Bootcamp 8: Continuous color scales

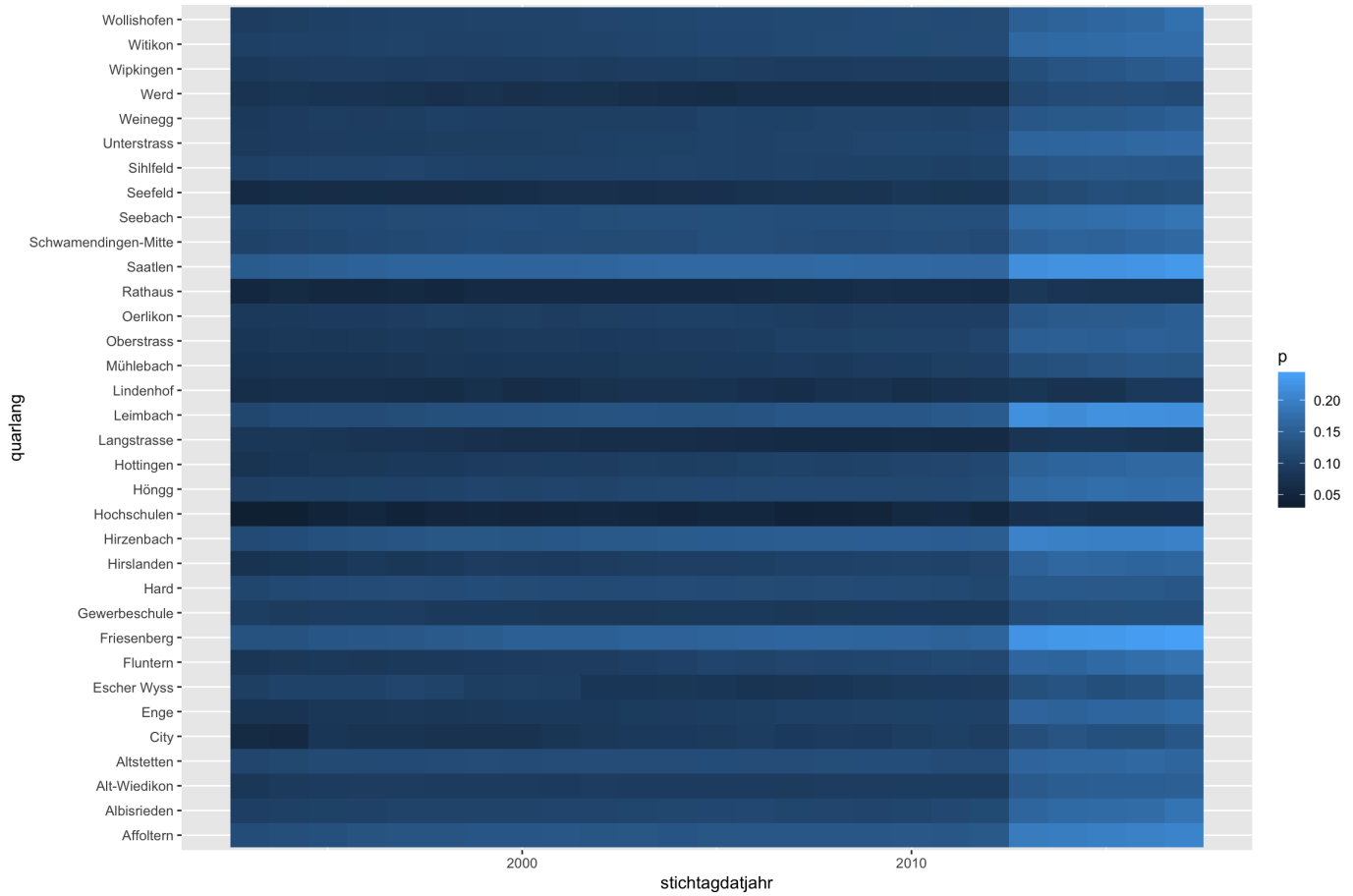
Lucien Baumgartner

11/6/2018


```
library(gplots)
library(ggplot2)
library(dplyr)
library(RColorBrewer)
library(viridis)
library(extrafont) # different fonts
library(scales)
library(reshape2)

rm(list=ls())

source('~/r-helpers/ggplot/ggplot-helper.R')

# set WD
setwd('~/ddj18/output/')

# load pop data
load('01-bevoelkerung-clean.RData')
```

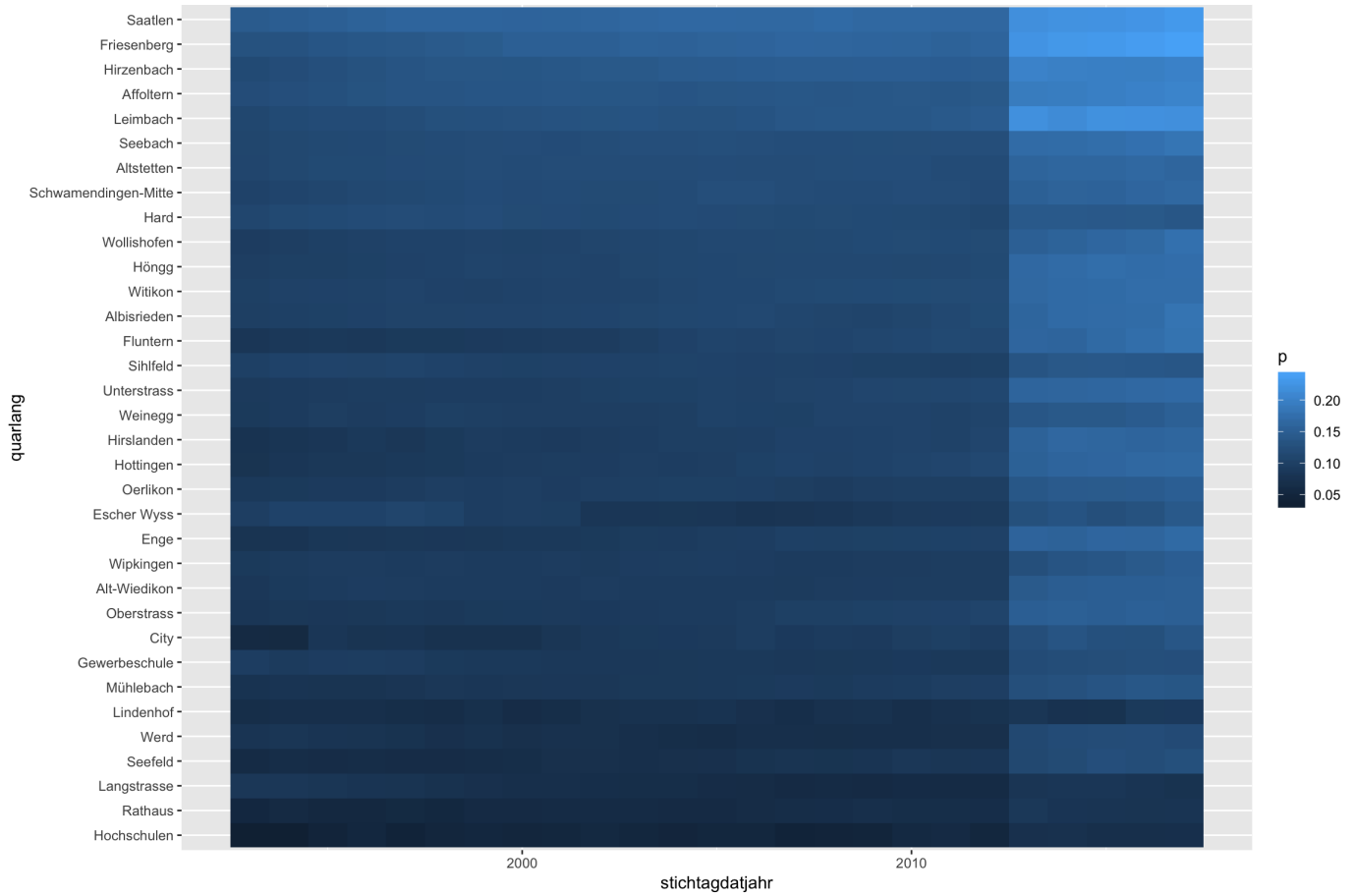
1 Baseline

```
# aggregate data
kids <- df %>% mutate(has.kids=ifelse(anzahlkinder>0,1,0)) %>%
  group_by(quarlang, stichtagdatjahr, has.kids) %>%
  summarise(n=n()) %>%
  mutate(p=n/sum(n)) %>%
  filter(has.kids==1)

# simple bar plot
ggplot(kids) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p))
```



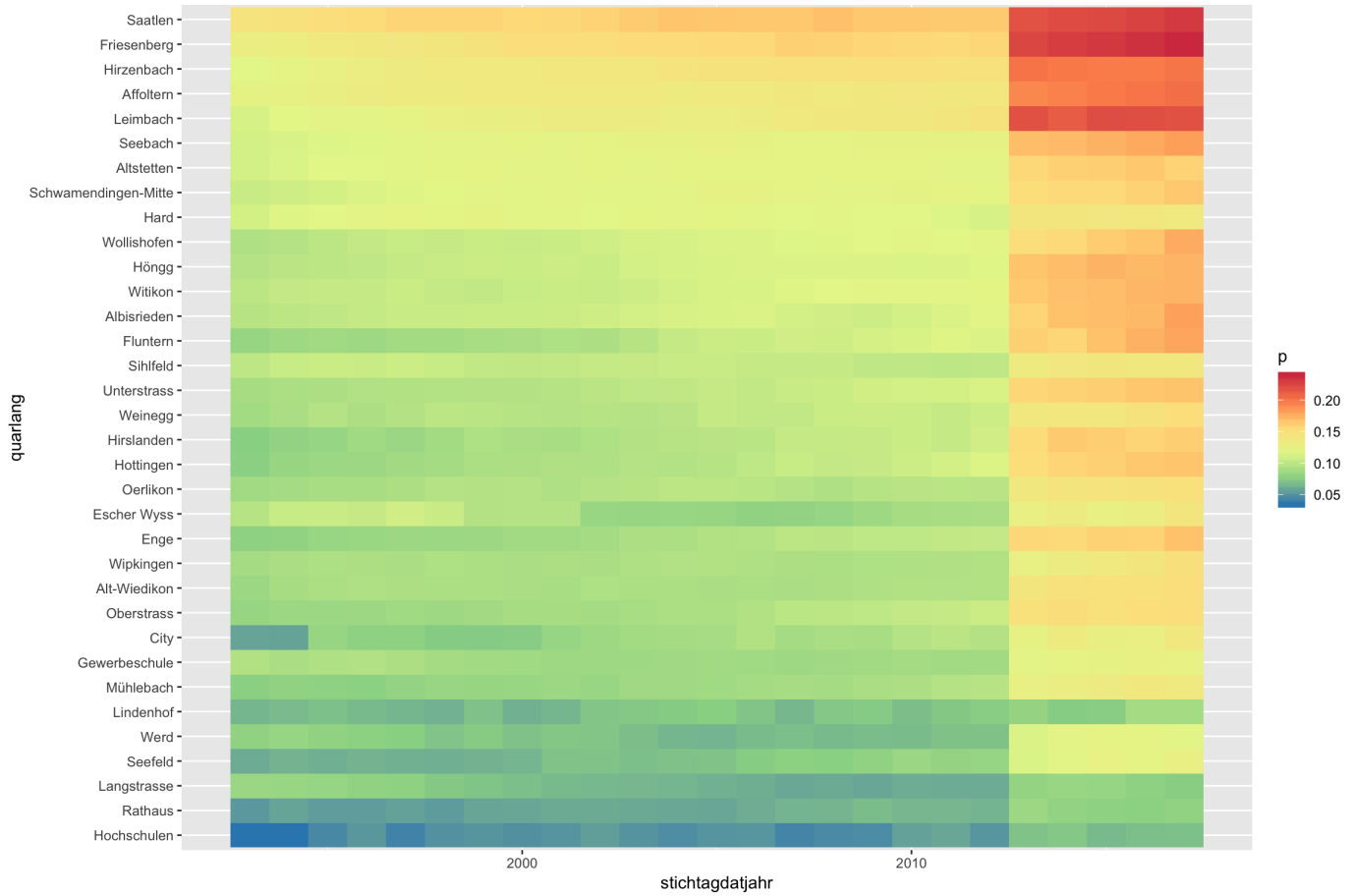
2 Sort levels

```
vec <- kids %>%
  ungroup %>%
  group_by(quarlang) %>%
  summarise(median.p=median(p))

kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)]
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p))
```



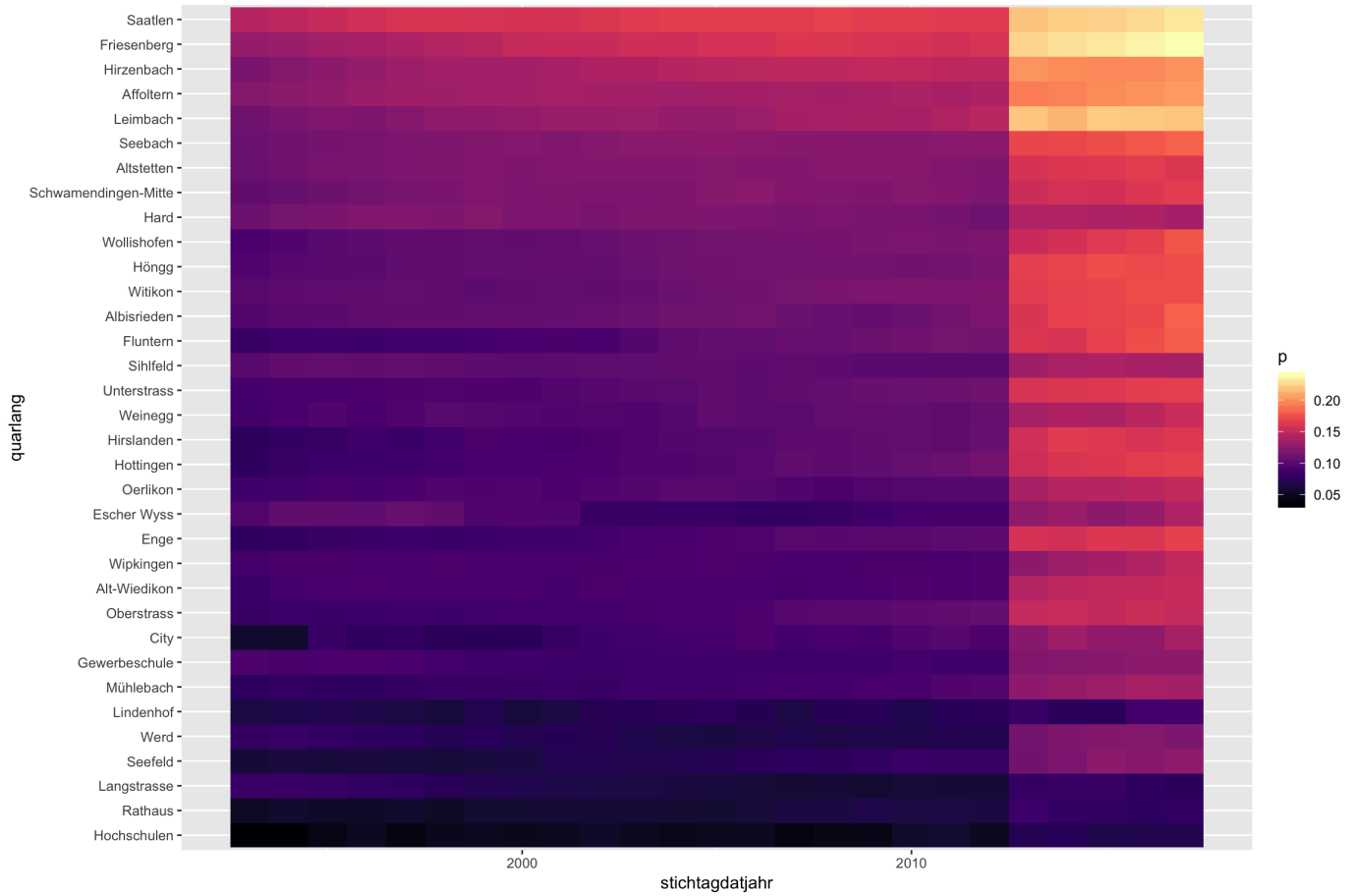
3 Continuous scales: brewer

```
kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)]
  ggplot(.) +
  scale_fill_distiller(palette='Spectral') +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p))
```



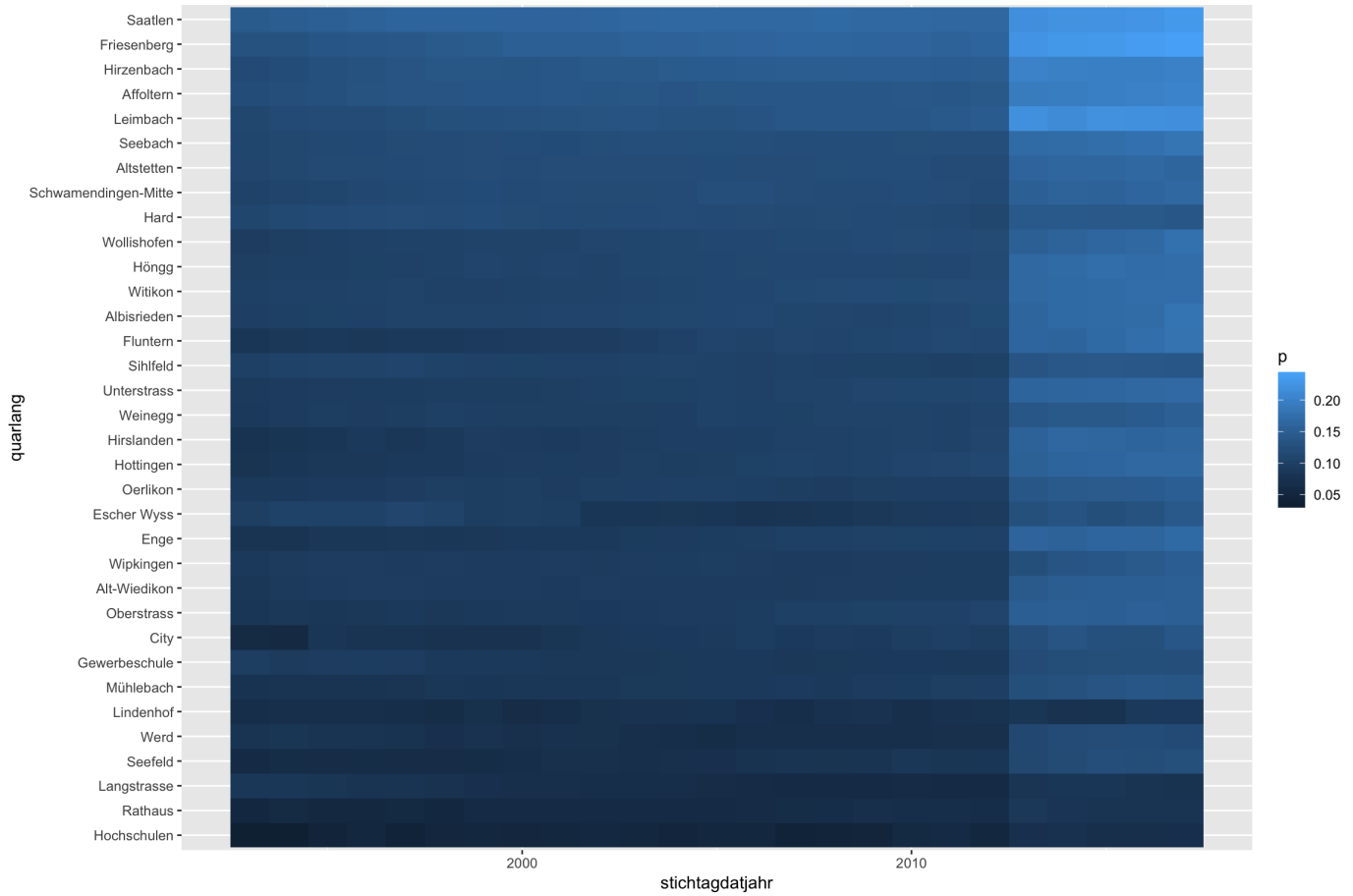
4 Continuous scales: viridis

```
kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)]
  ggplot(.) +
  scale_fill_viridis(option='A') +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p))
```

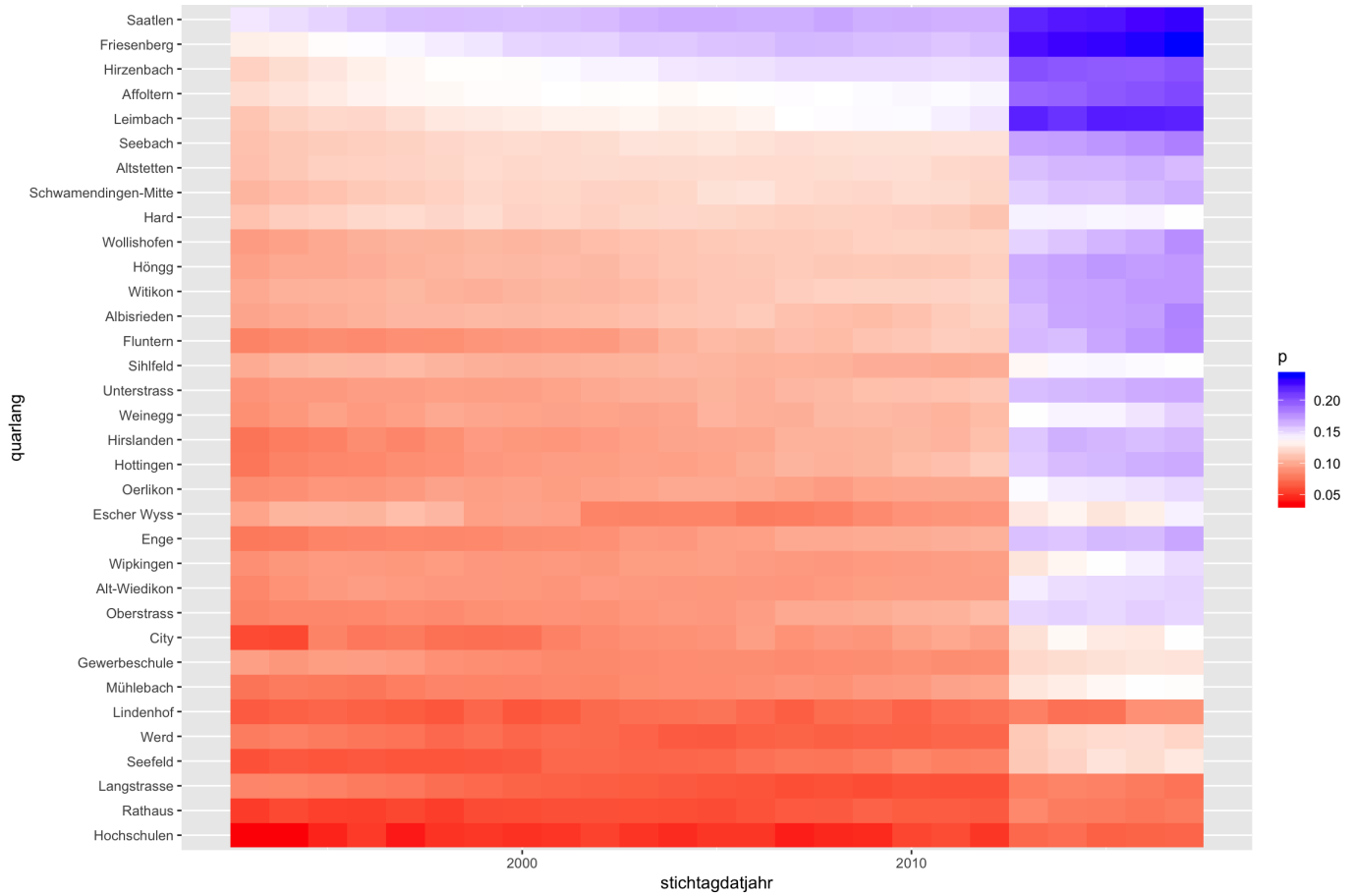


5 Continuous scales: scale_fill_gradient and scale_fill_gradient2

```
kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)])
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_gradient(low = "#132B43",
                      high = "#56B1F7",
                      na.value = "grey50")
```



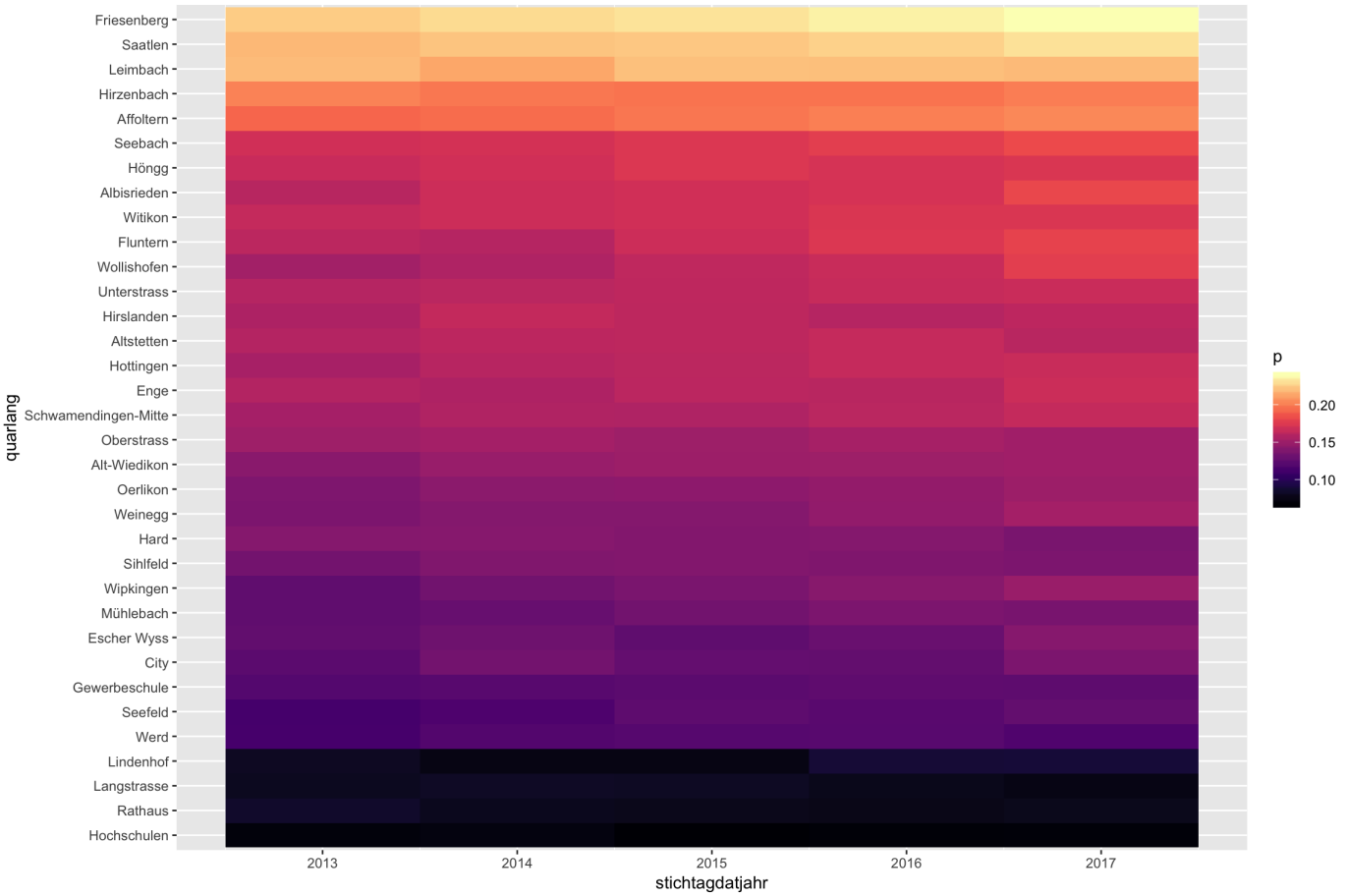

```
kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)])
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_gradient2(low = "red",
                       mid = "white",
                       high = "blue",
                       midpoint = mean(range(kids$p)),
                       na.value = "grey50")
```



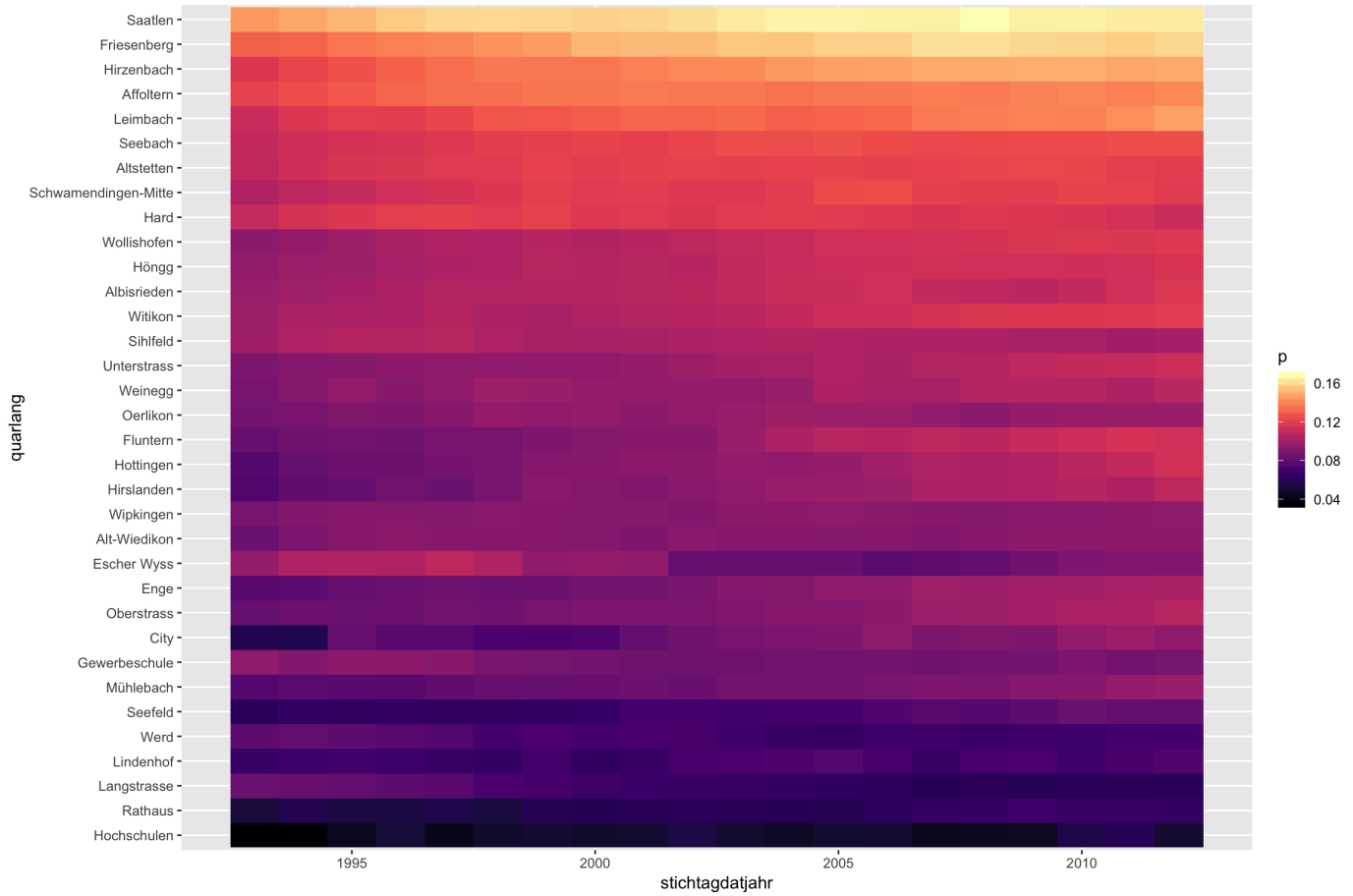
6 Tweakin: subsetting, rescaling, proper aggregation

6.1 subsetting

```
vec <- kids %>%
  ungroup %>%
  filter(stichtagdatjahr%in%2013:2018) %>%
  group_by(quarlang) %>%
  summarise(median.p=median(p))

kids %>%
  ungroup %>%
  filter(stichtagdatjahr%in%2013:2018) %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)]
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_viridis(option='A')
```

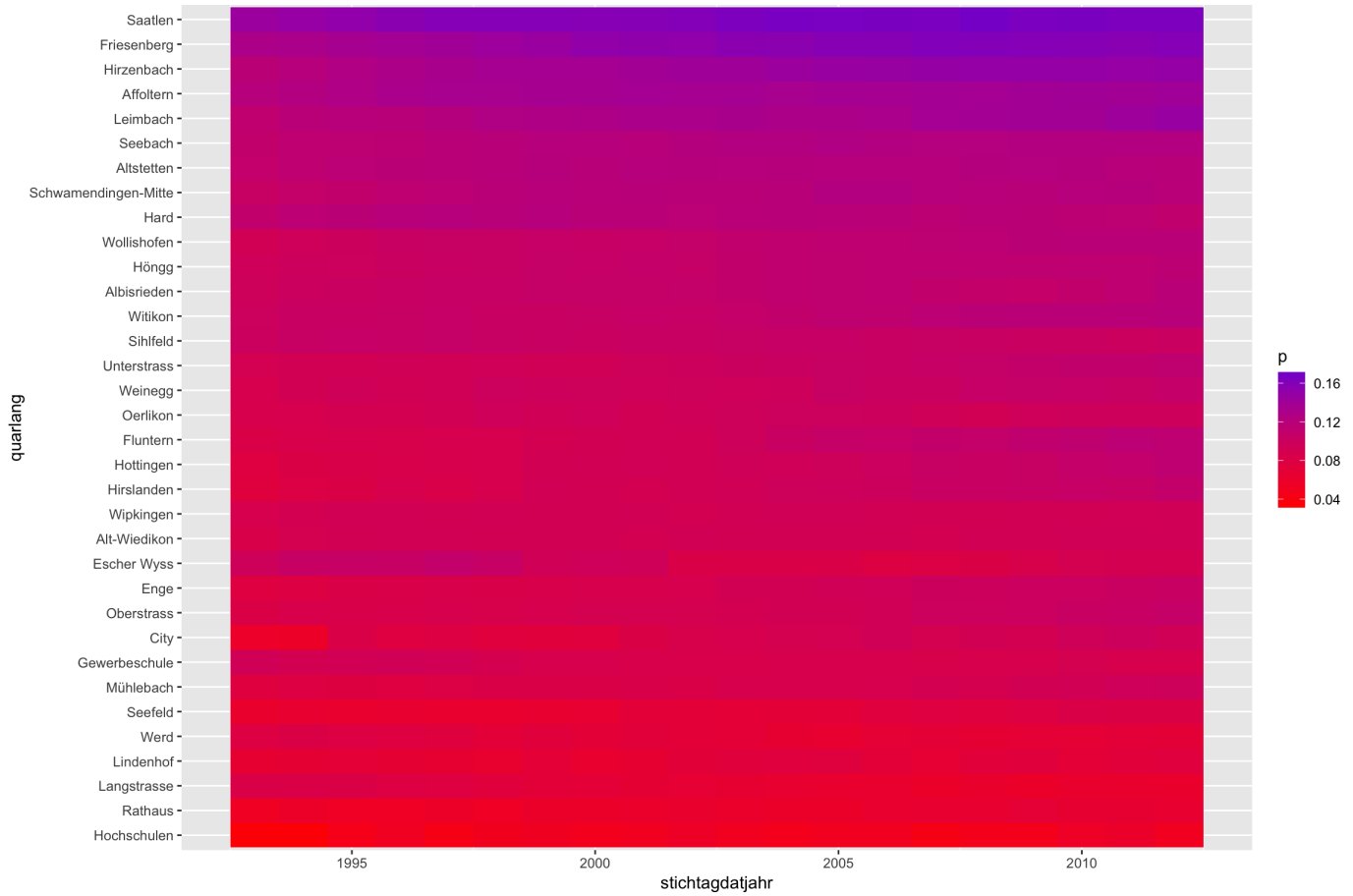



```
vec <- kids %>%
  ungroup %>%
  filter(!stichtagdatjahr%in%2013:2018) %>%
  group_by(quarlang) %>%
  summarise(median.p=median(p))

kids %>%
  ungroup %>%
  filter(!stichtagdatjahr%in%2013:2018) %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)])
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_viridis(option='A')
```

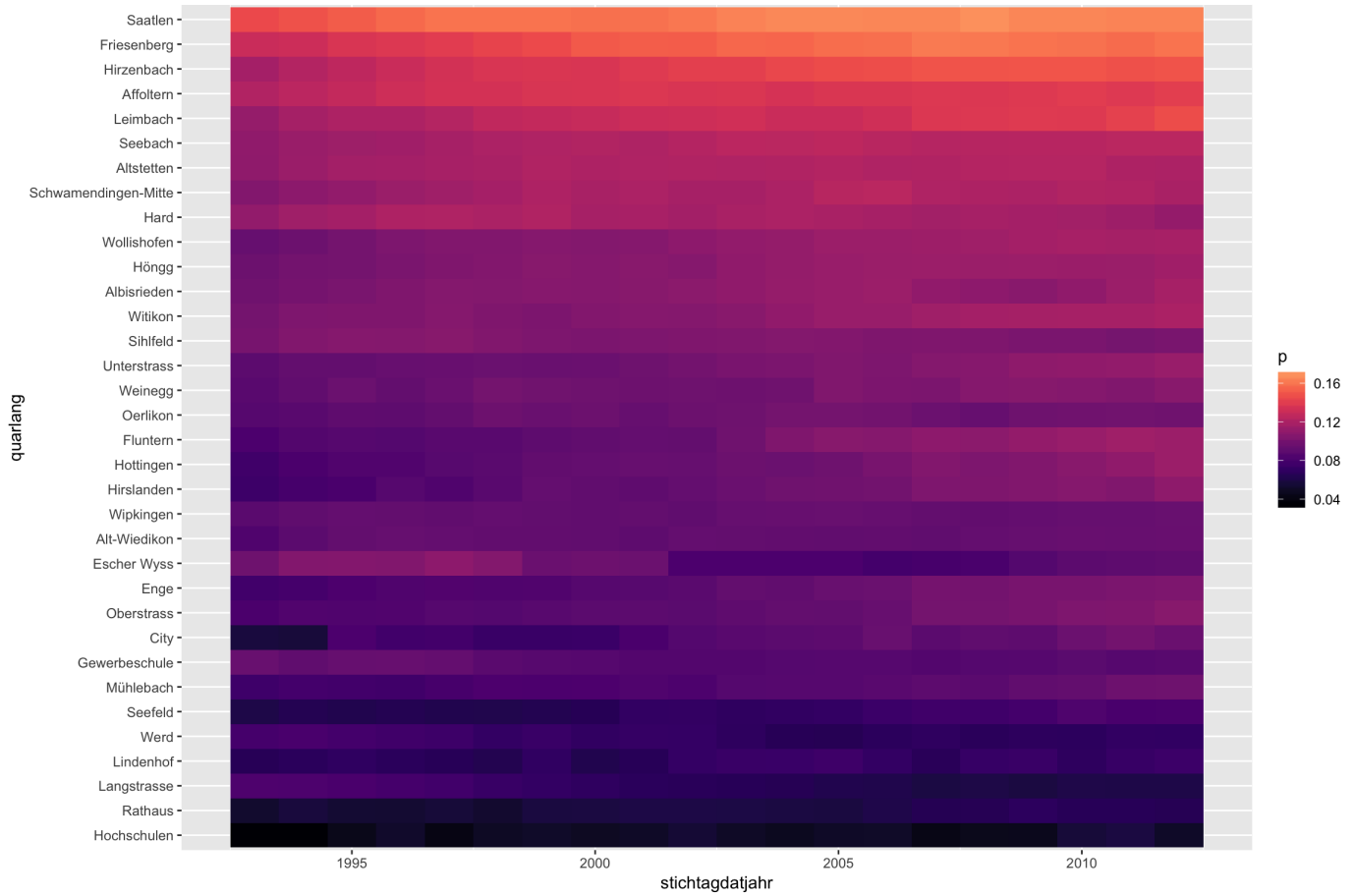


6.2 rescaling

```
kids %>%
  ungroup %>%
  filter(!stichtagdatjahr%in%2013:2018) %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)]
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_gradient(low = "red",
                      high = "blue",
                      na.value = "grey50",
                      rescaler = function(x, to = c(0, 1), from = NULL) {
    ifelse(x<0.2,
           scales::rescale(x,
                           to = to,
                           from = c(min(x, na.rm = TRUE), 0.2)),
           1)})
```




```
kids %>%
  ungroup %>%
  filter(!stichtagdatjahr%in%2013:2018) %>%
  mutate(quarlang=factor(quarlang, levels=vec$quarlang[order(vec$median.p)])
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_viridis(option='A',
                     rescaler = function(x, to = c(0, 1), from = NULL) {
    ifelse(x<0.2,
           scales::rescale(x,
                           to = to,
                           from = c(min(x, na.rm = TRUE), 0.2)),
           1)})
```



6.3 data-sensitive aggregation

```
lvls <- kids %>%
  ungroup %>%
  filter(!stichtagdatjahr%in%2013:2018) %>%
  mutate(group=ifelse(stichtagdatjahr>=round(mean(range(stichtagdatjahr)),0
  group_by(quarlang, group) %>%
  summarise(ht.median=median(p)) %>%
  ungroup %>%
  dcast(., quarlang ~ group, value.var = 'ht.median') %>%
  mutate(diff=second-first)
dim(kids)
```

```
## [1] 850   5
```

```
dim(lvls)
```

```
## [1] 34  4
```

```
kids %>%
  ungroup %>%
  mutate(quarlang=factor(quarlang, levels = lvls$quarlang[order(abs(lvls$di
  filter(!stichtagdatjahr%in%2013:2018) %>%
  ggplot(.) +
  geom_tile(aes(stichtagdatjahr, quarlang, fill=p)) +
  scale_fill_viridis(option='A',
                     rescaler = function(x, to = c(0, 1), from = NULL) {
    ifelse(x<0.15,
           scales::rescale(x,
                           to = to,
                           from = c(min(x, na.rm = TRUE), 0.15)),
           1)}) +
  geom_vline(aes(xintercept=round(mean(range(stichtagdatjahr)),0)))
```