drivers Involved In Fatal Collisions Who Were Speeding (2009)
##  [7] "# perc_fatl_alcohol = Percentage Of Drivers Involved In Fatal Collisions Who Were Alcohol-Impai
##  [8] "# perc_fatl_1st_time = Percentage Of Drivers Involved In Fatal Collisions Who Had Not Been Inv
##  [9] "##### DATA BEGIN #####"
## [10] "state|drvr_fatl_col_bmiles|perc_fatl_speed|perc_fatl_alcohol|perc_fatl_1st_time"
## [11] "Alabama|18.8|39|30|80"
## [12] "Alaska|18.1|41|25|94"
## [13] "Arizona|18.6|35|28|96"
```

```
## [14] "Arkansas|22.4|18|26|95"
## [15] "California|12|35|28|89"
## [16] "Colorado|13.6|37|28|95"
## [17] "Connecticut|10.8|46|36|82"
## [18] "Delaware|16.2|38|30|99"
## [19] "District of Columbia|5.9|34|27|100"
## [20] "Florida|17.9|21|29|94"
```

```
# Load the tidyverse library
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
# Read in road-accidents.csv and set the comment argument
car_acc <- read_delim("C:/Users/Shreya/Documents/Projects/road-accidents.csv", delim = '|', comment = '#
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   state = col_character(),
##   drvr_fatl_col_bmiles = col_double(),
##   perc_fatl_speed = col_double(),
##   perc_fatl_alcohol = col_double(),
##   perc_fatl_1st_time = col_double()
## )
```

```
# Generate an overview of the data frame
str(car_acc)
```

```
## spec_tbl_df [51 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ state               : chr [1:51] "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ drvr_fatl_col_bmiles: num [1:51] 18.8 18.1 18.6 22.4 12 13.6 10.8 16.2 5.9 17.9 ...
##  $ perc_fatl_speed     : num [1:51] 39 41 35 18 35 37 46 38 34 21 ...
##  $ perc_fatl_alcohol   : num [1:51] 30 25 28 26 28 28 36 30 27 29 ...
##  $ perc_fatl_1st_time  : num [1:51] 80 94 96 95 89 95 82 99 100 94 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   state = col_character(),
##   ..   drvr_fatl_col_bmiles = col_double(),
##   ..   perc_fatl_speed = col_double(),
##   ..   perc_fatl_alcohol = col_double(),
##   ..   perc_fatl_1st_time = col_double()
##   .. )
```

```r
# Display the last six rows of the data frame.
tail(car_acc, 6)
```

```
## # A tibble: 6 x 5
##   state     drvr_fatl_col_bmi~ perc_fatl_speed perc_fatl_alcoh~ perc_fatl_1st_t~
##   <chr>                  <dbl>           <dbl>            <dbl>            <dbl>
## 1 Vermont                 13.6              30               30               95
## 2 Virginia                12.7              19               27               88
## 3 Washingt~               10.6              42               33               86
## 4 West Vir~               23.8              34               28               87
## 5 Wisconsin               13.8              36               33               84
## 6 Wyoming                 17.4              42               32               90
```

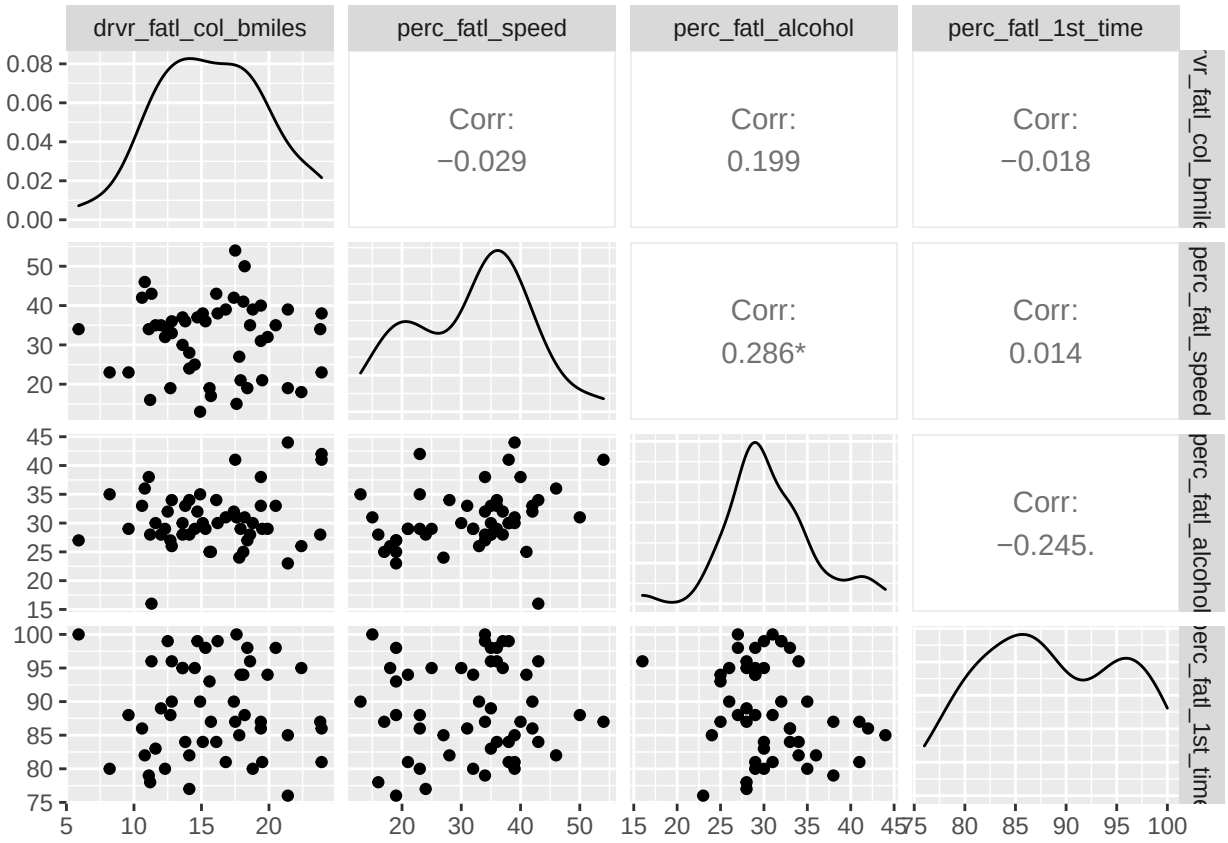
```r
# Compute summary statistics of all columns in the car_acc data frame
dat_summ <- summary(car_acc)
print(dat_summ)
```

```
##     state           drvr_fatl_col_bmiles perc_fatl_speed perc_fatl_alcohol
##  Length:51          Min.   : 5.90        Min.   :13.00   Min.   :16.00
##  Class :character   1st Qu.:12.75        1st Qu.:23.00   1st Qu.:28.00
##  Mode  :character   Median :15.60        Median :34.00   Median :30.00
##                     Mean   :15.79        Mean   :31.73   Mean   :30.69
##                     3rd Qu.:18.50        3rd Qu.:38.00   3rd Qu.:33.00
##                     Max.   :23.90        Max.   :54.00   Max.   :44.00
##  perc_fatl_1st_time
##  Min.   : 76.00
##  1st Qu.: 83.50
##  Median : 88.00
##  Mean   : 88.73
##  3rd Qu.: 95.00
##  Max.   :100.00
```

```r
# Deselect the state column and create a pairwise scatterplot
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
car_acc %>%
    select(-state) %>%
    ggpairs()
```




```
# Using pipes, remove the state column and then compute the correlation coefficient for all column pairs
corr_col <- car_acc %>% select(-state) %>% cor()
# Print the correlation coefficient for all column pairs
print(corr_col)
```

```
##                      drvr_fatl_col_bmiles perc_fatl_speed perc_fatl_alcohol
## drvr_fatl_col_bmiles           1.00000000     -0.02908015         0.1994263
## perc_fatl_speed               -0.02908015      1.00000000         0.2862442
## perc_fatl_alcohol              0.19942634      0.28624417         1.0000000
## perc_fatl_1st_time            -0.01794188      0.01406622        -0.2454551
##                      perc_fatl_1st_time
## drvr_fatl_col_bmiles        -0.01794188
## perc_fatl_speed              0.01406622
## perc_fatl_alcohol           -0.24545506
## perc_fatl_1st_time           1.00000000
```

From the correlation table, I see that the amount of fatal accidents is most strongly correlated with alcohol consumption (first row). But in addition, I also see that some of the features are correlated with each other, for instance, speeding and alcohol consumption are positively correlated. I, therefore, want to compute the association of the target with each feature while adjusting for the effect of the remaining features. This can be done using multivariate linear regression.

```
# Use lm to fit a multivariate linear regression model
fit_reg <- lm(drvr_fatl_col_bmiles~perc_fatl_speed+perc_fatl_alcohol+perc_fatl_1st_time, data=car_acc)
```

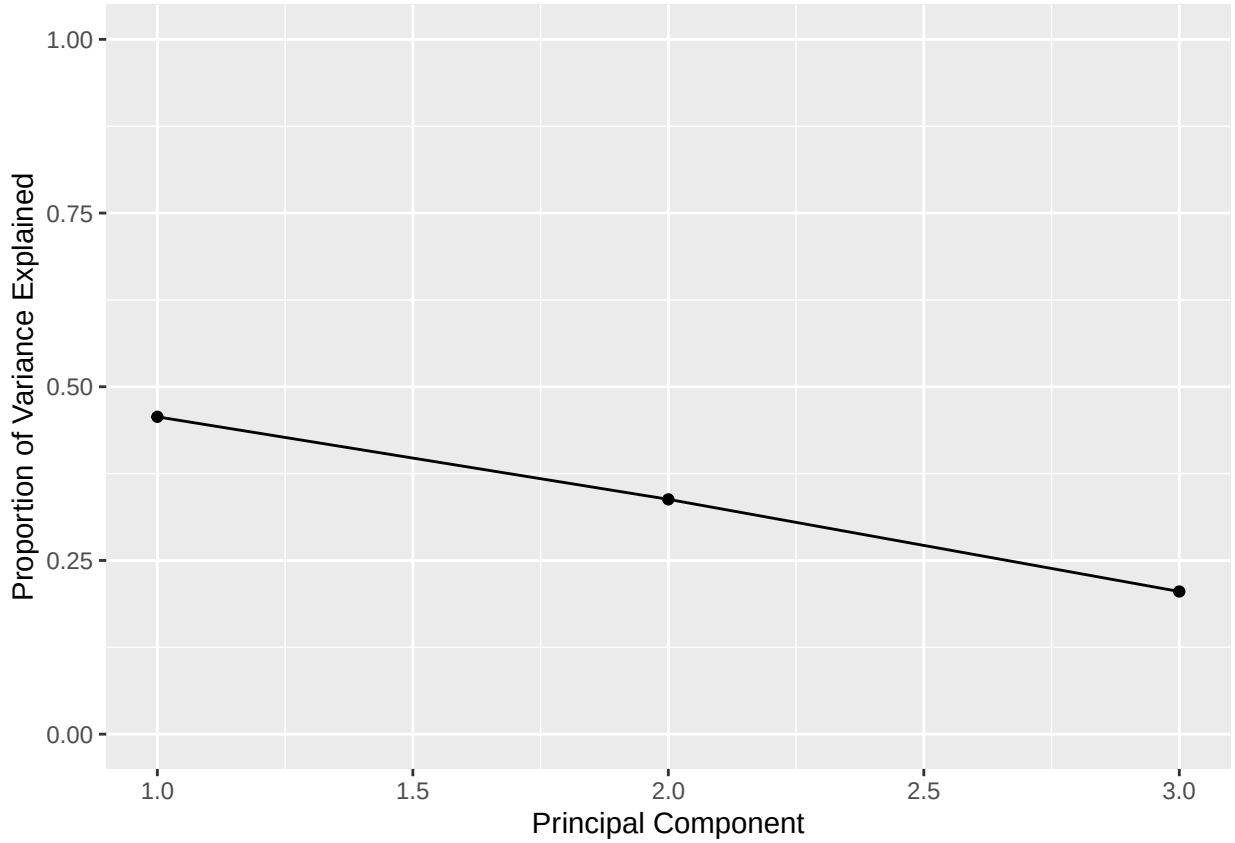
```r
# Retrieve the regression coefficients from the model fit
fit_coef <- coef(fit_reg)
print(fit_coef)
```

```
##        (Intercept)    perc_fatl_speed  perc_fatl_alcohol perc_fatl_1st_time
##         9.06498048        -0.04180041         0.19086404         0.02473301
```

PCA

```r
# Center and standardise the three feature columns
car_acc_standised <- car_acc %>%
    mutate(perc_fatl_speed=scale(perc_fatl_speed),
           perc_fatl_alcohol=scale(perc_fatl_alcohol),
           perc_fatl_1st_time=scale(perc_fatl_1st_time))

# Perform PCA on standardized features
pca_fit <- princomp(car_acc_standised[,c("perc_fatl_speed",
                    "perc_fatl_alcohol", "perc_fatl_1st_time")])

# Obtain the proportion of variance explained by each principle component
pr_var <- pca_fit$sdev^2
pve <- pr_var / sum(pr_var)

# Plot the proportion of variance explained, draw a point plot connected with lines
data_frame( comp_id=1:length(pve) , pve ) %>%
ggplot( aes(x=comp_id , y=pve) ) + geom_point() + geom_line() +
coord_cartesian(ylim=c(0,1)) +
labs(x="Principal Component",
     y="Proportion of Variance Explained")
```

```
## Warning: `data_frame()` was deprecated in tibble 1.1.0.
## Please use `tibble()` instead.
```

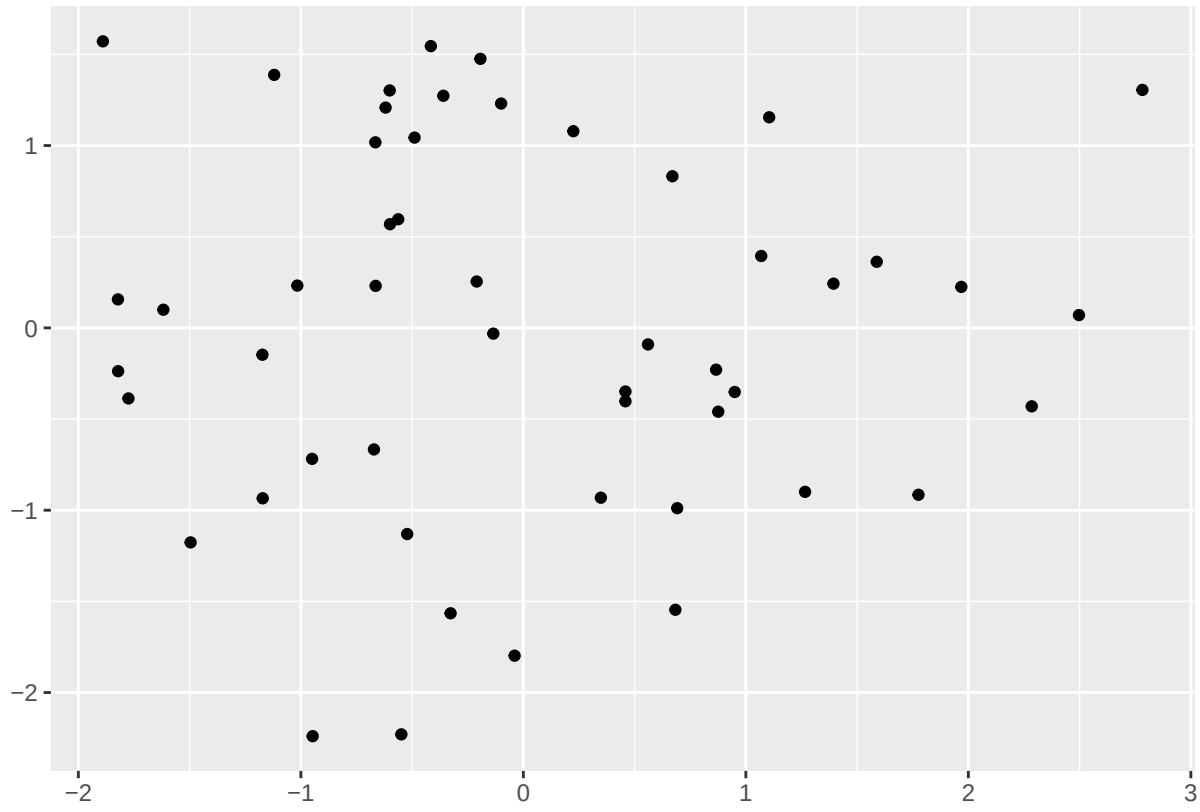



```
# Compute the cumulative proportion of variance and extract the variance
# explained by the first two principal components
cve <- cumsum(pve)
cve_pc2 <- cumsum(pve)
print(cve_pc2)
```

```
##    Comp.1    Comp.2    Comp.3
## 0.4567308 0.7946979 1.0000000
```

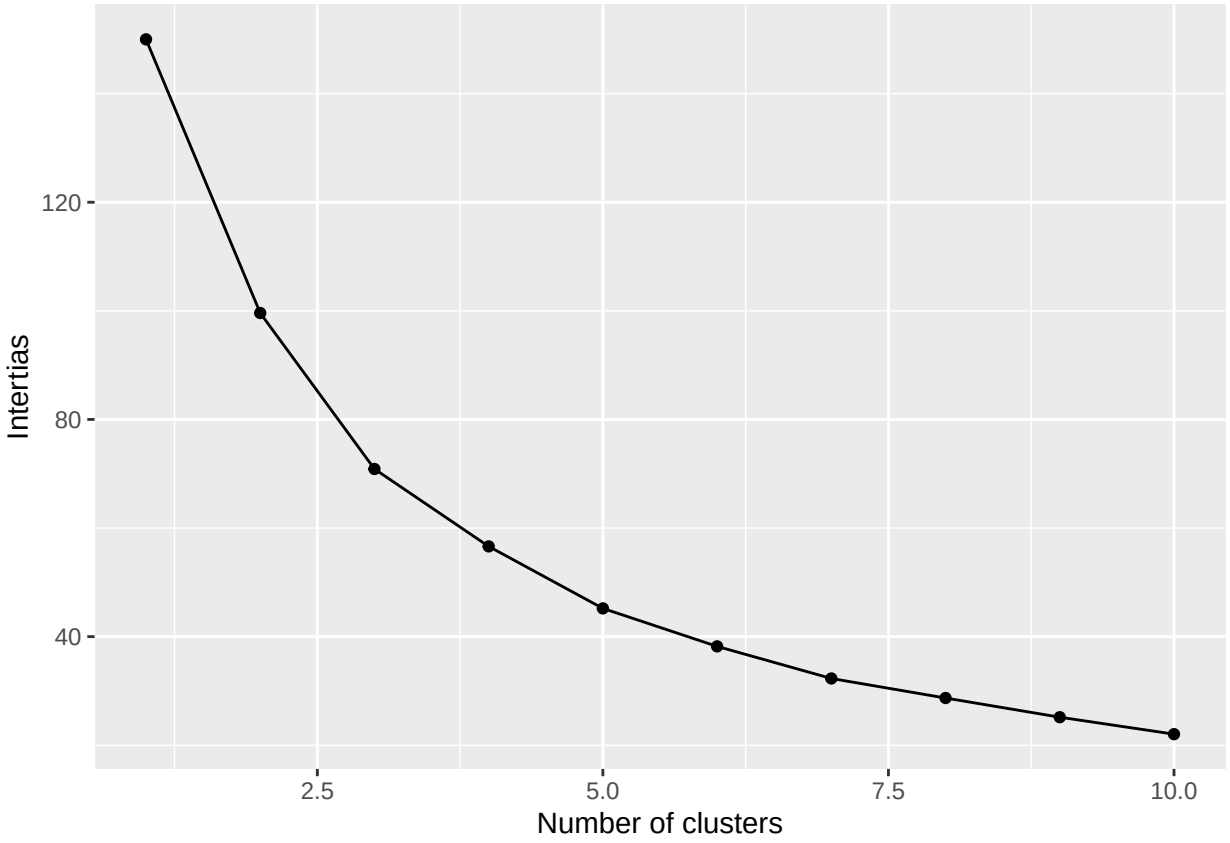
Visualize the First 2 Principle Components

```
# Get the principle component scores from the PCA fit
pcomp1 <- pca_fit$scores[,1]
pcomp2 <- pca_fit$scores[,2]

# Plot the first 2 principle components in a scatterplot using ggplot
data_frame(pcomp1,pcomp2) %>%
ggplot( aes(pcomp1, pcomp2)) +
    geom_point() +
    labs(x="",y="")
```

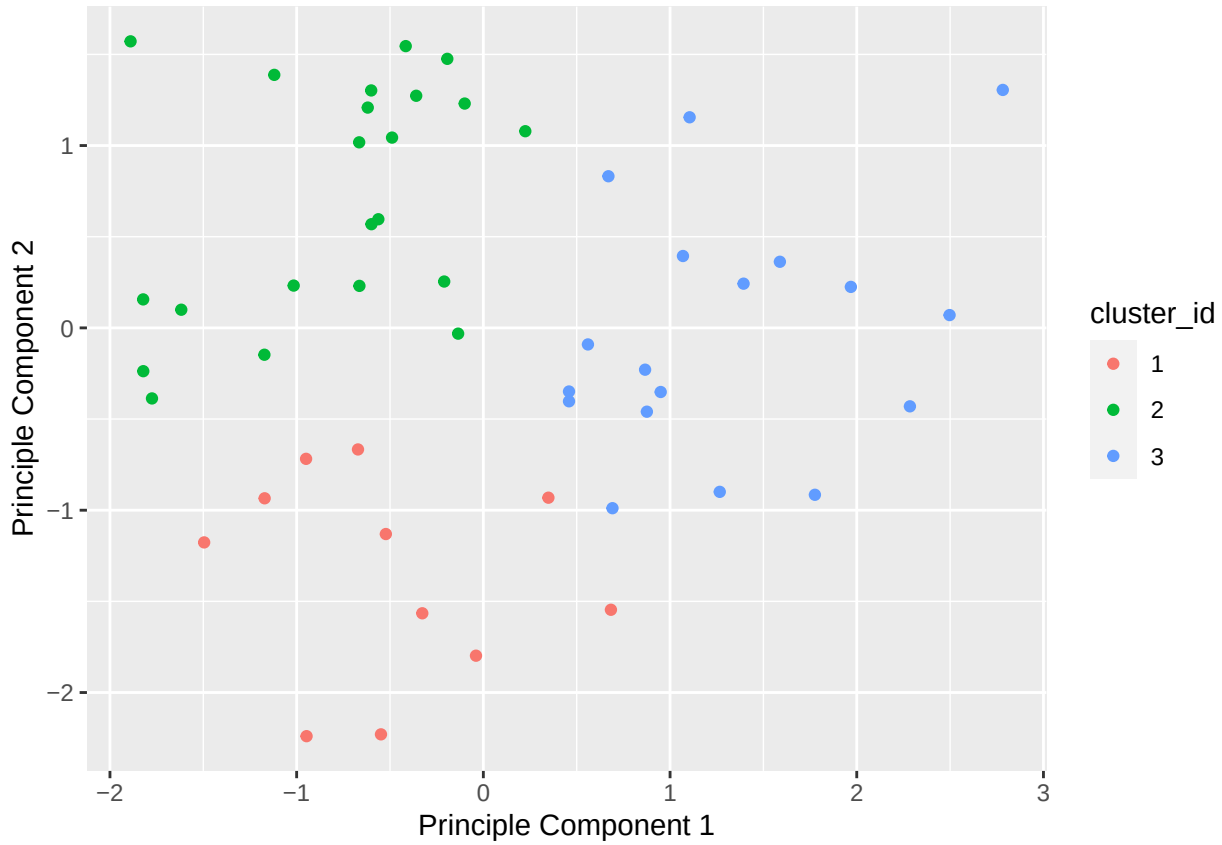


Find Clusters of Similar States in the Data

```
# Create a vector of 1 to 10
k_vec <- 1:10
# Initialise vector of inertias
inertias <- rep(NA, length(k_vec))
# Initialise empty list to save K-mean fits
mykm <- list()

for (k in k_vec) {
    # for each k, fit a K-mean model with k clusters and save it in the mykm list
    mykm[[k]] <- kmeans(car_acc_standised[,c(3,4,5)], centers=k, nstart=50)
    # for each k, get the within-cluster sum-of-squares and save
    inertias[k] <- mykm[[k]]$tot.withinss
}

# Plot the within-cluster sum-of-squares against the number of clusters used
data_frame(k_vec,inertias) %>%
ggplot( aes(k_vec, inertias) ) +
geom_point() + geom_line() +
labs(x="Number of clusters", y="Intertias")
```

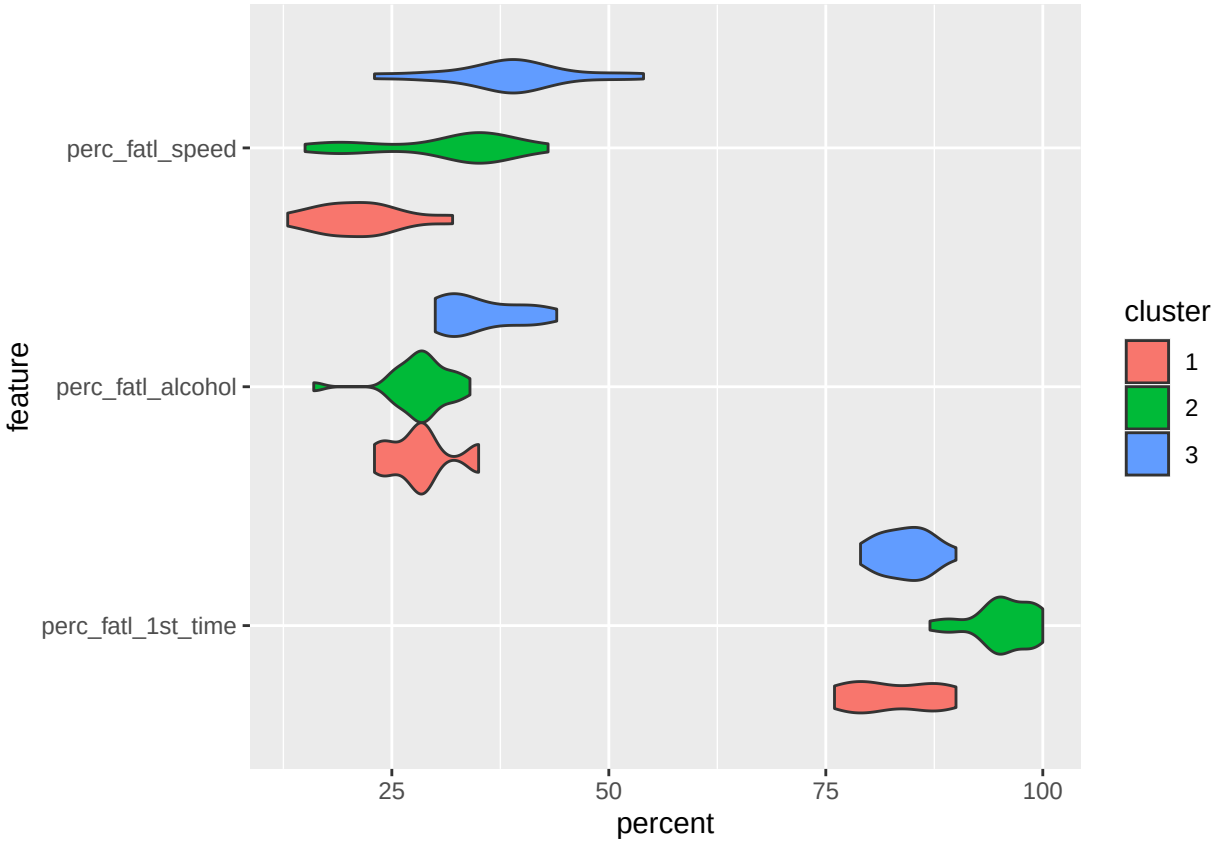



```
# Obtain cluster-ids from the kmeans fit with k=3
cluster_id <- as.factor(mykm[[3]]$cluster)

# Color the points of the principle component plot according to their cluster number
data_frame(pcomp1,pcomp2) %>%
ggplot(aes(x=pcomp1,y=pcomp2)) + geom_point(aes(col=cluster_id)) +
labs(x="Principle Component 1",
    y="Principle Component 2")
```



Visualize the Feature Differences between the Clusters

```
# Add cluster_id to the original data frame
car_acc$cluster <- cluster_id

# Get the data into long format and plot
car_acc %>%
    select(-drvr_fatl_col_bmiles) %>%
    gather(key=feature, value=percent, -state, -cluster) %>%
    ggplot(aes(x=feature,y=percent, fill=cluster)) +
    geom_violin() +
    coord_flip()
```


perc_fatl_speed
perc_fatl_alcohol
perc_fatl_1st_time
feature
25
50
75
100
percent
cluster
1
2
3