# Introduction to ggplot2

*Jason Kinyua*

*1/10/2020*

## Pre-requisites

- Install required libraries

```
install.packages(
  c('dplyr','ggplot2','magrittr','quantreg','ggridges'),
  deps=T,
  respos='https://cran.r-project.org'
  )
```

- Download datasets
  - Workshop dataset was provided together with other training materials on ICRAFuseR Slack channel
  - The dataset is an extract from Uber Movement
  - Explore more datasets at pathmind

## A recap of dplyr grammar

| Grammar | Description | Use Case |
|---|---|---|
| mutate() | adds new variables that are functions of existing variables | mutate(df, col_name=col_1+col_2) |
| select() | picks variables based on their names | select(df, -c("col_1","col_2")) |
| filter() | picks cases based on their values | filter(df ,col_1>5, col_2 <=20) |
| summarise() | reduces multiple values down to a single summary | summarise(df, count=n()) |
| arrange() | changes the ordering of the rows | arrange(df, desc(date)) |
| group_by() | allows you to perform any operation "by group" | df %>% group_by(column_1) %>% summarise(count=n(), mean=mean(column_2, na.rm=TRUE)) |

## Grammar of Graphics (ggplot) | Components

1. data

- The data used to produce the plot

2. aesthetic mappings

- between variables and visual properties

3. layer(s)

- usually through the geom function to produce geometric shape to be rendered

## Geoms for two continuous variables

| Geoms | Description | Code |
|---|---|---|
| jitter | Jitter points (to avoid overlapping) | geom_jitter() |
| point | Plot points at each x | y intersection |
| quantile | Plot lines from quantile regression | geom_quantile() |
| rug | Plot 1d scatterplot on margins (stripchart) | geom_rug() |
| smooth | Plot a smoothing function (many smoothers available) | geom_smooth() |
| text | Add text annotations | geom_text() |
| bin2d | Bin observations that are close together and color according the density | geom_bin2d() |
| density2d | Contour lines of the data density | geom_density2d() |
| contours | Surface contour plots | geom_contour() |
| hex | Hexagonal bins of data colored according to their density | geom_hex() |

## Geoms for One variable

| Geoms | Description | Code |
|---|---|---|
| area | Filled area plot | geom_area(stat = "bin") |
| density | Density plot | geom_density() |
| dotplot | Stacked dotplot, with each dot representing an observation | geom_dotplot() |
| polygon of Frequencies | Polygon of frequencies | geom_freqpoly |
| histogram | Standard histogram | geom_histogram |
| barplot | Standard barchart | geom_bar |

## Enough, Show me the Code!

```
# Load required libraries
# Uncomment below line to install if not installed
# install.packages(c('dplyr','ggplot2','magrittr'),deps=T,respos='https://cran.r-project.org')
library(dplyr)
library(ggplot2)
library(magrittr)
```

```
data <- read.csv('movement-speeds-quarterly-by-hod-nairobi-2019-Q2.csv')
dim(data)
```

```
## [1] 518005     13
```

### Inspect Dataset Variables

```
names(data)
```

```
##  [1] "year"              "quarter"           "hour_of_day"
##  [4] "segment_id"        "start_junction_id" "end_junction_id"
##  [7] "osm_way_id"        "osm_start_node_id" "osm_end_node_id"
## [10] "speed_kph_mean"    "speed_kph_stddev"  "speed_kph_p50"
## [13] "speed_kph_p85"
```

### Fewer data variables

```
clean_data <- data %>%
  select(c('hour_of_day','osm_start_node_id','osm_end_node_id','speed_kph_mean'))
dim(clean_data)
```

```
## [1] 518005      4
```

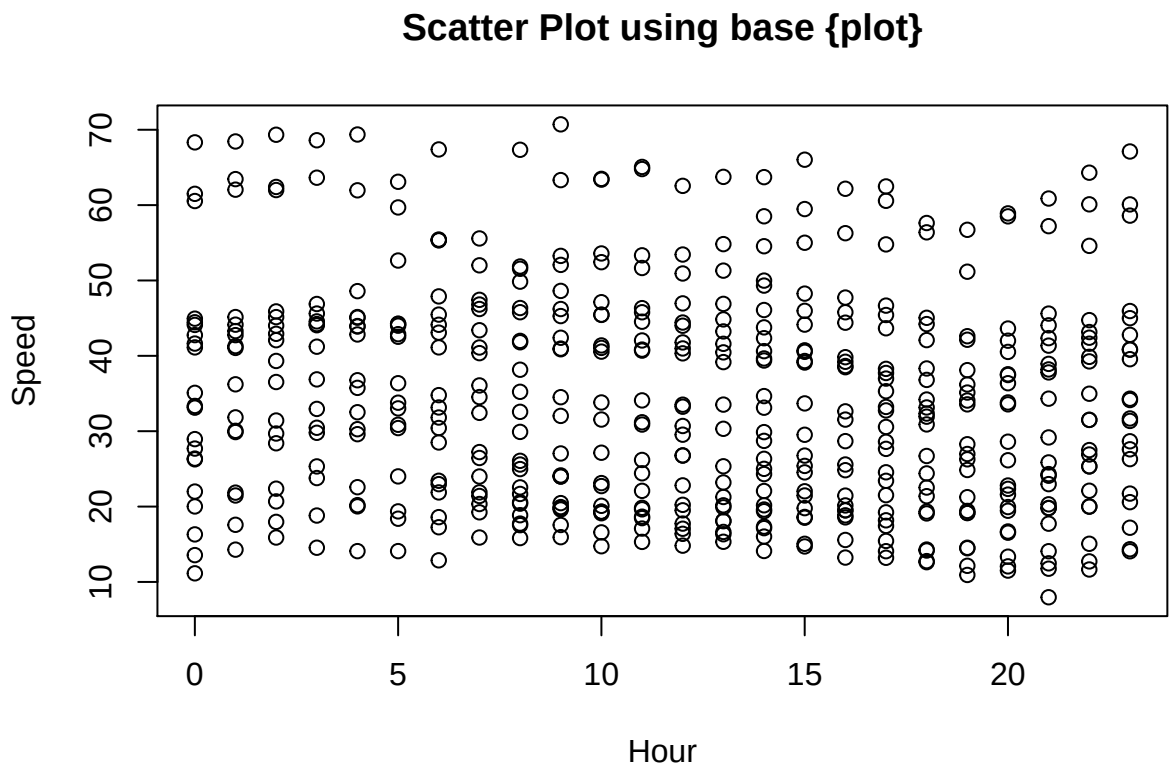
### Trim workshop dataset to 500 rows

```
lab_data <- clean_data %>% slice(c(1:500))
dim(lab_data)
```

```
## [1] 500   4
```

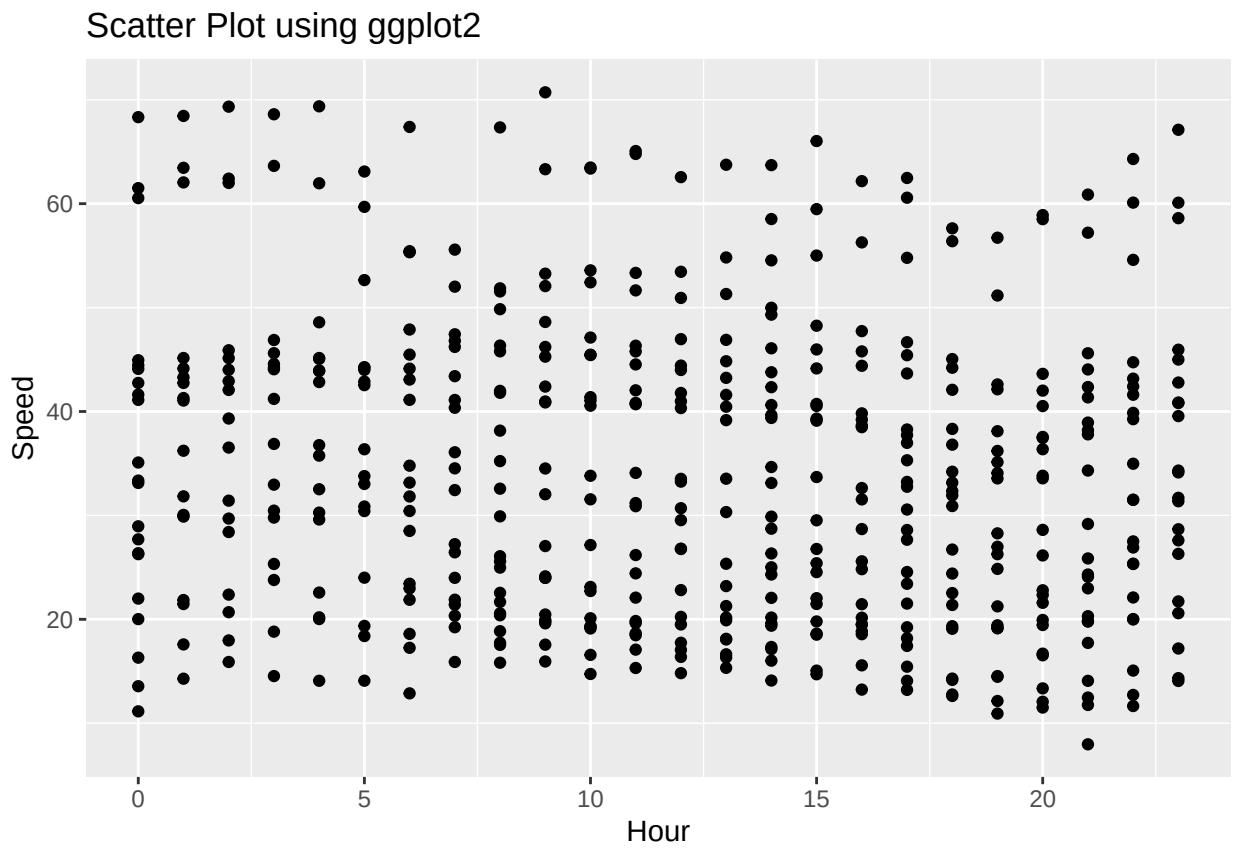
### Scatter plot (base {plot})

```
plot(
  x=lab_data$hour_of_day,
  y=lab_data$speed_kph_mean,
  xlab="Hour",ylab="Speed",
  main="Scatter Plot using base {plot}"
  )
```



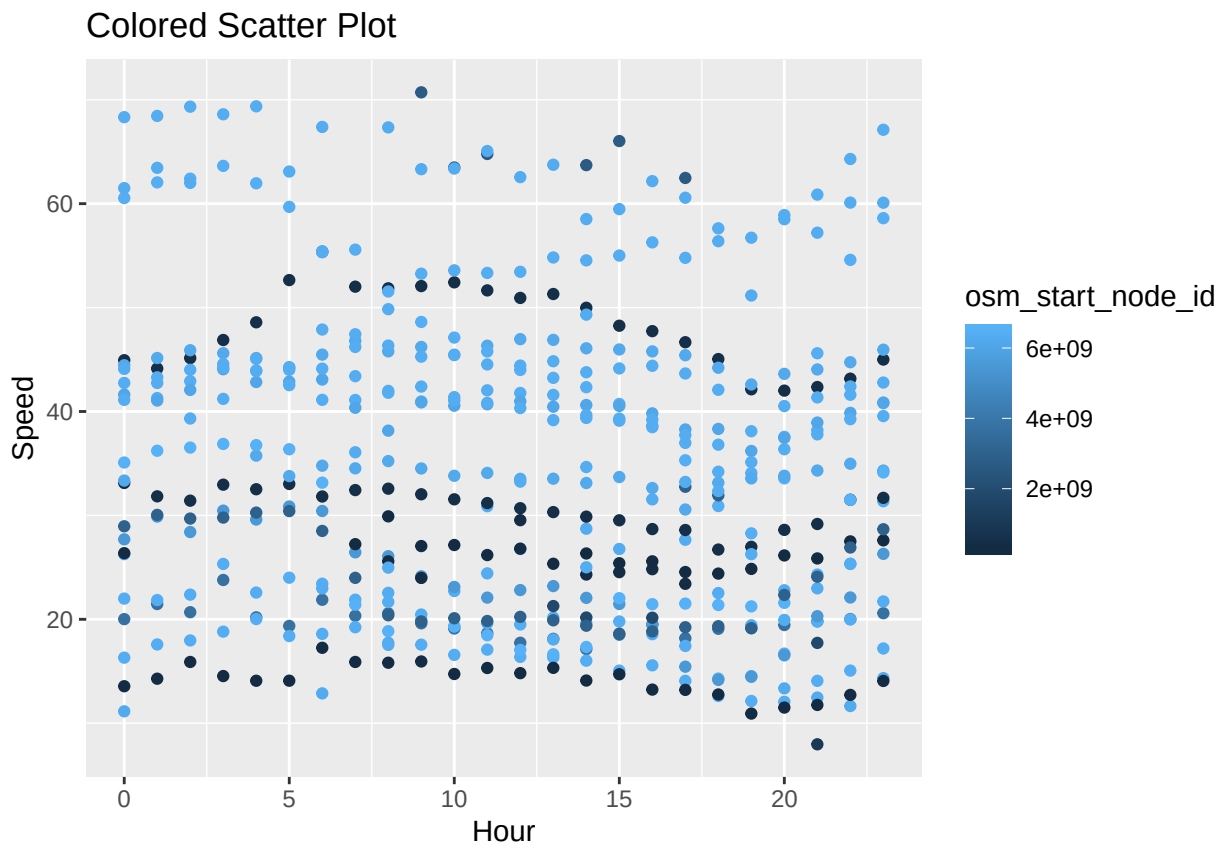

### Scatter Plot {ggplot}

```
ggplot(lab_data,aes(x=hour_of_day,y=speed_kph_mean)) +
  geom_point() +
  labs(
    title = "Scatter Plot using ggplot2",
    x = "Hour",
    y = "Speed"
  )
```



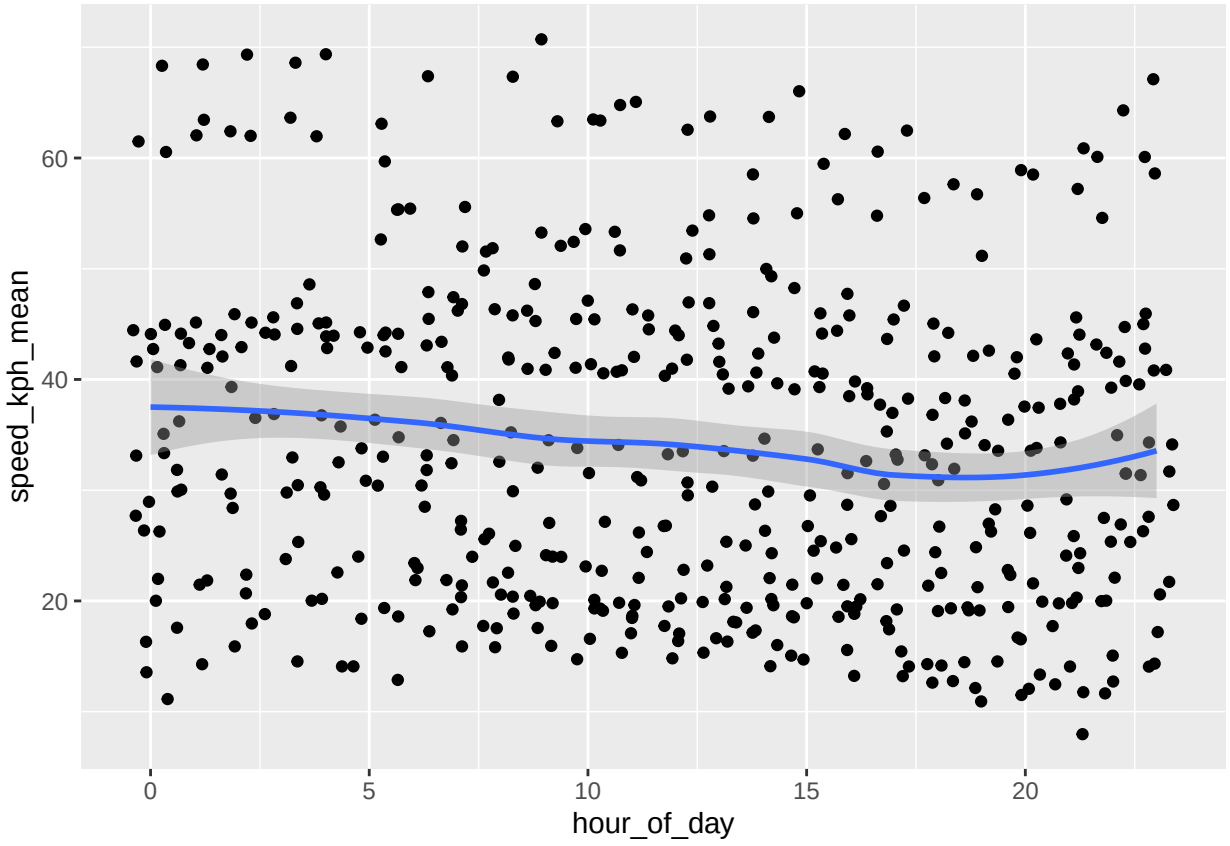

## Color Styling ggplot Graphics

```
ggplot(lab_data,aes(x=hour_of_day,y=speed_kph_mean, color=osm_start_node_id)) +
  geom_point() +
  labs(
    title = "Colored Scatter Plot",
    x = "Hour",
    y = "Speed"
  )
```



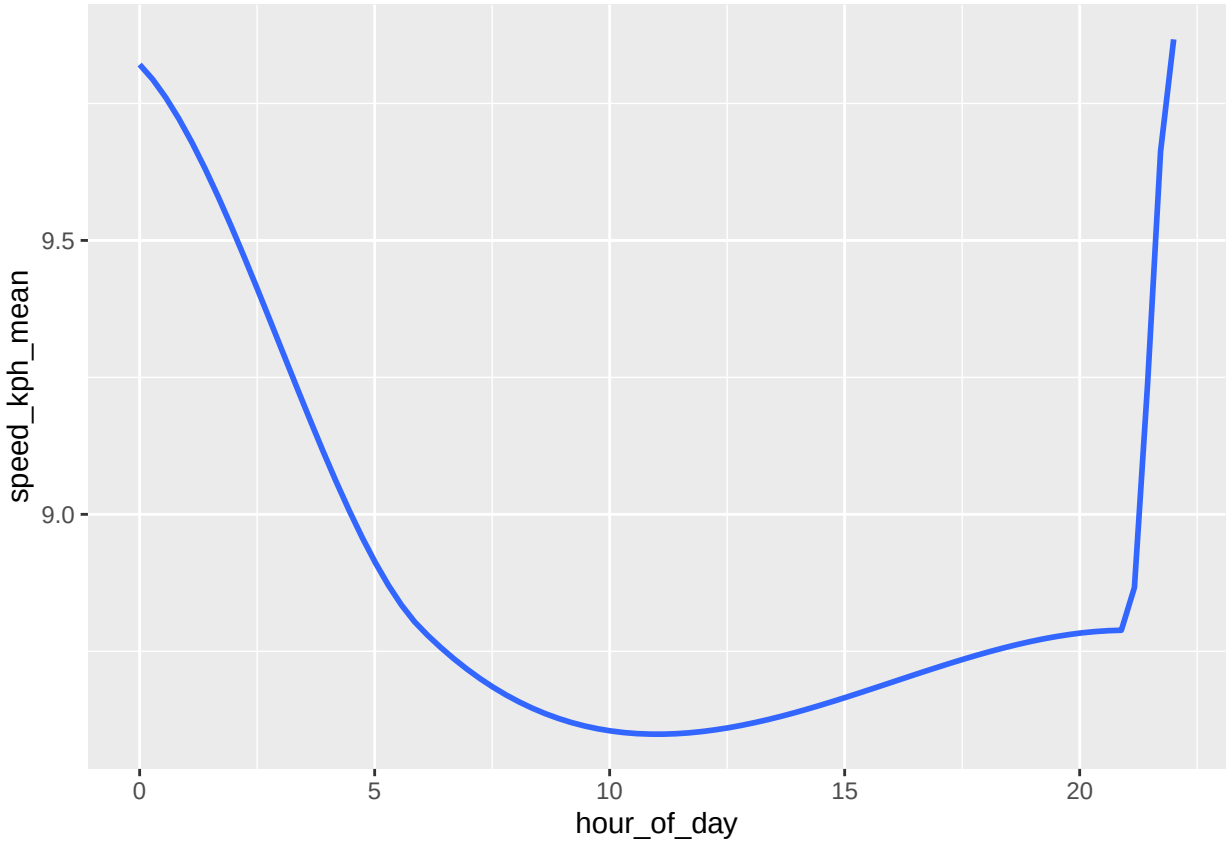

### Jitter + Smooth Line

```
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_jitter() +
  geom_smooth()
```



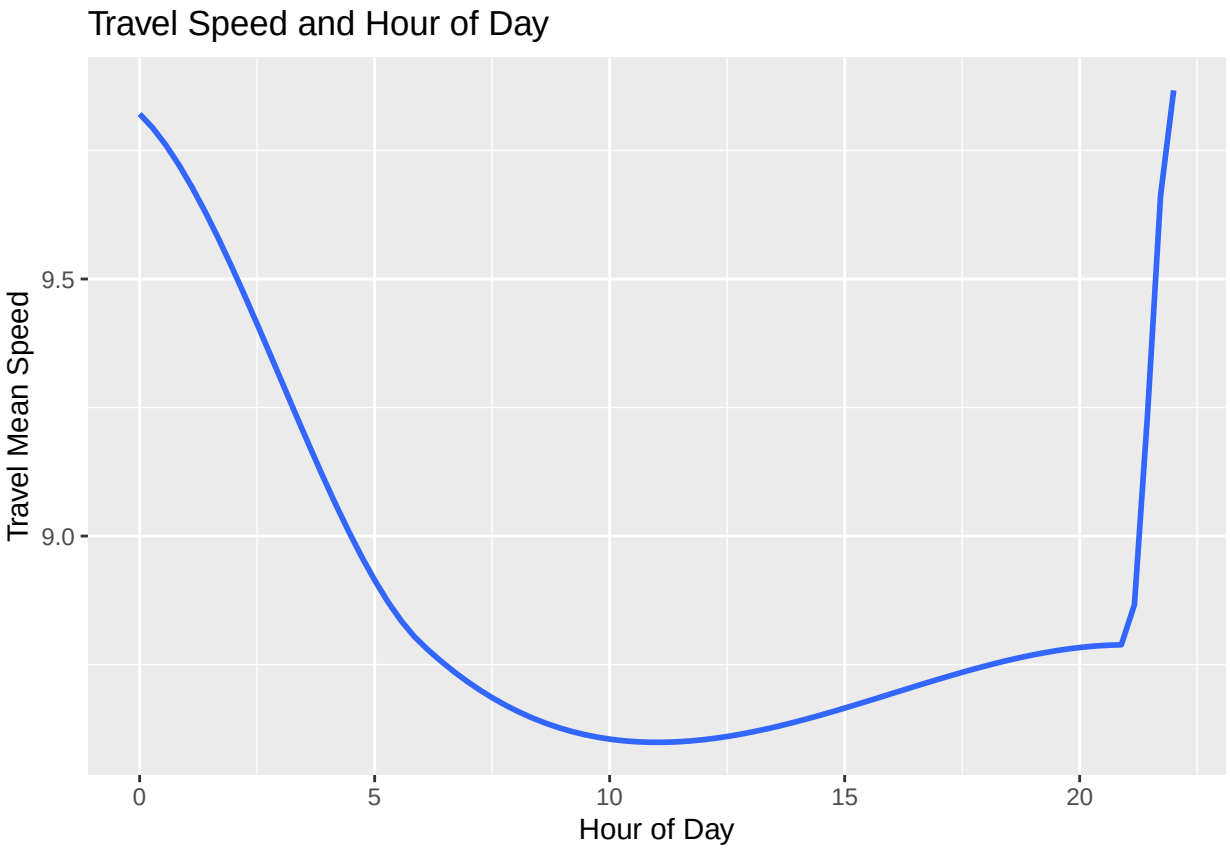

### Travel speed from a given start point

```
graph <- clean_data %>%
  filter(osm_start_node_id == lab_data$osm_end_node_id[1]) %>%
  ggplot(aes(hour_of_day,speed_kph_mean)) +
  geom_smooth(se=FALSE)
graph
```



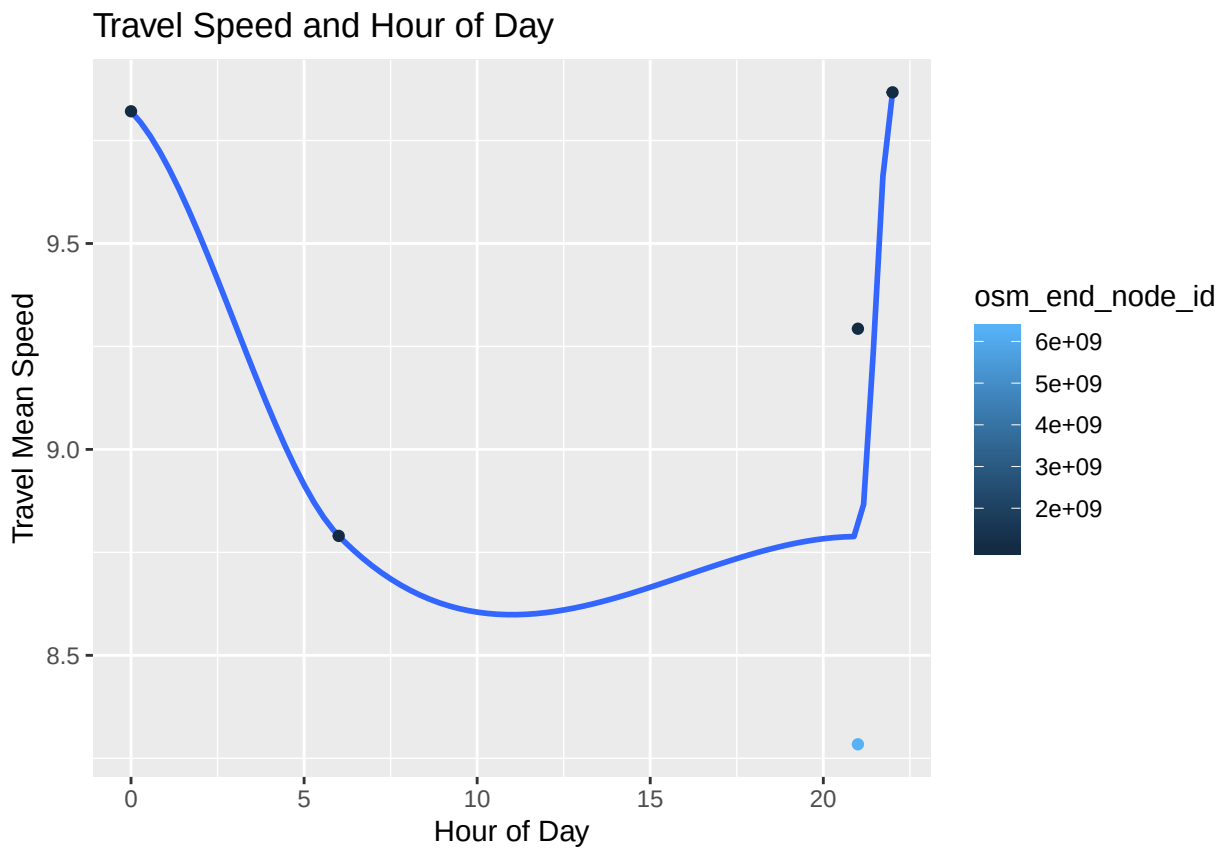

### Customize labels

```
labelled <- graph +
  xlab("Hour of Day") +
  ylab("Travel Mean Speed") +
  ggtitle("Travel Speed and Hour of Day")
labelled
```



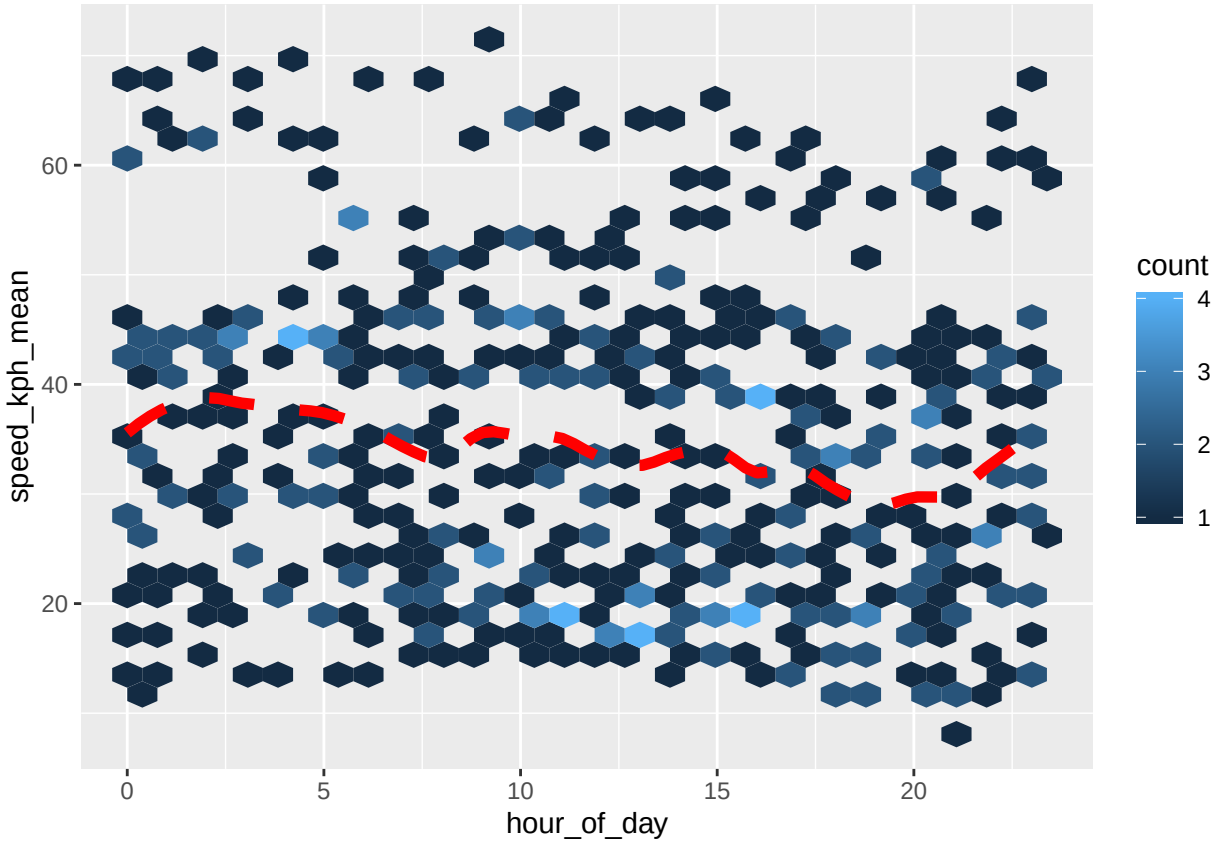

### Add Scatter points

```
pts <- labelled + geom_point(aes(color=osm_end_node_id))
pts
```



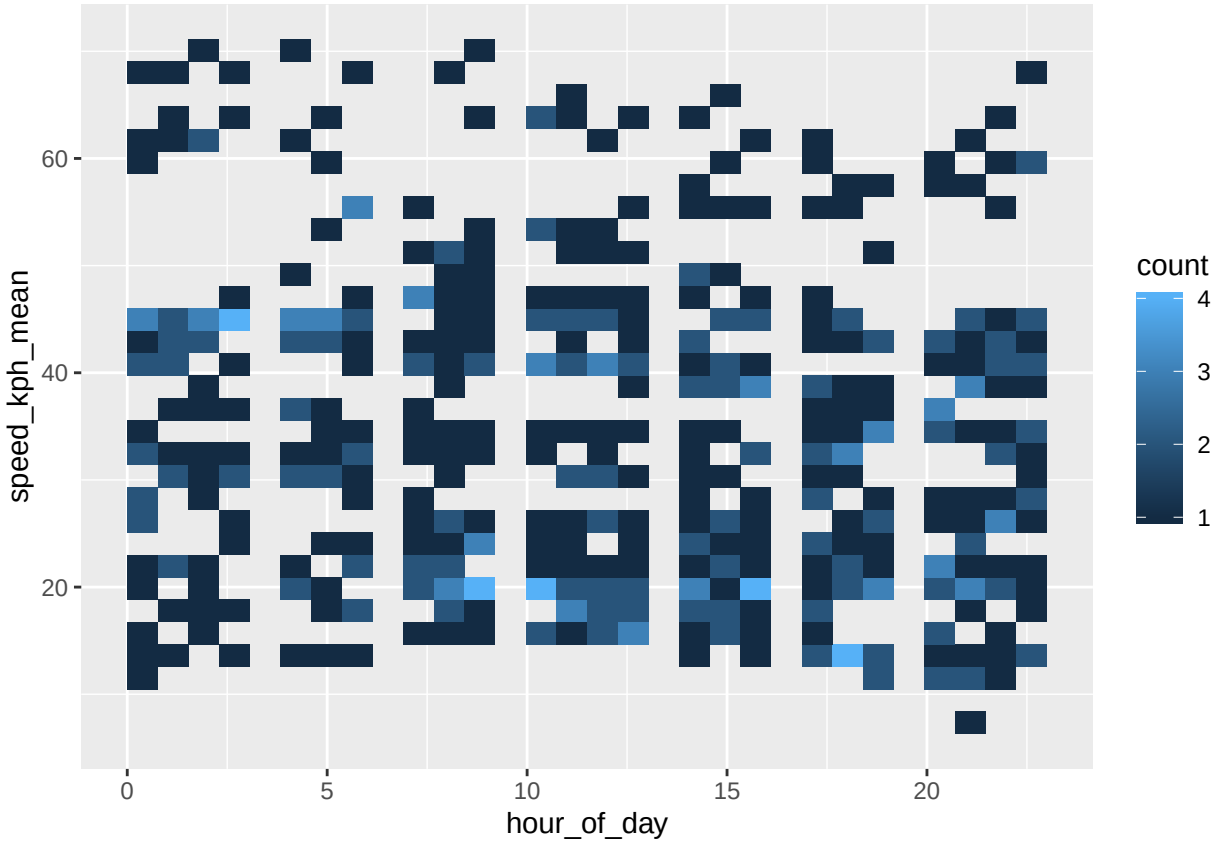

### Hex plot

```
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_hex() +
  geom_smooth(span=0.2,color='red',size=2,se=FALSE,linetype="dashed")
```



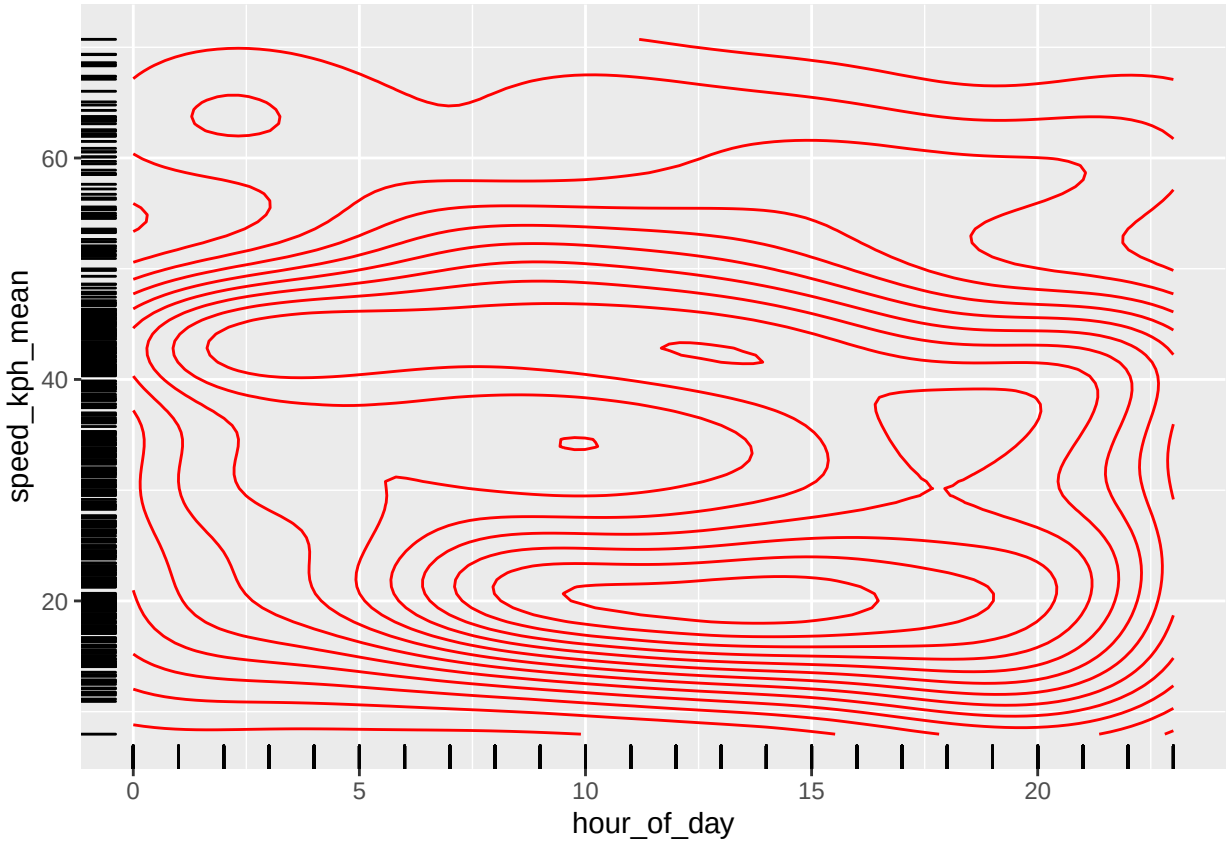

### Bin2d Plot

```
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_bin2d()
```



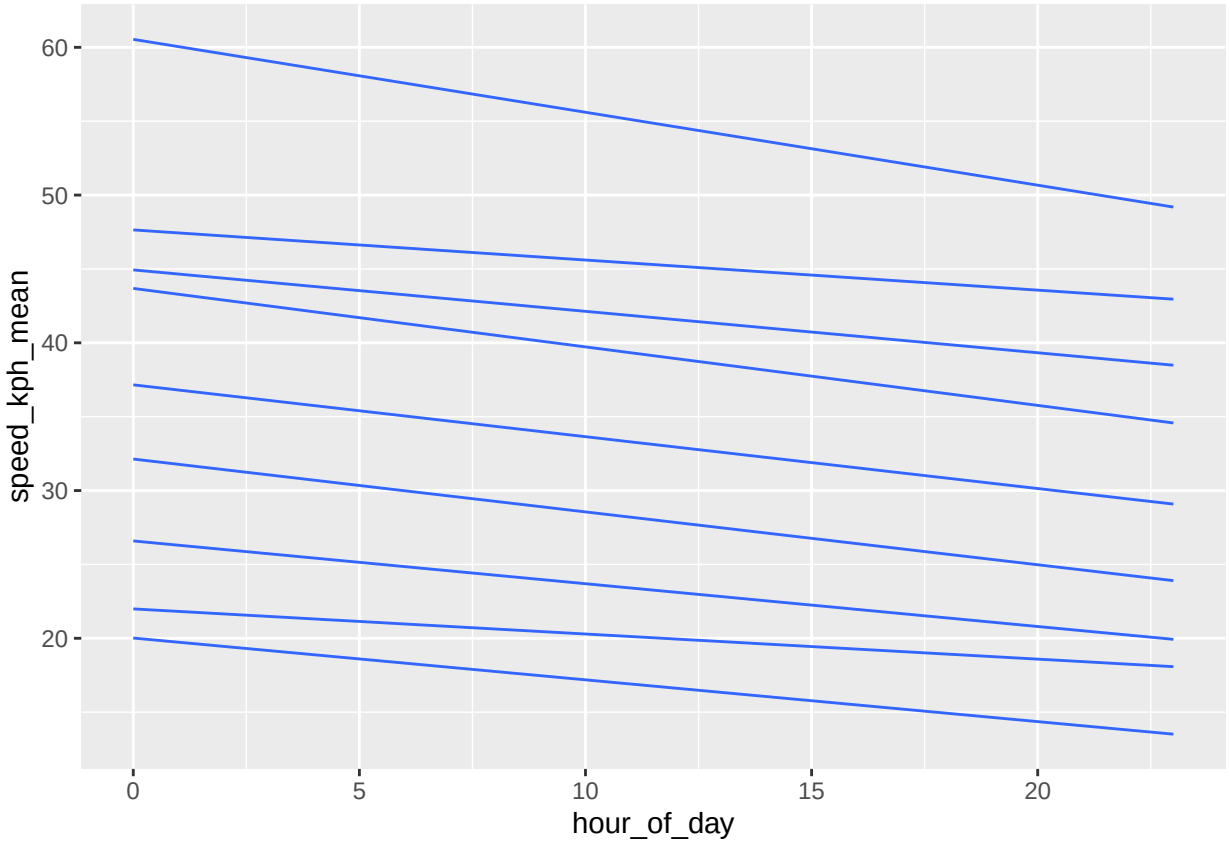

### Density2d + Rug Plot

```
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_density2d(color='red') +
  geom_rug()
```



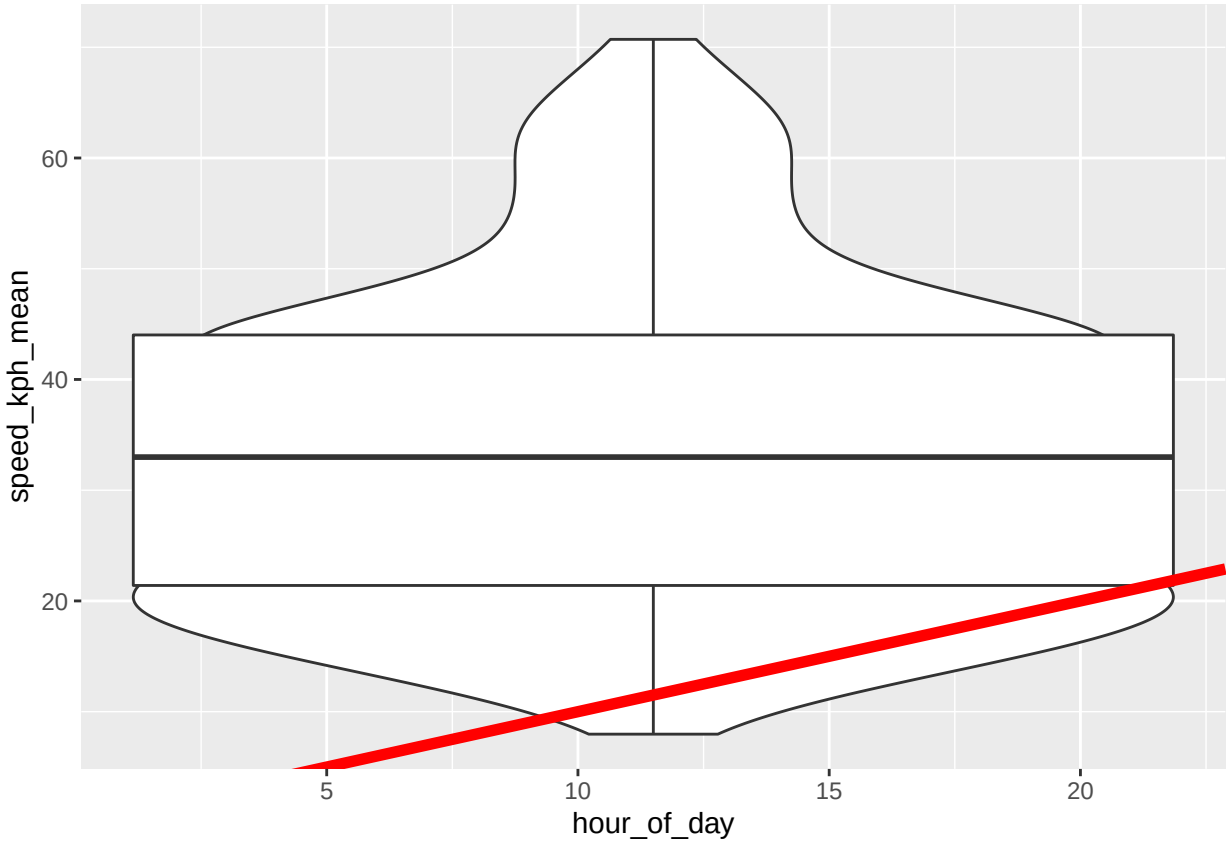

### 10th to 90th quantiles

```
# Install required package
# install.packages('quantreg',deps=T,respos='https://cran.r-project.org')
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_quantile(quantiles=seq(0.1,0.9,0.1))
```



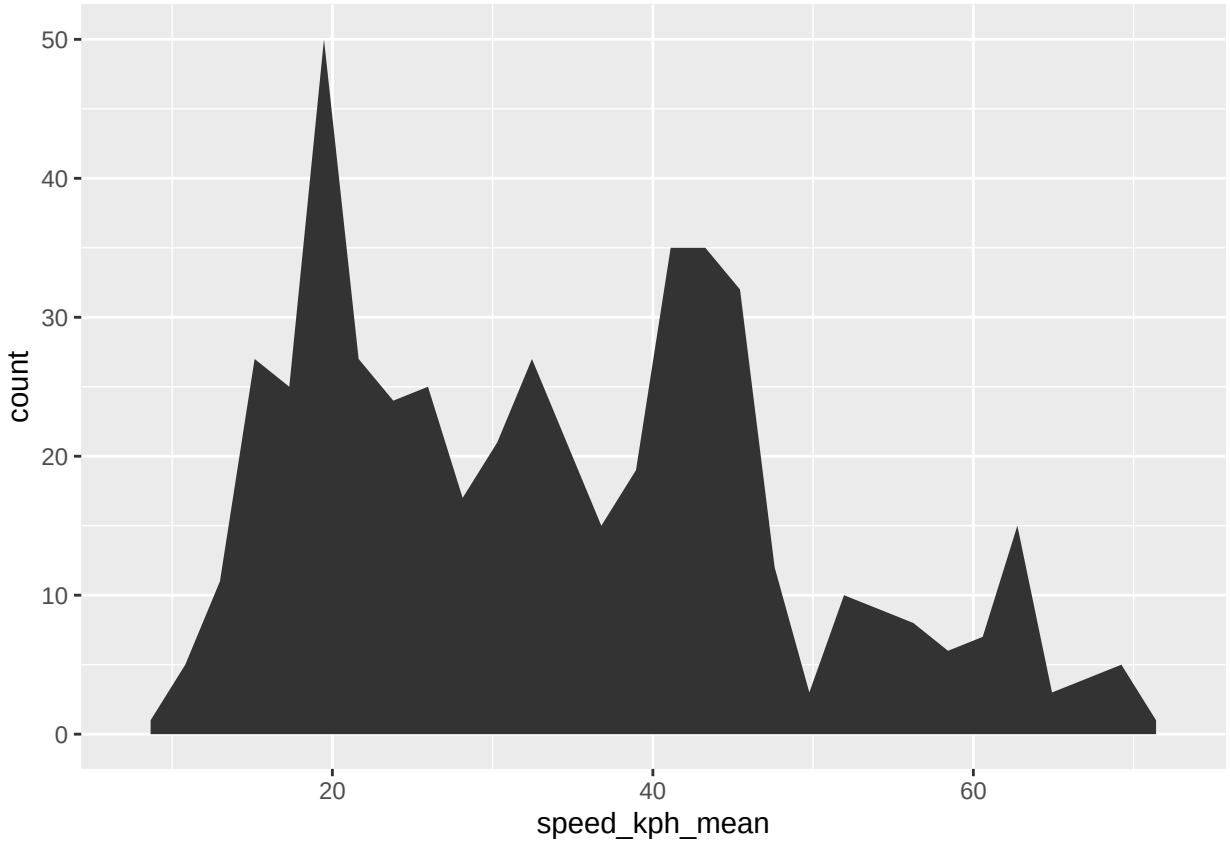

### Violin + Boxplot Plots Plot

```
ggplot(lab_data,aes(hour_of_day,speed_kph_mean)) +
  geom_violin() +
  geom_boxplot() +
  geom_abline(colour = "red", size = 2)
```



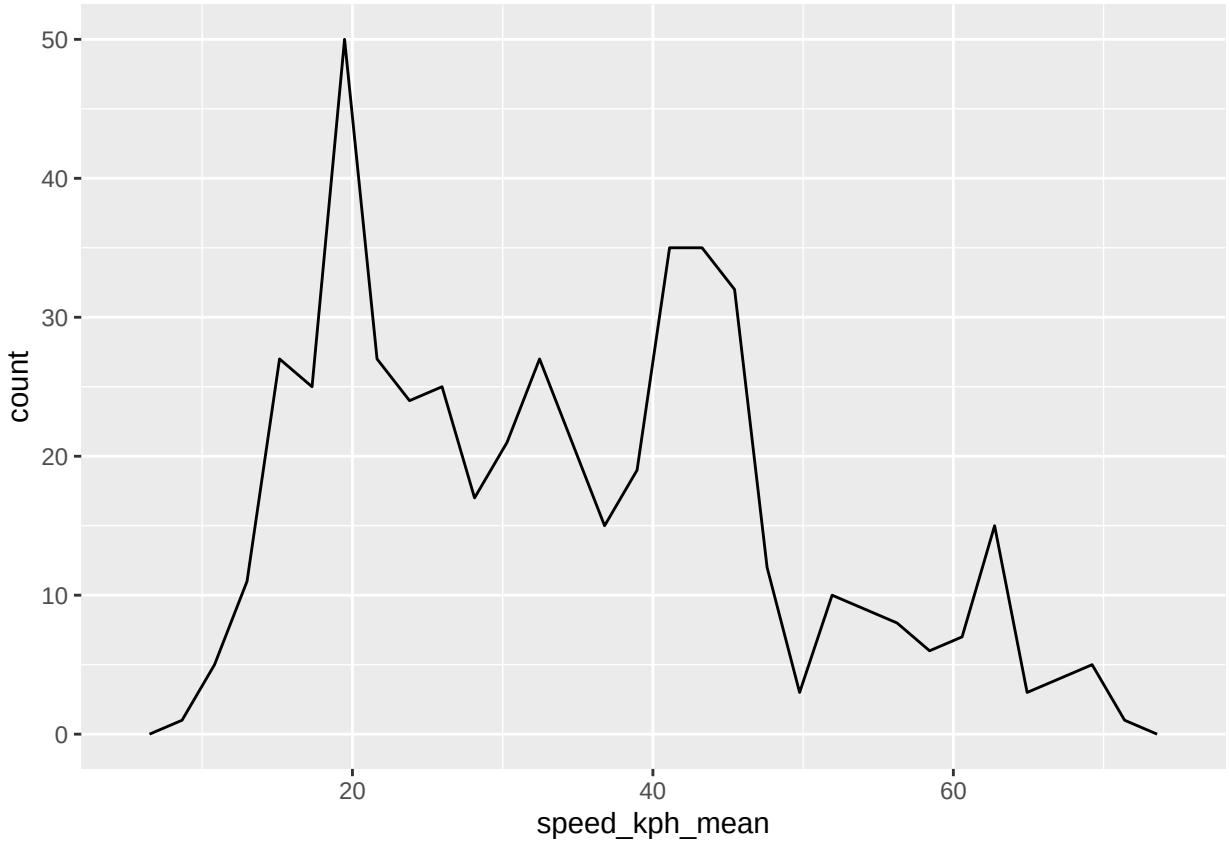

### Travel Speed Area Graph

```
speed <- ggplot(lab_data,aes(speed_kph_mean))
speed + geom_area(stat='bin')
```



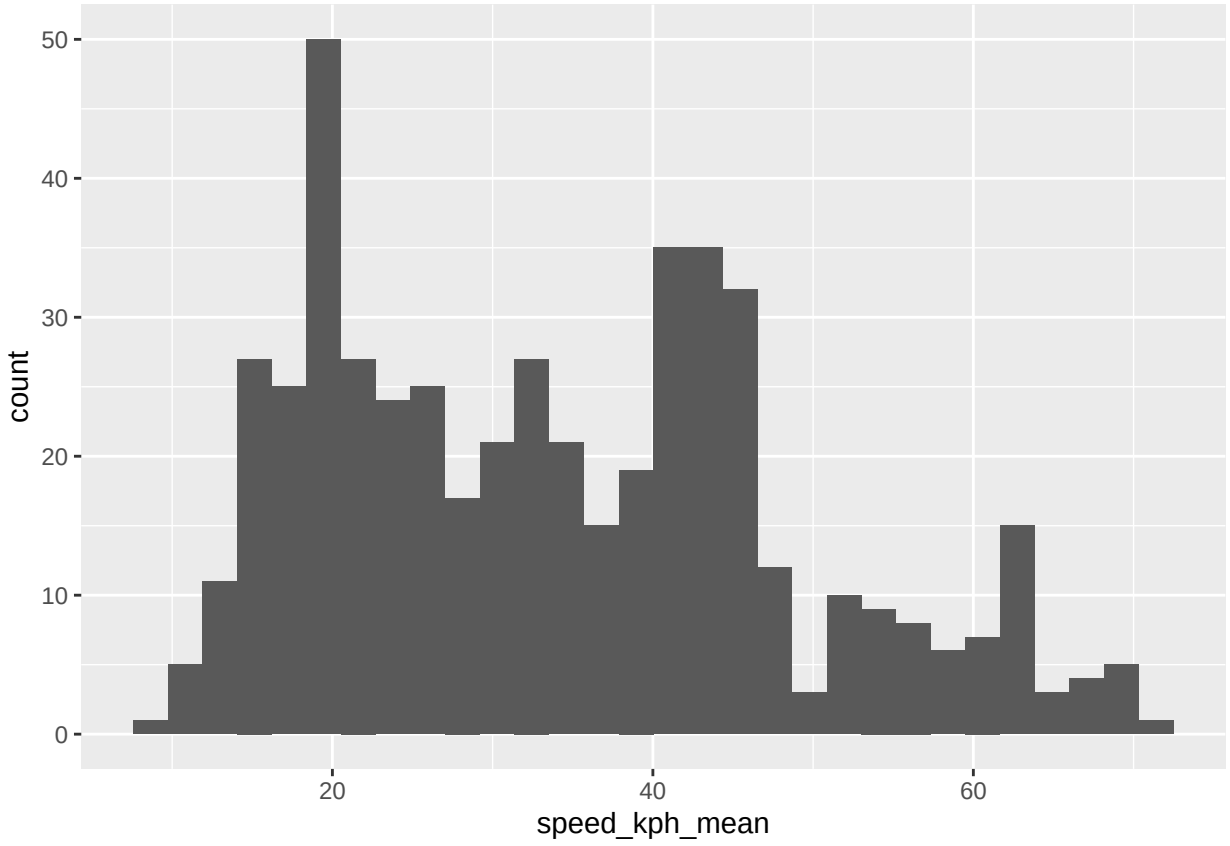

### Frequency Polygon

```
speed + geom_freqpoly()
```



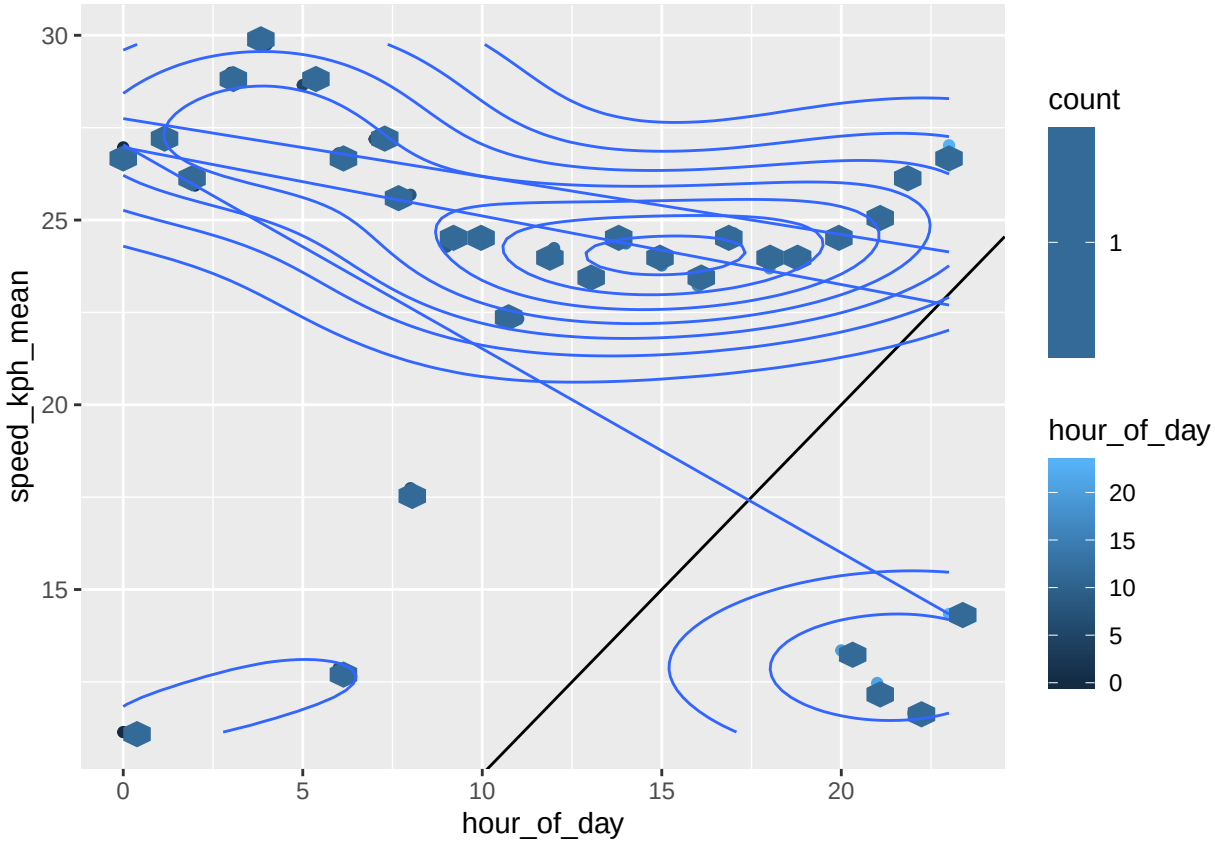

### Histogram

```
speed + geom_histogram()
```



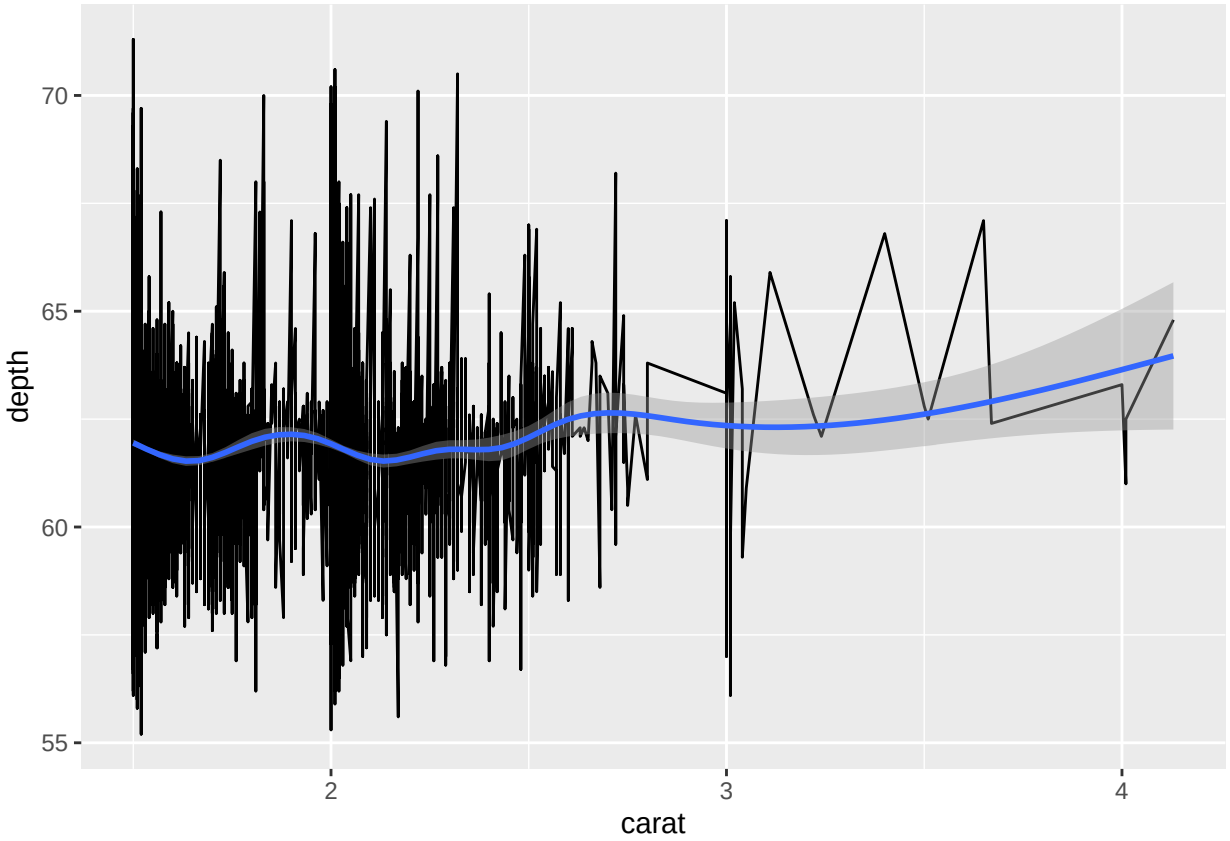

### lists + dplyr + ggplot2

```
geoms <- list(
  geom_point(),
  geom_abline(),
  geom_hex(),
  geom_quantile(),
  geom_density2d()
)
data %>% filter(osm_start_node_id==lab_data$osm_start_node_id[3]) %>%
  ggplot(aes(hour_of_day,speed_kph_mean,color=hour_of_day)) +
  geoms
```



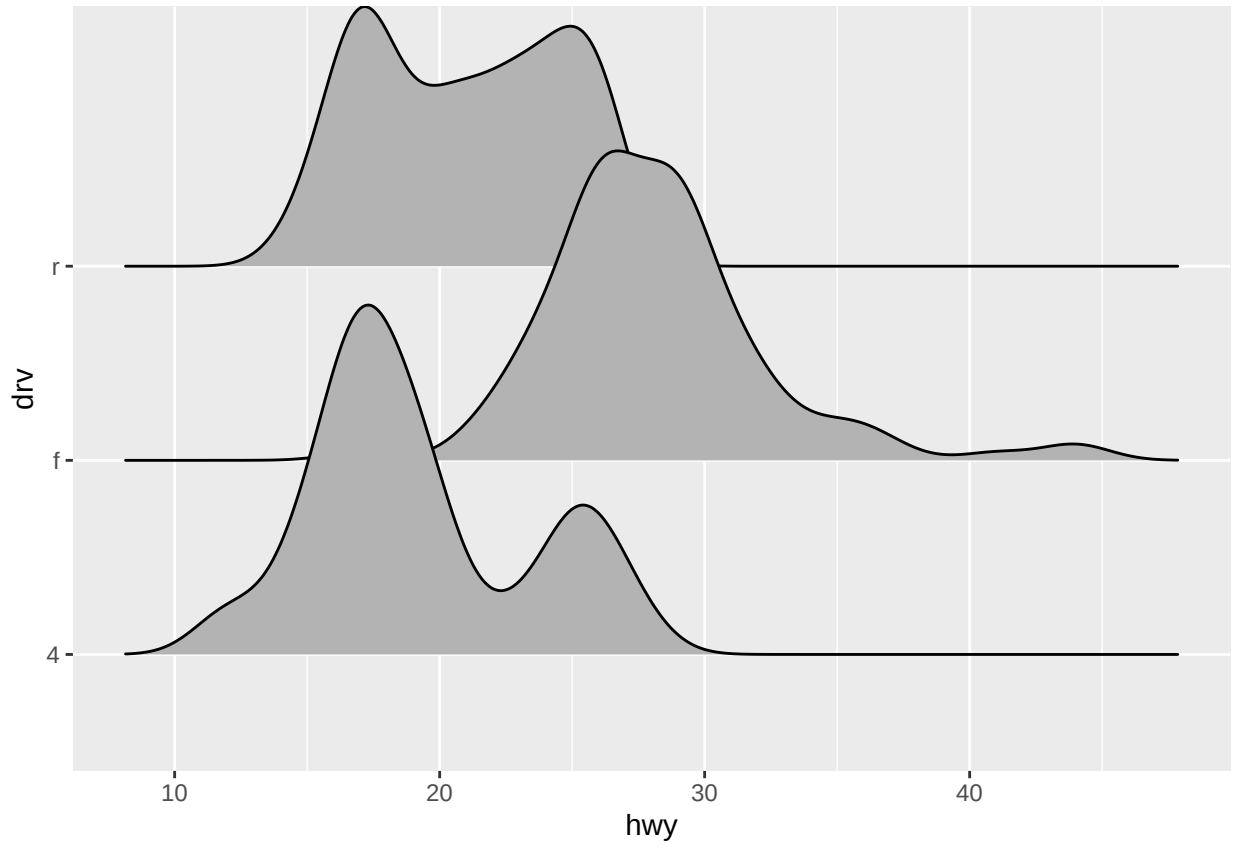

**Off Uber :)**

```
library(datasets)
diamonds %>%
  filter(depth>=45,depth<75)%>%
  filter (carat>=1.5,carat<4.5)%>%
  ggplot(aes(carat,depth)) +
  geom_line() +
  geom_smooth()
```



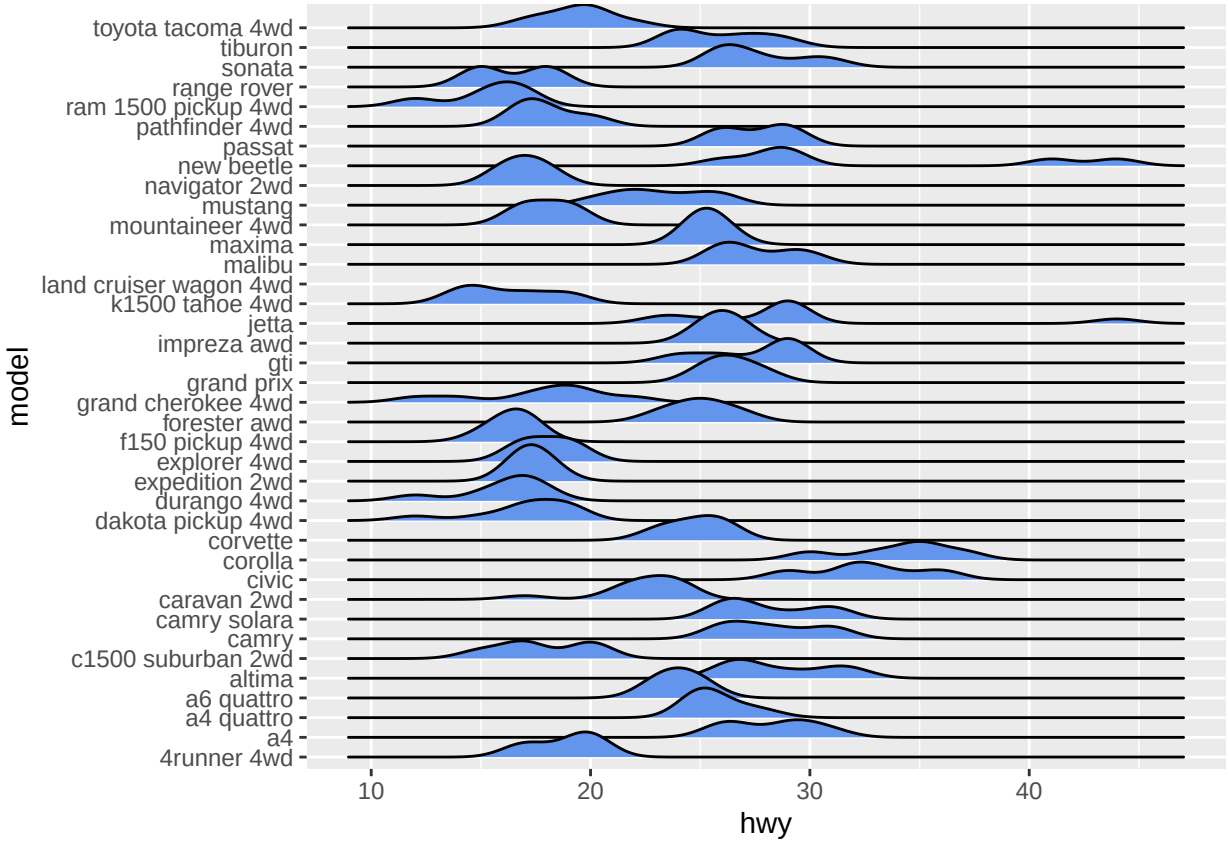

### ridgelines

```
# Install required package
#install.packages("ggridges",deps=T,repos="https://cran.r-project.org")
library(ggridges)
ggplot(mpg, aes(hwy, drv)) +
  geom_density_ridges()
```



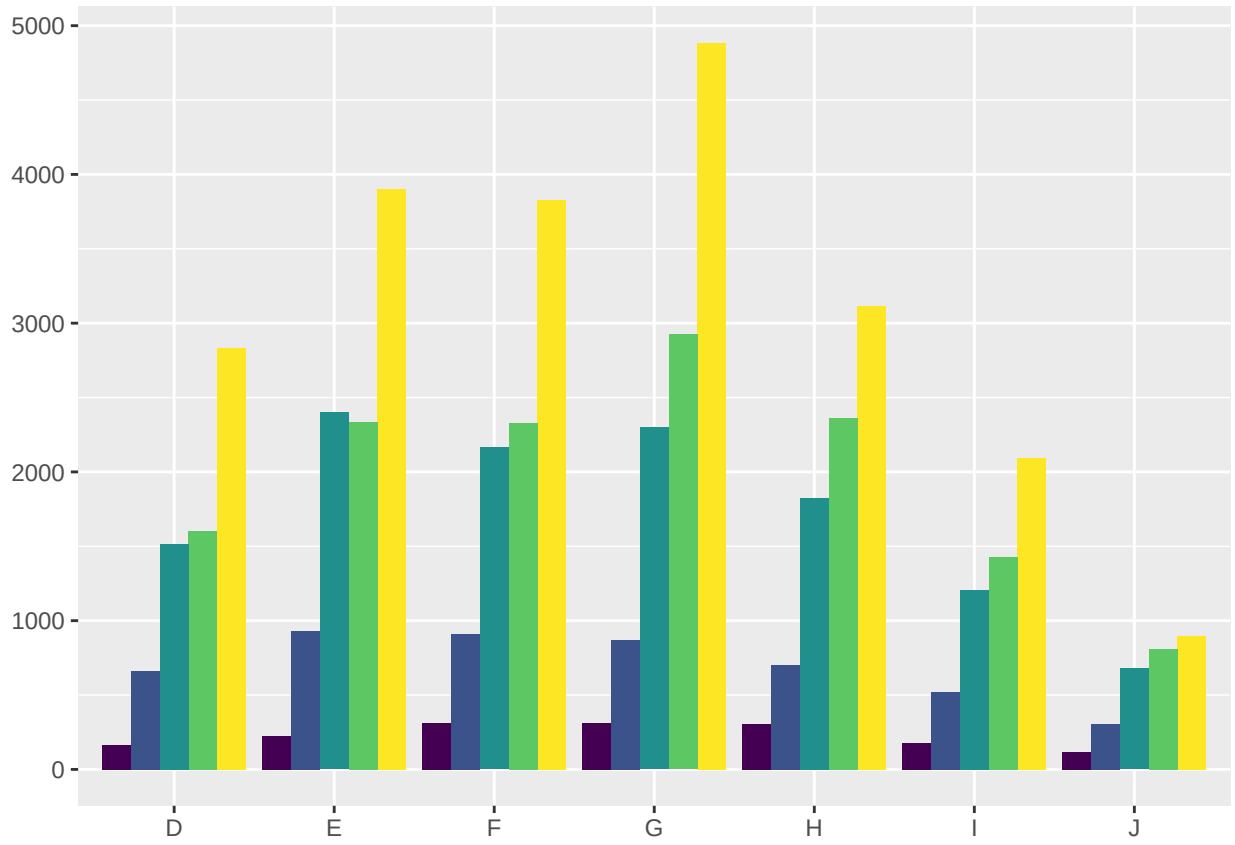

### More ridges

```
ggplot(mpg, aes(hwy, model)) +
  geom_density_ridges(fill = "cornflowerblue")
```




## Colored Bar Plot

```
ggplot(diamonds, aes(color, fill=cut))+
  xlab(NULL) + ylab(NULL) +
  theme(legend.position = "none")+
  geom_bar(position = "dodge")
```




Thank you!