

Wohnlagen in Bielefeld

Datenbasis für den Code Sprint sind die städtischen Daten zur Wohnlagenkarte [1] aus dem open-data Portal der Stadt Bielefeld. Weitere Teile der Stadt, die ausgezeichneten Wohnflächen, werden von diesem Datensatz durch bewertete Polygone überdeckt.

```
library(sf)
library(ggplot2)
library(ggmap)
library(magrittr)
library(dplyr)
library(tidyr)
library(units)
library(ggridges)
```

Die Bewertung im Ausgangsdatensatz

Die Definition der Wohnlage ist mit den Kennzahlen von 1 bis 4 angegeben. Die Stadt Bielefeld schreibt dazu in [1]:

1: Einfache Wohnlage

Als einfache Wohnlagen gelten Wohngebiete mit nachteilig wirkendem Charakter, z. B. durch die Lage an stark frequentierten Verkehrswegen, ein negatives Image als sozialer Brennpunkt und unzureichende Versorgungsinfrastruktur und/oder ihre Nähe zu größeren Gewerbe- und Industriegebieten

2: Normale Wohnlage

Der überwiegende Teil des Stadtgebietes wird durch die normale Wohnlage ohne besondere Vor- und Nachteile geprägt. Durchschnittliche Immissionsbelastungen kennzeichnen diese Gebiete. Grundstücke mit genügend Freiflächen sind charakteristisch für normale Wohnlagen, die den Nachteil der durchschnittlichen Immissionsbelastung wieder ausgleichen.

3: Gute Wohnlage

Gute Wohnlagen befinden sich meist in einem weitgehend abgeschlossenen, durchgrünten Wohngebiet mit aufgelockerter Bebauung, verkehrsgünstig gelegen und in der Nähe von Erholungsflächen, positives Image. Stark nachgefragte innerstädtische Wohnquartiere (Urbanes Wohnen) fallen ebenfalls in diese Kategorie.

4: Sehr gute/bevorzugte Wohnlage

Besonders imageträchtige und stark nachgefragte Innenstadtrandlagen. Sehr gute Wohnlagen weisen in der Regel alle Vorteile der guten Wohnlage auf, verfügen darüber hinaus noch über mindestens eine weiteren herausragenden Nutzungsvorteil wie Topographie oder eine besonders schöne Landschaft, Nachbarschaft und Umgebung. Sehr gute/bevorzugte Wohnlagen sind oft geprägt durch luxuriöse Bebauung mit großzügigen Außenanlagen und großen Abständen zur Nachbarbebauung. In diese Kategorie fallen auch Stadthäuser, häufig aus der Gründerzeit.

Laden der Daten und ein erster Blick

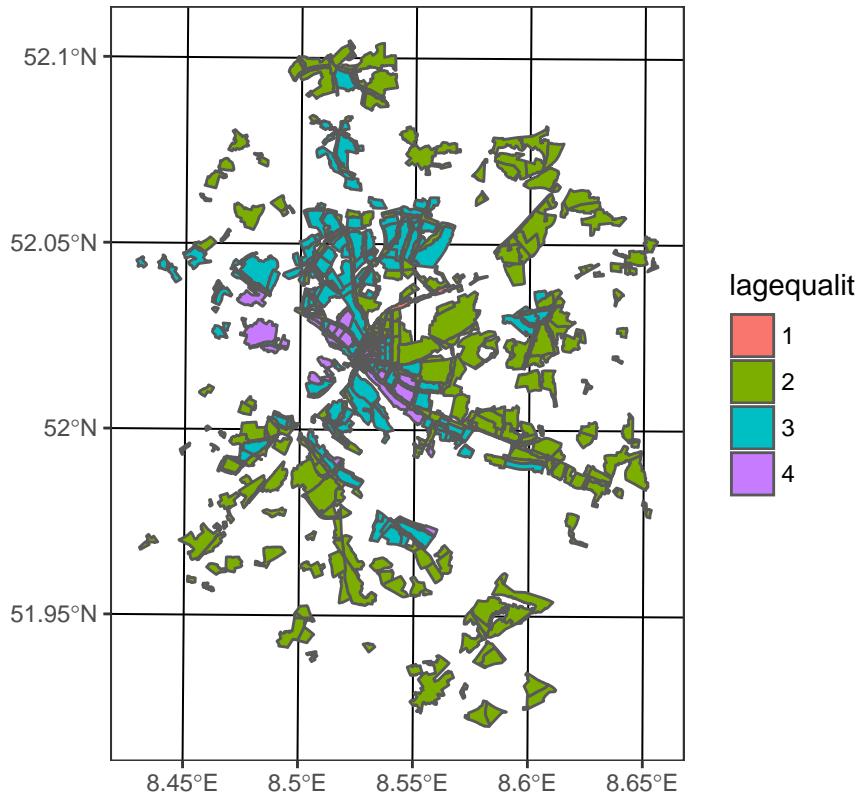
Recht puristisch ist der erste interessante Einblick auf die Daten. Es handelt sich um 809 bewertete Polygone, der Koordinaten im Koordinatenbezugssystem EPSG25832 gegeben sind:

```
wk <- st_read("wohnlagenkarte.gml")

## Reading layer `wohnlagenkarte_pl' from data source `/home/hanse/R/opendata_bielefeld/wohnlagenkarte.gml'
## Simple feature collection with 809 features and 5 fields
## geometry type:  MULTIPOINT
## dimension:      XY
## bbox:            xmin: 460795.4 ymin: 5752265 xmax: 476396.9 ymax: 5772789
## epsg (SRID):   25832
## proj4string:    +proj=utm +zone=32 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs

ggplot(wk) + geom_sf(data = wk, aes(fill = lagequalit)) +
  ggtitle("Wohnlagenkarte von Bielefeld") +
  theme_bw()
```

Wohnlagenkarte von Bielefeld



Wir bekommen auch die offiziellen Stadtbezirksgrenzen gemäß amtlichem Liegenschaftskataster:

```
bzrk <- st_read("stadtbezirke/result.gml")

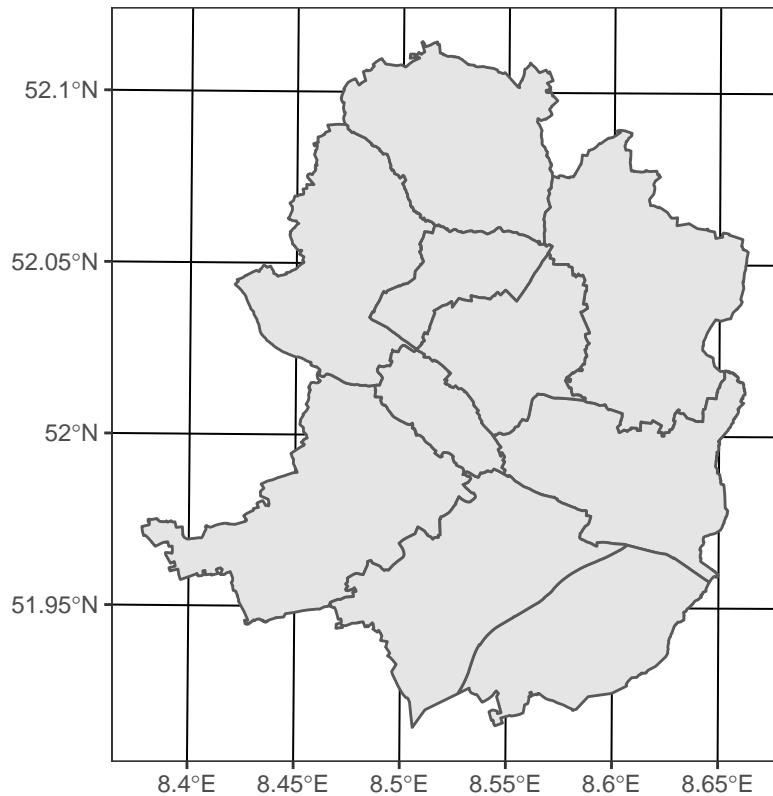
## Reading layer `stadtbezirke_pl' from data source `/home/hanse/R/opendata_bielefeld/stadtbezirke/result.gml'
## Simple feature collection with 10 features and 1 field
## geometry type:  POLYGON
## dimension:      XY
## bbox:            xmin: 457260.8 ymin: 5751685 xmax: 476916.8 ymax: 5773917
## epsg (SRID):   25832
```

```

## proj4string: +proj=utm +zone=32 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
ggplot(bzrk) + geom_sf(data = bzrk) +
  ggtitle("Stadtbezirksgrenzen von Bielefeld") +
  theme_bw()

```

Stadtbezirksgrenzen von Bielefeld



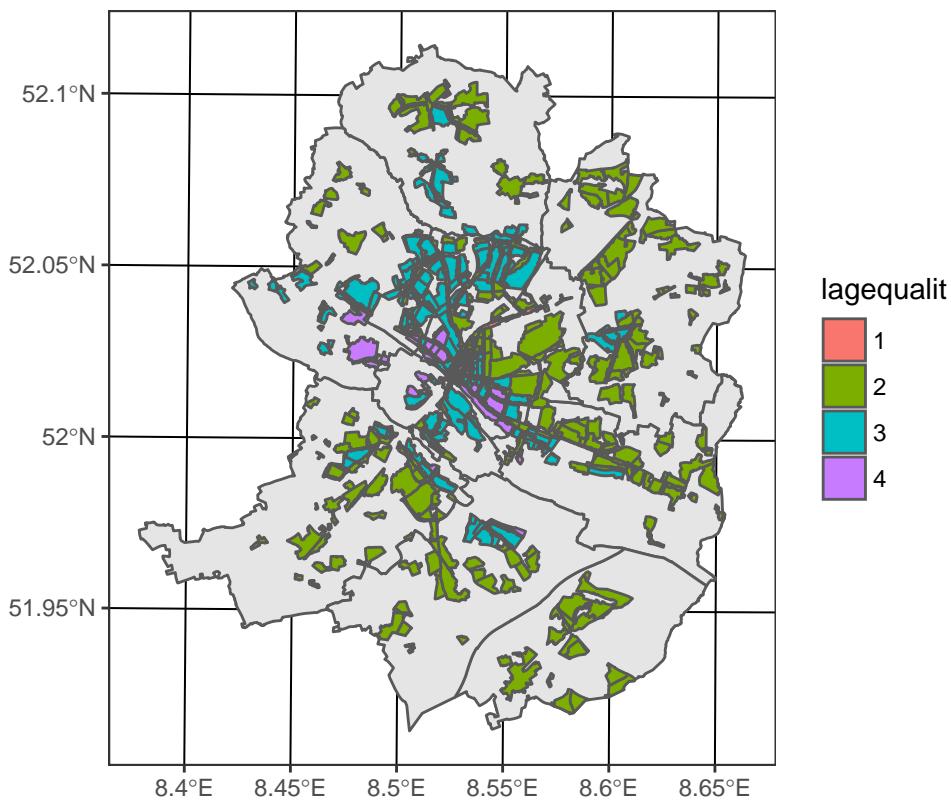
Kombiniert man diese beiden Datensätze, dann ergibt sich folgendes Bild der Wohlagenkarte:

```

ggplot(bzrk) + geom_sf(data = bzrk) +
  geom_sf(data = wk, aes(fill = lagequalit)) +
  ggtitle("Wohnlagenkarte von Bielefeld") +
  theme_bw()

```

Wohnlagenkarte von Bielefeld



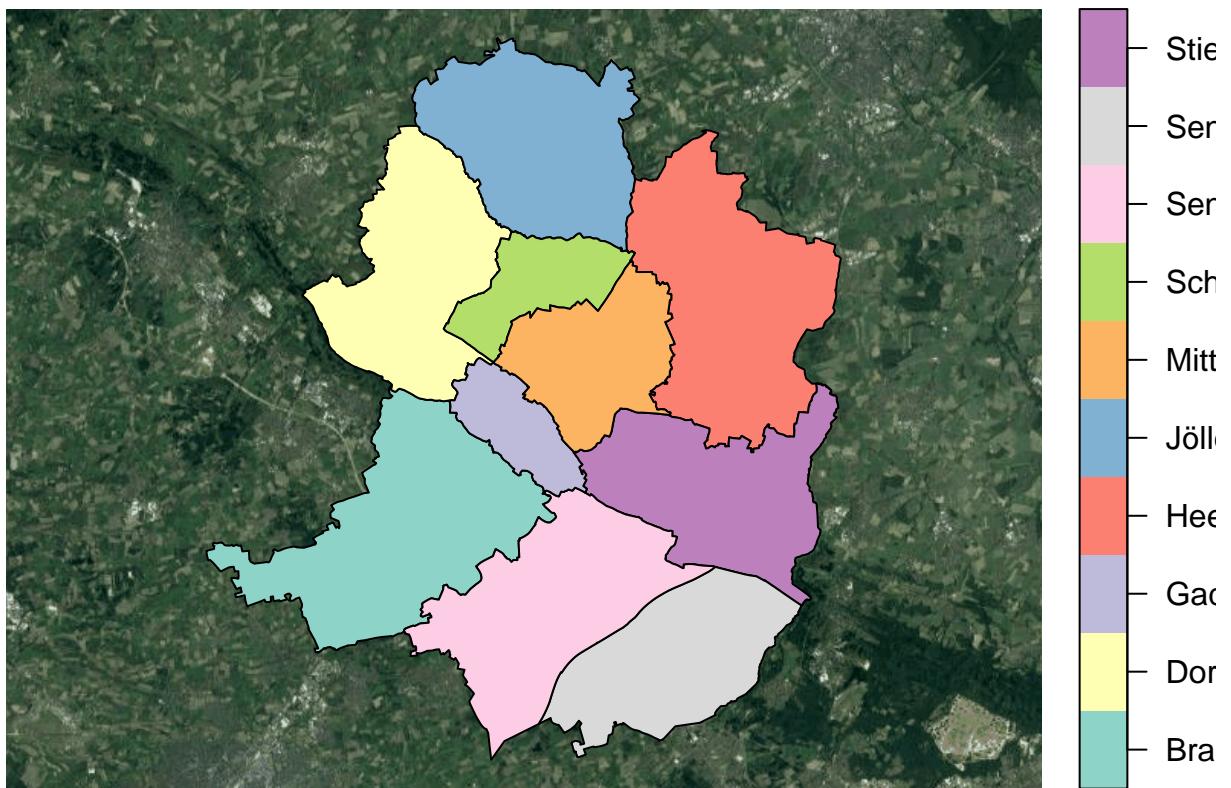
Zum Import von Kartenmaterial lässt sich das Paket ggmap nutzen:

```
bbox <- bzrk %>% st_transform(4326) %>% st_bbox
mapImageData <- get_googlemap(center = c(lon = median(bbox[c(1,3)]),
                                             lat = median(bbox[c(2,4)])), zoom = 10,
                                             size = c(500, 500),
                                             maptype = c("satellite"))

# Benenne die Stadtteile
bzrk$name <- ''
bzrk$name[1] <- 'Brackwede'
bzrk$name[2] <- 'Dornberg'
bzrk$name[3] <- 'Jöllenbeck'
bzrk$name[4] <- 'Heepen'
bzrk$name[5] <- 'Stieghorst'
bzrk$name[6] <- 'Sennestadt'
bzrk$name[7] <- 'Senne'
bzrk$name[8] <- 'Gadderbaum'
bzrk$name[9] <- 'Schildesche'
bzrk$name[10] <- 'Mitte'

bzrk %>% select(name,msGeometry) %>% st_transform(crs = 3857) %>% plot(main = "Stadtbezirke", bgMap =
```

Stadtbezirke

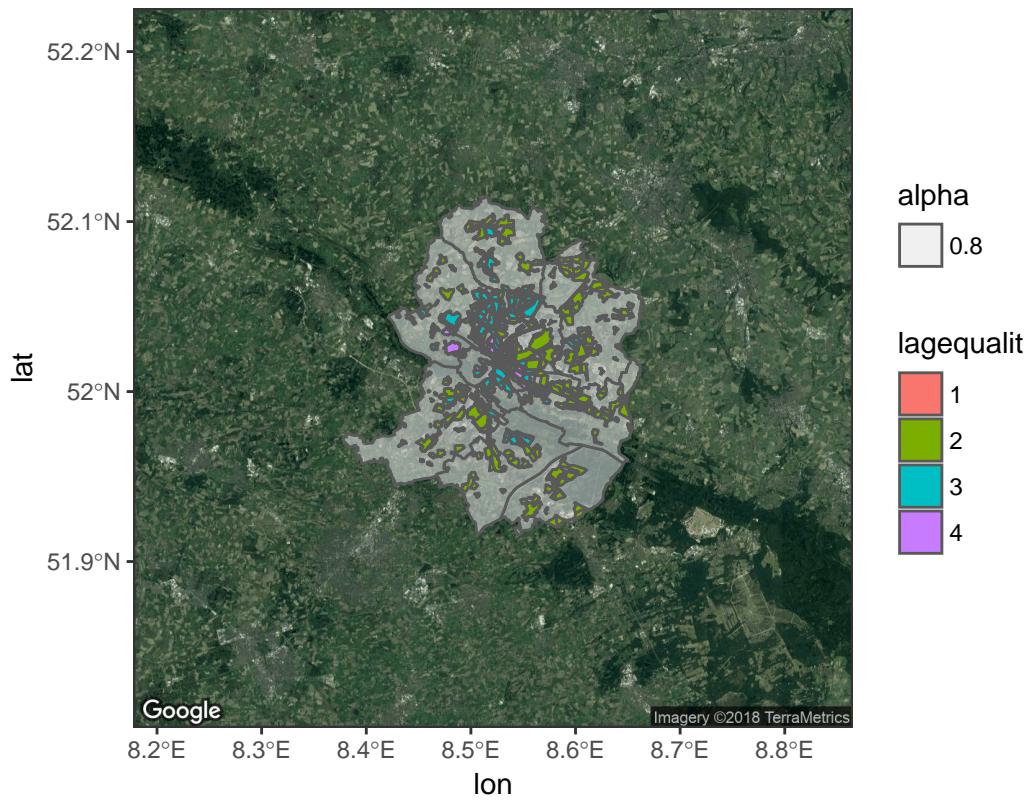


```
bg_google <- ggmap(mapImageData)

#googlemaps uses egps3857?
bzrk2 <- bzrk %>% st_transform(4326)
wk2 <- wk %>% st_transform(4326)
#nc_map = get_map(location = unname(st_bbox(bzrk)))

ggmap(mapImageData) + geom_sf(data = bzrk2, inherit.aes = FALSE, aes(alpha = 0.8)) +
  geom_sf(data = wk2, aes(fill = lagequalit), inherit.aes = FALSE) +
  ggtitle("Wohnlagenkarte von Bielefeld") +
  theme_bw()
```

Wohnlagenkarte von Bielefeld



Wohnlagenbewertung abfragen

Finde die Wohnlagenbewertung zu einer Adresse:

```
#besorge die Koordinaten eines PoI:
pt <- ggmap::geocode('Rolandstraße 16, Bielefeld', source = 'google')

## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Rolandstraße%C3%9Fe%20

# Alternative DataScienceToolkit:
#ggmap::geocode('Stapenhorststraße 100, Bielefeld', source = 'dsk')

# Erzeuge ein sf-Objekt, mit crs 4326
p <- pt %>% st_as_sf(coords = c("lon", "lat"), crs = 4326)

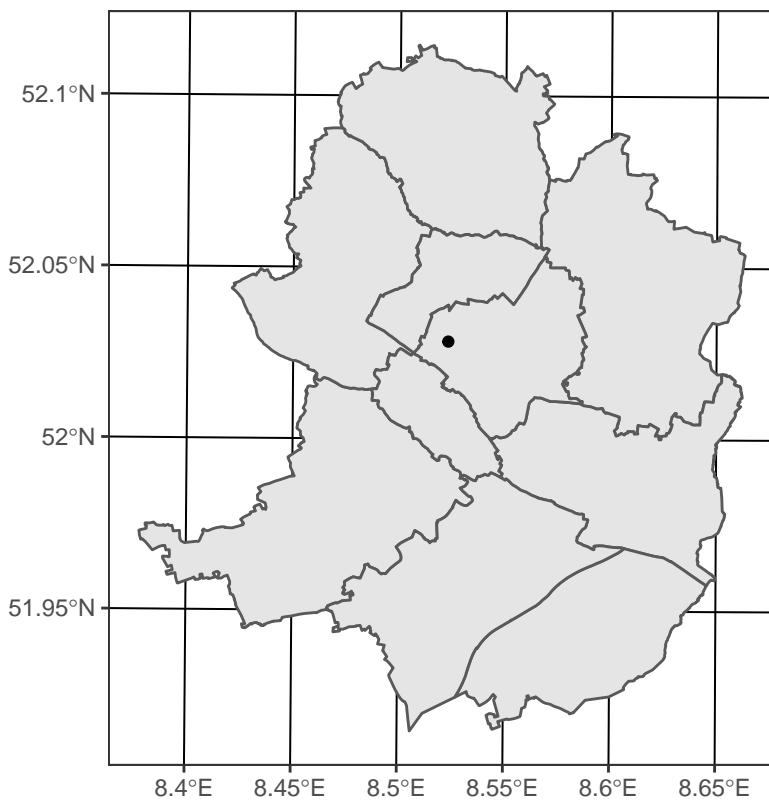
# Angleichen der Koordinatenreferenzsysteme
p %<>% st_transform(crs = st_crs(wk))

# Ausgeben der Lagequalität
st_join(p, wk)$lagequalit

## [1] 3
## Levels: 1 2 3 4

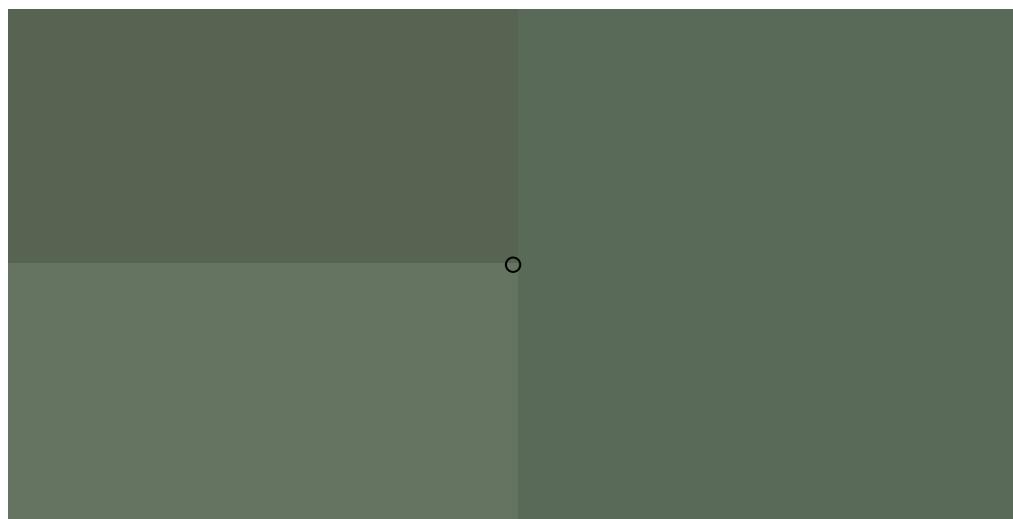
# Visuelle Überprüfung, ob die Koordinatentransformation richtig war:
ggplot(bzrk) + geom_sf(data = bzrk) +
  geom_sf(data = st_join(p, wk)) +
  ggtitle("Stadtbezirksgrenzen von Bielefeld") +
  theme_bw()
```

Stadtbezirksgrenzen von Bielefeld



```
# 2. visuelle Überprüfung anhand von Satellitenbild  
localImageData <- get_googlemap(center = c(lon = pt$lon,  
                                         lat = pt$lat), zoom = 10,  
                                         size = c(500, 500),  
                                         source = "osm",  
                                         maptype = c("satellite"))
```

```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=52.02841,8.52324&zoom=10&size=500x500
```



Qualität der Wohngebiete je Stadtbezirk:

Welchen Anzahl der Wohnfläche eines Stadtbezirks belegen die Wohnlagen der jeweiligen Qualität? Die unten stehende Tabelle ist so zu lesen, dass in Dornberg keine Wohnfläche mit Wohnqualität 1 vorhanden sind und 50% der Wohnfläche von Qualität 3 sind:

```
# Bringe beide Variablen in das gleich Bezugssystem:
wk2 <- wk %>% st_transform(crs = st_crs(bzrk)) %>%
    mutate(area = st_area(msGeometry))

#Hinzufügen einer Flächen-Spalte
bzrk %<%> mutate(area = st_area(msGeometry))

#Berechnung der Wohnfläche je Bezirk
temp <- st_join(wk2, bzrk) %>% st_buffer(0) %>% group_by(name) %>% summarise(warea = sum(area.x)) %>%
lf_warea <- function(bname){
  return(filter(temp, name == bname) %>% select(warea) %>% pull())
}

# Vectorize zerstört noch die Einheit m^2, daher wird diese anschließend
# wieder gesetzt.
lvf_warea <- Vectorize(lf_warea)
bzrk %<%> mutate(warea = set_units(lvf_warea(name), "m^2"))

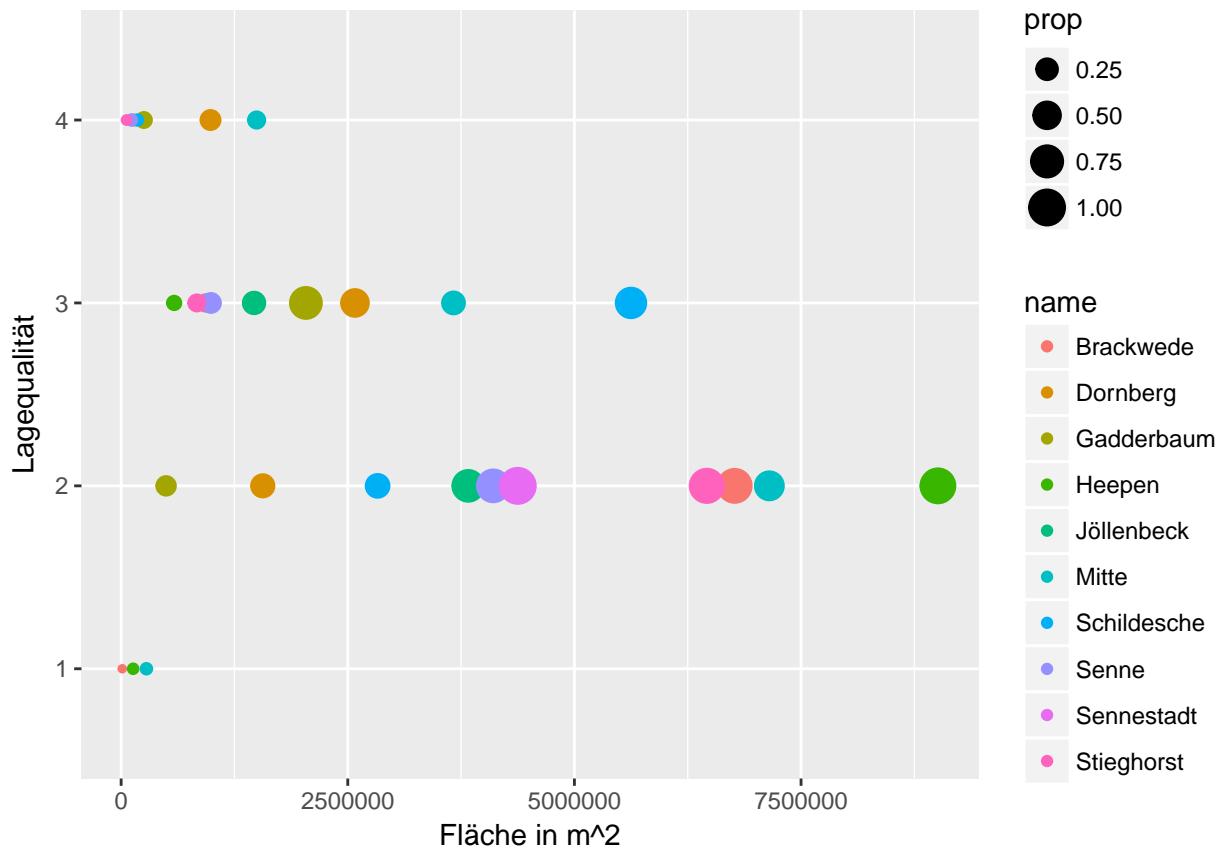
#Übersicht über die Wohngebiete nach Lagequalität je Stadtbezirk
#Die Spalte prop zeigt den Anzahl der Wohnfläche der jeweiligen Lagequalität an der Gesamtfläche des Bezirks
summary <- bzrk %>% #filter(name == 'Dornberg') %>%
  st_join(wk2) %>%
  group_by(name, lagequalit) %>% summarise(n = n(), area = sum(area.y), warea = mean(warea), prop = sum(
  as.data.frame() %>% select(name, lagequalit, n, area, warea, prop))

summary %>% select(name, lagequalit, prop) %>% spread(lagequalit, prop)

##          name      1      2      3      4
## 1    Brackwede 0.001597066 1 0.8625910 1 0.11847775 1 0.017334176 1
## 2    Dornberg      NA 1 0.3047000 1 0.50303473 1 0.192265316 1
## 3   Gadderbaum      NA 1 0.1780907 1 0.73179603 1 0.090113294 1
## 4     Heepen 0.013621755 1 0.9262608 1 0.06011747 1           NA 1
## 5   Jöllenbeck      NA 1 0.7230950 1 0.27690500 1           NA 1
## 6     Mitte 0.022169681 1 0.5679102 1 0.29121539 1 0.118704688 1
## 7 Schildesche      NA 1 0.3276798 1 0.65145063 1 0.020869586 1
## 8     Senne      NA 1 0.7880696 1 0.19026179 1 0.021668569 1
## 9  Sennestadt      NA 1 1.0000000 1           NA 1           NA 1
## 10 Stieghorst      NA 1 0.8783142 1 0.11345327 1 0.008232504 1
```

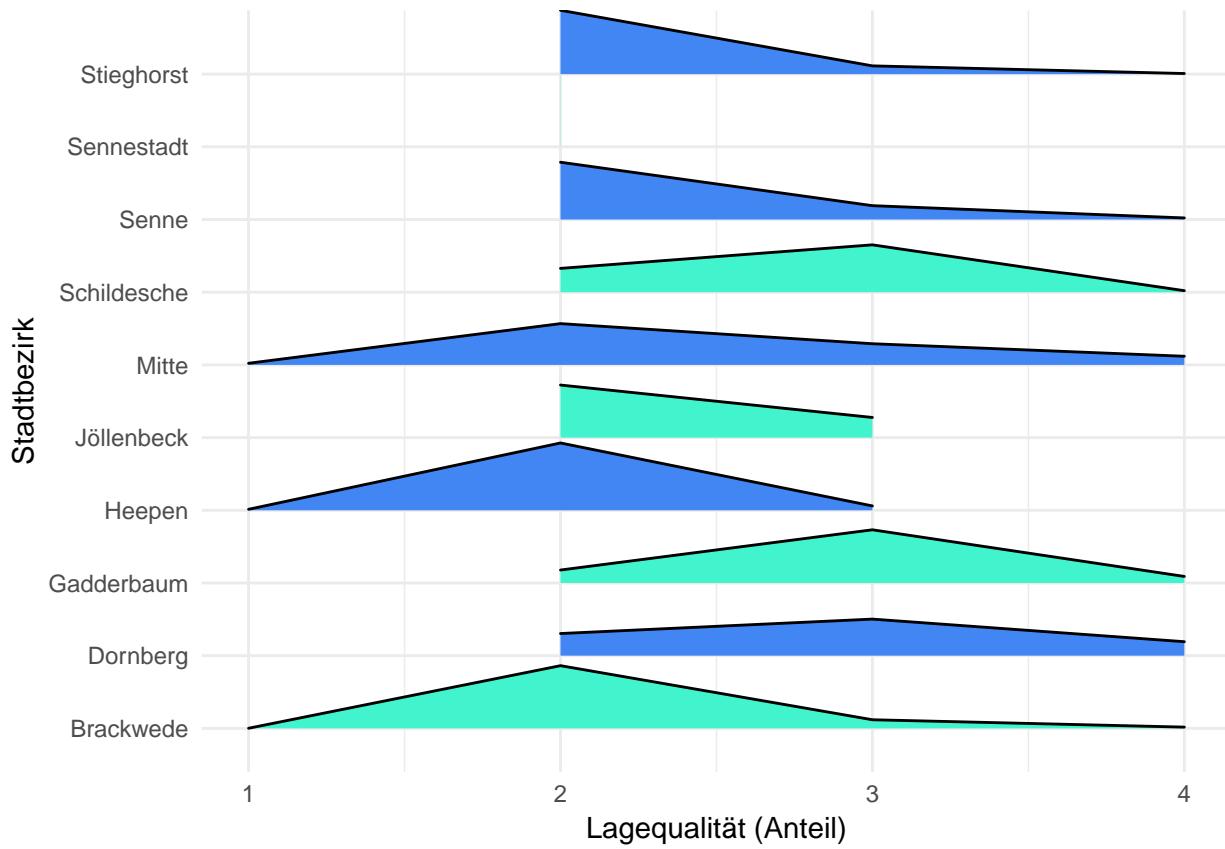
Grafisch aufbereitet ergibt sich folgendes Bild:

```
summary %>% mutate(area = as.numeric(area), prop = as.numeric(prop)) %>%
ggplot(aes(x = area, y = lagequalit)) + geom_point(aes(size = prop, color = name)) +
  labs(x = "Fläche in m^2", y = "Lagequalität")
```



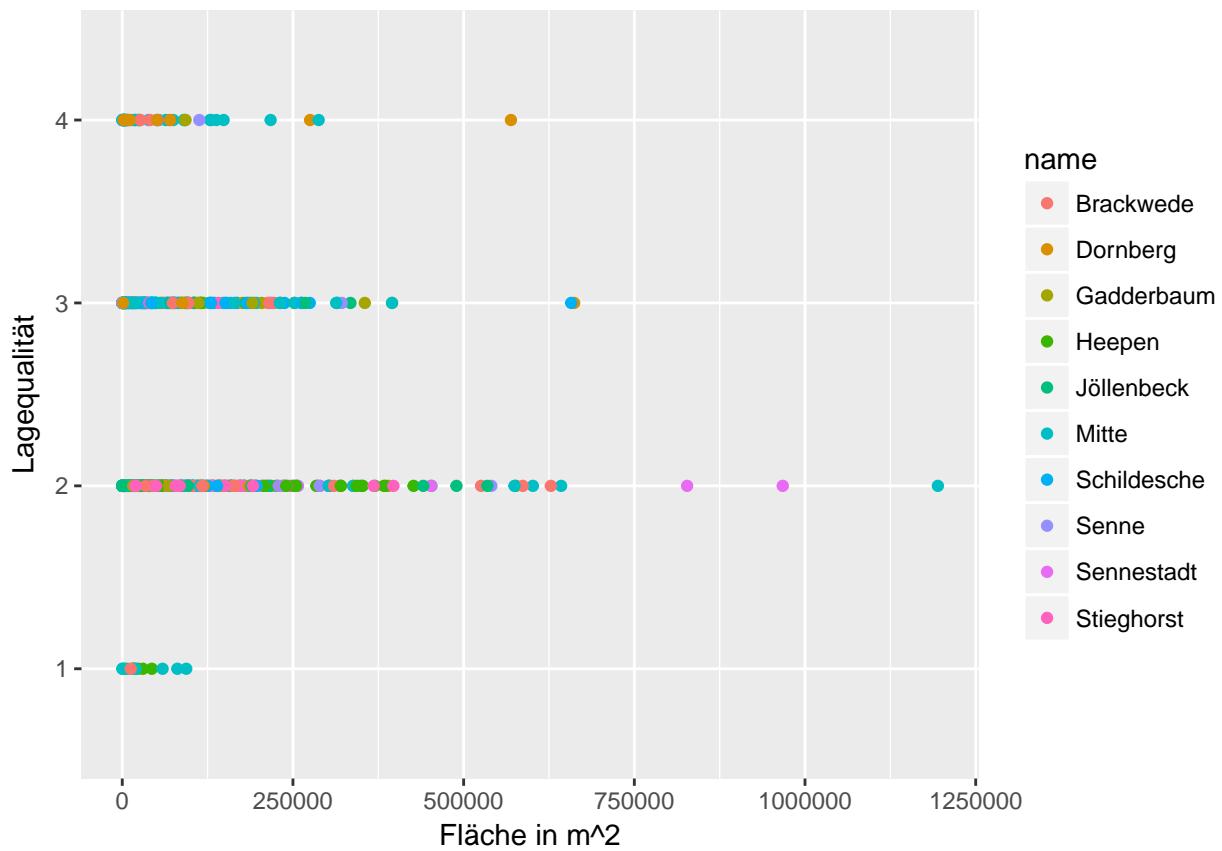
Eine visuell leichter zugängliche Darstellung dieser Zahlen ergibt sich vielleicht in einem gridplot:

```
summary %>% mutate(lagequalit = as.numeric(lagequalit), prop = as.numeric(prop)) %>%
ggplot(aes(x = lagequalit, y = name, height = prop, fill = name)) + geom_density_ridges(stat = "identity") +
  theme_minimal() + scale_fill_cyclical(values = c("#41f4cd", "#4286f4")) +
  labs(x = "Lagequalität (Anteil)", y = "Stadtbezirk")
```



Was macht eine hohe Lagequalität aus? Erkennt man Auffälligkeiten an der Größe der Gebiete? Gibt es interessante Ausreißer? Tatsächlich finden sich in den Bewertungsgruppen 2,3 und 4 Ausreißer, die sicher einen genaueren Blick wert sind.

```
st_join(wk2, bzrk) %>% mutate(area = as.numeric(area.x)) %>% select(area, lagequalit, name) %>% ggplot(
```



Quellen:

[1] Die Ausgangsdaten basieren auf der Wohnlagenkarte der Stadt Bielefeld, abgerufen von <https://open-data.bielefeld.de/dataset/wohnlagenkarte>, am 21.4.18. © Stadt Bielefeld, Amt für Geoinformation und Kataster (CC BY 4.0)