REGION
The Journal of ERSA
Powered by WU

Volume 12, Number 2, 2025, 23–50
DOI: 10.18335/region.v12i2.566

journal homepage: region.ersa.org
ISSN: 2409-5370

Sequence Analysis of Neighborhood Racial and Ethnic Changes: The Case of New York City 1980-2020

Elizabeth Delmelle[1], Eric Delmelle[2]

[1] University of Pennsylvania
[2] Lehigh University and Vrije University Brussels

Received: October 2, 2024/Accepted: October 12, 2025

Abstract. This paper demonstrates the application of sequence analysis to develop a typology of racial and ethnic trajectories in New York City neighborhoods from 1980 to 2020 using a reproducible R workflow. Our workflow begins with using an unsupervised classification method, k-means, at each decennial cross-section to derive 6 classes describing the racial and ethnic makeup of neighborhoods during the study period. These classes include four that depict a majority of Black, White, Hispanic, and Asian residents, and two mixed-race classes, Black and Hispanic, and a White majority with a mixture of other races. We then develop a sequence of classes for each census tract over the 5 decennial time stamps. Finally, we derive a longitudinal typology describing the predominant pathways of change using sequence analysis. This resulted in 14 distinct pathways including transitions to Hispanic and Asian majorities emerging from historically White or Black neighborhoods. The findings underscore the gradual nature of neighborhood racial transformations. Our approach is reproducible for researchers wanting to explore and visualize multidimensional neighborhood dynamics.

1 Introduction

Tracking and understanding neighborhood changes has been a central topic of urban studies and a fundamental concern for planning practitioners (Galster 2001, Landis 2016, Chapple, Zuk 2016). Neighborhood change can be comprehended according to various dimensions, from housing stock or built environment changes to residents' demographic and socioeconomic composition (Delmelle 2022). Thus, the study of neighborhood change involves analyzing multiple attribute dimensions through time for spatially situated units. Recent scholarship has progressed in the analytical strategies used to study neighborhood trajectories, introducing new methods for visualizing and mapping longitudinal pathways of change for multiple dimensions (Delmelle 2022).

In this article, we demonstrate one such technique, sequence analysis, in an illustrative, reproducible tutorial analyzing neighborhood change. We demonstrate the method using a case study of racial and ethnic changes in New York City census tracts from 1980-2020. While our focus is methodological, the empirical case study highlights the utility of sequence analysis in visualizing, detecting, and exploring longitudinal patterns of neighborhood changes.

Since the 1980s, the United States' demographic profile has become increasingly diverse in the past several decades, driven by sustained immigration and by the growing share of births to non-White populations, influencing demographic shifts in the overall population

structure (Frey 2022). Analyses following the release of the 2010 decennial census showed that increasing national diversity resulted in differing neighborhood trajectories, largely contingent on the broader metropolitan context (Terbeck 2023, Wright et al. 2014).

Much of the empirical scholarship on neighborhood racial and ethnic changes has developed indices to categorize the makeup or diversity of racial and ethnic groups and then explored changes in a neighborhood's diversity categorization over time. For example, a neighborhood might transition from 'low diverse' to 'moderately diverse' from one decade to the next (Farrell, Lee 2011, Wright et al. 2014). Alternately, another set of techniques aims to describe the longitudinal sequences or trajectories that the racial and ethnic groups in a neighborhood have followed. Three methods have been adopted in the literature for this purpose: statistical curve fitting, time-series clustering, and sequence analysis. With curve-fitting models like growth mixture models or latent growth models, mathematical functions fit each racial and ethnic group under study that summarize the predominant trends (Zwiers et al. 2018, Hipp, Kim 2023). Time series clustering is an unsupervised classification technique aimed at clustering continuous longitudinal data (Delmelle et al. 2025). Finally,in sequence analysis, neighborhoods are first grouped into similar categorical clusters at each time stamp. Then, each a sequence of neighborhood categories is constructed over time. Finally, the sequences of these clusters are grouped using a sequence alignment technique (Delmelle 2016, González-Leonardo et al. 2023).

While all three approaches - curve fitting, time-series clustering, and sequence analysis aim to characterize change over time, they differ in what types of data and research questions they are best suited for. Curve fitting methods are well-suited for summarizing trends in continuous outcomes, especially when the trajectory is expected to follow a smooth or parametric form, which must be specified *a priori*. Time-series clustering also retains the continuous nature of longitudinal data, but must be performed on single variables at a time. Cross-sectional classes of trajectories can be constructed to form multivariate typologies (Delmelle et al. 2025), however, as the number of variables increases, this becomes an arduous workflow.

Sequence analysis is oriented towards categorical trajectories that can be constructed using multiple input variables, as demonstrated in this article. The method is particularly useful in visualizing and analyzing the timing, order, and transitions between qualitatively distinct neighborhood states rather than in the smoothness of changes.

In this article, we explore trajectories of neighborhood racial and ethnic changes in the largest and one of the most diverse cities in the United States, New York City, using longitudinal census data up to the latest 2020 decennial release. This notebook showcases a workflow that introduces sequence analysis to the study of multidimensional neighborhood changes. We begin by processing the raw longitudinal census data, then performing a k-means classification, and finally classify and map sequences clusters. Our case study illustrates the gradual progression that neighborhoods follow in when undergoing racial and ethnic transformations. We observe an overall decline in the share of neighborhoods categorized by a large White majority population, in exchange for increasing diversity, especially Hispanic and Asian populations.

2 Computational environment

The main libraries used in this paper include `dplyr` for processing tabular census data, `sf` for mapping the resulting clusters, `cluster` for performing the `k-means` cluster analysis. Finally, to perform the sequence analysis, we use `TraMineR` (Gabadinho et al. 2011). This is a popular and continually updated R package for performing sequence analysis for a host of social science applications, including analyses of neighborhood change (Delmelle 2016, 2017, Patias et al. 2020).

[1]:
```
# Set CRAN mirror
options(repos = c(CRAN = "https://cloud.r-project.org/"))

# For data management
if (!require('knitr')) install.packages('knitr'); library('knitr')
if (!require('rmarkdown')) install.packages('rmarkdown'); library('rmarkdown')
if (!require('dplyr')) install.packages('dplyr'); library('dplyr')
```

```
#for reading in data
if (!require('here')) install.packages('here'); library('here')

#for reading in census data
if (!require('tidycensus')) install.packages('tidycensus'); library('tidycensus')
if (!require('tigris')) install.packages('tigris'); library('tigris')
options(tigris_class = "sf") # returns data in sf format

#for data table formatting
if (!require('knitr')) install.packages('knitr'); library('knitr')

#for data table pivoting
if (!require('tidyverse')) install.packages('tidyverse'); library('tidyverse')

#mapping packages
if (!require('sp')) install.packages('sp'); library('sp')
if (!require('sf')) install.packages('sf'); library('sf')

#For data visualization
if (!require('ggplot2')) install.packages('ggplot2'); library('ggplot2')

#for facet plots
if (!require('gridExtra')) install.packages('gridExtra'); library('gridExtra')

#For sequence clustering
if (!require('TraMineR')) install.packages('TraMineR'); library('TraMineR')

# For k-means and hierarchical cluster analysis
if (!require('cluster')) install.packages('cluster'); library('cluster')

#For visualizing k-means outputs
if (!require('factoextra')) install.packages('factoextra'); library('factoextra')

# For creating heat map to describe k-means cluster results
if (!require('pheatmap')) install.packages('pheatmap'); library('pheatmap')

# For creating heat map to describe k-means cluster results
if (!require('RColorBrewer')) install.packages('RColorBrewer'); library('RColorBrewer')

# Other packages (cowplot:inset maps; extrafont for additional sans serif fonts)
if (!require('cowplot')) install.packages('cowplot'); library('cowplot')
if (!require('patchwork')) install.packages('patchwork'); library('patchwork')
if (!require('extrafont')) install.packages('extrafont'); library('extrafont')
loadfonts(device = "win")
if (!require('ggspatial')) install.packages('ggspatial'); library('ggspatial')
loadfonts(device = "win")
```

3 Data

We use decennial census tract data to examine neighborhood racial and ethnic changes. Census tracts serve as imperfect, yet well-used neighborhood proxies. Census tract boundaries change over time, further complicating the study of population dynamics within these boundaries. There are several sources of data that have been harmonized using interpolation techniques to a consistent set of boundaries over time. We use the Longitudinal Tract Database (LTDB) which uses areal and population interpolation techniques alongside ancillary data on water cover to derive estimates (Logan et al. 2014). Analyses of the errors produced by three popular longitudinal data providers suggest that LTDB performs similarly to the dataset produced by the National Historic Geographic Information System (NHGIS) and both perform better than the Neighborhood Change Database which relies solely on areal interpolation without the inclusion of ancillary data (Logan et al. 2014). Therefore, for this type of analysis either the LTDB or NHGIS would be suitable dataset for this analysis.

We obtained the full count decennial data from LTDB from 1980-2020 from the Diversity and Disparties project at Brown University. The census variables have been interpolated to 2010 tract boundaries. Because the coding of census race and ethnicity changes over time, we opted to begin in 1980 as 1970, the earliest dataset available, did

not record a count of Latino or Hispanic residents. The raw data contains all census tracts throughout the United States. For visualization purposes, we also import a shapefile of 2010 census tract boundaries using the `tidycensus` package and setting the geometry to `true`.

[2]:
```
setwd(here())  #current working directory

#csv tables for longitudinal data
census20<- read.csv("data/ltdb_std_2020_fullcount.csv")
census10<- read.csv("data/LTDB_Std_2010_fullcount.csv")
census00<- read.csv("data/LTDB_Std_2000_fullcount.csv")
census90<- read.csv("data/LTDB_Std_1990_fullcount.csv")
census80<- read.csv("data/LTDB_Std_1980_fullcount.csv")

#geometry data
#filter for NYC counties (5 boroughs)
nyc_counties <- c("Kings", "Queens", "New York", "Richmond", "Bronx")
tract <- get_decennial(geography = "tract",
                       variables = "P001001",
                       year = 2010,
                       state = "NY",
                       county = nyc_counties,
                       geometry = TRUE,
                       progress = FALSE)
```

We next calculate the share of `White`, `Black`, `Hispanic`, and `Asian` residents in each tract for each decade from the raw count using the total population as the denominator. We filter out tracts with no population and select only the relevant columns to create our data frame. We then join all columns from the five decennial data frames into one data frame called `census_all` and finally select only census tracts from the five counties that comprise New York City's five boroughs: Bronx County, Kings County (Brooklyn), New York County (Manhattan), Queens County, Richmond County (Staten Island).

[3]:
```
census80 <- census80 %>% filter (POP80 >0)
census80$perwhite80 <- census80$NHWHT80/census80$POP80
census80$perblack80 <- census80$NHBLK80/census80$POP80
census80$perhisp80 <- census80$HISP80/census80$POP80
census80$perasian80 <- census80$ASIAN80/census80$POP80
census80<- census80 %>% select(c("TRTID10","perwhite80", "perblack80", "perhisp80",
                                 "perasian80"))

census90 <- census90 %>% filter (POP90 >0)
census90$perwhite90 <- census90$NHWHT90/census90$POP90
census90$perblack90 <- census90$NHBLK90/census90$POP90
census90$perhisp90 <- census90$HISP90/census90$POP90
census90$perasian90 <- census90$ASIAN90/census90$POP90
census90<- census90 %>% select(c("TRTID10","state","county","perwhite90", "perblack90",
                                 "perhisp90", "perasian90"))

census00 <- census00 %>% filter (POP00 >0)
census00$perwhite00 <- census00$NHWHT00/census00$POP00
census00$perblack00 <- census00$NHBLK00/census00$POP00
census00$perhisp00 <- census00$HISP00/census00$POP00
census00$perasian00 <- census00$ASIAN00/census00$POP00
census00<- census00 %>% select(c("TRTID10","perwhite00", "perblack00", "perhisp00",
                                 "perasian00"))

census10 <- census10 %>% rename("TRTID10" = "tractid")
census10$perwhite10 <- census10$nhwht10/census10$pop10
census10$perblack10 <- census10$nhblk10/census10$pop10
census10$perhisp10 <- census10$hisp10/census10$pop10
census10$perasian10 <- census10$asian10/census10$pop10
census10<- census10 %>% select(c("TRTID10","perwhite10", "perblack10", "perhisp10",
                                 "perasian10"))

census20 <- census20 %>% rename("TRTID10" = "TRTID2010")
census20$perwhite20 <- census20$nhwt20/census20$pop20
census20$perblack20 <- census20$nhblk20/census20$pop20
census20$perhisp20 <- census20$hisp20/census20$pop20
```

```
census20$perasian20 <- census20$asian20/census20$pop20
census20<- census20 %>% select(c("TRTID10","perwhite20", "perblack20", "perhisp20",
                                 "perasian20"))

#join all data frames from each decade
census_all<- census90 %>%
  left_join(census00) %>%
  left_join(., census10) %>%
  left_join(., census20)%>%
  left_join(., census80)

#Select NYC Counties. These include Bronx County, Kings County (Brooklyn),
#New York County (Manhattan), Queens County, Richmond County (Staten Island)

census_select <- census_all %>% filter((state == "NY" & county == "Bronx County")|
                               (state == "NY" & county == "Kings County")|
                               (state == "NY" & county == "New York County")|
                               (state == "NY" & county == "Queens County")|
                               (state == "NY" & county == "Richmond County"))

##remove NA values and state and county columns
census_nyc <- na.omit(census_select)%>% select(-state, -county)

#Select only the tractID column from the shapefile. Rename the field for ease of joining
#and convert to double to match the csv data.
tract<- tract %>% select("GEOID")
tract<- rename(tract, TRTID10 = GEOID)
tract$TRTID10<- as.double(tract$TRTID10)
```

[4]:
```
# Set tigris options to return the data in sf format (spatial data frame)
options(tigris_class = "sf")

# Download county boundaries for New York state
ny_counties <- counties(state = "NY", cb = TRUE,
                        progress = FALSE)

# Filter for only the New York City counties
nyc_counties <- ny_counties %>%
  filter(NAME %in% c("Bronx", "Kings", "New York", "Queens", "Richmond"))

# View the structure of the data
print(nyc_counties)
```

[4]:
```
Simple feature collection with 5 features and 12 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: -74.25563 ymin: 40.4961 xmax: -73.70036 ymax: 40.91771
Geodetic CRS:  NAD83
  STATEFP COUNTYFP COUNTYNS        GEOIDFQ GEOID     NAME        NAMELSAD
1      36      005 00974101 0500000US36005 36005    Bronx    Bronx County
2      36      047 00974122 0500000US36047 36047    Kings    Kings County
3      36      061 00974129 0500000US36061 36061 New York New York County
4      36      081 00974139 0500000US36081 36081   Queens   Queens County
5      36      085 00974141 0500000US36085 36085 Richmond Richmond County
  STUSPS STATE_NAME LSAD     ALAND    AWATER                       geometry
1     NY   New York   06 109235672  39353304 MULTIPOLYGON (((-73.77242 4...
2     NY   New York   06 179684481  71158757 MULTIPOLYGON (((-74.04171 4...
3     NY   New York   06  58683879  29010416 MULTIPOLYGON (((-74.00641 4...
4     NY   New York   06 281594051 188444349 MULTIPOLYGON (((-73.96262 4...
5     NY   New York   06 148982679 117441532 MULTIPOLYGON (((-74.16154 4...
```

4 Basic conceptual intuition

4.1 Categorizing Neighborhoods Using k*-means*

Sequence analysis requires categorical input states. Given that neighborhood demographic and socioeconomic data is very often continuous, the first step is to transform the data into discrete categories for each year of the analysis. In this example, we opt to use a data-driven, unsupervised classification approach for this stage by applying k-means to cluster the racial and ethnic composition of each census tract in each decade into one of

several mutually exclusive categories. This step serves as a preprocessing stage, not an analysis of changes itself.

As an alternative, for the case of racial and ethnic compositions, for example, pre-specified thresholds (e.g., >50% of a population group) could be used to define categories like "majority Black" or "majority Hispanic". This approach works well in situations where clear theoretical cutoffs exist. However, other applications involving socio-economic indicators or multivariate neighborhood characteristics may not lend themselves to clear analyst-defined cutoffs, making a data-driven approach preferable. Since the purpose of this article is illustrative, we proceed with an overview of implementing k-means to construct the initial discrete classes of neighborhoods from the original racial and ethnic makeup data. However, if the analyst already has theoretically grounded thresholds, they can bypass this step and proceed directly to the sequence analysis.

In the context of neighborhood racial and ethnic classification, Reibel, Regelson (2011) introduced the idea of using an unsupervised classification approach for studying neighborhood racial change as an alternative to the use of neighborhood diversity indices. The objective of the k-means algorithm is to group observations in such a way that maximizes the similarity of observations within groups or clusters while maximizing the dissimilarity between each cluster. In other words, the goal is to group neighborhoods so that those most similar to each other according to their racial and ethnic makeup are assigned to the same cluster and the clusters themselves are distinct from one another in terms of their makeup.

With the k-means algorithm, the number of clusters, k must be determined *a priori*. To make this determination, it is customary to evaluate multiple solutions using various fit statistics in conjunction with domain and geographic knowledge of the data (Delmelle 2015). It is also commonly recommended that input variables first be normalized to avoid placing unequal emphasis on variables that may be on different measurement scales. However, in our case study, all of the racial and ethnic variables represent percentages of the population and so this step is not performed in the case study.

Our ultimate goal is to understand the major pathways of neighborhood change and so each neighborhood will be classified five times for 1980, 1990, 2000, 2010, and 2020 to establish its longitudinal sequence. To ensure that the clusters are temporally stable, we will perform the clustering for all years at once. New neighborhood typologies may emerge over time with this approach. In that instance, only tracts from the later years would be assigned to the new cluster.

4.2 Identifying pathways of change using sequence analysis

Like k-means, sequence analysis is an unsupervised classification technique but instead of grouping observations based on cross-sectional similarity, it clusters them based on the similarity of longitudinal categorical sequences. A central component of sequence analysis is defining how similar or dissimilar two sequences are. There are multiple methods to compute sequence dissimilarity. Studer, Ritschard (2016) provides a comprehensive overview of techniques for determining sequences dissimilarity for social science applications.

In this case study, our goal is to group neighborhoods that follow similar racial and ethnic change trajectories, specifically in which order of categories traverse through. This gives us an overview of the various pathways of change neighborhoods may take. For example sequences that progress from a majority White composition to Mixed Race and eventually to majority Hispanic may represent one frequent change pathway. Because our sequences are measured at fixed intervals (every decade from 1980 to 2020), and each neighborhood has the same number of observations, we are less concerned with irregular timing or duration, which are often central considerations in life-course studies.

To prioritize this kind of ordered similarity, we use OMstrans, a variation of the popular Optimal Matching (OM) algorithm (Delmelle 2017). OM treats sequences as strings and calculates the 'edit distance', or minimal cost to transform one sequence into another using insertions, deletions, and substitutions. In OMstrans, the algorithm operations not on the raw states, but on the transitions between states by merging eaech state in a sequence with its predecessor. This approach emphasizes how one state leads into another and helps preserve the sequencing of change.

For example, one neighborhood may have a sequence of *White*, *White*, *Mixed Race*, *Hispanic* while another neighborhood might follow the sequence *White*, *White*, *Mixed Race*, *Mixed Race*. Between these two strings, there is one entry that differs: the final state. To transform one to another, we could substitute *Hispanic* for *Mixed Race* in the second sequence, and the dissimilarity of the two sequences would be equal to the cost of that substitution.

In the OMstrans variant, distances between sequences of transitions are computed. This means that each state is merged with its previous state to create a subsequence. In the previous illustrative example, the first sequence becomes (*White-White*, *White-MixedRace*, *MixedRace-Hispanic*) and the OM cost evaluation is then applied to these subsequences.

The OMstrans approach allows us to incorporate empirical transition probabilities into the substitution cost matrix. In this way, frequent transitions (e.g., *White* to *Mixed*) are penalized less than rare ones. Insertions and deletions can be discoraged by assigning a higher cost than the empirical transitions so that disruptions to the temporal alignment of sequences is discouraged. The balance between preserving state similarity and order is further governed by the parameter w, the origin-transition parameter. When $w = 1$, OMstrans approximates the traditional OM algorithm. A lower value places greater emphasis on the ordering or sequencing of events than on the specific states themselves.

5 Application

5.1 Study Area

Our study area consists of census tracts within the five Boroughs of New York (see Figure 1 for an overview map).

[5]:
```
# Filter for only the New York City counties
nyc_counties <- ny_counties %>%
  filter(NAME %in% c("Bronx", "Kings", "New York", "Queens", "Richmond"))

# Modify labels for specific counties
nyc_counties$label_main <- ifelse(nyc_counties$NAME == "New York", "New York",
                                  ifelse(nyc_counties$NAME == "Kings", "Kings",
                                         ifelse(nyc_counties$NAME == "Richmond",
                                                "Richmond", nyc_counties$NAME)))

nyc_counties$label_paren <- ifelse(nyc_counties$NAME == "New York", "(Manhattan)",
                                   ifelse(nyc_counties$NAME == "Kings", "(Brooklyn)",
                                          ifelse(nyc_counties$NAME == "Richmond",
                                                 "(Staten Island)", "")))

# Calculate centroids for each county to get coordinates for labels
centroids <- st_centroid(nyc_counties)
coords <- st_coordinates(centroids)

# Add the coordinates to the nyc_counties data frame
nyc_counties <- nyc_counties %>%
  mutate(X = coords[, 1],
         Y = coords[, 2])

# Adjust coordinates manually for specific counties
nyc_counties <- nyc_counties %>%
  mutate(
    X = ifelse(NAME == "New York", X - 0.11,              # Move "New York" left,
               ifelse(NAME == "Richmond", X + 0.012, X)), # Richmond right
    Y = ifelse(NAME == "New York", Y + 0.005,
               ifelse(NAME == "Richmond", Y + 0.03, Y)) # Adjust Richmond label higher
  )

#you can mask water area if you would like.
#nyc_counties <- nyc_counties %>%  erase_water(area_threshold = 0.9)

# Create the large map for the state without a scale bar
zoomed_map <- ggplot() +
  geom_sf(data = ny_counties, fill = "gray85", color = "white", size = 1) +
```

```
            # New York State outline
  geom_sf(data = nyc_counties, fill = "#85B0A9", color = "white", size = 3) +
            # NYC counties in teal
  geom_segment(aes(x = -73.98, y = 40.775, xend = -74.03, yend = 40.775),
            color = "gray25", size = 0.5) +
            # Line from New York label to the boundary
  geom_text(data = nyc_counties, aes(x = X, y = Y, label = label_main),
            color = "black", size = 4, fontface = "bold") +
            # Add main county names
  geom_text(data = nyc_counties, aes(x = X, y = Y - 0.02, label = label_paren),
            color = "gray25", size = 3.75) +
            # Add parentheses labels in lighter color
  geom_text(aes(x = -73.6, y = 40.725, label = "Nassau"),
            color = "gray55", size = 4, fontface = "bold") +
            # Corrected Nassau label position
  geom_text(aes(x = -73.8, y = 40.98, label = "Westchester"),
            color = "gray55", size = 4, fontface = "bold") +
            # Corrected Westchester label
  geom_text(aes(x = -73.6, y = 40.705, label = "(Long Island)"),
            color = "gray70", size = 3.5, fontface = "bold") +
            # Add Long Island label below Nassau
  coord_sf(xlim = c(-74.25, -73.5), ylim = c(40.4, 41.05), expand = FALSE) +
            # Crop to focus on NYC
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(), # Remove major grid lines
    panel.grid.minor = element_blank(), # Remove minor grid lines
    axis.title = element_blank(), # Remove axis titles
    axis.text = element_blank(), # Remove axis text (labels)
    legend.position = "none", # Remove the legend
    plot.margin = margin(0, 0, 0, 0) # Remove margins for a tighter crop
  )

# Create the large map for NY and surrounding counties with text annotation
inset_map <- ggplot() +
  geom_sf(data = ny_counties, fill = "gray90", color = NA, size = 1) +
            # All counties in gray, including Nassau and Westchester
  geom_sf(data = nyc_counties, fill = "#569289", color = "#569289", size = 5) +
            # NYC counties in darker teal
  annotate("text", x = -76.2, y = 42.8, label = "New York State",
           size = 5.25, color = "gray49", angle = 0,
           alpha = 0.7) + # Add text label
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(), # Remove major grid lines
    panel.grid.minor = element_blank(), # Remove minor grid lines
    plot.title = element_text(hjust = 0), # Align title to the left
    plot.subtitle = element_text(hjust = 0), # Align subtitle to the left
    axis.title = element_blank(), # Remove axis titles
    axis.text = element_blank(), # Remove axis text (labels)
    plot.margin = margin(0, 0, 0, 0) # Tighten margins
  )

# Combine the maps, placing the inset map within the zoomed-in map
final_plot <- ggdraw() +
  draw_plot(zoomed_map) +
  draw_plot(inset_map, x = 0.07, y = 0.65, width = 0.3, height = 0.3)
            # Adjust position and size of the inset

# Display the combined plot
print(final_plot)
```

[5]:
```
Output in Figure 1
```

To begin our case study, we start by preparing the data for the k-means clustering. This involves pivoting the data frame so that each census tract is represented with five distinct rows, once for each decennial value. To do so, we pivot from a wide to a long format and select out just the four race and ethnicity values to be used in the clustering.

Figure 1: Overview map of New York City's five boroughs and surrounding counties

[6]:
```
# Convert the data frame from wide to long format
census_long <- census_nyc %>%
  pivot_longer(cols = starts_with("per"),
               names_to = c(".value", "year"),
               names_pattern = "per(\\w+)(\\d{2})") %>%
  mutate(year = case_when(
    year == "80" ~ 1980,
    year == "90" ~ 1990,
    year == "00" ~ 2000,
    year == "10" ~ 2010,
    year == "20" ~ 2020,
    TRUE ~ as.integer(year)
  ))

data_for_clustering <- census_long %>%
  select(white, black, hisp, asian)
```

As explained above, the k-means clustering procedure requires that the number of clusters, k be specified *a priori*. There are fit statistics that attempt to determine the optimal number of clusters, considering the similarity of observations within each cluster and the distinctiveness of the clusters from each other. However, these mathematically-derived solutions are devoid of any contextual or theoretical understanding of the problem under study. Therefore, the selection of k often becomes more akin to art than a science (Von Luxburg et al. 2012), considering the objective of the study. We begin with two common data-driven approaches that explore multiple clustering solutions for different k values and then examines the *within sum of squares* (WSS) and the *average silhouette score* for each solution. The WSS assesses how compact a clustering solution is, or how homogeneous the observations assigned to each cluster are, and the average silhouette score measures how well separated each cluster is from each other. Our objective is to derive a typology of neighborhoods according to their racial and ethnic makeup, for four groups: percent White, Black, Hispanic, and Asian. Figures 2 and 3 show the results of the clustering analysis; the Elbow method plot (Figure 2) suggests the optimal number of clusters, while the Silhouette method plot (Figure 3) helps validate the cluster separation.

[7]:
```
# Now do the k-means clustering on all

data_for_clustering <- census_long %>%
  select(white, black, hisp, asian)

# Function to calculate total within-cluster sum of squares for different k
wss <- function(k) {
  kmeans(data_for_clustering, k, nstart = 10)$tot.withinss
}
```

```
# Compute and plot wss for k = 1 to k = 10
k.values <- 1:10
wss_values <- map_dbl(k.values, wss)

# Elbow method plot
plot(k.values, wss_values, type = "b", pch = 19, frame = FALSE,
     xlab = "Number of clusters K",
     ylab = "Total within-clusters sum of squares")
```

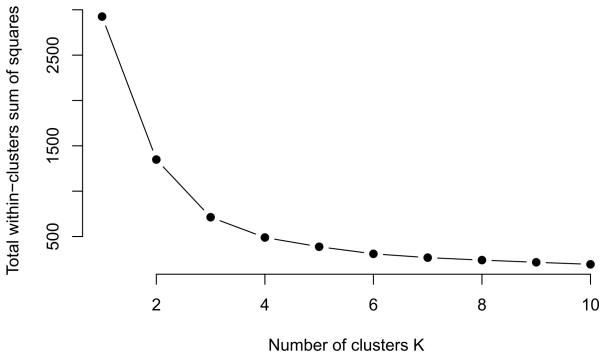

Figure 2: Elbow method plot – Clustering analysis

[7]:
```
Output in Figure 2
```

[8]:
```
# Silhouette method for determining the optimal number of clusters
fviz_nbclust(data_for_clustering, kmeans, method = "silhouette")
```

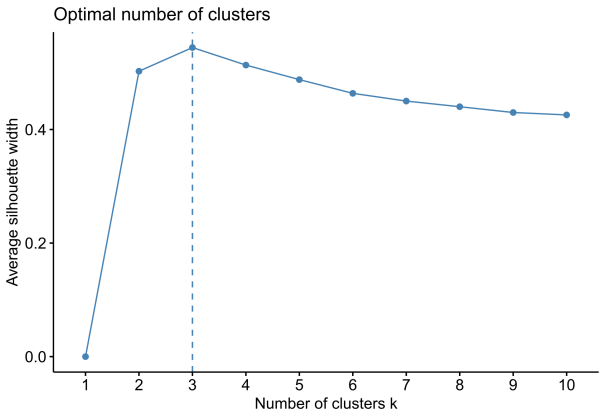

Figure 3: Silhouette method plot – Clustering analysis

[8]:
```
Output in Figure 3
```

5.2 Exploring k*-means Cluster Solutions*

According to these plots, the mathematically optimal number of neighborhood clusters for racial makeup is three. We can further explore the makeup of neighborhoods within these three clusters a few ways to determine if, in fact, three clusters provides a meaningful segmentation of four distinct racial and ethnic groups. In the plots below, we can visualize the average silhouette value for each cluster. The *Silhouette* values range from -1 to 1; values close to 1 suggest that the observations are well clustered while negative values suggest that an observation might be assigned to the wrong cluster. From the plot, Clusters 2 and 3 appears to be the most cohesive clusters, with average silhouette widths of 0.64, while cluster 1 has some potentially poorly classified neighborhoods. Descriptions of the racial and ethnic makeup of the clusters are obtained from the associated stacked

Table 1: Cluster Profiles: Average Demographics

Cluster	White	Black	Hispanic	Asian
1	0.15	0.18	0.50	0.15
2	0.07	0.75	0.14	0.02
3	0.73	0.04	0.12	0.10

bar charts. We can see that Cluster 1 is characterized as being nearly 50 percent Hispanic, with near equal shares of Whites, Blacks, and Asians. Cluster 2 is majority Black with approximately 15 percent Hispanics and few Whites and Asians. Finally, Cluster 3 is majority White. Therefore, this segmentation provides us with clusters indicating the dominant racial groups, but may miss some nuances of other racial and ethnic neighborhood compositions. We can therefore explore how increasing k may portray a richer portrait of neighborhood demographic profiles. Figure 4 illustrates the Silhouette Analysis to assess cluster separation, and the demographic makeup of the three clusters.

[9]:
```r
# Assume the optimal number of clusters (k) is 3 from the previous steps
set.seed(123)
kmeans_result <- kmeans(data_for_clustering, centers = 3, nstart = 25)

# Add the cluster assignments to the original data
census_long$cluster <- kmeans_result$cluster

# Silhouette Analysis
sil <- silhouette(kmeans_result$cluster, dist(data_for_clustering))
fviz_silhouette(sil, print.summary=FALSE)
```

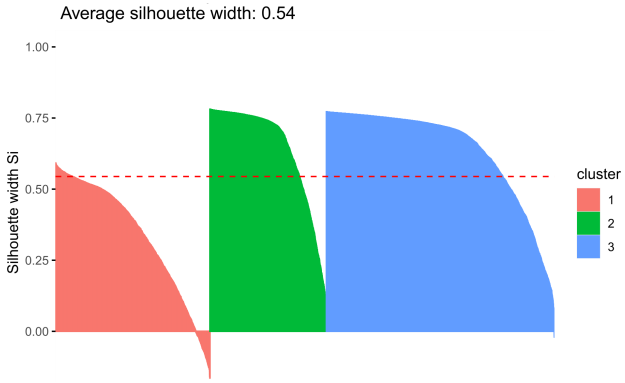

Figure 4: Cluster silhouette plot

[9]:
```
Output in Figure 4
```

[10]:
```r
# Extract silhouette information to a data frame (df)
sil_df <- as.data.frame(sil[, 1:3])
colnames(sil_df) <- c("Cluster", "Silhouette Width", "Neighboring Cluster")

# Cluster profiles
cluster_profiles <- census_long %>%
  group_by(cluster) %>%
  summarise(across(c(white, black, hisp, asian), ~ round(mean(.), 2)))

# Print the table as a formatted table
kable(cluster_profiles, caption = "Cluster Profiles: Average Demographics",
      col.names = c("Cluster", "White", "Black", "Hispanic", "Asian"),
      format = "markdown")
```

[10]:
```
Output in Table 1
```

[11]:
```
# Reshape the data from wide to long format
cluster_profiles_long <- cluster_profiles %>%
  pivot_longer(cols = c("white", "black", "hisp", "asian"),
               names_to = "Demographic", values_to = "Proportion")

# Create the stacked bar chart
ggplot(cluster_profiles_long, aes(x = factor(cluster), y = Proportion,
                                  fill = Demographic)) +
  geom_bar(stat = "identity") +
  labs(title = "Cluster Demographic Makeup - 3 clusters", x = "Cluster",
       y = "Proportion") +
  scale_fill_brewer(palette = "Set3") +  # Adjust color palette if desired
  theme_minimal()
```

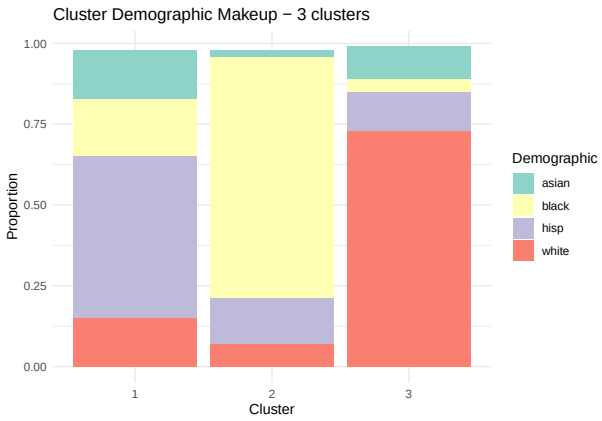

Figure 5: Cluster Analysis Results: Silhouette Analysis and Cluster Demographic Makeup

[11]:
```
Output in Figure 5
```

Next, we compare 4, 5, and 6 cluster solutions. We can see that the average silhouette of the solutions declines as the number of clusters increases. The demographic profiles show several new neighborhood typologies emerge with more clusters added. With a four cluster solution, we observe neighborhood typologies for each of the three dominant racial and ethic groups: White, Hispanic, and Black along with one mixed neighborhood type, shown in cluster 4. As expected, that cluster displays the lowest silhouette value, with some potentially mis-classified neighborhoods. The 5 cluster solution adds a cluster showing a majority Asian population, alongside two majority White populations - one showing more diversity than the other), and a majority Black, and Hispanic group. Finally, a 6 cluster solution shows more racially mixed groups, but at the expense of less well-defined or separated clusters. In the code below, each variable name ends with a number (4, 5 or 6), indicating which number of clusters (k) it represents. The results displays the silhouette analysis results for clustering into (a) four clusters, (b) five clusters, and (c) six clusters, illustrating the cohesion and separation of clusters at different sizes.

[12]:
```
# Perform k-means clustering for 4, 5, and 6 clusters
set.seed(123)
kmeans_4 <- kmeans(data_for_clustering, centers = 4, nstart = 25)
kmeans_5 <- kmeans(data_for_clustering, centers = 5, nstart = 25)
kmeans_6 <- kmeans(data_for_clustering, centers = 6, nstart = 25)

# Add cluster assignment to original data
census_long$cluster_4 <- kmeans_4$cluster
census_long$cluster_5 <- kmeans_5$cluster
census_long$cluster_6 <- kmeans_6$cluster

# Silhouette analysis for 4, 5, and 6 clusters
sil_4 <- silhouette(kmeans_4$cluster, dist(data_for_clustering))
sil_5 <- silhouette(kmeans_5$cluster, dist(data_for_clustering))
sil_6 <- silhouette(kmeans_6$cluster, dist(data_for_clustering))
```

```
# Plot silhouette for 4 clusters
plot_4 <- fviz_silhouette(sil_4, print.summary=FALSE) +
  ggtitle("Silhouette Plot - Four Clusters")

# Plot silhouette for 5 clusters
plot_5 <- fviz_silhouette(sil_5, print.summary=FALSE) +
  ggtitle("Silhouette Plot - Five Clusters")

# Plot silhouette for 6 clusters
plot_6 <- fviz_silhouette(sil_6, print.summary=FALSE) +
  ggtitle("Silhouette Plot - Six Clusters")

# Combine the three plots into a faceted view
grid.arrange(plot_4, plot_5, plot_6, ncol = 1)
```

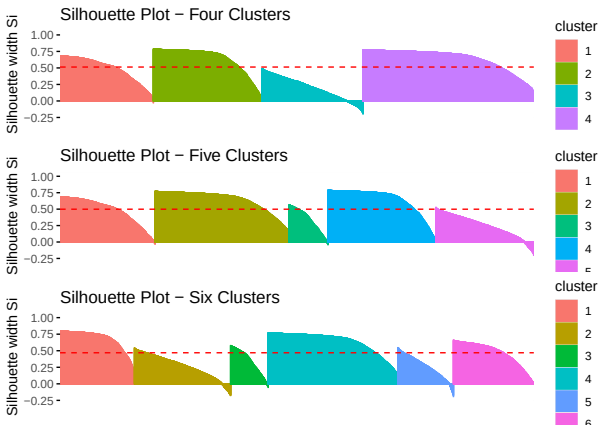

Figure 6: Silhouette Analysis for Different Cluster Sizes: (top) Four Clusters, (middle) Five Clusters, and (bottom) Six Clusters.

[12]:
```
Output in Figure 6
```

Figure 6 illustrates the demographic composition of clusters for different clustering solutions: (top) four clusters, (middle) five clusters, and (bottom) six clusters, highlighting how demographic groups are distributed across various cluster labels.

[13]:
```
# We need to hard code the labels to be able to compare the demographic profile
# of each solution. This is because R randomly assigns the label each time,
# even if the clustering solution is the same because we set the seed.

label_clusters <- function(cluster_profiles) {
  cluster_profiles %>%
    mutate(
      label = case_when(
        white > 0.75 ~ "3 White",
        hisp > 0.45 ~ "2 Hispanic",
        black > 0.75 ~ "1 Black",
        asian > 0.45 ~ "5 Asian",
        white < 0.56 & hisp > 0.20 & black < 0.20 ~ "4 Mixed",
        black > 0.49 & hisp > 0.25 & white < 0.15 ~ "6 Black and Hispanic",
        TRUE ~ "Other"  # Default label if none of the conditions are met
      )
    )
}

# Create cluster profiles and label them according to your custom rules
cluster_profiles_4 <- census_long %>%
  group_by(cluster_4) %>%
  summarise(across(c(white, black, hisp, asian), ~ round(mean(.), 2))) %>%
  mutate(`Clustering_Solution` = "4 Clusters", Cluster = cluster_4) %>%
  label_clusters()

cluster_profiles_5 <- census_long %>%
```

```
  group_by(cluster_5) %>%
  summarise(across(c(white, black, hisp, asian), ~ round(mean(.), 2))) %>%
  mutate(`Clustering_Solution` = "5 Clusters", Cluster = cluster_5) %>%
  label_clusters()

cluster_profiles_6 <- census_long %>%
  group_by(cluster_6) %>%
  summarise(across(c(white, black, hisp, asian), ~ round(mean(.), 2))) %>%
  mutate(`Clustering_Solution` = "6 Clusters", Cluster = cluster_6) %>%
  label_clusters()

# Combine all cluster profiles into one table
cluster_profiles_combined <-
  bind_rows(cluster_profiles_4, cluster_profiles_5, cluster_profiles_6)

# Map the labels back to the original tracts by joining based on cluster assignments
census_long <- census_long %>%
  left_join(cluster_profiles_4 %>% select(Cluster, label) %>% rename(label_4 = label),
            by = c("cluster_4" = "Cluster")) %>%
  left_join(cluster_profiles_5 %>% select(Cluster, label) %>% rename(label_5 = label),
            by = c("cluster_5" = "Cluster")) %>%
  left_join(cluster_profiles_6 %>% select(Cluster, label) %>% rename(label_6 = label),
            by = c("cluster_6" = "Cluster"))

# Reshape data for plotting
cluster_profiles_long <- cluster_profiles_combined %>%
  pivot_longer(cols = c("white", "black", "hisp", "asian"),
               names_to = "Demographic", values_to = "Proportion")

# Plot stacked bar charts of demographic makeup for each labeled cluster
ggplot(cluster_profiles_long, aes(x = label, y = Proportion, fill = Demographic)) +
  geom_bar(stat = "identity", position = "stack") +
  facet_grid(Clustering_Solution ~ ., scales = "free_x") +
      # Facet vertically by clustering solution
  scale_y_continuous(labels = scales::percent_format(accuracy = 1)) +
  labs(
    title = "",
    x = "Cluster Label",
    y = "Proportion of Demographic Group",
    fill = "Demographic Group"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )
```

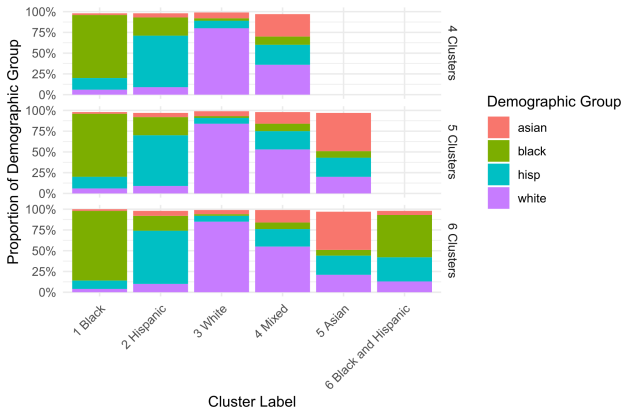

Figure 7: Geographic variation of Labeled Clusters for Different Clustering Solutions.

[13]:
```
Output in Figure 7
```

We can explore how this plays out for a specific observation. For example, take the first census tract in the data frame for the year 2020. This tract's racial composition

was 58% Black and 30% Hispanic with small shares of Whites and Asians. With the four cluster solution, this tract was classified into class 1, `Majority Black`. For the five cluster solution, it was also classified as `Majority Black`, and for the six cluster solution, `Black and Hispanic`. In 1990, that same tract was 34% black and 61% Hispanic, resulting in the tract being classified as `Majority Hispanic` by all three clustering solutions. When looking at change over time, the neighborhood will either be registered as transitioning from `Majority Hispanic` to either `Majority Black` or `Black and Hispanic`, depending on the final cluster solution. In this instance, the 6 cluster solution provides a more accurate portrayal of the dynamics - while the neighborhood did technically become majority Black, there is still a significant Hispanic presence, a detail that would have been omitted by limiting the number of groups.

Finally, we can examine the spatial distribution of these clustering solutions to help aid in the final determination for the sequence analysis. Since we did the cluster analysis on all decades, for this purpose, we will contrast the 4, 5, and 6 cluster solutions just for 2020. All three maps depict a similar spatial pattern, but with a more fragmented pattern in the case of the 6 cluster solution. For example, the first two maps show contiguous tracts of the predominantly Black cluster, but the third map shows the emergence of the `Black and Hispanic` group largely forming on the outskirts of this spatial cluster. Another apparent distinction is the large spatial cluster of the `Asian` cluster in Queens, which had been labeled as `Mixed` in the 4 cluster solution.

To better understand details on racial neighborhood transitions, we will go with the larger number of clusters, 6, despite the mathematical preference for a 3 cluster solution. Our result illustrates the demographic distribution of labeled clusters for different clustering solutions: four clusters, five clusters, and six clusters.

[14]:
```r
# Filter data for the year 2020 and merge labels with tract data for mapping
tract_clusters_2020 <- tract %>% erase_water(area_threshold = 0.9) %>%
  left_join(census_long %>%
      filter(year == 2020) %>%  # Filter for the year 2020
      select(TRTID10, cluster_4, cluster_5, cluster_6, label_4, label_5, label_6),
            by = "TRTID10") %>%
  pivot_longer(cols = c("cluster_4", "cluster_5", "cluster_6"),
      names_to = "ClusterSolution", values_to = "Cluster") %>%
  pivot_longer(cols = c("label_4", "label_5", "label_6"),
      names_to = "LabelSolution", values_to = "Label") %>%
  filter(str_replace(ClusterSolution, "cluster_", "") == str_replace(LabelSolution,
                                                                     "label_", "")) %>%
  mutate(Label = factor(Label)) %>%
  mutate(ClusterSolution = recode(ClusterSolution,
                                  "cluster_4" = "4 Clusters",
                                  "cluster_5" = "5 Clusters",
                                  "cluster_6" = "6 Clusters")) %>%
  mutate(ClusterSolution = factor(ClusterSolution, levels = c("4 Clusters", "5 Clusters",
                                                              "6 Clusters")))

# Create the base plot with the labeled clusters
base_plot <- ggplot(tract_clusters_2020) +
  geom_sf(aes(fill = Label), color = NA) +  # Use the Label field for fill
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add county outlines
  facet_wrap(~ ClusterSolution, nrow = 1) +  # Facet horizontally by clustering solution
  scale_fill_manual(
    values = c("3 White" = "#F0E442", "2 Hispanic" = "#D55E00",
               "1 Black" = "#0072B2", "5 Asian" = "#CC79A7",
               "4 Mixed" = "#009E73", "6 Black and Hispanic" = "#56B4E9"),
    labels = c("Black", "Hispanic", "White", "Asian", "Mixed", "Black and Hispanic")
      # Remove numbers from labels
  ) +
  labs(x = "", y = "", fill = NULL) +
    # Remove the title and set fill to NULL to remove "Label"
  theme_void() +
  theme(
    strip.text = element_text(hjust = .5, vjust = .1, face = "italic", size = 12),
      # Center facet labels
    legend.position = "bottom",  # Place the legend at the bottom
    legend.title = element_blank(),  # Ensure legend title is blank
```

```
    legend.text = element_text(size = 8)
  ) +
  guides(fill = guide_legend(nrow = 1, byrow = TRUE, label.position = "bottom"))
    # Place category names under the boxes

# Display the plot
grid.arrange(base_plot)
```

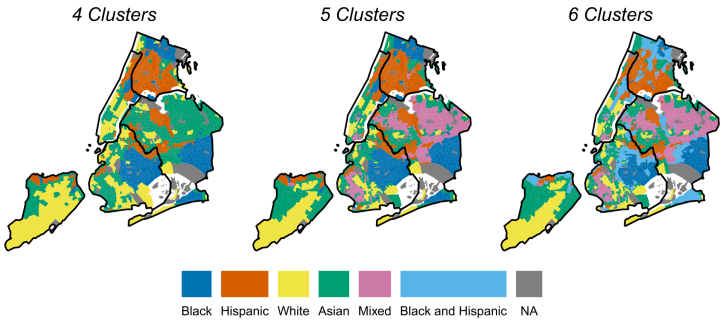

Figure 8: Demographic Distribution of Labeled Clusters for Different Clustering Solutions: Four Clusters (top), Five Clusters (middle), and Six Clusters (bottom).

[14]:
```
Output in Figure 8
```

We can begin with a simple exploration of the spatial changes over time in the maps showing the six clusters from 1980-2020 in a series of small multiples. From these maps, we can pick out some general spatial patterns over time. For example, we can see the expanse of neighborhoods classified as majority White from 1980 significantly diminishes by 2020. The share of Hispanics is shown to increase over time in the northern sections of the City. Towards the East, we see neighborhoods generally transition from predominantly White in 1980 to White and Mixed Race and eventually to Asian by 2020. There are also some evident stable clusters. For instance, the two clusters of predominantly black neighborhoods in the South and Southeast appear quite stable over time. However, the cluster of majority Black neighborhoods in towards the north of Manhattan is diminished, replaced by a Black and Hispanic classification.

[15]:
```
decades <- c(1980, 1990, 2000, 2010, 2020)

# Filter and prepare data for mapping
tract_clusters_decades <- tract %>%
  left_join(census_long %>%
              select(TRTID10, year, label_6), by = "TRTID10") %>%
  filter(!is.na(label_6))

# Set factor levels for year to ensure proper ordering
tract_clusters_decades <- tract_clusters_decades %>%
  mutate(year = factor(year, levels = decades))

# Plot faceted maps for each decade (year)
ggplot(tract_clusters_decades) +
  geom_sf(aes(fill = label_6), color = NA) +
    # Use factor to ensure proper color assignment
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add county outlines
  facet_wrap(~ year, ncol = 5) +  # Facet by year with 5 columns
  scale_fill_manual(
    values = c("3 White" = "#F0E442", "2 Hispanic" = "#D55E00", "1 Black" = "#0072B2",
               "5 Asian" = "#CC79A7", "4 Mixed" = "#009E73",
               "6 Black and Hispanic" = "#56B4E9"),
    labels = c("Black", "Hispanic", "White", "Asian", "Mixed", "Black and Hispanic"),
      # Remove numbers from labels
    name = ""  # Remove legend title
  ) +
  labs(title = " ", x = "", y = "") +
  theme_void() +
```

```
  theme(
    plot.title = element_text(hjust = 0, color = "gray"),  # Set the title color to gray
    legend.position = "bottom",  # Move legend to the bottom
    legend.direction = "horizontal",  # Make the legend horizontal
    legend.title = element_blank(),  # Ensure the legend title is blank
    legend.key.size = unit(0.6, "cm"),  # Set consistent size for legend boxes
    legend.key.height = unit(0.6, "cm"),  # Set consistent height for legend boxes
    legend.key.width = unit(0.5, "cm")  # Set consistent width for legend boxes
  ) +
  guides(
    fill = guide_legend(
      nrow = 1, byrow = TRUE, label.position = "bottom"
        # Place category names under the legend boxes
    )
  )
```

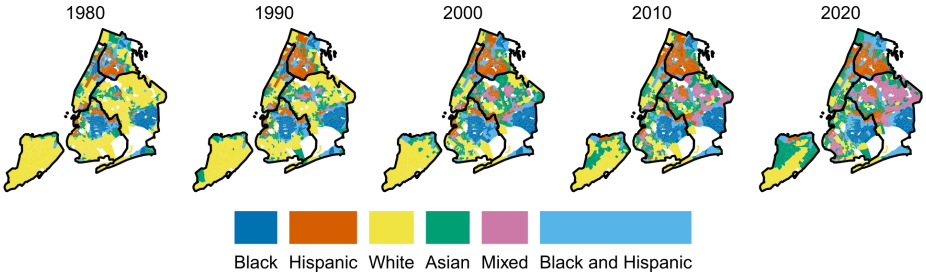

Figure 9: Change in the cluster membership over five decades

[15]:
```
Output in Figure 9
```

5.3 *Sequence Analysis*

Our objective with the sequence classification is to come up with a typology of neighborhood sequences over time to describe general pathways of change. Each neighborhood has a sequence of classes over time, one of 6 categorical groups for each of the five decennial census values from 1980-2020. The first step in the analysis is to convert the data frame into a sequence for each neighborhood. We can see an illustrative example of the longitudinal sequences from the first five records. Classes are separated by a dash (-). There are 2122 sequences in the dataset and 197 unique sequences; all sequences are displayed in the plot below. Thus, the purpose of clustering the sequences is to extract the general patterns present from the set of all sequences.

[16]:
```
# Convert the data from long to wide format to have a sequence for each neighborhood
# Assuming census_long is your data frame
census_wide <- tract_clusters_decades %>%
  st_drop_geometry() %>%
  select(TRTID10, year, label_6) %>%
  pivot_wider(names_from = year, values_from = label_6)

# Rename the columns to show only the year (remove "cluster_" prefix if it was added
# during previous steps)
colnames(census_wide) <- sub("cluster_", "", colnames(census_wide))

# Ensure columns are in the correct order by year
census_wide <- census_wide %>%
  select(TRTID10, `1980`, `1990`, `2000`, `2010`, `2020`)

# Ensure the sequence columns are factors
census_wide <- census_wide %>%
  mutate(across(starts_with("19") | starts_with("20"), as.factor))

# Check the distinct states (categories) in your sequences
unique_states <- unique(unlist(census_wide[, -1]))  # Exclude TRTID10 column
num_states <- length(unique_states)
print(unique_states)
```

[16]:
```
[1] 6 Black and Hispanic 2 Hispanic           4 Mixed
[4] 3 White              1 Black              5 Asian
Levels: 1 Black 2 Hispanic 3 White 4 Mixed 5 Asian 6 Black and Hispanic
```

[17]:
```
print(num_states)  # Number of unique states
```

[17]:
```
[1] 6
```

[18]:
```
# Define the custom color palette with the correct colors
custom_palette <- c(
  "1 Black" = "#0072B2",
  "2 Hispanic" = "#D55E00",
  "3 White" = "#F0E442",
  "4 Mixed" = "#009E73",
  "5 Asian" = "#CC79A7",
  "6 Black and Hispanic" = "#56B4E9"
)

# Create the sequence object with the renamed columns
sequence_data <- seqdef(census_wide[, -1], cpal = custom_palette)
# Exclude the TRTID10 column

# Check the number of distinct sequences
num_sequences <- seqtab(sequence_data, idx = 0) %>% nrow
print(num_sequences)
```

[18]:
```
[1] 197
```

[19]:
```
# Display the first few sequences
head(sequence_data)
```

[19]:
```
  Sequence
1 6 Black and Hispanic-2 Hispanic-6 Black and Hispanic-6 Black and Hispanic-6 Black and
                                                                                Hispanic
2 2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic
3 2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic
4 2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic
5 2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic-2 Hispanic
6 4 Mixed-2 Hispanic-2 Hispanic-2 Hispanic-5 Asian
```

[20]:
```
# Plot the sequences
seqIplot(sequence_data,                          # Sequence object
         with.legend = "right",                  # Display legend on right side of plot
         cex.legend = 0.6,                       # Change size of legend
         main = "Neighborhood Racial and Ethnic Trajectories") # Plot title
```

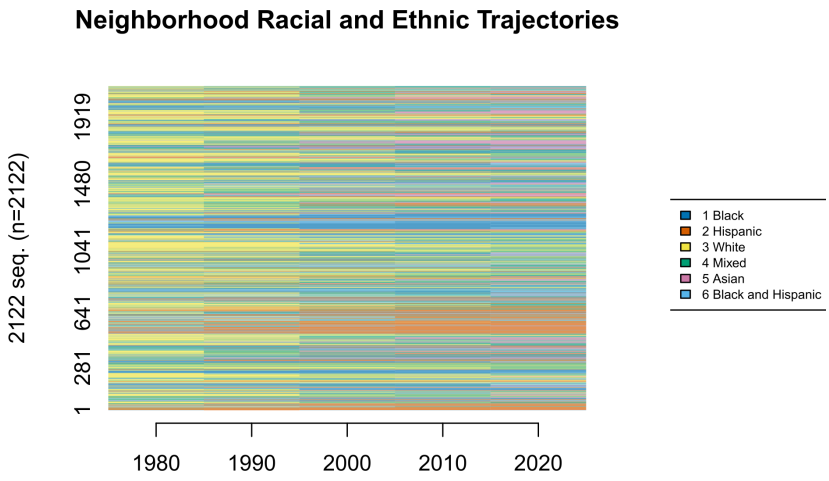

Figure 10: Sequence of each neighborhood.

[20]:
```
Output in Figure 10
```

The next objective is to compute the dissimilarity between all sequences. As described previously, we use the OMstrans algorithm to preserve the ordering of events as we are most interested in describing how neighborhoods have generally transitioned over time. We set a low value (`0.1`) for the parameter `otto` in the `seqdist` command which is the origin-transition trade-off weight. This emphasizes the ordering of sequence states. We also set a high `indel` cost of `3`. Finally, substitution costs are a function of the transition rate `TRATE` between states. This places a lower cost on more frequent transitions.

The substitution cost matrix is shown in Table 2. The table indicates that more frequent transitions from, for example, Black & Hispanic to Black, Mixed to White, White to Mixed, and Black to Black and Hispanic have lower costs than rarer transitions from, for example, Asian to Black, Hispanic to Black, or Black to Asian. Generally, the more mixed race groups are more transitional white the majority race groups tend to transition through a mixed race state. This will become more evident as we examine the resulting sequence clusters.

Once the cost matrix is established, we then cluster the sequences using the dissimilarity matrix as an input to generate the typology. We follow an iterative process in determining the optimal solution like the one described above for the k means clustering. In short, multiple solutions are tested and the resulting sequence clusters are visualized to inspect for heterogeneity. Because our clustering and distance matrix are optimized for describing transitions over time, we end up with one cluster that contains all neighborhoods that remained constant over time. These sequences are very different from one another in the neighborhood types they describe, but they all represent a pathway or sequences of no change.

To describe the resulting sequence clusters, we plot a Sequence Frequency Plot for each cluster. We settled on 14 trajectory clusters describing neighborhood racial and ethnic transitions from 1980 to 2020 in New York City. The Sequence Frequency Plots illustrate the sequences belonging to each cluster and the sequence bars are scaled to visualize the frequency of each sequence. A summary of the trajectories is as follows:

1. Hispanic Majority to Black and Hispanic
2. Stability Cluster
3. White and Mixed Race to Hispanic Majority
4. White to White and Mixed Race to Hispanic Majority
5. Black and Hispanic to Hispanic Majority
6. White Majority to White and Mixed Race
7. White and Mixed Race to Black and Hispanic
8. Black and Hispanic to Black Majority
9. White and Mixed Race to Asian Majority
10. White and Mixed Race to White Majority
11. Black and Hispanic to White and Mixed Race
12. White and Mixed Race to Asian Majority
13. Black Majority to Black and Hispanic
14. Black and Hispanic to Asian Majority

[21]:
```
# Pass this palette to seqdef
# Define sequence data with custom color palette
sequence_data <- seqdef(census_wide[, -1], cpal = custom_palette)
  # Exclude the TRTID10 column

# Compute Optimal Matching (OM) distances using the TRATE cost method
costs <- seqcost(sequence_data, method = "TRATE")
om_distances <- seqdist(sequence_data, method = "OMstran", indel = 3, sm = costs$sm,
                        otto = 0.1)

#Extract the substitution cost matrix
sub_matrix <- round(costs$sm, 2)  # round to 2 decimal places for clarity

# Convert to a data frame for export
sub_df <- as.data.frame(sub_matrix)

# Add row names as a column for better display
sub_df$From <- rownames(sub_df)
sub_df <- sub_df[, c("From", setdiff(names(sub_df), "From"))]

# Optional: reorder columns to match row order
sub_df <- sub_df[, c("From", rownames(sub_matrix))]
```

Table 2: Substitution Cost Matrix

From	1 Black	2 Hispanic	3 White	4 Mixed	5 Asian	6 Black & Hisp
1 Black	0.00	2.00	1.99	1.99	2.00	1.77
2 Hispanic	2.00	0.00	2.00	1.86	1.94	1.86
3 White	1.99	2.00	0.00	1.70	2.00	1.99
4 Mixed	1.99	1.86	1.70	0.00	1.80	1.91
5 Asian	2.00	1.94	2.00	1.80	0.00	1.97
Black & Hisp	1.77	1.86	1.99	1.91	1.97	0.00

```
# Output to table
sub_df[6,1] <- "Black & Hisp"
names(sub_df)[names(sub_df) == '6 Black and Hispanic'] <- '6 Black & Hisp'
knitr::kable(sub_df, row.names=FALSE)
```

[21]:
```
Output in Table 2
```

[22]:
```
# Perform hierarchical clustering using Ward's method on the OM distances
clusterward <- agnes(om_distances, diss = TRUE, method = "ward")

# Define the number of clusters and assign clusters to the sequence data
num_clusters <- 14
clusters <- cutree(clusterward, k = num_clusters)
census_wide$sequence_cluster <- clusters

# Define cluster names for reference
cluster_names1 <- c(
  "1 Black & Hispanic to Hispanic Majority",
  "2 Stability",
  "3 White Mixed Race to Hispanic",
  "4 White to Mixed Race to Hispanic Majority",
  "5 Majority White to Increasing Diversity"
)

# Plot sequences for clusters in groups, adjusting legend settings
plot_sequence_clusters <- function(cluster_range) {
  seqfplot(
   sequence_data[census_wide$sequence_cluster %in% cluster_range, ],
   group = census_wide$sequence_cluster[census_wide$sequence_cluster %in% cluster_range],
   sortv = "from.start",
   border = NA,
   with.legend = "right",         # Place the legend on the right
   legend.prop = 0.2,             # Set the proportion of the plot area for the legend
   legend.border = FALSE          # Remove the border around the legend
  )
}

# Plot sequences for specified cluster ranges
plot_sequence_clusters(1:2)
```

[22]:
```
Output in Figure 11
```

[23]:
```
plot_sequence_clusters(3:4)
```

[23]:
```
Output in Figure 12
```

[24]:
```
plot_sequence_clusters(5:6)
```

[24]:
```
Output in Figure 13
```

[25]:
```
plot_sequence_clusters(7:8)
```

[25]:
```
Output in Figure 14
```

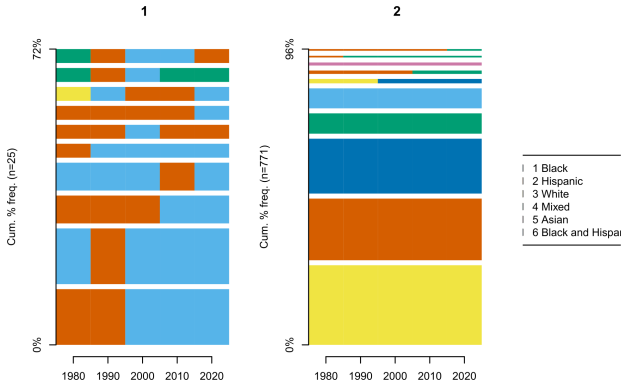

Figure 11: Sequences for Clusters 1 and 2

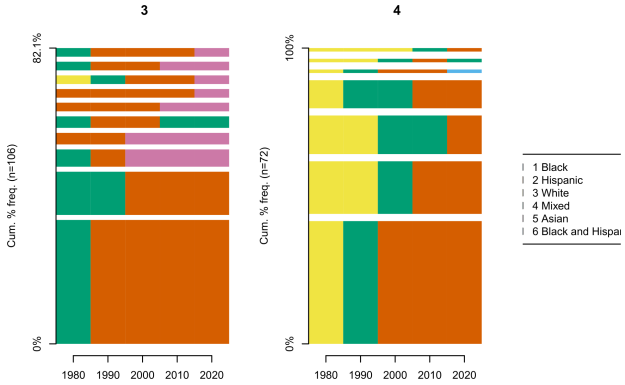

Figure 12: Sequences for Clusters 3 and 4

```
[26]: plot_sequence_clusters(9:10)
```

```
[26]: Output in Figure 15
```

```
[27]: plot_sequence_clusters(11:12)
```

```
[27]: Output in Figure 16
```

```
[28]: plot_sequence_clusters(13:14)
```

```
[28]: Output in Figure 17
```

Of these 14 pathways, there are 3 that lead to the formation of a neighborhood

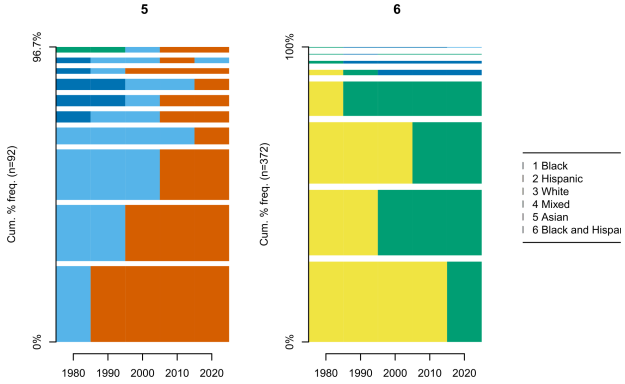

Figure 13: Sequences for Clusters 5 and 6

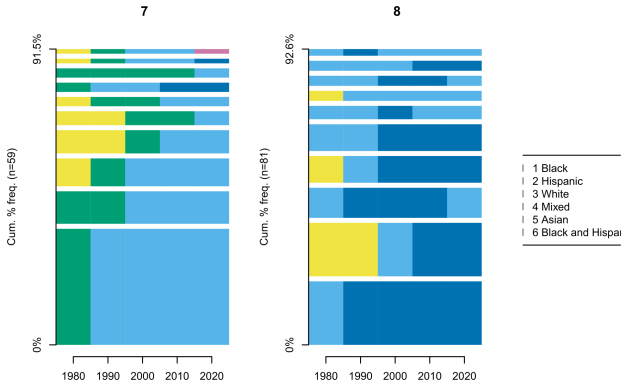

Figure 14: Sequences for Clusters 7 and 8

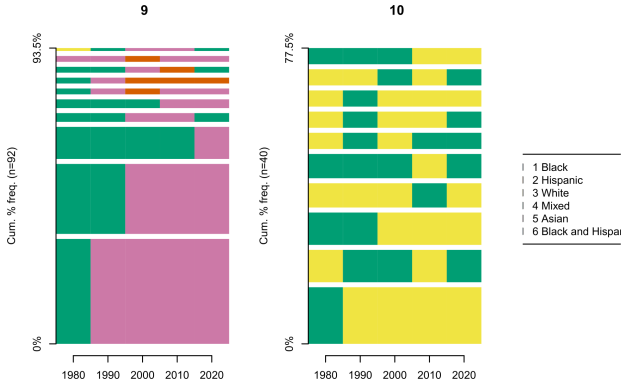

Figure 15: Sequences for Clusters 9 and 10

transitioning into a Hispanic Majority cluster by 2020. This includes Cluster 3 - showing neighborhoods that went from being a slight majority White, but with a mixture of other races in 1980 and 1990 to transitioning to majority Hispanic by 1990 or 2000. Some of these sequences indicate a continued transition towards becoming majority Asian in the later years. Further segmenting the sequences into more clusters may have separated out those trajectories, but for the sake of brevity, we leave them mixed in. Spatially, these are shown in orange on the map below and can be see clustered in Staten Island, Queens, and in the Northern portion of the city. They are also notably adjacent to neighborhoods indicated by the red color, those representing sequence cluster 4, transitioning from majority White to White and Mixed Race and then to Majority Hispanic. This latter cluster might represent the precursor to cluster 3, but are neighborhoods that made this transition from majority White later, where the changes took time to spatially spillover

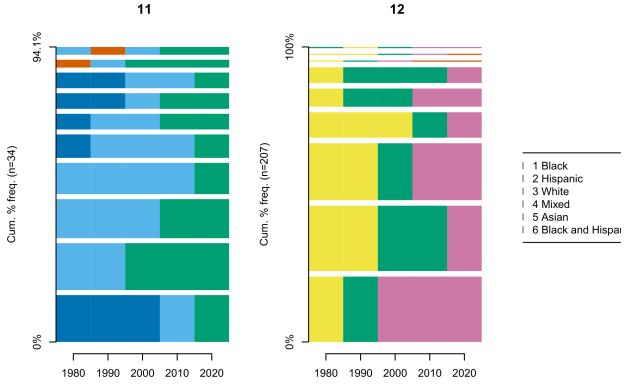

Figure 16: Sequences for Clusters 11 and 12

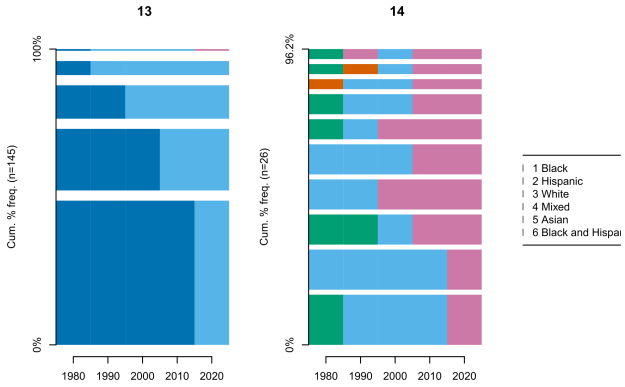

Figure 17: Sequences for Clusters 13 and 14

to adjacent neighborhoods, as indicated by the map. Both of these pathways depict a transition from largely White populations to largely Hispanic.

The third pathway depicting a transition to majority Hispanic is distinct. It is represented by cluster 5 showing a transition from either majority Black neighborhoods towards a mixed Hispanic and Black group and eventually majority Hispanic or, beginning the 1980 time stamp, in a more mixed Black and Hispanic state. Geographically, these neighborhoods are shown more in the northern sections of Manhattan and the Bronx.

From these two sets of sequence clusters, we can see that majority Hispanic neighborhoods in New York City have emerged out of either majority Black or Majority White neighborhoods over time.

[29]:
```
# Define the Hispanic majority clusters for plotting
hispanic_majority_clusters <- census_wide %>%
  filter(sequence_cluster %in% c(4, 5, 6))  # Filter clusters 4, 5, and 6

# Join the filtered data with the tract shapefile based on TRTID10
tract_hispanic_majority <- tract %>%
  left_join(hispanic_majority_clusters, by = "TRTID10") %>%
  erase_water(area_threshold = 0.75)

# Add county borders and color to the plot
ggplot(tract_hispanic_majority) +
  geom_sf(aes(fill = factor(sequence_cluster)), color = NA) +  # Map sequence clusters
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add white county borders
  scale_fill_manual(values = c("#FB8072", "#80B1D3", "#FDB462"),  # Custom color palette
                    labels = c("3 White - White, Mixed Race - Hispanic",
                               "5 Black - Black & Hispanic - Hispanic",
                               "4 White, Mixed Race - Hispanic"),
                    name = "Hispanic Majority Pathway") +
  labs(title = "",
       x = "", y = "") +
  theme_void() +
  theme(legend.position = "right",  # Place legend on the right
        legend.direction = "vertical",  # Arrange legend items vertically
        legend.title = element_text(size = 12),  # Customize legend title size
        legend.key.width = unit(2, "cm"),  # Adjust legend key width
        legend.box = "vertical")  # Place legend title on top of the legend
```

There are also 3 pathways leading to an Asian majority neighborhood type. These include clusters 9 and 12 which are also likely continuations of longer trajectories, that show a gradual transition from Majority White to White mixed race and eventually to Asian Majority. Geographically, these are clustered in the northern section of Queens. Sequence cluster 14 is more distinct in that the Asian majority transitioned from the Black and Hispanic mixed group and spatially, they are generally located in upper Manhattan and the Bronx.

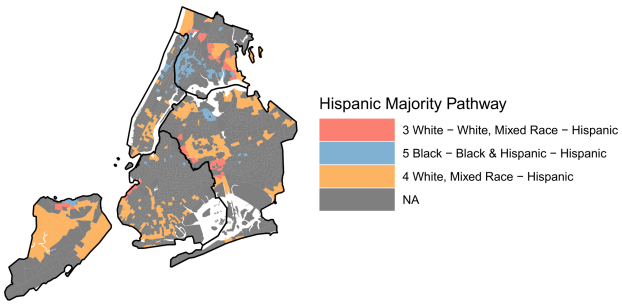

Figure 18: Neighborhoods Following Pathway to Hispanic Majority with White County Borders.

[30]:
```
# Filter census data to include only clusters 14, 12, and 9
# (Asian Majority Pathway clusters)

asian_majority_clusters <- census_wide %>%
  filter(sequence_cluster %in% c(14, 12, 9))

# Join the filtered data with the tract shapefile (here, use TRTID10)
tract_asian_majority <- tract %>% erase_water(area_threshold = 0.75) %>%
  left_join(asian_majority_clusters, by = "TRTID10")  # Join with spatial data

# Create a map of the neighborhoods with clusters 14, 12, and 9 with new labels
ggplot(tract_asian_majority) +
  geom_sf(aes(fill = factor(sequence_cluster)), color = NA) +  # Map sequence clusters
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add white county borders
  scale_fill_manual(values = c("#8DD3C7", "#FFFFB3", "#BEBADA"),
    # Custom color palette for Asian Majority clusters
            labels = c("14 White, Mixed Race to Black & Hispanic to Asian Majority",
                       "12 White Majority to White Mixed Race to Asian",
                       "9 White Mixed Race to Asian"),
            name = "Asian Majority Pathway") +
  labs(title = "",
       x = "", y = "") +
  theme_void() +
  theme(legend.position = "right",  # Place legend at the bottom
        legend.direction = "vertical",  # Arrange legend items horizontally
        legend.title = element_text(size = 12),  # Customize legend title size
        legend.key.width = unit(2, "cm"),  # Adjust legend key width
        legend.box = "vertical")  # Place legend title on top of the legend
```

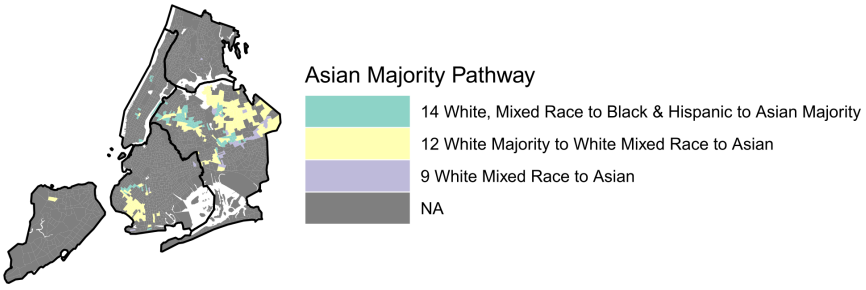

Figure 19: Neighborhoods Following Pathway to Asian Majority

[30]:
```
Output in Figure 19
```

[31]:
```
# Filter data for Increasing White (clusters 10 and 11)
increasing_white_clusters <- census_wide %>%
  filter(sequence_cluster %in% c(10, 11))  # Clusters 10 and 11 for increasing White

# Filter data for Increasing Black (clusters 1 and 8)
increasing_black_clusters <- census_wide %>%
```

```
  filter(sequence_cluster %in% c(1, 8))  # Clusters 1 and 8 for increasing Black

# Join the filtered data with the tract shapefile for each group
tract_increasing_white <- tract %>%  erase_water(area_threshold = 0.75) %>%
  left_join(increasing_white_clusters, by = "TRTID10")
    # Join spatial data for increasing White

tract_increasing_black <- tract %>%   erase_water(area_threshold = 0.75) %>%
  left_join(increasing_black_clusters, by = "TRTID10")
    # Join spatial data for increasing Black
```

There are two pathways for increasing both Black and White shares in a neighborhoods. Neighborhoods that became increasingly White either followed a trajectory from White mixed race to majority White (Cluster 10) or from either all Black or Black and Hispanic to White and Mixed race (11). Notably, this transition largely took place within the past 1-2 decades, aligning with when gentrification trends became accentuated in some cities, including New York. We see a clear cluster of this latter group in Brooklyn, a borough whose gentrification trends have been well documented (Chronopoulos 2020, Halasz 2023).

For the case of increasing Black populations, Cluster 1 shows a pathway from Hispanic majority to mixed Black and Hispanic and Cluster 8 shows a gradual transition from Black and Hispanic to majority Black; a trend that largely begin towards the middle of the study period, around the 2000 census data mark. Spatially, the former is more dispersed, while the latter is depicted more clearly in the southern neighborhoods of Brooklyn.

[32]:
```
# Adjust plotting settings to improve map visibility
# Create the map for Increasing White, zoomed in on the five boroughs
ggplot(tract_increasing_white) +
  geom_sf(aes(fill = factor(sequence_cluster)), color = NA) +  # Map sequence clusters
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add white county borders
  scale_fill_manual(values = c("#8DD3C7", "#FFFFB3"),           # Custom color palette
            labels = c("White Mixed Race to Majority White",    # for White clusters
                       "Black and Hispanic to White Mixed Race"),
            name = "Increasing White Pathway") +
  coord_sf(xlim = c(-74.3, -73.7), ylim = c(40.5, 40.9), expand = FALSE) +
    # Zoom into the NYC area
  theme_void() +
  theme(legend.position = "right",  # Place legend at the right
        legend.direction = "vertical",  # Arrange legend items vertically
        legend.title = element_text(size = 12),  # Customize legend title size
        legend.key.width = unit(2, "cm"),
        legend.box = "vertical")  # Place legend title on top
```

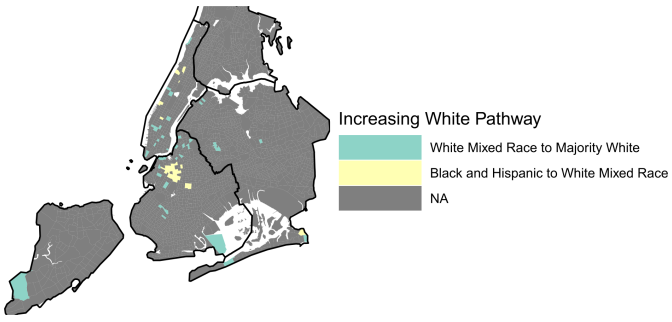

Figure 20: Neighborhoods Following Pathway to White Majority

[32]:
```
Output in Figure 20
```

[33]:
```
# Create the map for Increasing Black, zoomed in on the five boroughs
ggplot(tract_increasing_black) +
  geom_sf(aes(fill = factor(sequence_cluster)), color = NA) +  # Map sequence clusters
  geom_sf(data = nyc_counties, fill = NA, color = "black", size = 0.5) +
    # Add white county borders
```

```
  scale_fill_manual(values = c("#FB8072", "#80B1D3"),           # Custom color palette
            labels = c("Hispanic Majority to Black & Hispanic", # for Black clusters
                       "Black and Hispanic to Black Majority"),
            name = "Increasing Black Pathway") +
  coord_sf(xlim = c(-74.3, -73.7), ylim = c(40.5, 40.9), expand = FALSE) +
    # Zoom into the NYC area
  theme_void() +
  theme(legend.position = "right",  # Place legend at the right
        legend.direction = "vertical",  # Arrange legend items vertically
        legend.title = element_text(size = 12),  # Customize legend title size
        legend.key.width = unit(2, "cm"),
        legend.box = "vertical")  # Place legend title on top
```

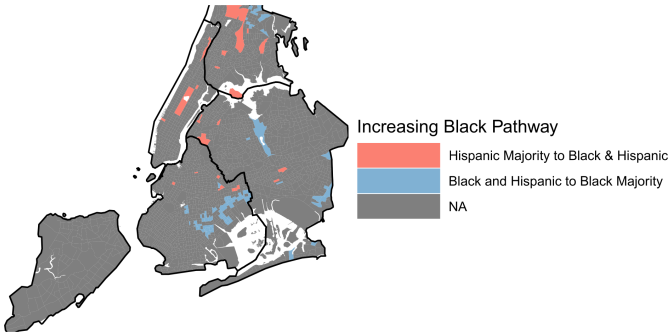

Figure 21: Neighborhoods Following Pathway to Black Majority

[33]:
```
Output in Figure 21
```

Finally, of the sequences depicting no change from 1980 to 2020, shown in Cluster 2, the majority of those are for tracts with a racial majority: White, Hispanic, and Black. But those are followed by two stable sequences of neighborhoods with some racial diversity. The first is the majority White, but with a mixture of other races and the second is the mixed Black and Hispanic cluster. There is some debate in the literature whether racially mixed neighborhoods can be stable over time, or are they simply depicting a point along a pathway of change when one group will eventually become a majority. Research has shown that highly diverse neighborhoods are quite unstable over time, likely to transition to a less diverse state. In particular the transition from predominantly White towards majority Hispanic, results in a period of unstable racial mixture while that transition takes place (Wright et al. 2020). Others have identified a small, but persistent set of racially diverse neighborhoods throughout the United States, but particularly in racially diverse metropolitan areas (Hipp, Kim 2023). Here, we find some evidence of stable, non-racially homogeneous Census Tracts.

6 Conclusions

This analysis showcases a method for developing and visualizing a typology of neighborhood change pathways. We used a case study of decennial racial and ethnic changes in New York City census tracts from 1980-2020. The workflow first involves developing a cross-sectional typology of classes describing the racial mixture of neighborhoods. To do so, we used an unsupervised classification approach, k-means to derive six such clusters.

We demonstrated how determining the number of clusters often falls to more of an art than a precise science as our final clustering for this analysis exceeded the mathematically optimal three clusters, but provided us with more nuance on the racial mixture of neighborhoods. We performed the cluster analysis on all census tracts in the city for each of the five decennial census time stamps at once to ensure a temporally consistent set of groupings. We then created sequences of neighborhood clusters over the study period and developed a typology of sequences that grouped them based on the similarity of how they changed over time. To do so, we used the OMstrans algorithm for computing sequence dissimilarity to ensure that the ordering or sequencing of events was preserved. Finally,

we mapped sequence clusters to spatially visualize neighborhoods that followed similar pathways of change. For our case study of the largest city in the United States, and one of the most diverse, we observed a decline in the number of majority White census tracts over time. We identified several pathways of change leading to majority Hispanic neighborhoods - emerging either out of previously majority White or Black neighborhoods. We also observed pathways leading to majority Asian tracts, transitioning from Black and Hispanic neighborhoods, largely in Queens. A smaller share of neighborhoods became either increasingly White or Black. Neighborhoods that saw an increase in the share of Whites, saw a notable increase in the transition from Black and Hispanic to White and mixed race over the past two decades. Neighborhoods that increased in the share of Blacks largely transitioned from Black and Hispanic to majority Black or from White and mixed race to Black and Hispanic.

The strength of the sequence analysis technique lies in its ability to clearly visualize common pathways of neighborhood change. One of its limitations, however, is the need to segment continuous, longitudinal data into discrete, categorical states. In this article, we devoted attention to this step, illustrating the use of a k-means algorithm as an unsupervised classification method for grouping multiple variables. This phase involves decisions, particularly, the number of classes, as that directly influences the resulting sequences and subsequent interpretations. Alternative methods for clustering time series data that preserve the continuous nature of the data may be preferable in cases where the number of neighborhood variables is limited (Delmelle et al. 2025).

Beyond describing trajectories, sequence analysis also opens up avenues for further inquiry. The resulting sequence clusters can serve as the basis for additional analyses like modeling the predictors of specific trajectories or analyzing their spatial patterns using categorical spatial autocorrelation measures such as the join-count statistic.

References

Chapple K, Zuk M (2016) Forewarned: The use of neighborhood early warning systems for gentrification and displacement. *Cityscape* 18: 109–130

Chronopoulos T (2020) What's happened to the people? Gentrification and racial segregation in Brooklyn. *Journal of African American Studies* 24: 549–572. CrossRef

Delmelle E (2015) Five decades of neighborhood classifications and their transitions: A comparison of four US cities, 1970–2010. *Applied Geography* 57: 1–11. CrossRef

Delmelle E (2017) Differentiating pathways of neighborhood change in 50 US metropolitan areas. *Environment and planning A* 49: 2402–2424. CrossRef

Delmelle EC (2016) Mapping the DNA of urban neighborhoods: Clustering longitudinal sequences of neighborhood socioeconomic change. *Annals of the American Association of Geographers* 106: 36–56. CrossRef

Delmelle EC (2022) GIScience and neighborhood change: Toward an understanding of processes of change. *Transactions in GIS* 26: 567–584. CrossRef

Delmelle EC, Nilsson I, Duma N (2025) Time series clustering for exploring neighborhood dynamics: The case of US neighborhood racial and ethnic trends, 1990–2020. *Geographical Analysis*. CrossRef

Farrell CR, Lee BA (2011) Racial diversity and change in metropolitan neighborhoods. *Social Science Research* 40: 1108–1123. CrossRef

Frey WH (2022) A new great migration is bringing black Americans back to the South. Brookings institution, https://www.brookings.edu/research/a-new-great-migration-is-bringing-black-americans-back-to-the-south/

Gabadinho A, Ritschard G, Müller NS, Studer M (2011) Analyzing and visualizing state sequences in R with TraMineR. *Journal of statistical software* 40: 1–37. CrossRef

Galster G (2001) On the nature of neighbourhood. *Urban Studies* 38: 2111–2124. CrossRef

González-Leonardo M, Newsham N, Rowe F (2023) Understanding population decline trajectories in Spain using sequence analysis. *Geographical Analysis* 55: 495–516. CrossRef

Halasz JR (2023) Between gentrification and supergentrification: Hybrid processes of socio-spatial upscaling. *Journal of Urban Affairs* 45: 771–796. CrossRef

Hipp JR, Kim JH (2023) Persistent racial diversity in neighborhoods: What explains it and what are the long-term consequences? *Urban Geography* 44: 640–667. CrossRef

Landis JD (2016) Tracking and explaining neighborhood socioeconomic change in US metropolitan areas between 1990 and 2010. *Housing Policy Debate* 26: 2–52. CrossRef

Logan JR, Xu Z, Stults BJ (2014) Interpolating US decennial census tract data from as early as 1970 to 2010: A longitudinal tract database. *The Professional Geographer* 66: 412–420. CrossRef

Patias N, Rowe F, Cavazzi S (2020) A scalable analytical framework for spatio-temporal analysis of neighborhood change: A sequence analysis approach. In: *Geospatial Technologies for Local and Regional Development: Proceedings of the 22nd AGILE Conference on Geographic Information Science 22*, 223–241. Springer

Reibel M, Regelson M (2011) Neighborhood racial and ethnic change: The time dimension in segregation. *Urban Geography* 32: 360–382. CrossRef

Studer M, Ritschard G (2016) What matters in differences between life trajectories: A comparative review of sequence dissimilarity measures. *Journal of the Royal Statistical Society Series A: Statistics in Society* 179: 481–511. CrossRef

Terbeck FJ (2023) The impact of regional and local population trends on suburban poverty and ethnoracial composition change: A shift-share analysis of the Chicago metropolitan area in the 2000s. *Population, Space and Place* 28: e2549. CrossRef

Von Luxburg U, Williamson RC, Guyon I (2012) Clustering: Science or art? In: *Proceedings of ICML workshop on unsupervised and transfer learning*, 65–79. JMLR Workshop and Conference Proceedings

Wright R, Ellis M, Holloway SR, Catney G (2020) The instability of highly racially diverse residential neighborhoods in the United States. *Sociology of Race and Ethnicity* 6: 365–381. CrossRef

Wright R, Ellis M, Holloway SR, Wong S (2014) Patterns of racial diversity and segregation in the United States: 1990–2010. *The Professional Geographer* 66: 173–182. CrossRef

Zwiers M, van Ham M, Manley D (2018) Trajectories of ethnic neighbourhood change: Spatial patterns of increasing ethnic diversity. *Population, Space and Place* 24: e2094. CrossRef