

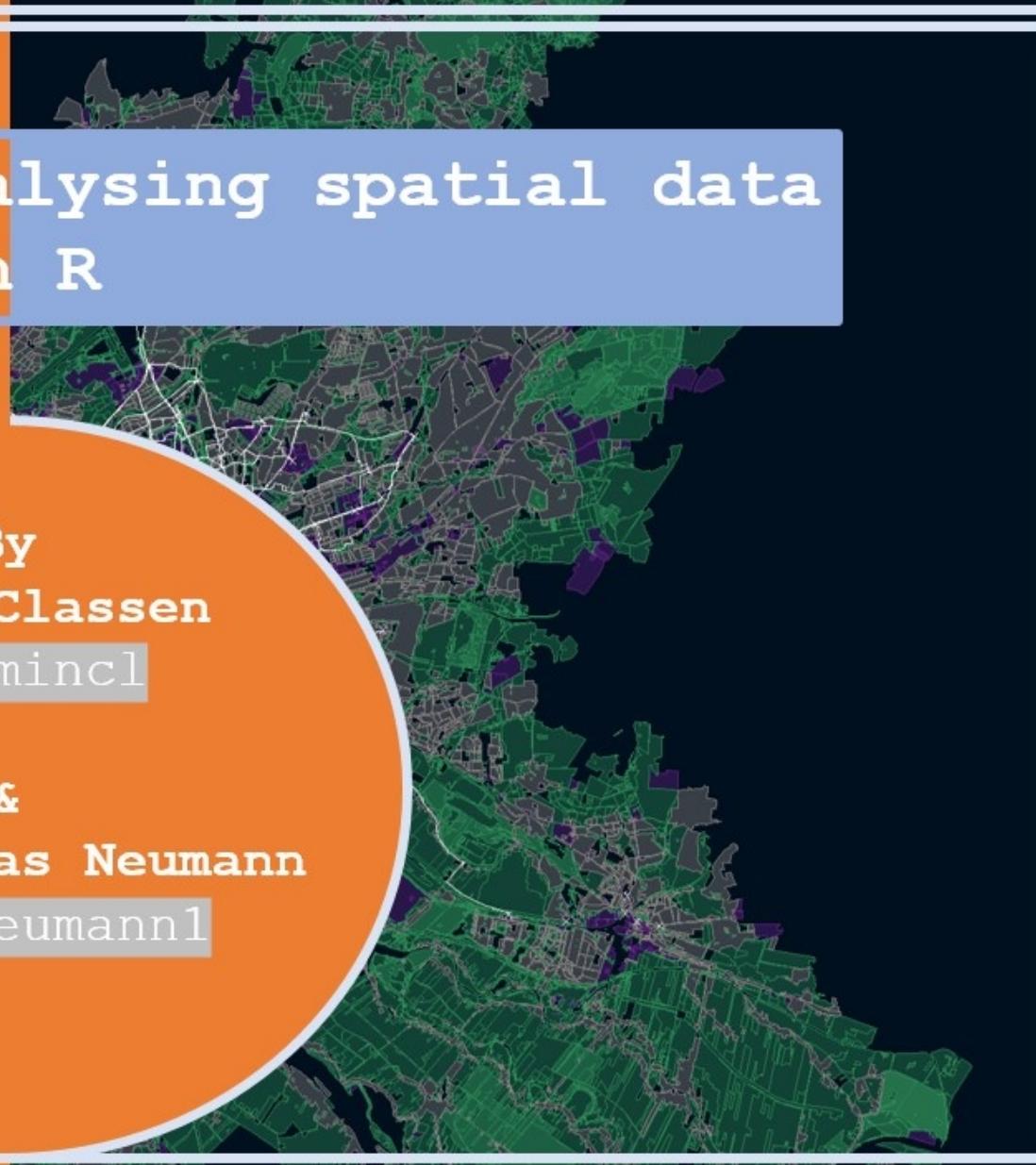
Introduction to analysing spatial data in R

By
Jasmin Classen

 [@jasmincl](https://github.com/jasmincl)

&
Andreas Neumann

 [@anneumann1](https://github.com/anneumann1)



HAMBURG's

most prominent routes for shared bike users



Find our Resources

Our complete talk is on Github:

You can either:

A: Clone the repo: <https://github.com/anneumann1/CorrelCon21>

B: Download the zip folder, unpack it and open Rproj.
File:

https://github.com/anneumann1/CorrelCon21/blob/main/spatial_analysis_R_intro.zip



id	name	lat	lon
131543	Landungsbrücke/Hafentor	53.5	9.97
131546	Fischmarkt/Breite Straße	53.5	9.95
131547	Paulinerplatz/Wohlwillstraße	53.6	9.97
131639	Burgstraße/Haner Landstraße	53.6	9.97
131640	Innocentiapark/Oberstraße	53.6	9.97

1. Load Data



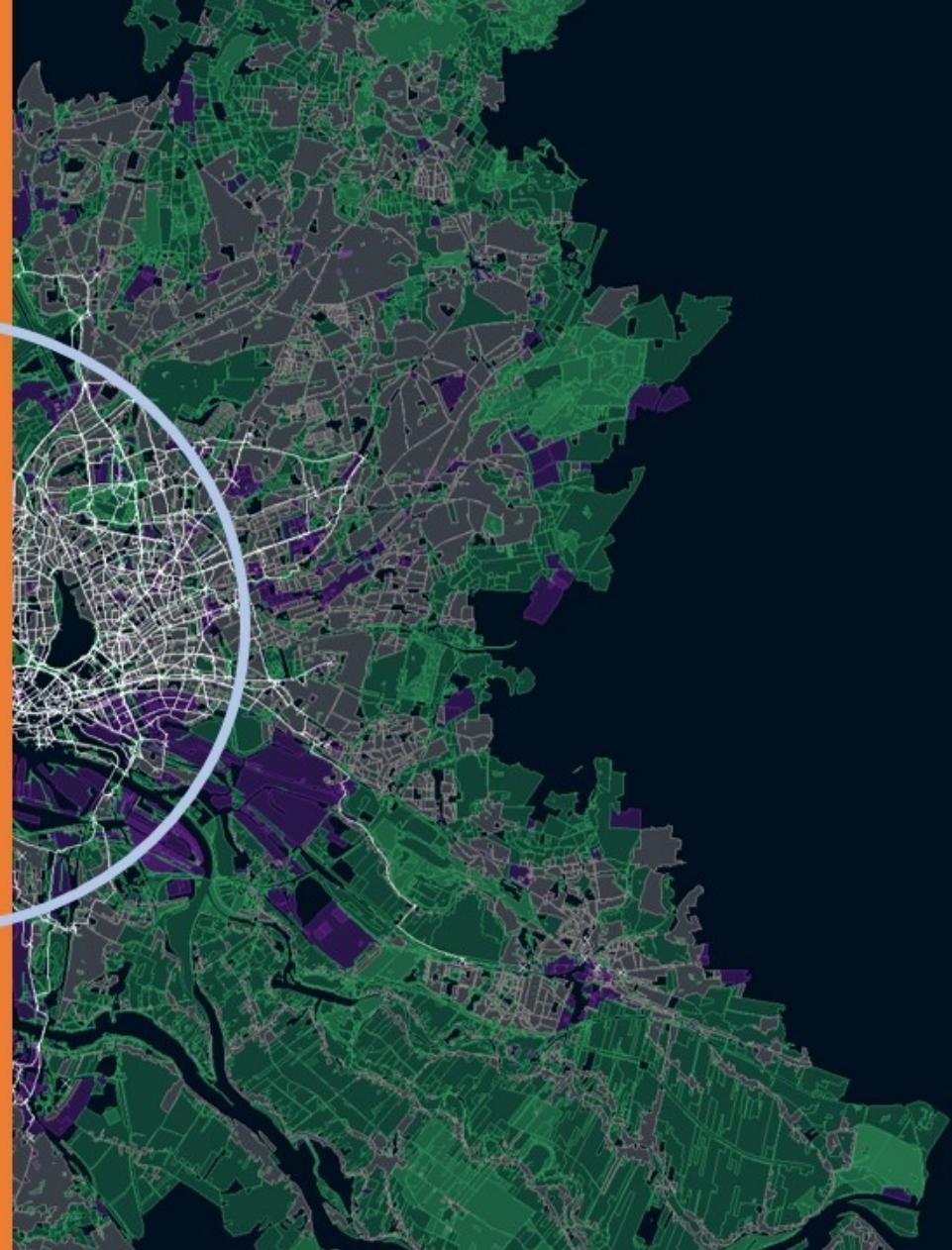
2. Prepare Data



3. Create Plot

Load Data

1



Input

1

Load packages

```
install.packages(  
  c("Rmisc", "sf", "sp", "ggplot2", "dplyr"))
```

```
library("ggplot2")  
library("readr")  
library("sf")  
library("Rmisc")  
library("dplyr")
```

Spatial Data Shapes

Points: Stations



"bike_station"

Lines: Bike routes

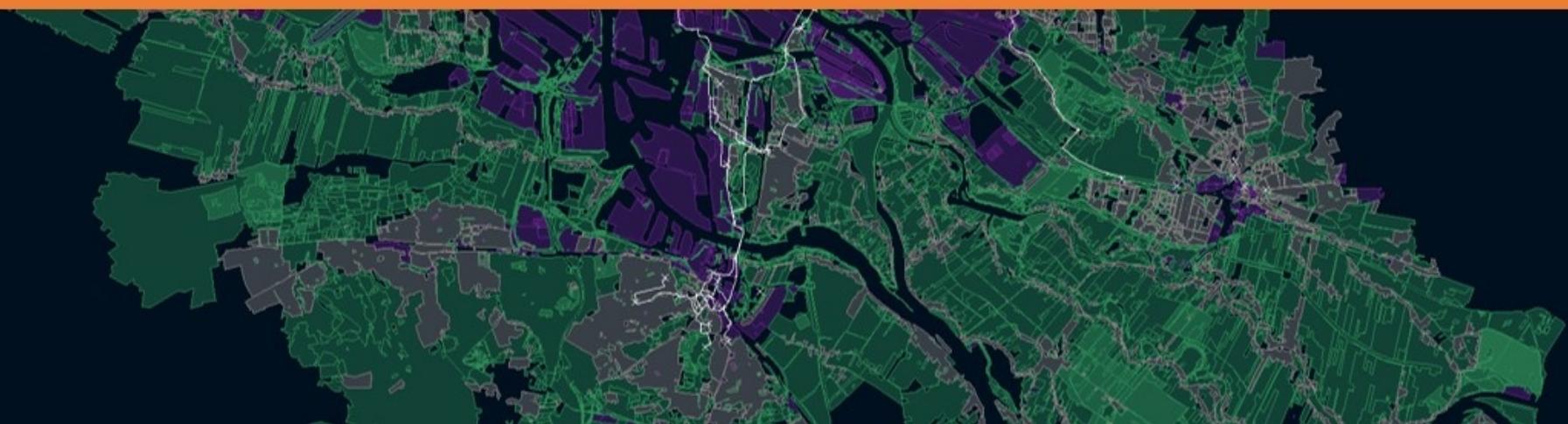


"shortest_cycle_paths"

Polygons: Landuse



"land_use"



Data Sources

- Bikesharing trips & stations: Open Data Portal from Deutsche Bahn



- Landuse: Openstreetmap data downloaded for Hamburg from Geofabrik Download Server
- Shortest paths between stations: Extracted from CycleStreetsAPI using Rpackage "cyclestreets"

Note: All data sources were already slightly prepared & cleaned and sampled for easier plotting, contact us for details.

Input

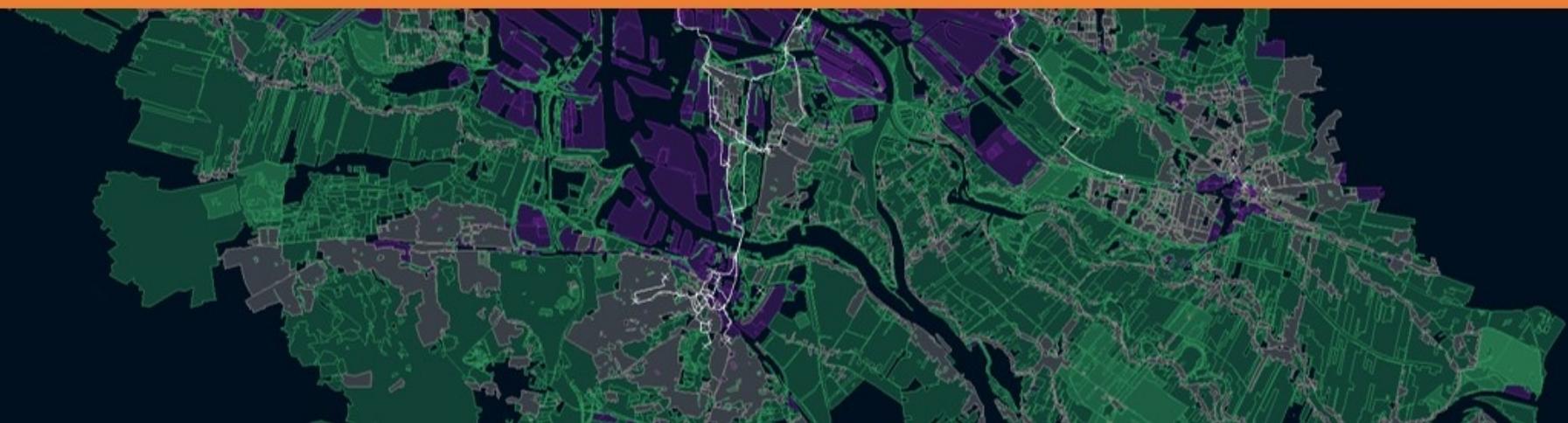


```
bikeshare_station <- read_csv("datasets/bikeshare_stations_hh.csv")  
land_use <- readRDS("datasets/land_use.Rds")  
shortest_cycle_paths <- readRDS("datasets/shortest_cycle_paths.rds")
```

We are using an R-Project so all file paths are universal.

Geometry Column

- list column
- collection of coordinates (points)
- lines & polygons & points



Input

3

```
head(bikeshare_station)
```

Output

Input

3

```
head(bikeshare_station)
```

Output

```
> head(bikeshare_station)
# A tibble: 6 × 4
  station_id name                lat    lon
      <dbl> <chr>              <dbl>  <dbl>
1     131543 Landungsbrücke/Hafentor 53.5   9.97
2     131546 Fischmarkt/Breite Straße 53.5   9.95
3     131547 Paulinenplatz/Wohlwillstraße 53.6   9.96
4     131639 Burgstraße/Hammer Landstraße 53.6  10.0 
5     131640 Innocentiapark/Oberstraße 53.6   9.98
6     131641 Goebenstraße/Eppendorfer Weg 53.6   9.96
```

Coordinates for stations are not yet converted to a spatial object.

Input

3

```
head(land_use)
```

Output

Input

3

```
head(land_use)
```

Output

```
> head(land_use)
Simple feature collection with 6 features and 5 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: 9.817368 ymin: 53.46784 xmax: 10.05681 ymax: 53.57648
Geodetic CRS:  WGS 84
# A tibble: 6 × 6
  osm_id   code fclass      name                           geometry cat
  <chr>   <int> <chr>      <chr>                         <MULTIPOLYGON [°]> <chr>
1 3220314  7209 commercial NA    (((9.899728 53.47843, 9.899961 53.47908, 9.9... busin...
2 4223465  7207 allotments NA    (((9.817368 53.47478, 9.817529 53.47697, 9.8... green...
3 4275521  7202 park        August-Lütgens-Park  (((9.946126 53.55659, 9.946467 53.55671, 9.9... green...
4 4277224  7212 retail      NA    (((10.02617 53.57126, 10.02691 53.57159, 10... busin...
5 4279326  7202 park        Friedrichsberger Park ((10.04953 53.57438, 10.04981 53.57451, 10... green...
6 4299049  7201 forest      NA    (((9.921254 53.47042, 9.923303 53.47051, 9.9... green...
```

Polygons are saved as a geometry in multipolygon format, a spatial data format.

Input

3

```
head(shortest_cycle_paths)
```

Output

Input

3

```
head(shortest_cycle_paths)
```

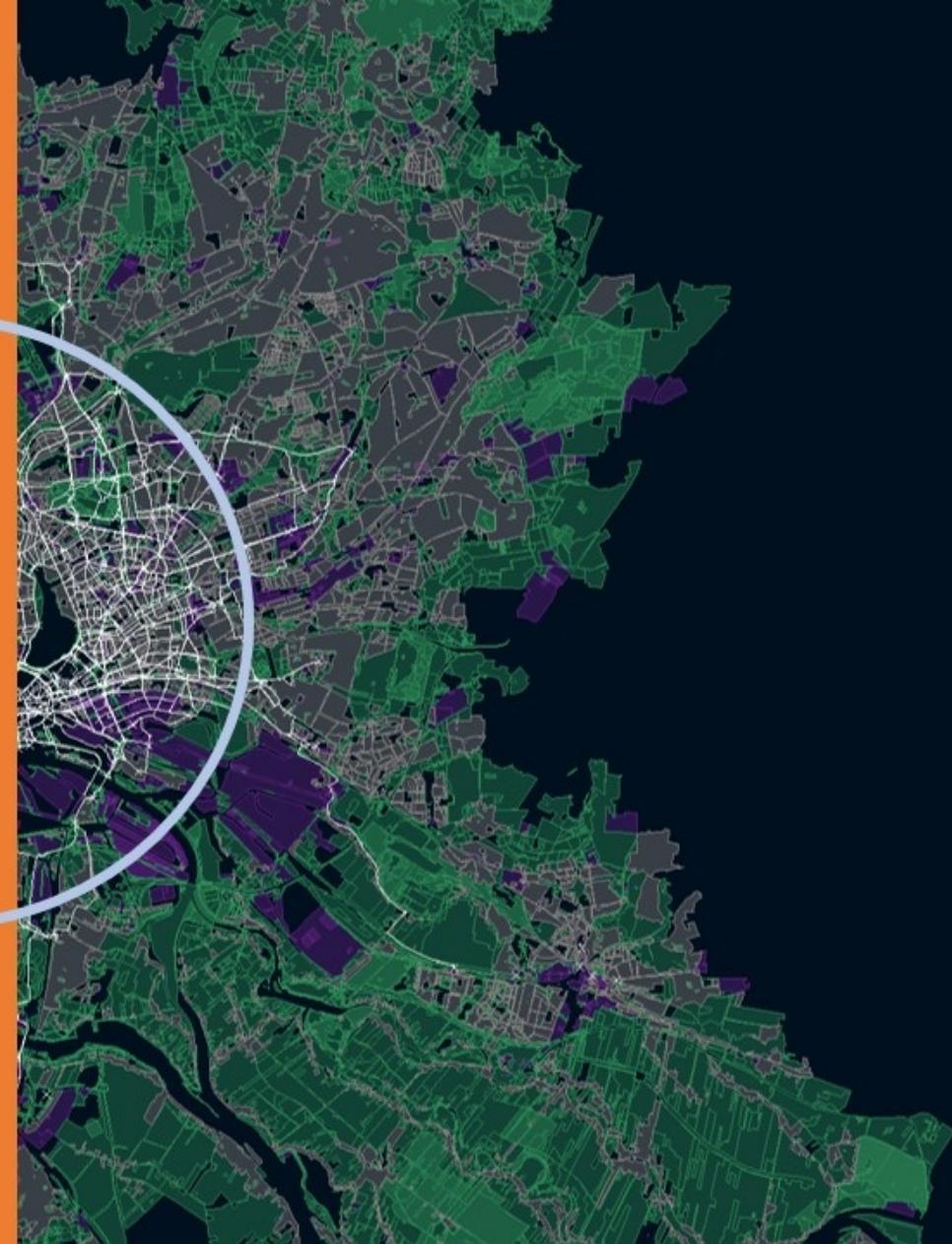
Output

```
> head(shortest_cycle_paths)
Simple feature collection with 6 features and 5 fields
Active geometry column: geometry
Geometry type: LINESTRING
Dimension: XY
Bounding box: xmin: 9.95088 ymin: 53.5457 xmax: 10.0413 ymax: 53.57758
Geodetic CRS: WGS 84
# A tibble: 6 × 7
# Groups:   start_rental_zone_hal_id [1]
  start_rental_zone... end_rental_zone_h... trip_count      geometry start_name   end_name
    <int>           <int>     <int>      <LINESTRING [°]> <chr>       <chr>
1     131543         131546      80 (9.9723 53.5457, 9.95088 ... Landungsbrü... Fischmarkt...
2     131543         131547      24 (9.9723 53.5457, 9.96246 ... Landungsbrü... Paulinenpl...
3     131543         131639      12 (9.9723 53.5457, 10.0413 ... Landungsbrü... Burgstraße...
4     131543         131640       3 (9.9723 53.5457, 9.98211 ... Landungsbrü... Innocentia...
5     131543         131641      17 (9.9723 53.5457, 9.963063... Landungsbrü... Goebenstra...
6     131543         131642      12 (9.9723 53.5457, 9.962395... Landungsbrü... Osterstraß...
# ... with 1 more variable: geom_bike <LINESTRING [°]>
```

A `dataframe` can have multiple geometries.

Prepare Data

2



Input

4

Prepare Bar Chart Data

```
bikeshare_station<-bikeshare_station %>%  
st_as_sf(coords = c("lon", "lat"))%>%  
st_set_crs(bikeshare_station,4326) %>%  
st_transform(4326)
```

```
buffers <- st_transform(land_use,  
CRS("+init=epsg:3068 +datum=WGS84  
+units=m"))
```

```
buffers <- st_buffer(buffers, 12)
```

```
buffers <- buffers %>% st_transform(4326)
```

We loaded a data.frame -> we need to convert our df into a sf-object

Object needs to be projected to a coordinate reference system

Input

4

Prepare Bar Chart Data

```
bikeshare_station<-bikeshare_station %>%
st_as_sf(coords = c("lon", "lat"))%>%
st_set_crs(bikeshare_station, 4326) %>%
st_transform(4326)

buffers <- st_transform(land_use,
CRS ("+init=epsg:3068 +datum=WGS84
+units=m"))

buffers <- st_buffer(buffers, 12)

buffers <- buffers %>% st_transform(4326)
```

Add 12 meter buffer around land use to make stations more likely to intersect (slightly hacky).

Input

4

Prepare Bar Chart Data

```
bikeshare_station<-bikeshare_station %>%
st_as_sf(coords = c("lon", "lat"))%>%
st_set_crs(bikeshare_station, 4326) %>%
st_transform(4326)

buffers <- st_transform(land_use,
CRS ("+init=epsg:3068 +datum=WGS84
+units=m"))

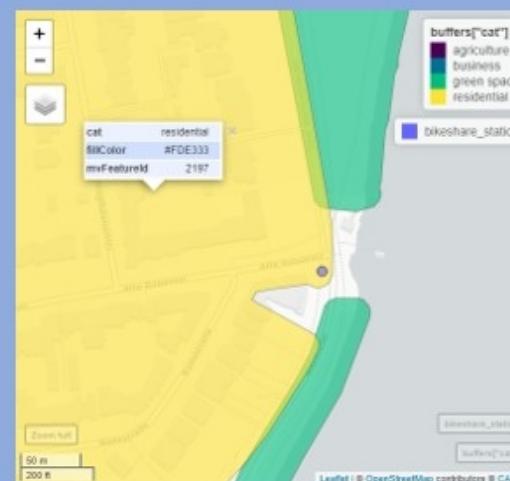
buffers <- st_buffer(buffers, 12)

buffers <- buffers %>% st_transform(4326)
```

Output



Before



After

Input

4

Prepare Bar Chart Data

```
bikeshare_station<-bikeshare_station%>%
st_as_sf(coords = c("lon", "lat"))%>%
st_set_crs(bikeshare_station, 4326)%>%
st_transform(4326)

buffers <- st_transform(land_use,
CRS("+init=epsg:3068 +datum=WGS84
+units=m"))

buffers <- st_buffer(buffers, 12)

buffers <- buffers %>% st_transform(4326)
```

Output

```
Simple feature collection with 6 features and 2 fields
Geometry type: POINT
Dimension:      XY
Bounding box:  xmin: 9.95088 ymin: 53.5457 xmax: 10.0413 ymax
Geodetic CRS:  WGS 84
# A tibble: 6 × 3
  station_id name                           geometry
     <dbl> <chr>                         <POINT [°]>
1    131543 Landungsbrücke/Hafentor      (9.9723 53.5457)
2    131546 Fischmarkt/Breite Straße     (9.95088 53.5462)
3    131547 Paulinenplatz/Wohlwillstraße (9.96246 53.5542)
4    131639 Burgstraße/Hammer Landstraße (10.0413 53.55569)
5    131640 Innocentiapark/Oberstraße    (9.98211 53.57725)
6    131641 Goebenstraße/Eppendorfer Weg (9.963063 53.57758)
```

Stations data also has a
geometry column now.

Input

4

Prepare Bar Chart Data

Find out which bike-station is located in which landuse polygon

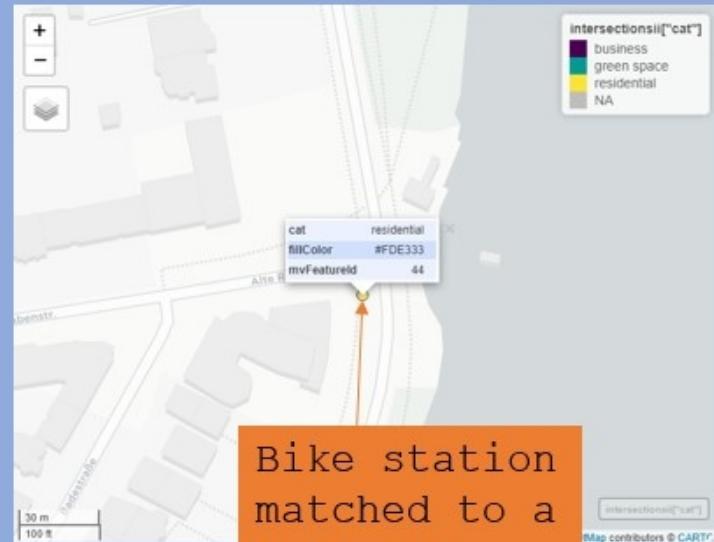
```
intersections <-  
st_join(bikeshare_station,buffers)
```

Calculate frequency of stations per land use type. Add "unclassified" for stations without landuse.

```
station_landuse_count <- intersections%>%  
group_by(cat)%>%  
dplyr::summarise(n = n())%>%  
mutate(freq = n / sum(n))
```

```
station_landuse_count$cat[is.na(station_l  
anduse_count$cat)] <- "unclassified"
```

Output



Input

4

Prepare Bar Chart Data

Find out which bike-station is located in which landuse polygon

```
intersections <-  
st_join(bikeshare_station,buffers)
```

Calculate frequency of stations per land use type. Add "unclassified" for stations without landuse.

```
station_landuse_count <- intersections %>%  
group_by(cat) %>%  
dplyr::summarise(n = n()) %>%  
mutate(freq = n / sum(n))
```

```
station_landuse_count$cat[is.na(station_l  
anduse_count$cat)] <- "unclassified"
```

Output

```
> intersections
```

Simple feature collection with 249 features and 5 fields

Geometry type: POINT

Dimension: XY

Bounding box: xmin: 9.862013 ymin: 53.44762 xmax: 10.22 ymax: 53.65429

Geodetic CRS: WGS 84

A tibble: 249 × 6

	station_id	geometry	osm_id	code	fclass	cat
1	131543	<POINT [°]>	9.9723 53.5457	NA	NA NA	NA
2	131546	(9.95088 53.5462)	137392019	Z203	residential	residential
3	131547	(9.96246 53.5542)	NA	NA	NA	NA
4	131639	(10.0413 53.55569)	NA	NA	NA	NA
5	131640	(9.98211 53.57725)	20362343	Z202	park	green space
6	131640	(9.98211 53.57725)	37399701	Z217	scrub	green space
7	131640	(9.98211 53.57725)	136965863	Z203	residential	residential
8	131641	(9.963063 53.57758)	54584830	Z203	residential	residential
9	131641	(9.963063 53.57758)	54684012	Z203	residential	residential
10	131642	(9.962395 53.57318)	678240859	Z203	residential	residential
						# ... with 239 more rows

```
> station_landuse_count
```

Simple feature collection with 4 features and 3 fields

Geometry type: MULTIPOLY

Dimension: XY

Bounding box: xmin: 9.862013 ymin: 53.44762 xmax: 10.22 ymax: 53.65429

Geodetic CRS: WGS 84

A tibble: 4 × 4

	cat	n	geometry	freq
1	* <chr>	<int>	<MULTIPOLY [°]>	<dbl>
2	business	34	((9.878527 53.57454), (9.930344 53.55...	0.135
3	green space	84	((9.878527 53.57454), (9.86216 53.558...	0.333
4	residential	64	((9.925653 53.55082), (9.927945 53.55...	0.254
5	NA	70	((9.862423 53.57163), (9.862013 53.54...	0.278

Input

4

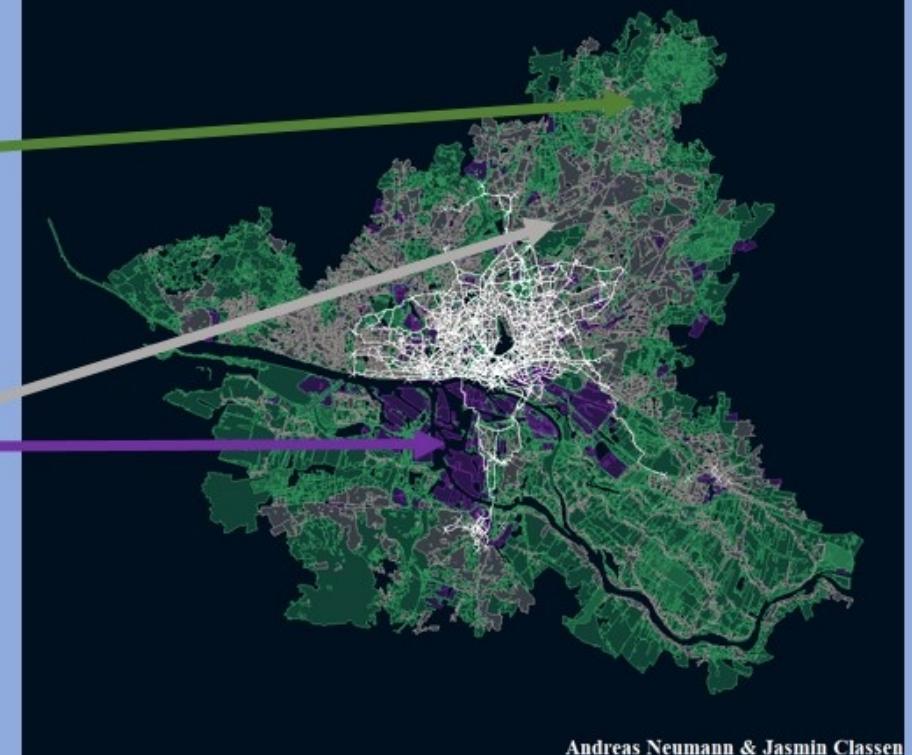
Prepare Landuse Data

```
land_use <- st_set_crs(land_use, 4326)

land_use$cat<-
  case_when(
    str_detect(land_use$fclass, "allotments|fore
    st|grass|heath|meadow|nature_reserve|park|re
    creation_ground|scrub|park|orchard|cemetary|
    farmland|farmyard|vineyard") ~ "green
    space",
    str_detect(land_use$fclass,
    "retail|commercial|industrial|quarry") ~
    "business",
    str_detect(land_use$fclass,
    "residential|military") ~ "residential")
```

Output

*HAMBURG's
most prominent routes for shared bike users*



Andreas Neumann & Jasmin Classen

Input

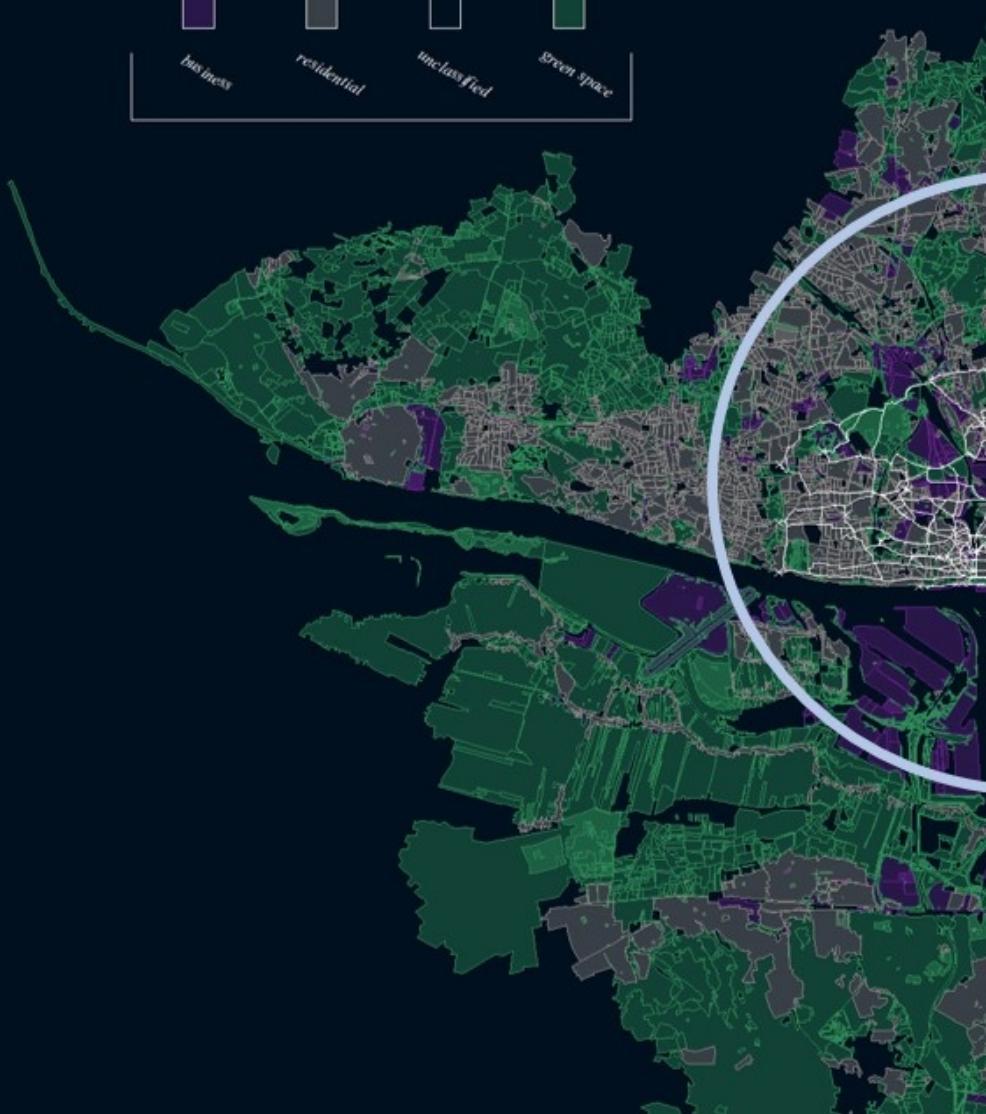
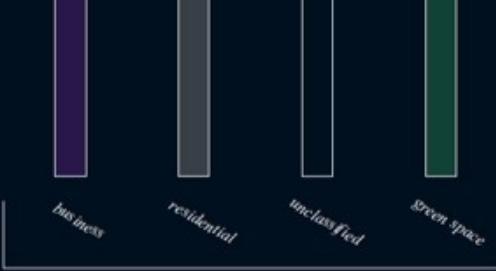
4

Prepare Map Data

Sample route data for better visibility
of single routes.

```
set.seed(123)
shortest_cycle_paths_sampled <-
shortest_cycle_paths %>%
  slice_sample(prop=0.08)
```

Output



3

Create Plot

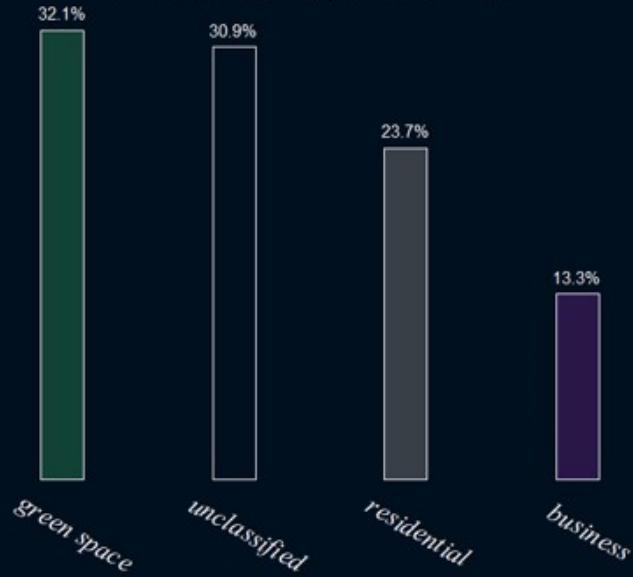
HAMBURG's

most prominent routes for shared bike users

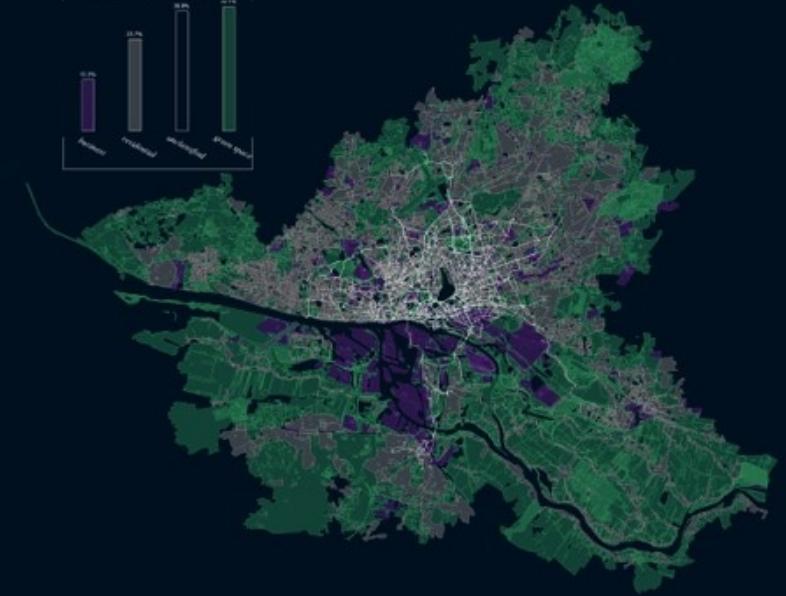
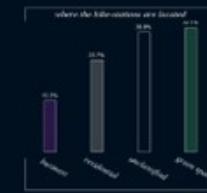


Andreas Neumann & Jasmin Classen

where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

1

2

3

HAMBURG's

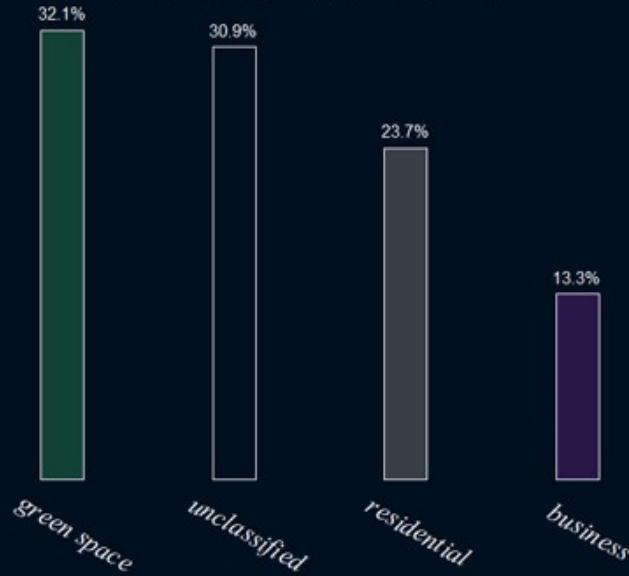
most prominent routes for shared bike users



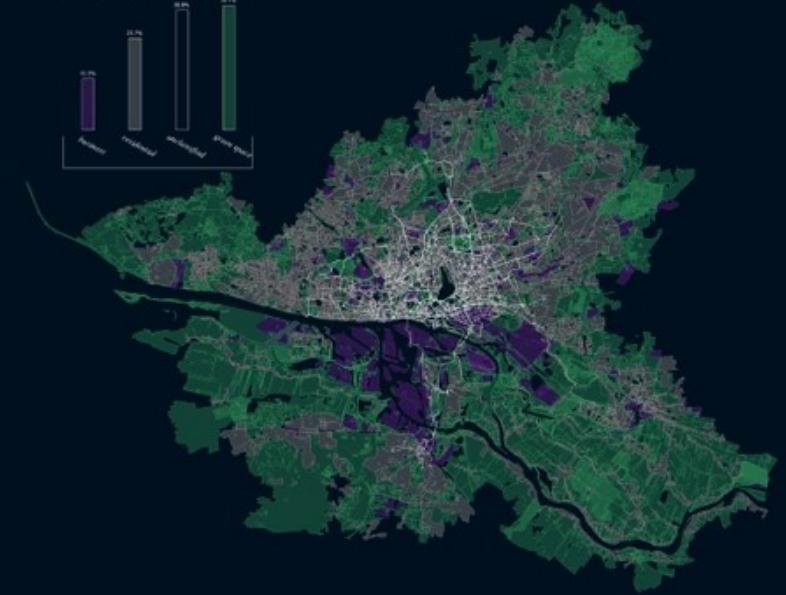
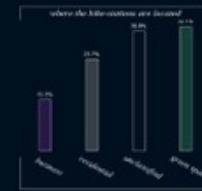
Andreas Neumann & Jasmin Classen

1

where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

2

3

Hamburg_map<-

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat), col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle = "most prominent routes  
for shared bike users", caption=glue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential"  
  ="seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "residential" =  
  "seashell4", "business"="darkorchid4"), guide = "none") +
```

A

```
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, face="italic"))
```

B

Code block 1

Code block 1

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat), col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle = "most prominent routes for shared bike users", caption=glue("Andreas  
Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential"  
  ="seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "residential"  
  ="seashell4", "business"="darkorchid4"), guide = "none") +
```

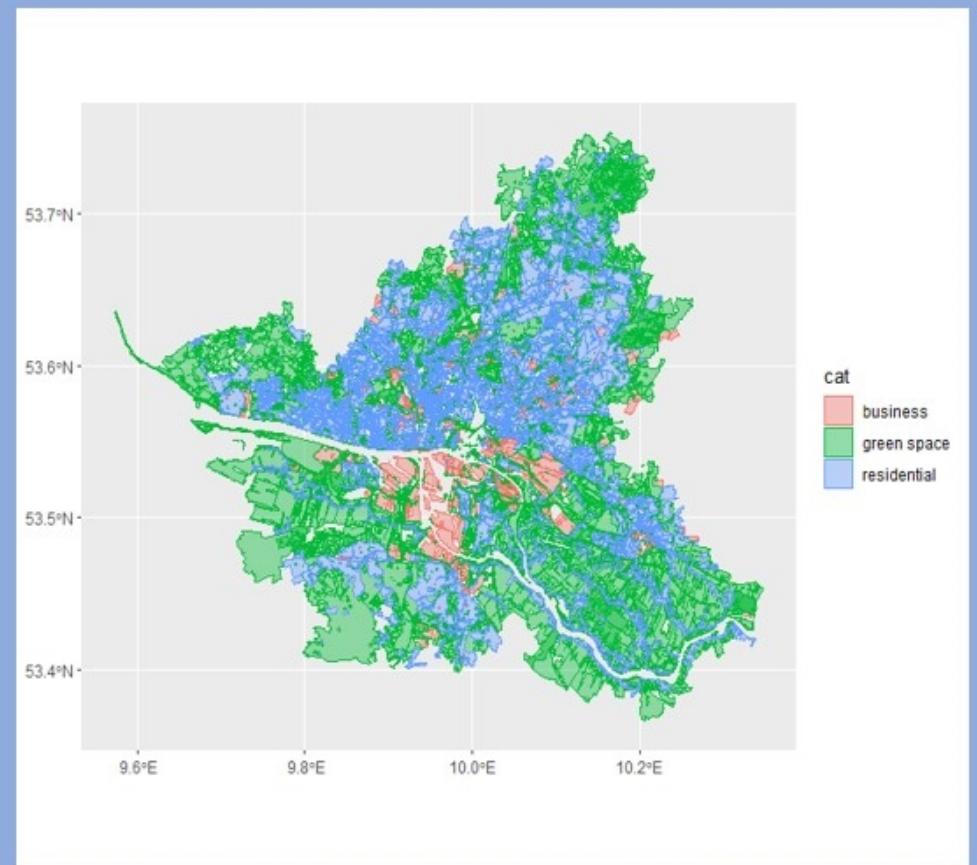
A



Input

```
ggplot()+
  geom_sf(land_use,mapping=aes(fill=cat,col=cat),alpha =
  .4)+  
  geom_point(bikeshare_station,mapping=aes(x=lon,y=lat),c  
ol="white",shape = 4,size=2,alpha=5)+  
  geom_sf(paths_street_direct_sampled,mapping=aes(geometr  
y=geom_bike),col=alpha("white",5))+  
  labs(title = "\nHAMBURG's",subtitle =  
"most prominent routes for shared bike users",caption=g  
lue("Andreas Neumann & Jasmin Classen"))+  
  scale_fill_manual(name="Category:",values=c("green spac  
e"="seagreen4","residential"  
="seashell4","business"="darkorchid4"),guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "  
residential"  
="seashell4","business"="darkorchid4"), guide =  
"none") +...
```

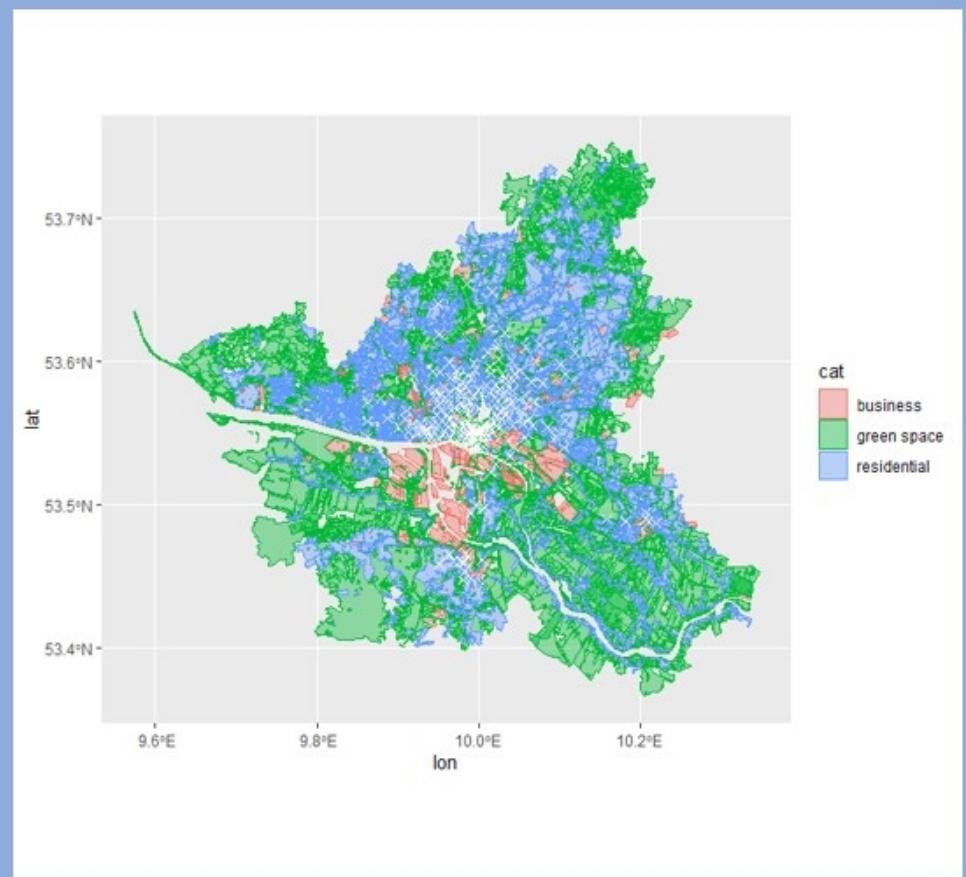
Output



Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha =  
    .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat), c  
    ol="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometr  
    y=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle =  
    "most prominent routes for shared bike users", caption=g  
    lue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green spac  
    e"="seagreen4", "residential"  
    ="seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "  
    residential"  
    ="seashell4", "business"="darkorchid4"), guide =  
    "none") +...
```

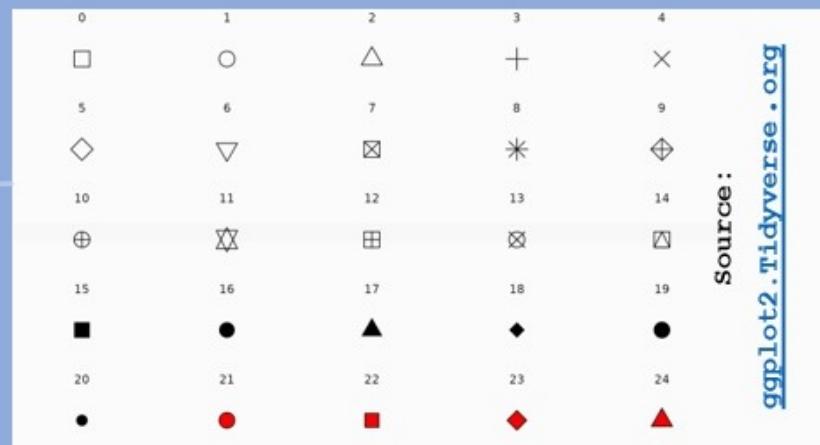
Output



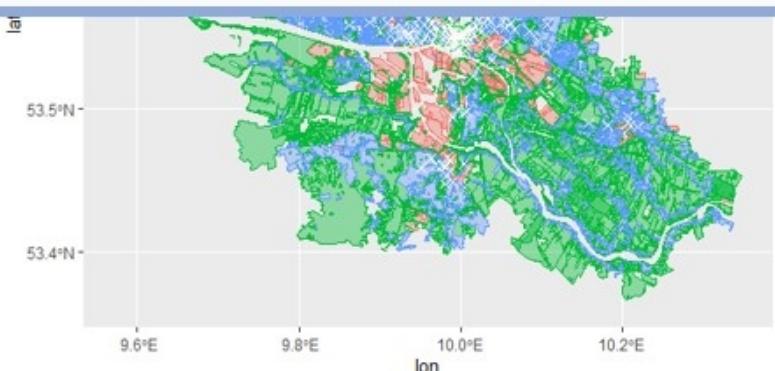
Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha =  
  .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon,y=lat), c  
  ol="white", shape = 4, size=z, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometr  
  y=geom_bike), col=alpha("white",5)) +  
  labs(title = "\nHAMBURG's", subtitle =  
  "most prominent routes for shared bike users", caption=g  
  lue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green spac  
  e"="seagreen4", "residential"  
  ="seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "  
  residential"  
  ="seashell4", "business"="darkorchid4"), guide =  
  "none") +...
```

Output



Source:
ggplot2.Tidyverse.org



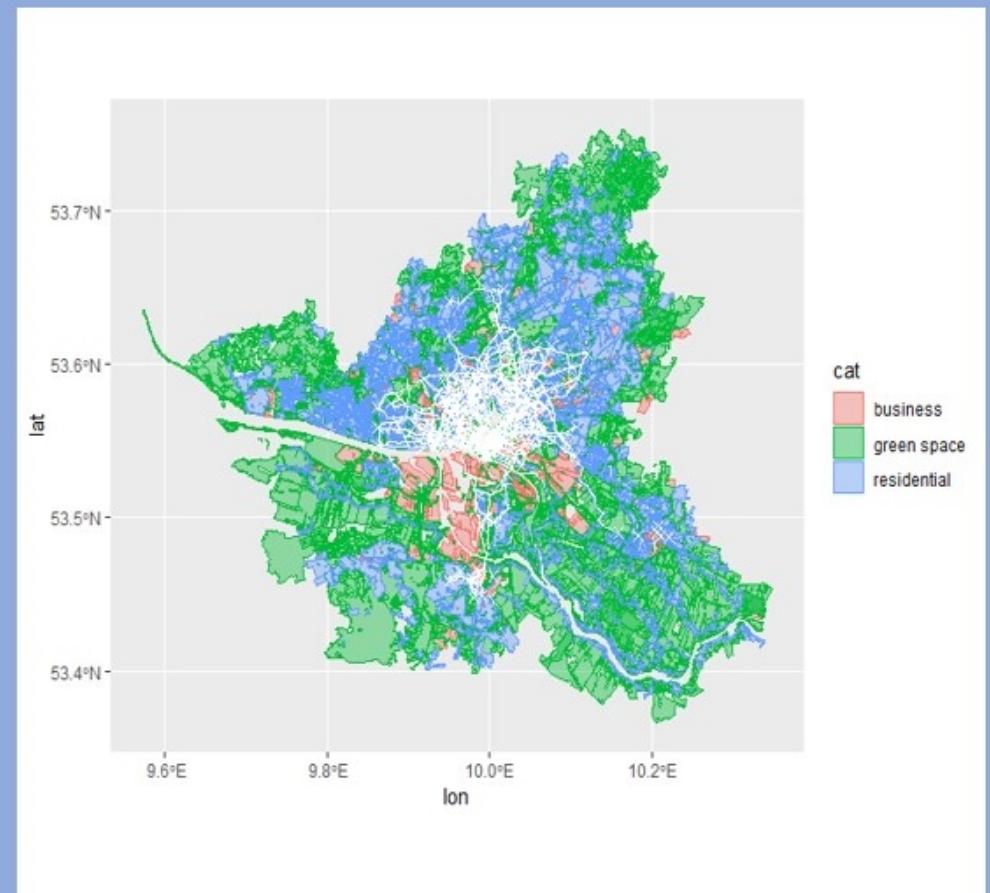
cat

- business
- green space
- residential

Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat),  
             col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white",5)) +  
  labs(title =  
    "\nHAMBURG's", subtitle = "most prominent routes for sh  
    ared bike users", caption=glue("Andreas Neumann &  
    Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green spa  
    ce"="seagreen4", "residential"  
    ="seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4",  
    "residential"  
    ="seashell4", "business"="darkorchid4"), guide =  
    "none") +...
```

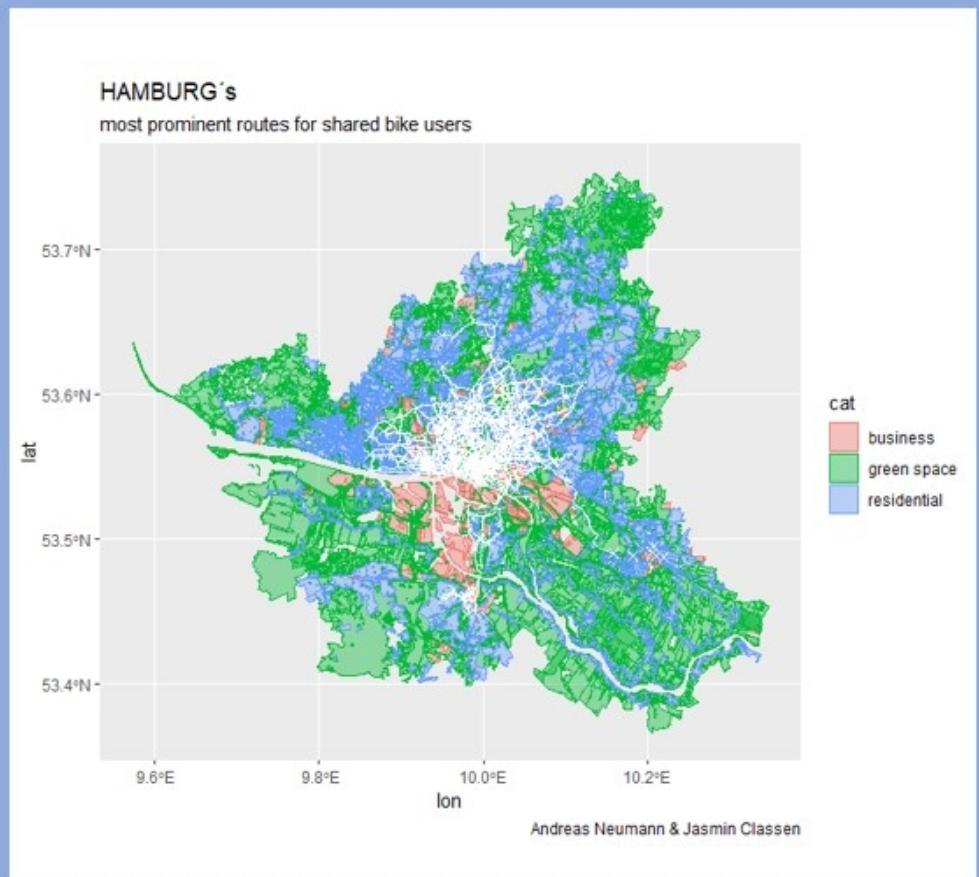
Output



Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat),  
             col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle = "most prominent routes for shared bike users", caption=glue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential" = "seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "residential" = "seashell4", "business"="darkorchid4"),  
                     guide = "none") + ...
```

Output

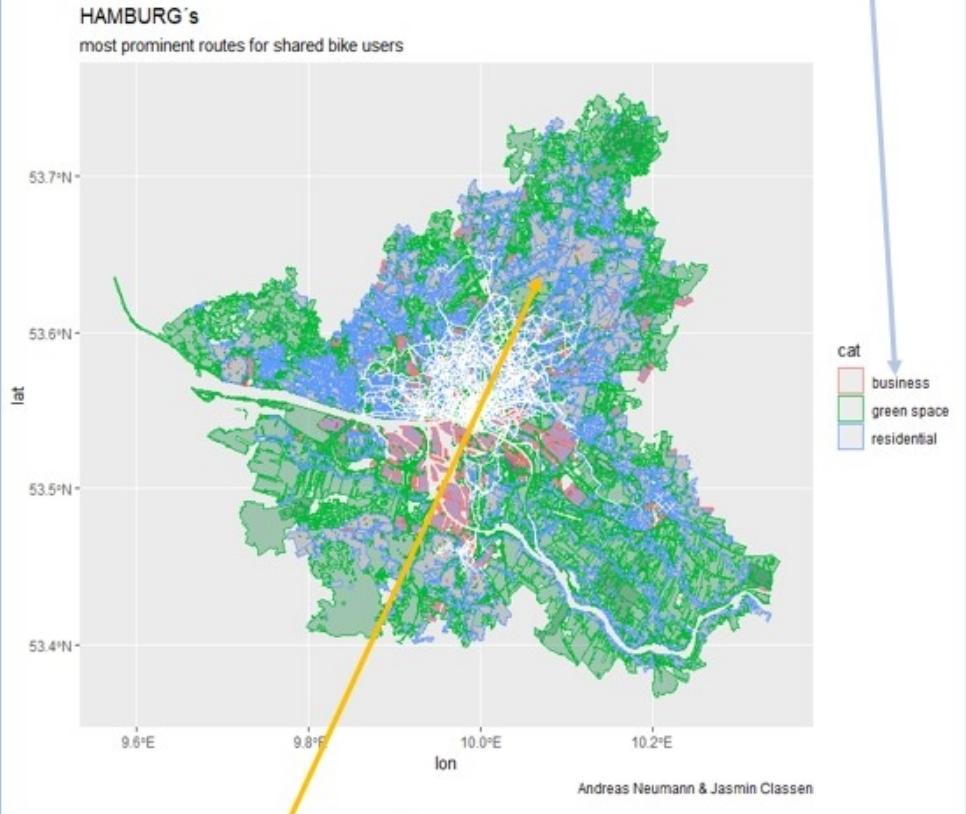


Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat),  
             col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle = "most prominent routes for shared bike users", caption=glue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential" = "seashell4", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "residential" = "seashell4", "business"="darkorchid4"),  
                     guide = "none") + ...
```

Output

Empty filling because the guide has been removed



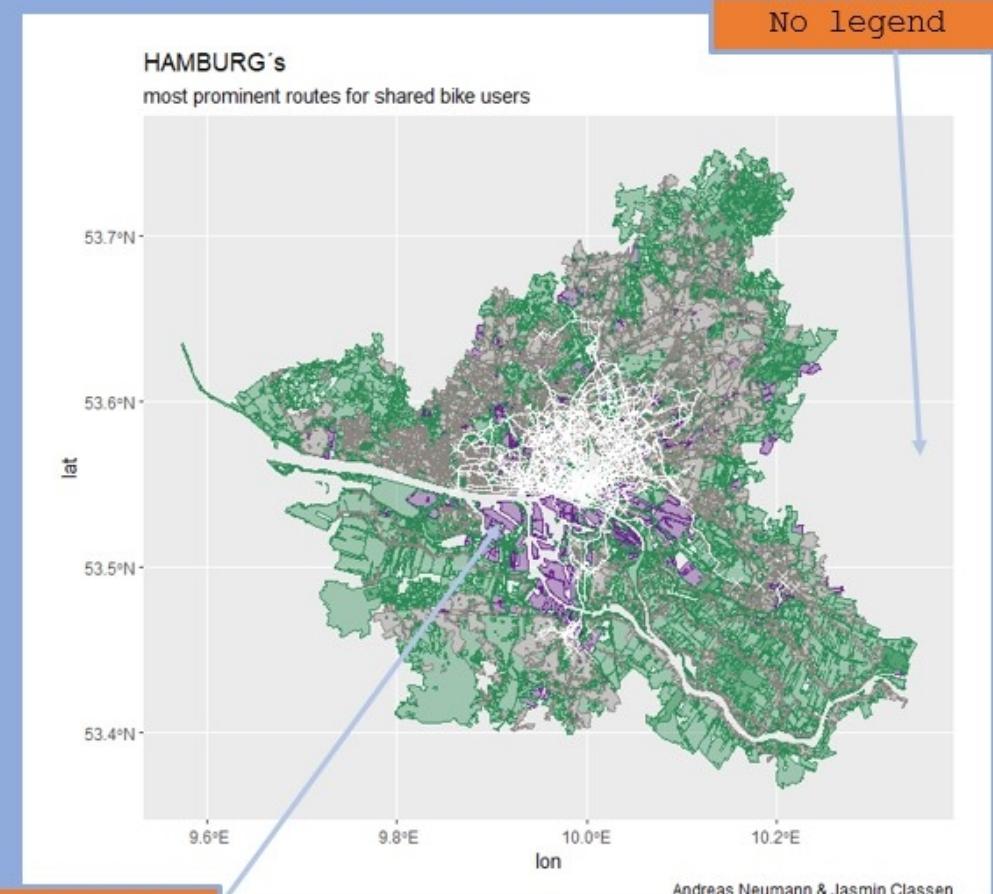
Right filling - wrong contour colours

Input

```
ggplot() +  
  geom_sf(land_use, mapping=aes(fill=cat, col=cat), alpha = .4) +  
  geom_point(bikeshare_station, mapping=aes(x=lon, y=lat),  
             col="white", shape = 4, size=2, alpha=5) +  
  geom_sf(paths_street_direct_sampled, mapping=aes(geometry=geom_bike), col=alpha("white", 5)) +  
  labs(title = "\nHAMBURG's", subtitle = "most prominent routes for shared bike users", caption=glue("Andreas Neumann & Jasmin Classen")) +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential" = "seashell14", "business"="darkorchid4"), guide="none") +  
  scale_color_manual(values=c("green space"="seagreen4", "residential" = "seashell14", "business"="darkorchid4"),  
                     guide = "none") + ...
```

Output

No legend



Filling and contour colours match

Code block 1

B

```
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, face="italic"))
```



Code block 1

Plot & Panel
background

```
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, face="italic"))
```

B



Code block 1

```
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, face="italic"))
```

X,Y axis

B





Code block 1

```
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, face="italic"))
```

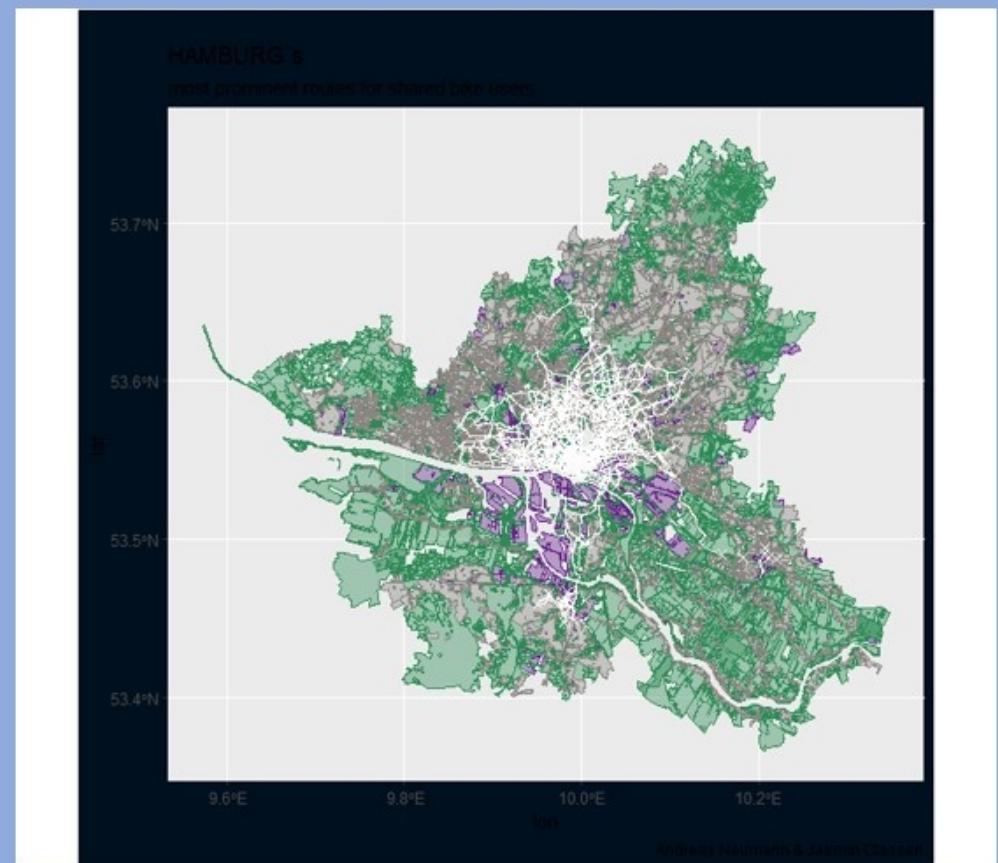
font

B

Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times  
New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, fa  
ce="italic"))
```

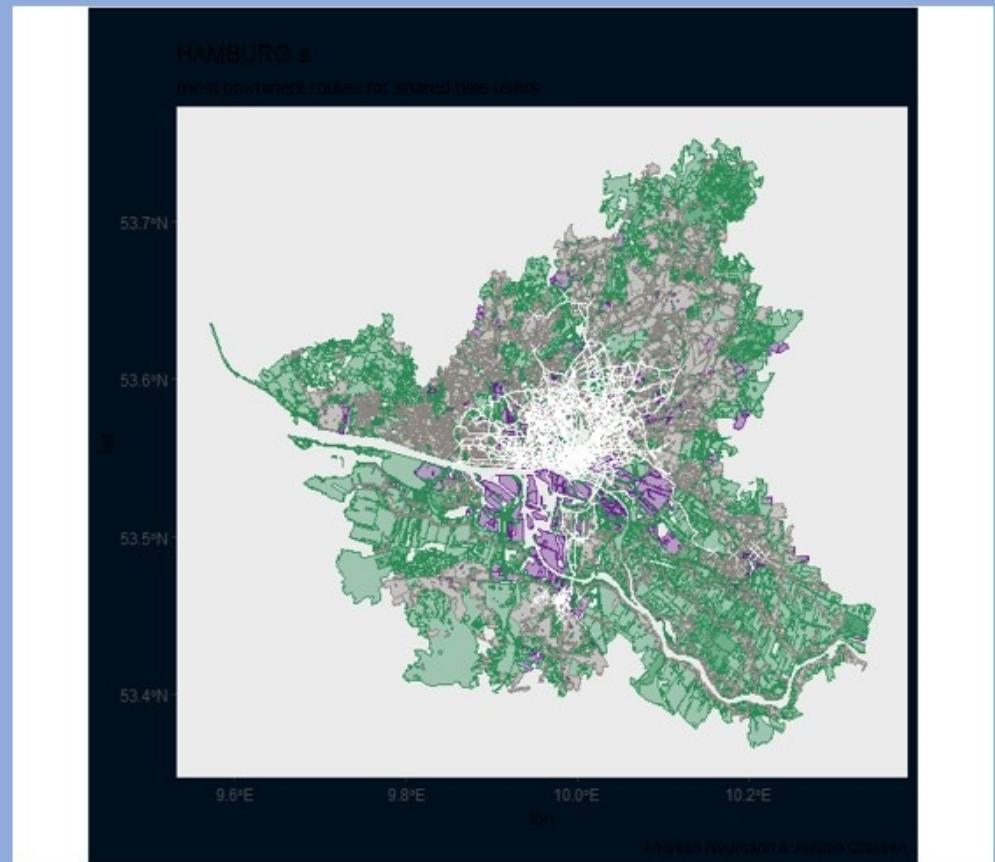
Output



Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times  
New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, fa  
ce="italic"))
```

Output



Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times  
New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, fa  
ce="italic"))
```

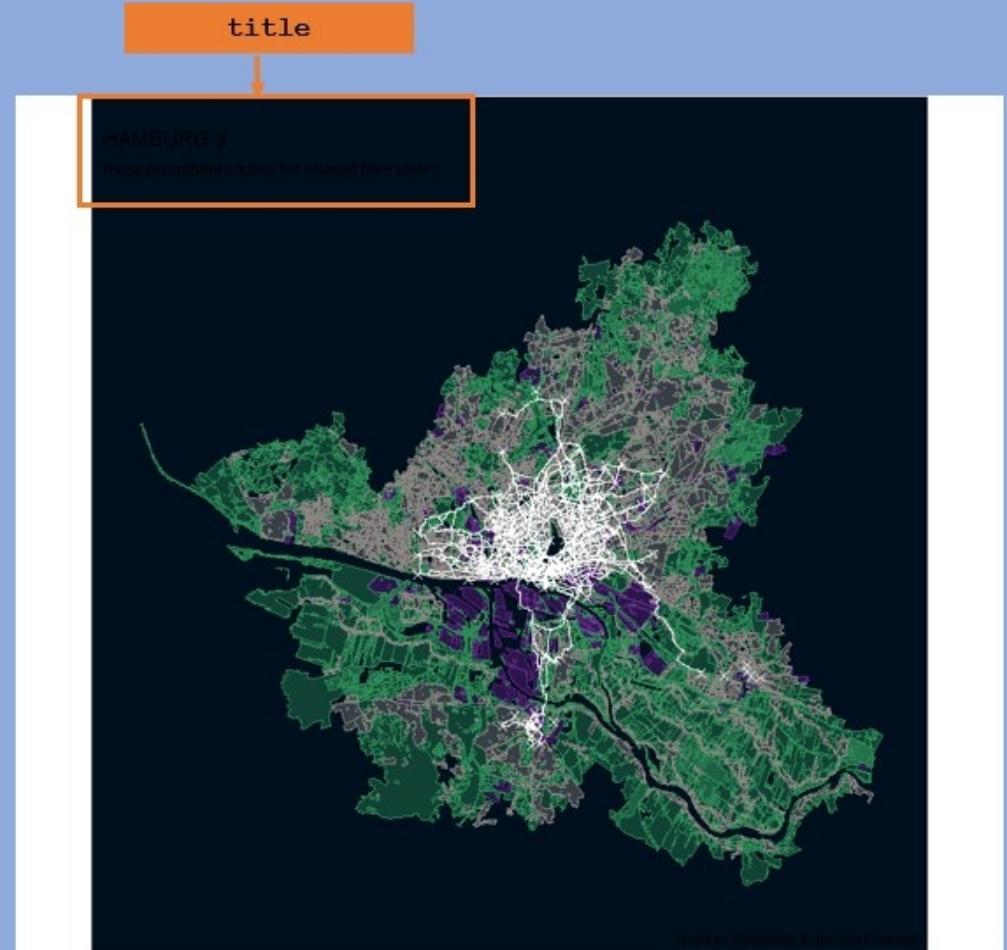
Output



Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times  
New Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, fa  
ce="italic"))
```

Output

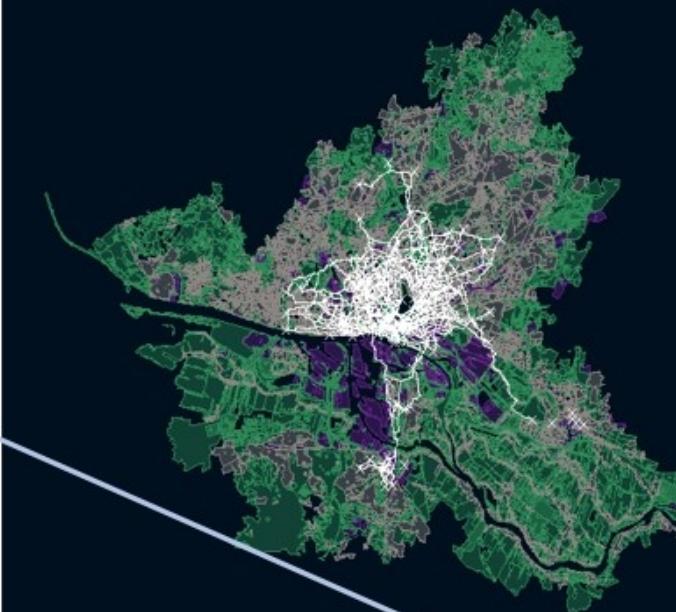


Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New  
Roman", face="bold", size=21,color="white"),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85",hjust=.5, fa  
ce="italic"))
```

Output

HAMBURG's
most prominent routes for shared bike users

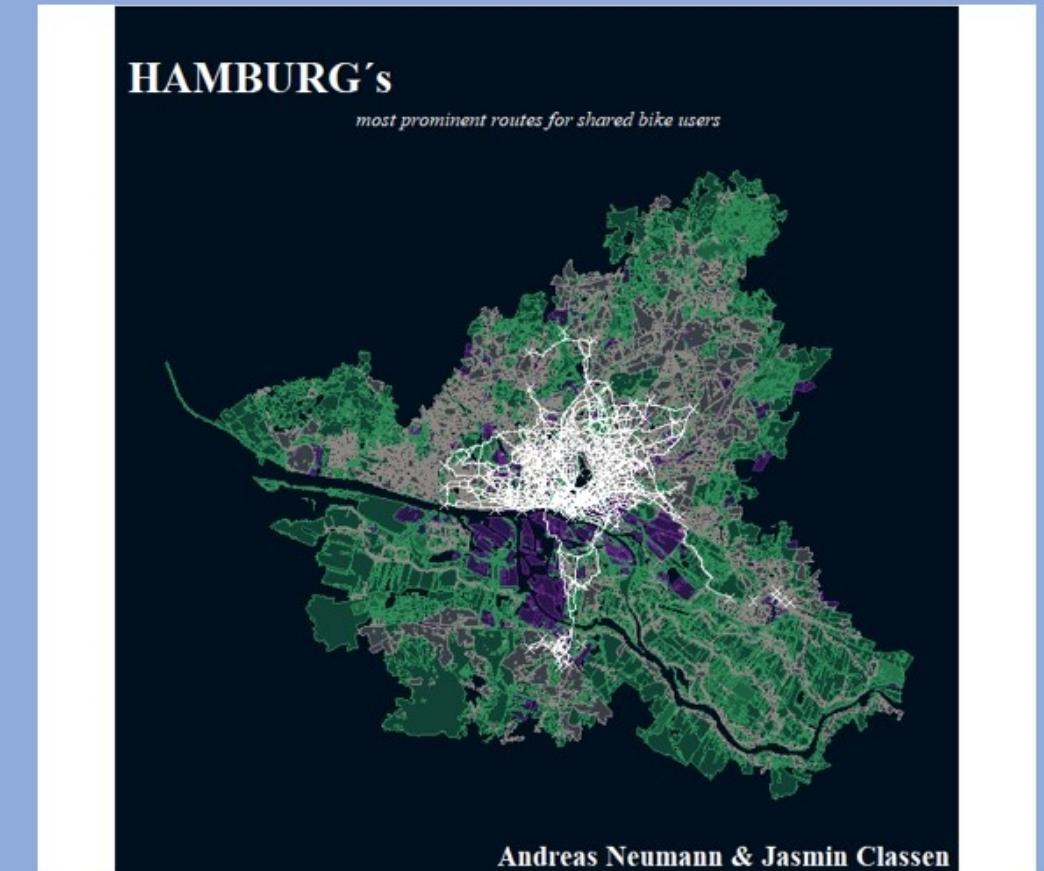


Andreas Neumann & Jasmin Classen

Input

```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New  
Roman", face="bold", size=21),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85", hjust=.5, fa  
ce="italic"))
```

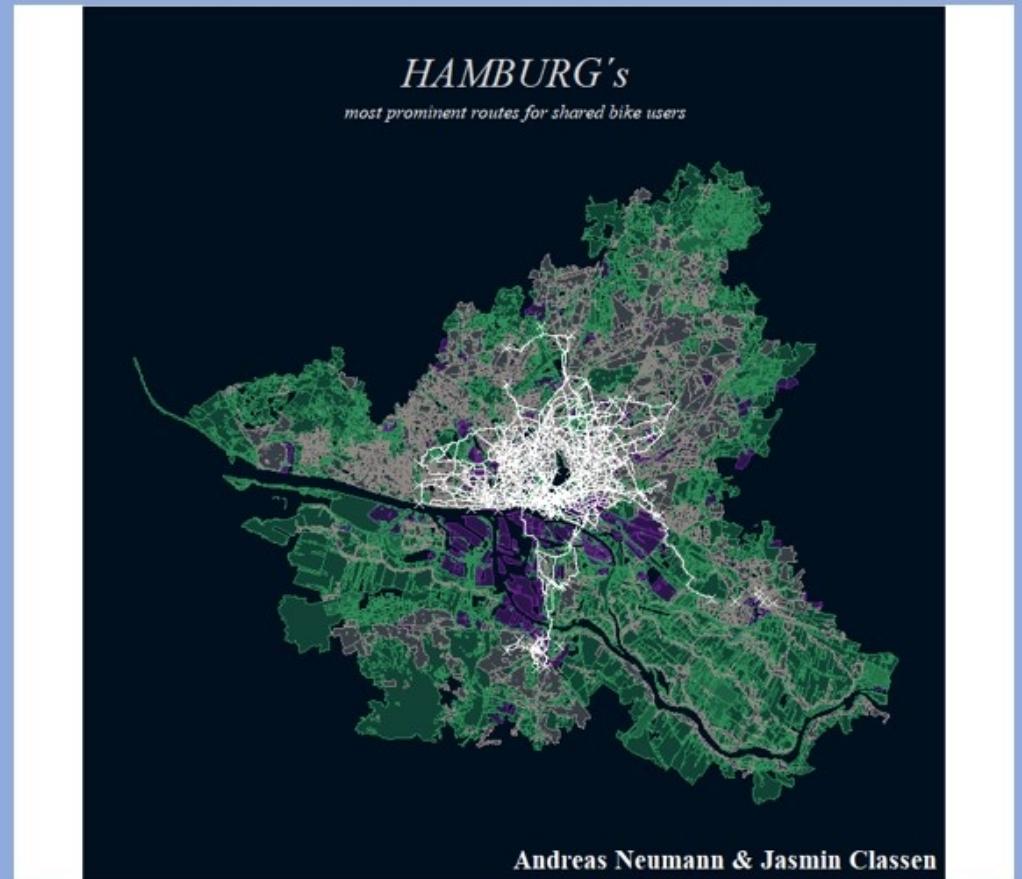
Output



Input

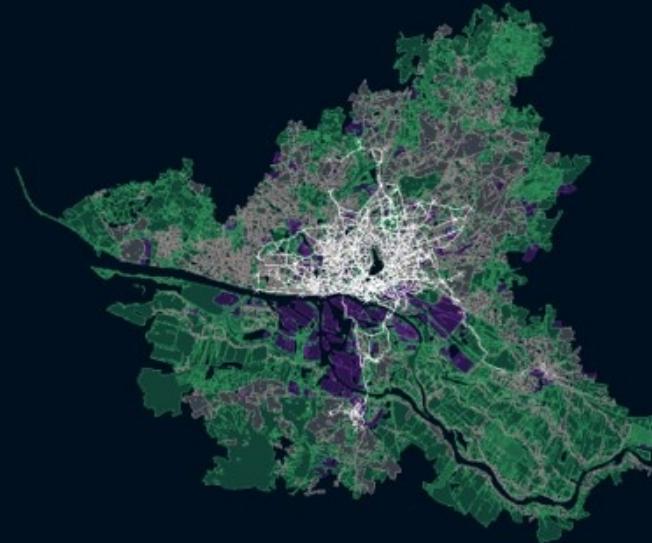
```
+  
theme(  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.background = element_rect("#00101f"),  
  axis.text = element_blank(),  
  axis.ticks = element_blank(),  
  axis.title = element_blank(),  
  text=element_text(family="Times New  
Roman", face="bold", size=21),  
  plot.subtitle=element_text(size=12, hjust=0.5, face="  
italic", color="white"),  
  plot.title = element_text(color="grey85", hjust=.5, fa  
ce="italic"))
```

Output



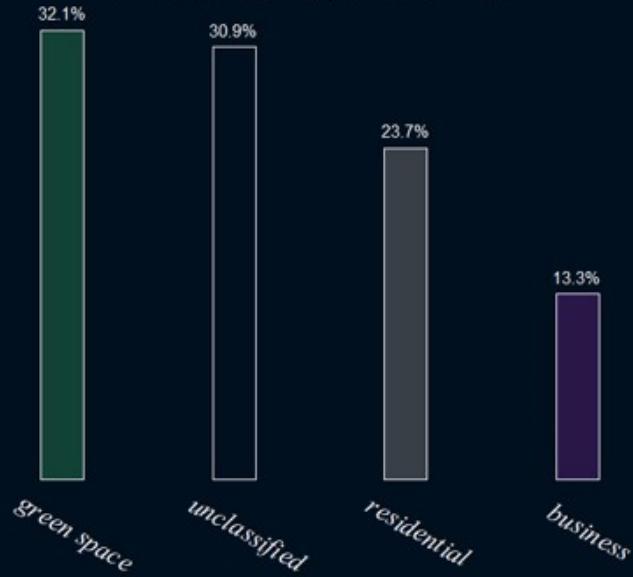
HAMBURG's

most prominent routes for shared bike users

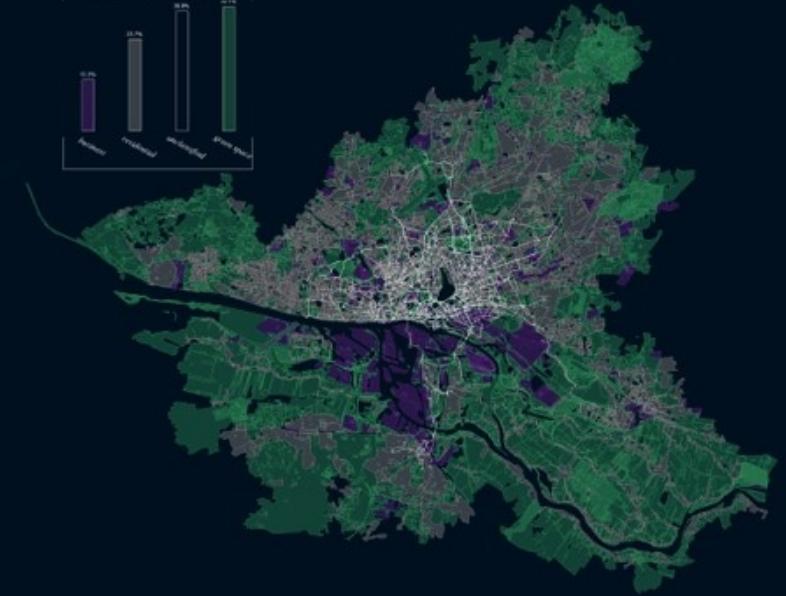
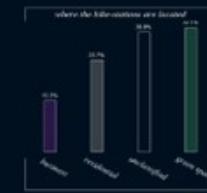


Andreas Neumann & Jasmin Classen

where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

1

2

3

HAMBURG's

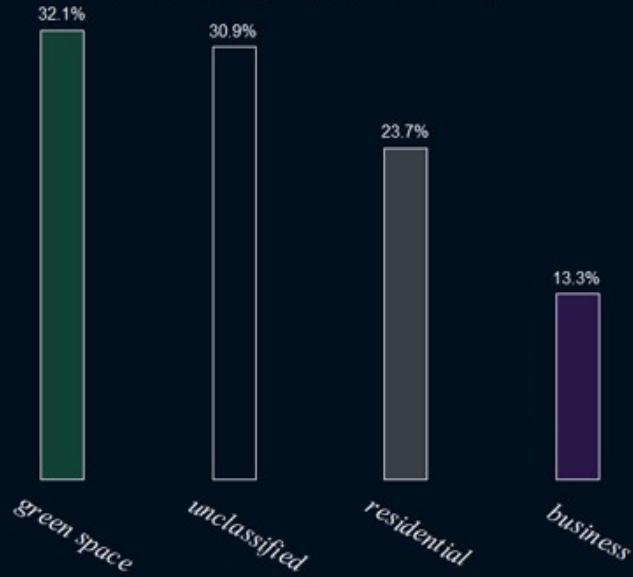
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

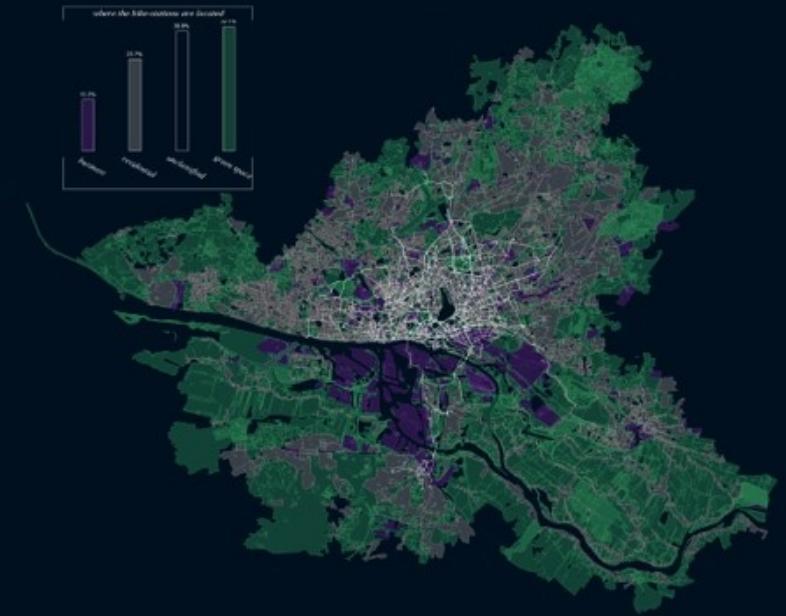
1

where the bike-stations are located



2

HAMBURG's
most prominent routes for shared bike users



3

Station_count_barchart<-

```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-freq), freq, fill=cat), stat="identity", width=.25, alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are located") +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential" ="seashell4",  
  "unclassified"="#00101f", "business"="darkorchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label = scales::percent(freq)), colour = "white",  
  size = 4, vjust = -.8) +
```

A

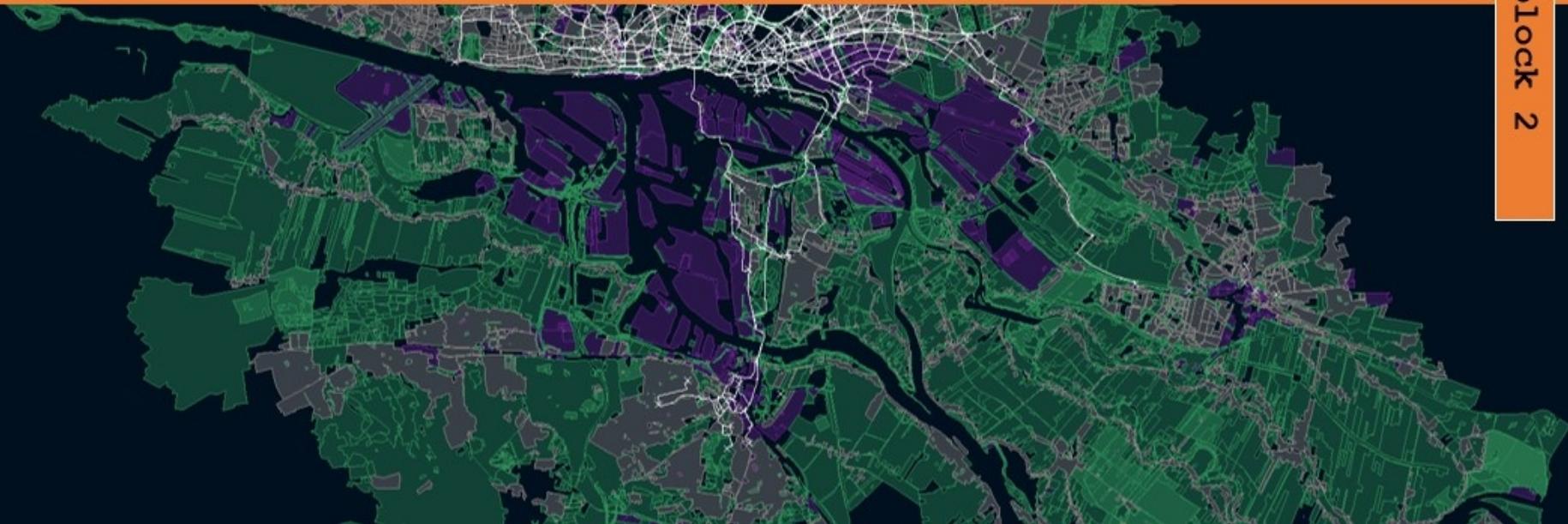
```
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5, face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17, face="italic", angle=-30, hjust = .48, vjust = -.8, color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman", face="bold", size=21))
```

Code block 2

B

Code block 2

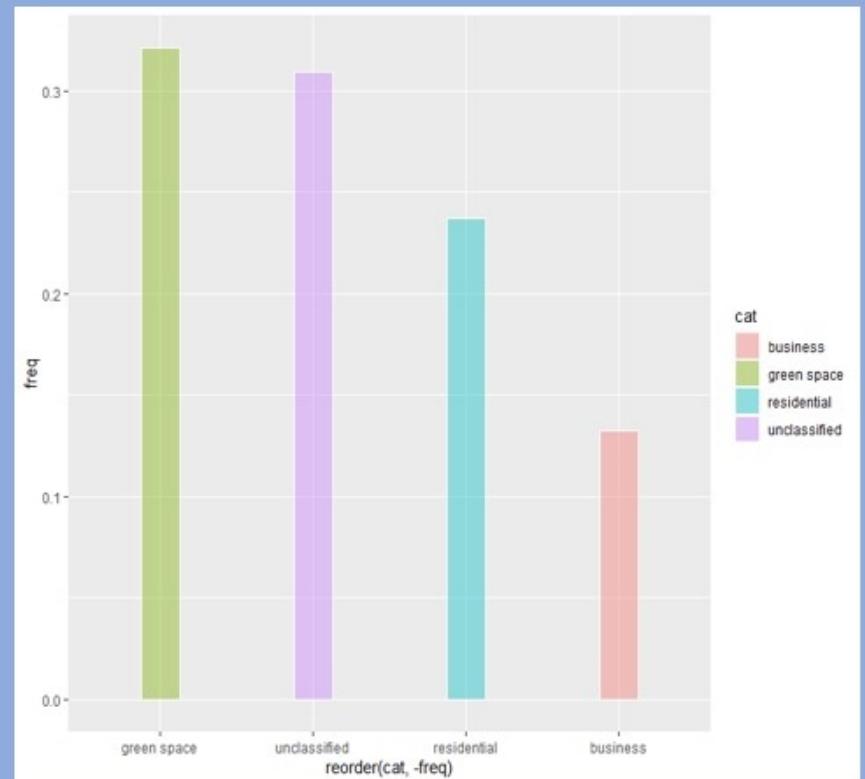
```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-freq), freq, fill=cat), stat="identity", width=.25, alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are located") +  
  scale_fill_manual(name="Category:", values=c("green space"="seagreen4", "residential"  
  ="seashell14", "unclassified"="#00101f", "business"="darkorchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label = scales::percent(freq)), colour = "white",  
  size = 4, vjust = -.8) +
```



Input

```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-  
    freq), freq, fill=cat), stat="identity", width=.25,  
  alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are  
  located") +  
  scale_fill_manual(name="Category:", values=c("green  
  space"="seagreen4", "residential"  
  ="seashell4", "unclassified"="#00101f", "business"="dar  
  korchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label =  
    scales::percent(freq)), colour = "white",  
    size = 4, vjust = -.8) +
```

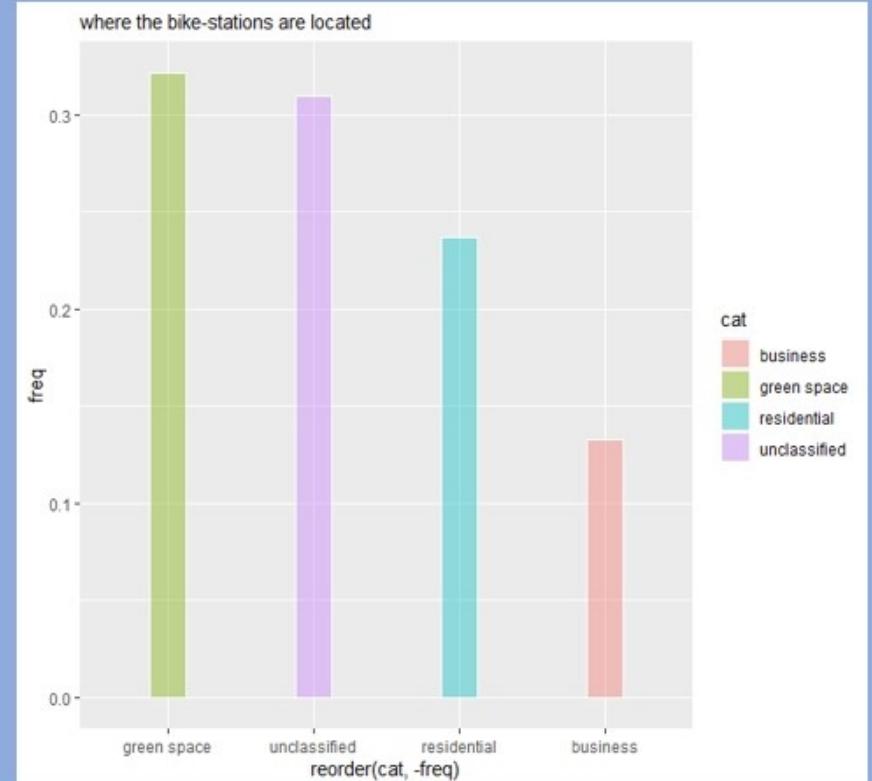
Output



Input

```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-  
    freq), freq, fill=cat), stat="identity", width=.25,  
  alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are  
  located") +  
  scale_fill_manual(name="Category:", values=c("green  
  space"="seagreen4", "residential"  
  ="seashell4", "unclassified"="#00101f", "business"="dar  
  korchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label =  
    scales::percent(freq)), colour = "white",  
  size = 4, vjust = -.8) +
```

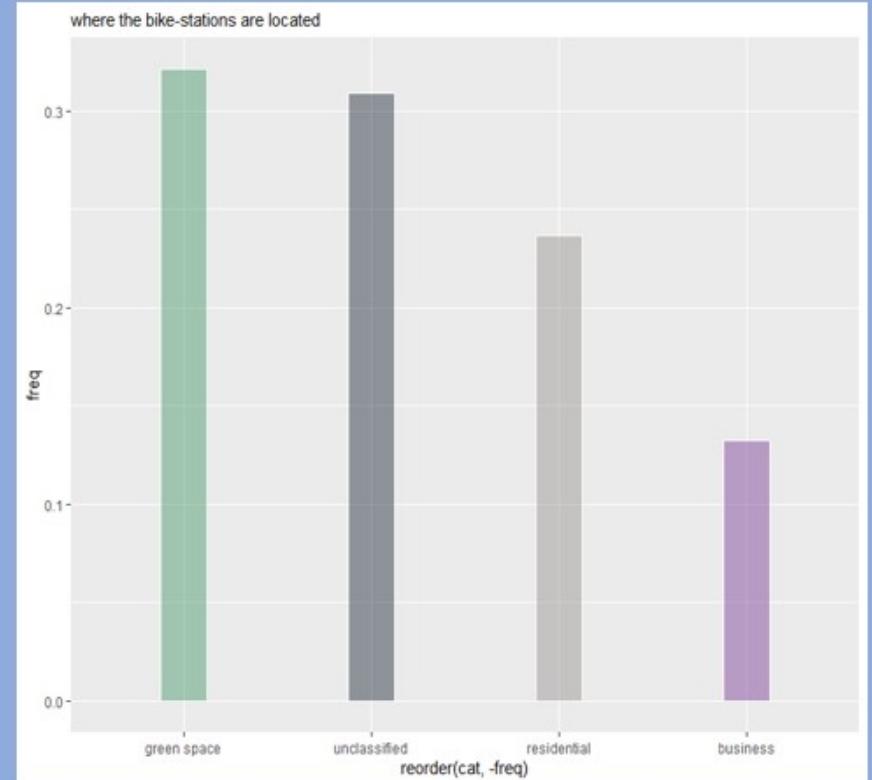
Output



Input

```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-  
    freq), freq, fill=cat), stat="identity", width=.25,  
    alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are  
    located") +  
  scale_fill_manual(name="Category:", values=c("green  
    space"="seagreen4", "residential"  
    ="seashell4", "unclassified"="#00101f", "business"="dar  
    korchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label =  
    scales::percent(freq)), colour = "white",  
    size = 4, vjust = -.8) +
```

Output

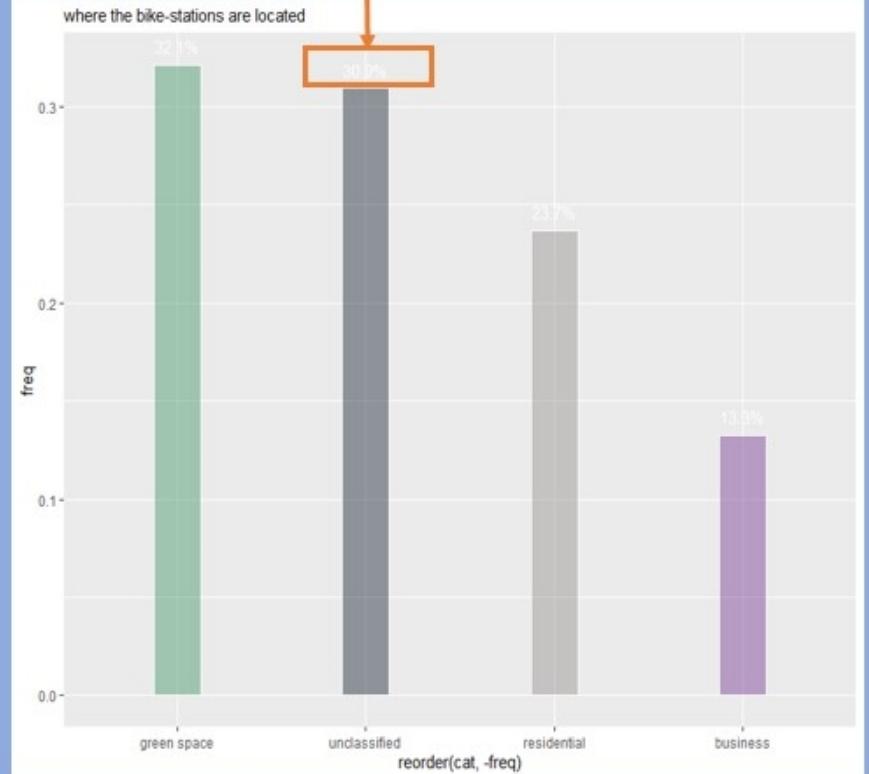


Input

```
ggplot(station_landuse_count)+  
  geom_bar(mapping=aes(reorder(cat,-  
    freq), freq, fill=cat), stat="identity", width=.25,  
    alpha=.4, col="white") +  
  labs(subtitle = "where the bike-stations are  
  located") +  
  scale_fill_manual(name="Category:", values=c("green  
  space"="seagreen4", "residential"  
  ="seashell4", "unclassified"="#00101f", "business"="dar  
  korchid4"), guide="none") +  
  geom_text(mapping=aes(reorder(cat,-freq), freq, label =  
    scales::percent(freq)), colour = "white",  
    size = 4, vjust = -.8) +
```

Output

Percentages added



Code block 2

B

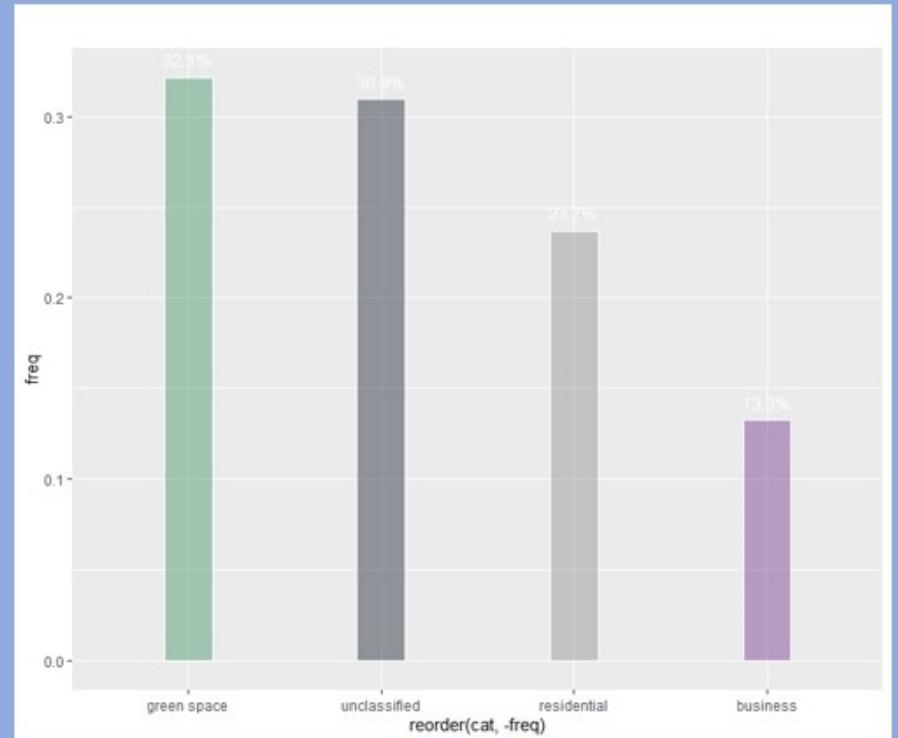
```
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5, face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17, face="italic",angle=-30,hjust = .48,vjust = -.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman", face="bold", size=21))
```



Input

```
+  
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5,  
  face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17,  
  face="italic",angle=-30,hjust = .48,  
  vjust =-.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman",  
  face="bold", size=21))
```

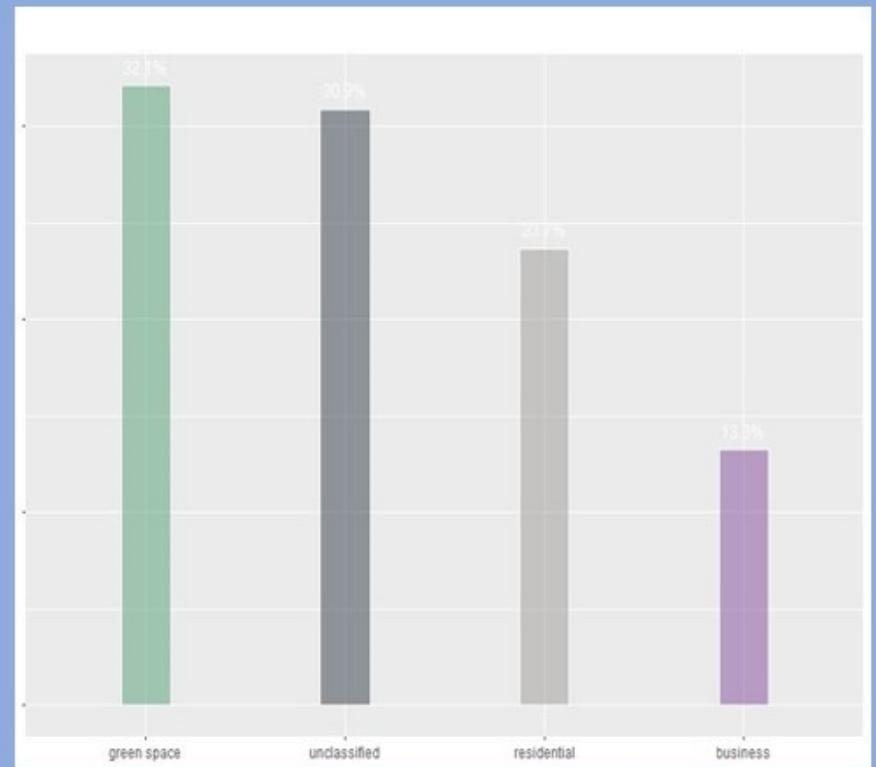
Output



Input

```
+  
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5,  
  face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17,  
  face="italic",angle=-30,hjust = .48,  
  vjust =-.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman",  
  face="bold", size=21))
```

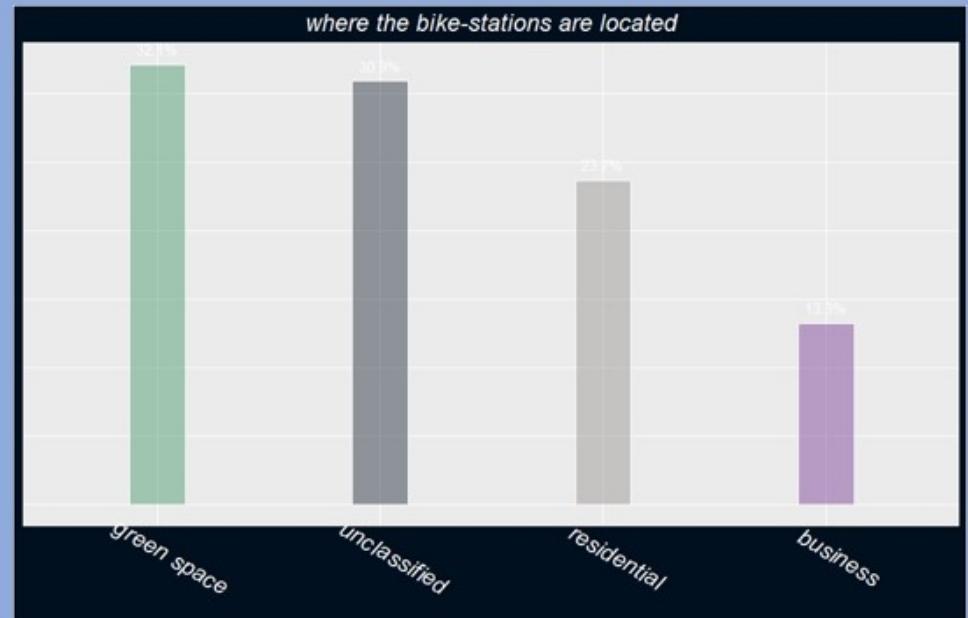
Output



Input

```
+  
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5,  
  face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17,  
  face="italic",angle=-30,hjust = .48,  
  vjust =-.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman",  
  face="bold", size=21))
```

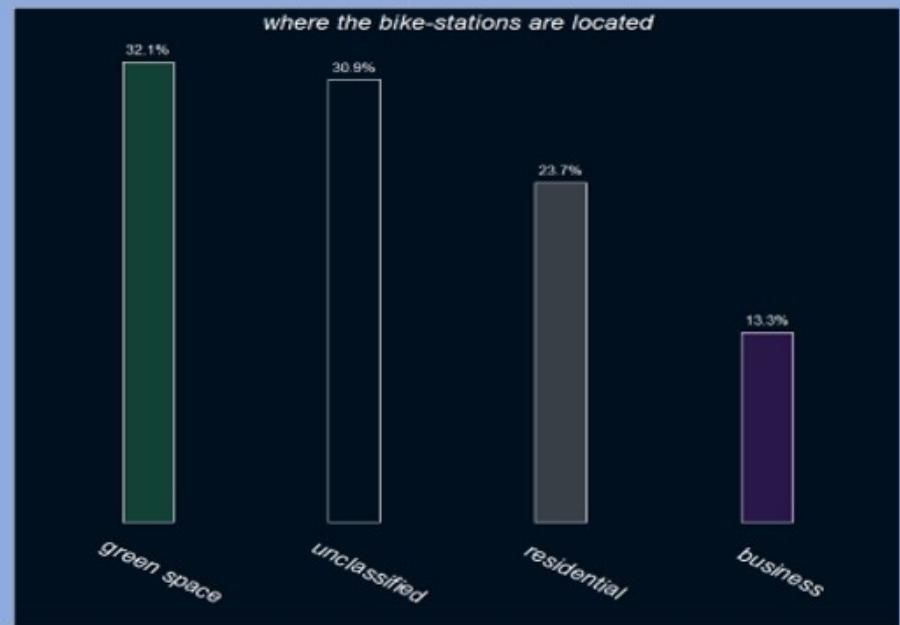
Output



Input

```
+  
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5,  
  face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17,  
  face="italic",angle=-30,hjust = .48,  
  vjust =-.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman",  
  face="bold", size=21))
```

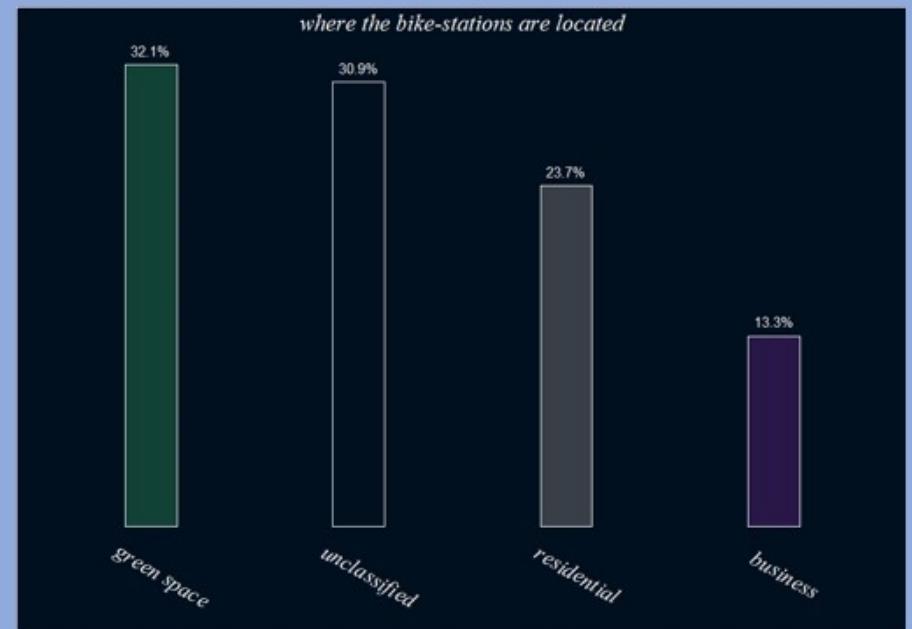
Output



Input

```
+  
theme(  
  plot.subtitle=element_text(size=18, hjust=0.5,  
  face="italic", color="white"),  
  axis.title = element_blank(),  
  axis.title.y = element_blank(),  
  axis.text.y = element_blank(),  
  axis.text.x = element_text(size=17,  
  face="italic",angle=-30,hjust = .48,  
  vjust =-.8,color="white"),  
  axis.ticks = element_blank(),  
  plot.background = element_rect("#00101f"),  
  panel.grid.major = element_blank(),  
  panel.grid.minor = element_blank(),  
  panel.background = element_rect("#00101f"),  
  text=element_text(family="Times New Roman",  
  face="bold", size=21))
```

Output



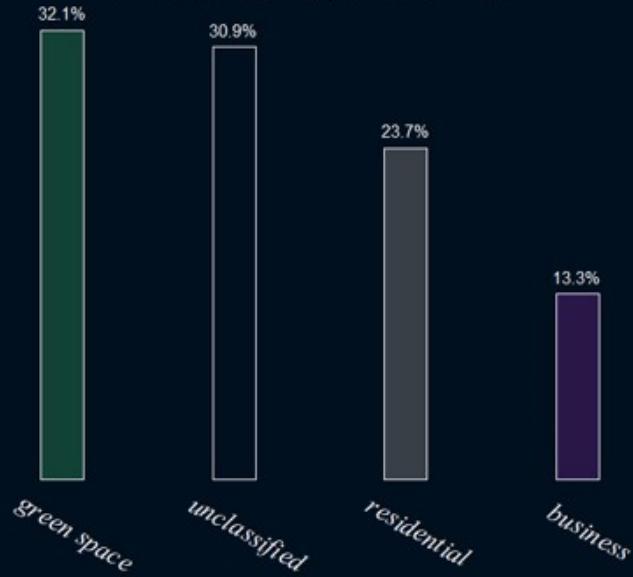
HAMBURG's

most prominent routes for shared bike users

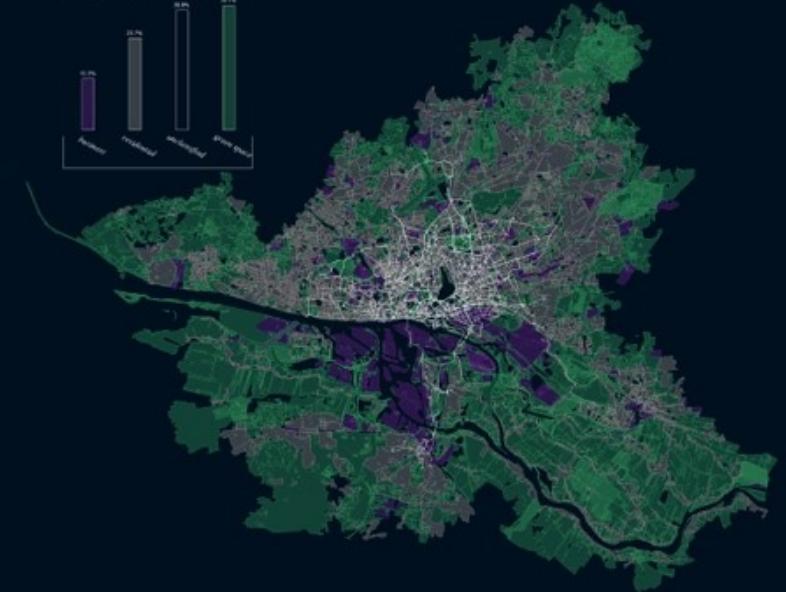
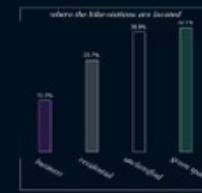


Andreas Neumann & Jasmin Classen

where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

1

2

3

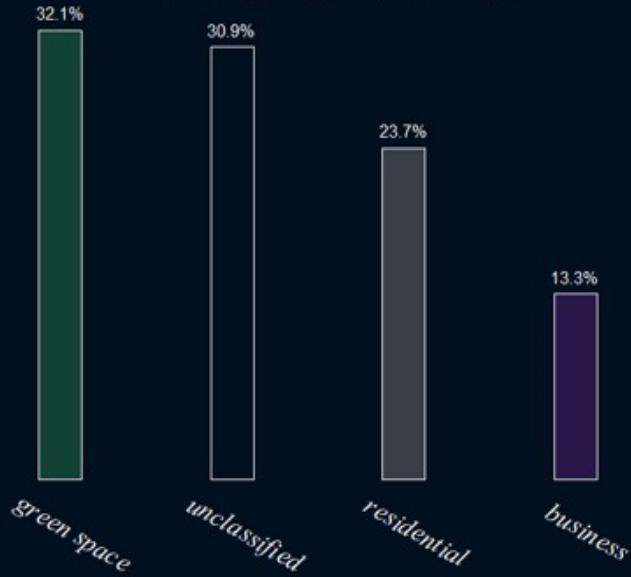
HAMBURG's

most prominent routes for shared bike users

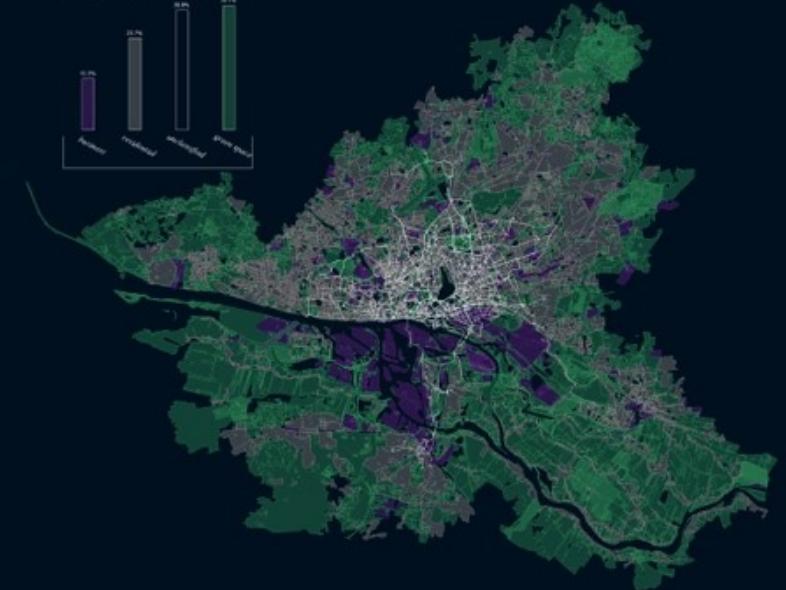
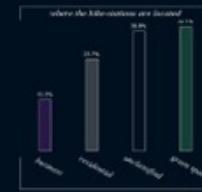


Andreas Neumann & Jasmin Classen

where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



Andreas Neumann & Jasmin Classen

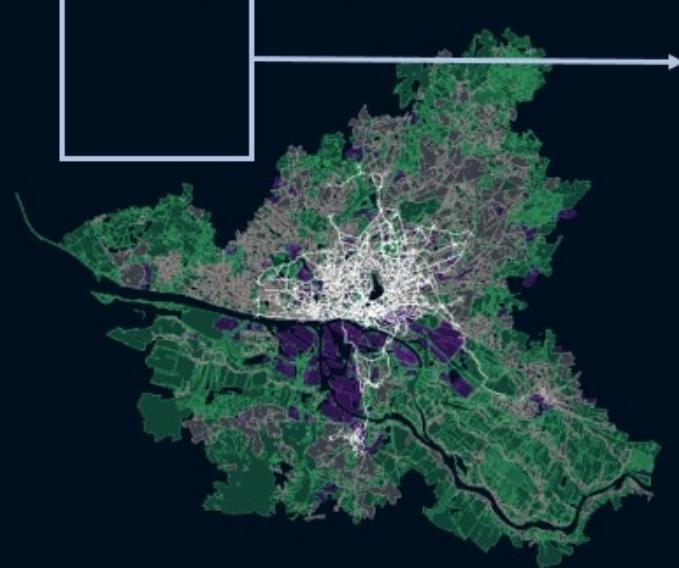
1

2

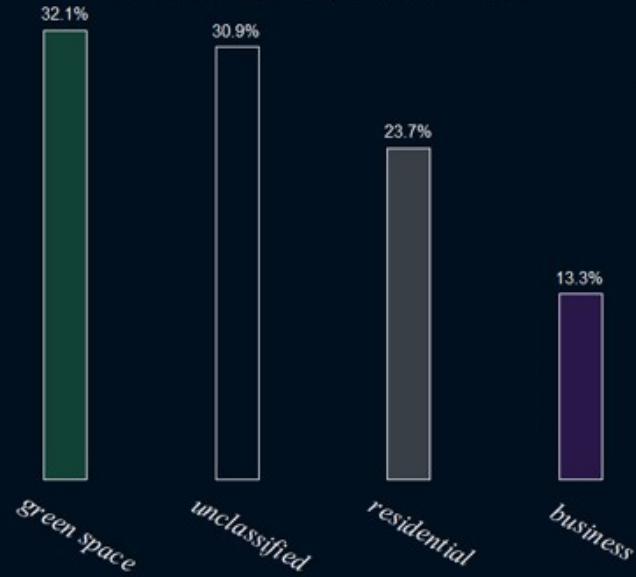
3

HAMBURG's

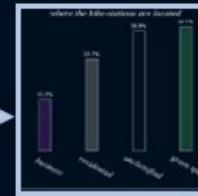
most prominent routes for shared bike users



where the bike-stations are located



HAMBURG's
most prominent routes for shared bike users



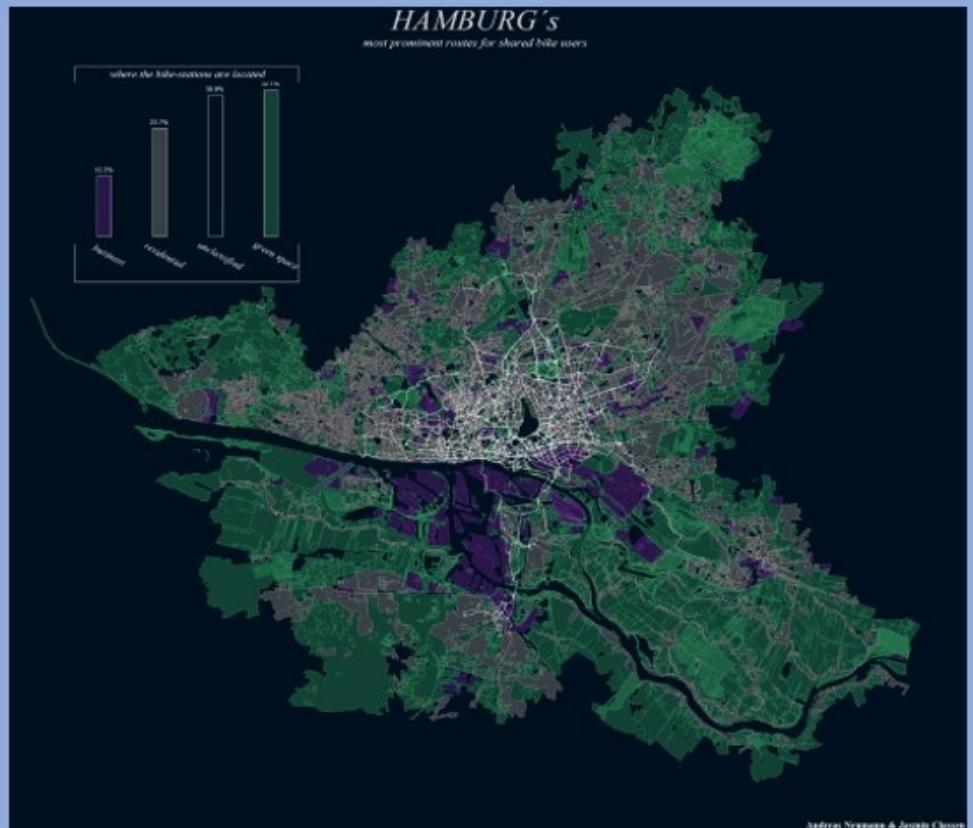
Andreas Neumann & Jasmin Classen

Andreas Neumann & Jasmin Classen

Input

```
hamburg_map+inset_element(station_count_barchart,  
left = .3,  
bottom = .7,  
right = .1,  
top = .98)
```

Output



Input

```
hamburg_map+inset_element(station_count_barchart,  
left = .3,  
bottom = .7,  
right = .1,  
top = .98)  
  
Or  
  
library("egg")  
  
hamburg_map+annotation_custom(  
  ggplotGrob(station_count_barchart),  
  xmin = 9.6, xmax = 9.85, ymin = 53.65, ymax = 53.75)
```

Output

