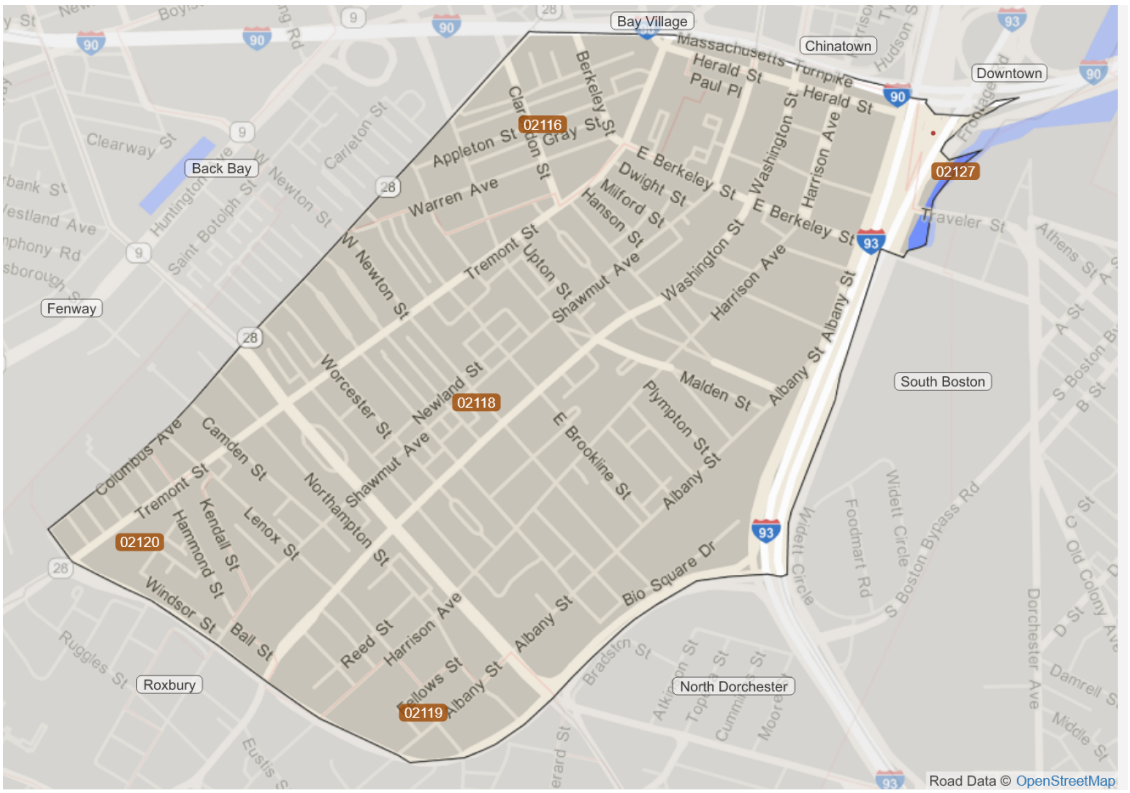
south_end

February 1, 2020

1 Background

The census tract for the South End begins just on the SE side of Columbus Avenue, as shown in this map:



Let's explore the demographics of the South End.

2 Setup

[1]:
```
import pandas as pd
import altair as alt
from altair import datum
```

3 Loading Data

[2]:
```
df = pd.read_csv('south_end.csv')
df
```

[2]:

	Category	Subcategory	Decade	Count	Percent
0	Population	NaN	1950	49753	NaN
1	Age	0-9 years	1950	5870	0.12
2	Age	10-19 years	1950	4387	0.09
3	Age	20-34 years	1950	11947	0.24
4	Age	35-54 years	1950	14374	0.29
..	…	…	…	…	…
149	Labor Force (age 16+)	Male	2010	-	NaN
150	Labor Force (age 16+)	Female	2010	-	NaN
151	Housing Tenure	Occupied Housing Units	2010	15629	NaN
152	Housing Tenure	Owner-occupied	2010	5702	0.36
153	Housing Tenure	Renter-occupied	2010	9927	0.64

[154 rows x 5 columns]

[3]:
```
df[['Category','Subcategory']]
```

[3]:

	Category	Subcategory
0	Population	NaN
1	Age	0-9 years
2	Age	10-19 years
3	Age	20-34 years
4	Age	35-54 years
..	…	…
149	Labor Force (age 16+)	Male
150	Labor Force (age 16+)	Female
151	Housing Tenure	Occupied Housing Units
152	Housing Tenure	Owner-occupied
153	Housing Tenure	Renter-occupied

[154 rows x 2 columns]

[4]:
```
df[['Category','Subcategory']].drop_duplicates()
```

[4]:

	Category	Subcategory
0	Population	NaN
1	Age	0-9 years
2	Age	10-19 years
3	Age	20-34 years
4	Age	35-54 years
5	Age	55-64 years
6	Age	65 years and over

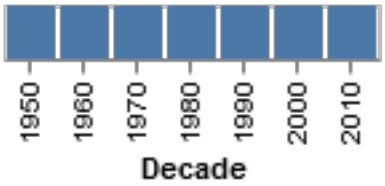
7	Educational Attainment (age 25+)	less than High School
8	Educational Attainment (age 25+)	High School or GED
9	Educational Attainment (age 25+)	Some College or Associate's Degree
10	Educational Attainment (age 25+)	Bachelor's Degree or Higher
11	Nativity	Foreign Born
12	Race/ Ethnicity	White
13	Race/ Ethnicity	Black/ African American
14	Race/ Ethnicity	Hispanic
15	Race/ Ethnicity	Asian/PI
16	Race/ Ethnicity	Other
17	Labor Force (age 16+)	Male
18	Labor Force (age 16+)	Female
19	Housing Tenure	Occupied Housing Units
20	Housing Tenure	Owner-occupied
21	Housing Tenure	Renter-occupied

4 Creating Visualizations

4.1 Age Distribution by Decade

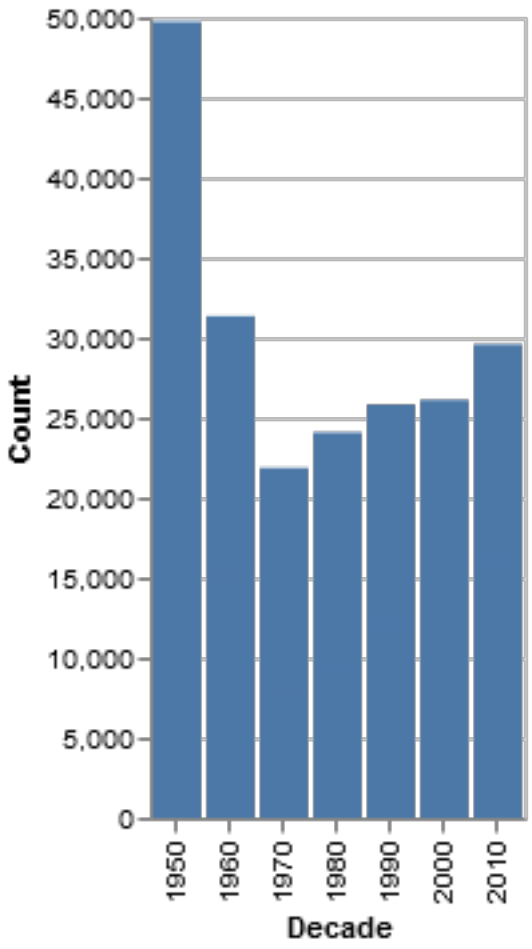
[5]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
)
```

[5]:



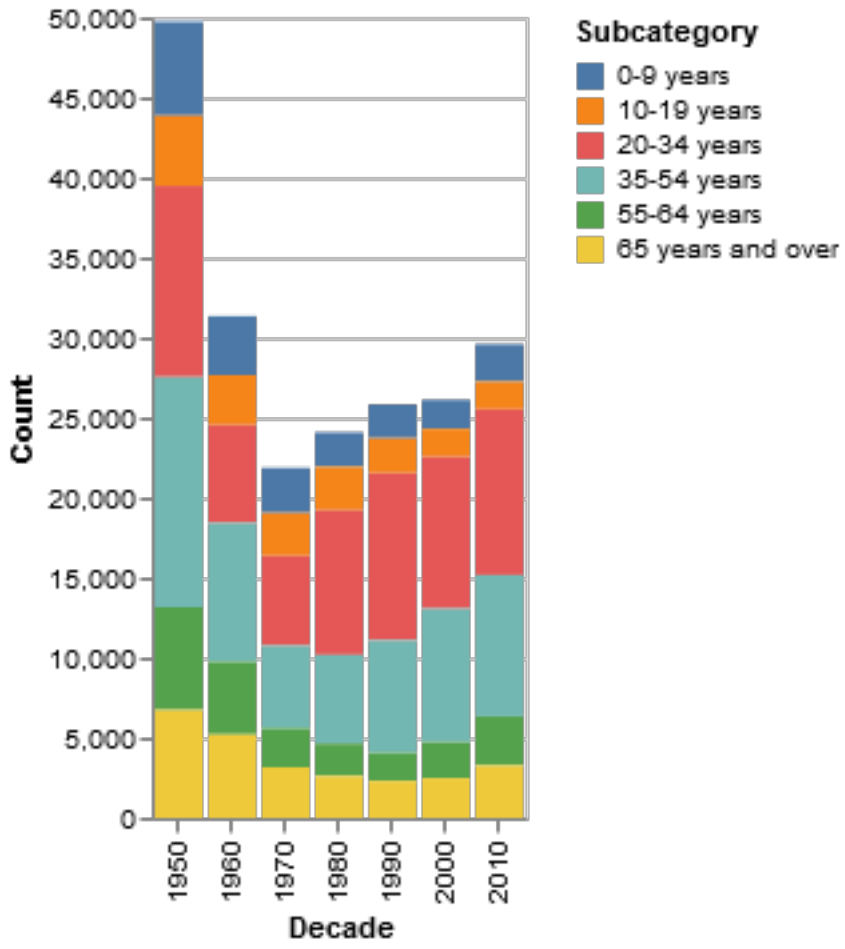
[6]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q'
)
```

[6]:



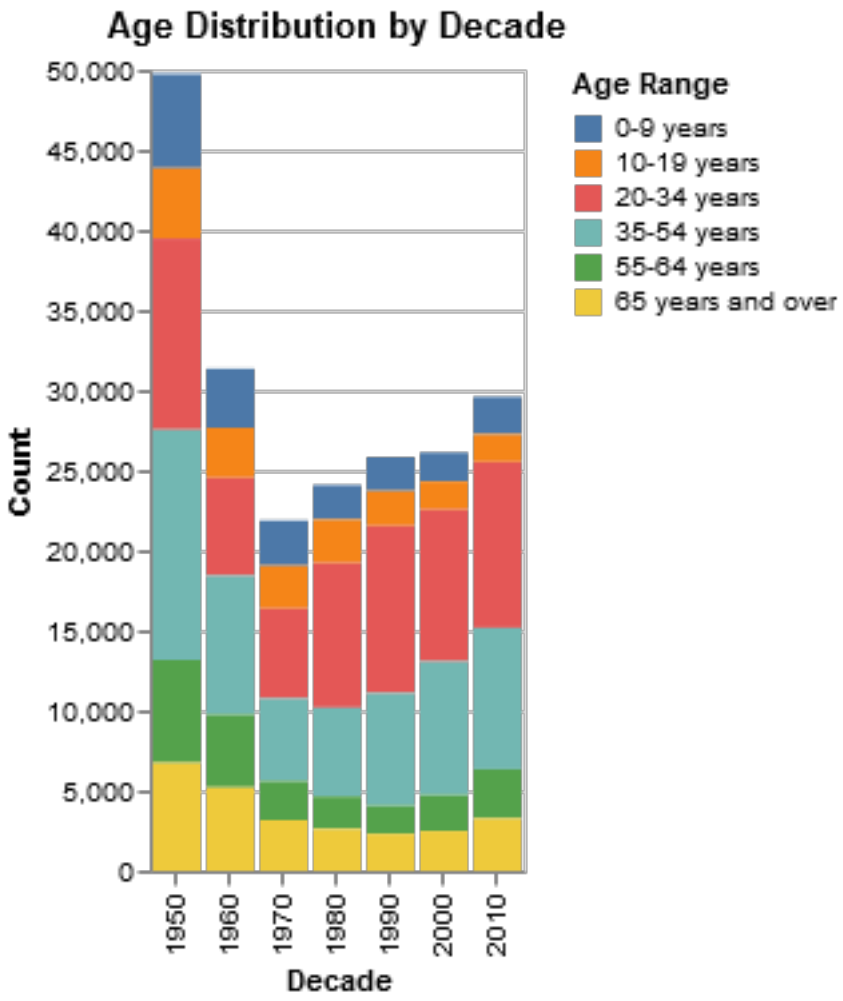
[7]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory:N'
).transform_filter(
    (datum.Category == 'Age')
)
```

[7]:



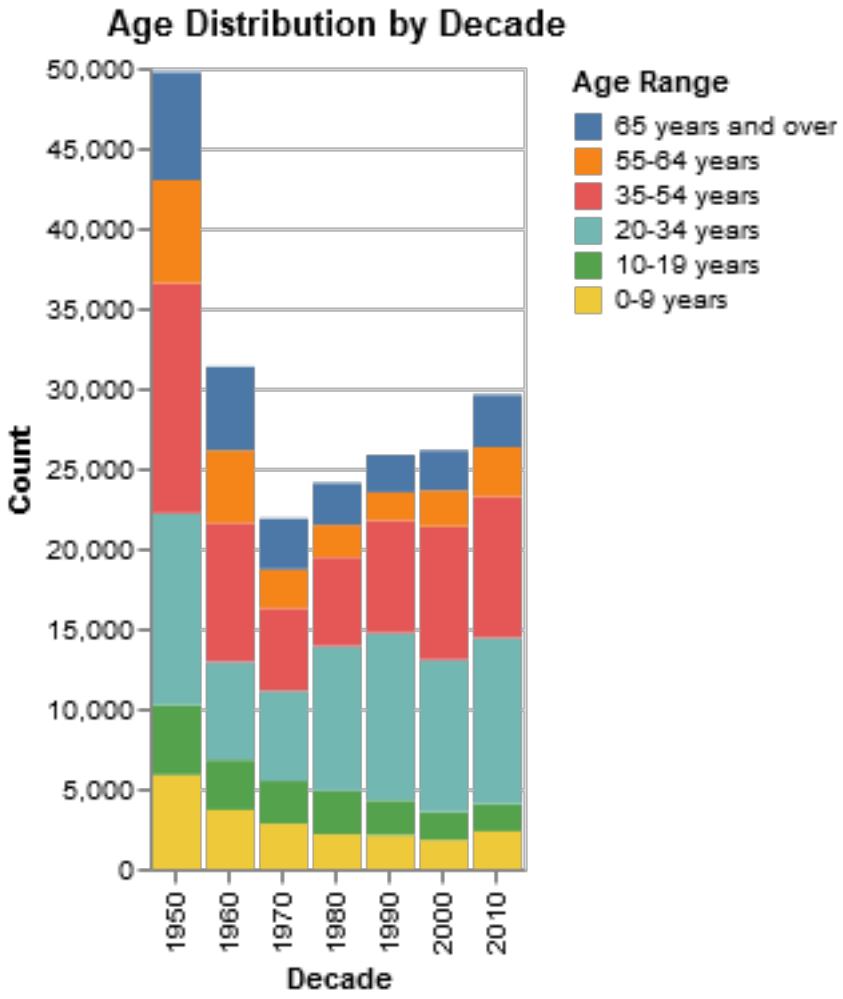
[8]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory', title='Age Range'),
).transform_filter(
    (datum.Category == 'Age')
).properties(
    title='Age Distribution by Decade'
)
```

[8]:



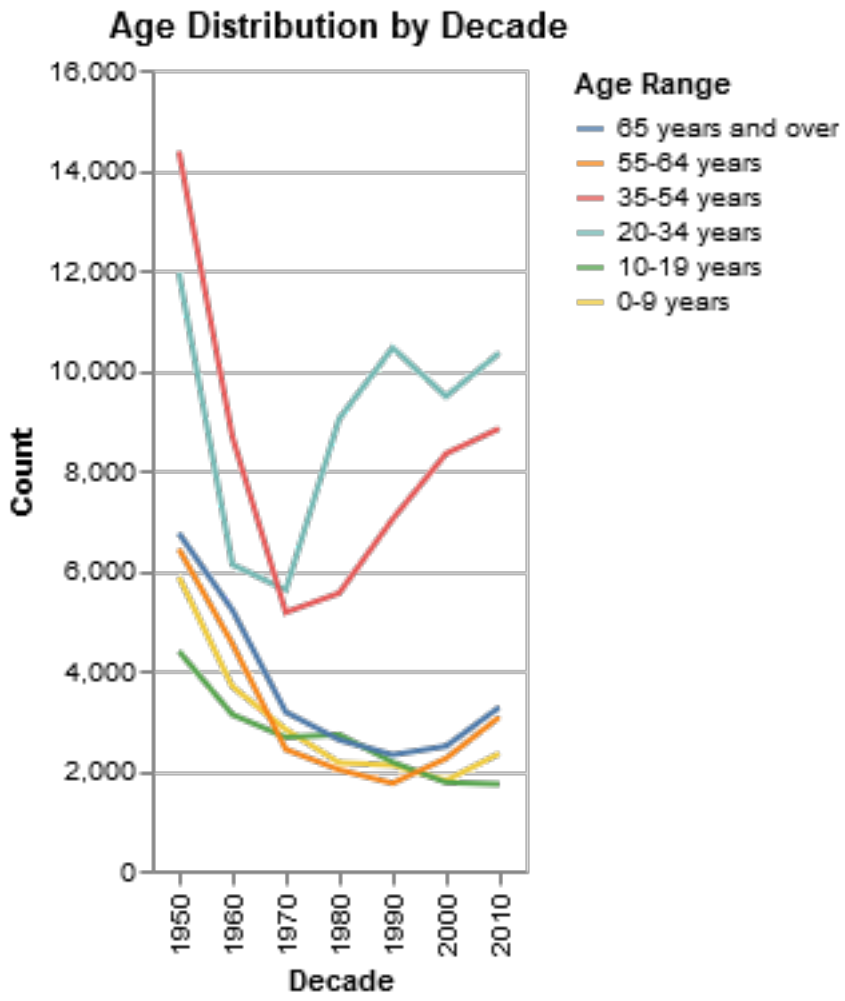
[9]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title='Age Range', sort='descending'),
    order=alt.Order(
        'Subcategory:N',
        sort='ascending'
    )
).transform_filter(
    (datum.Category == 'Age')
).properties(
    title='Age Distribution by Decade'
)
```

[9]:



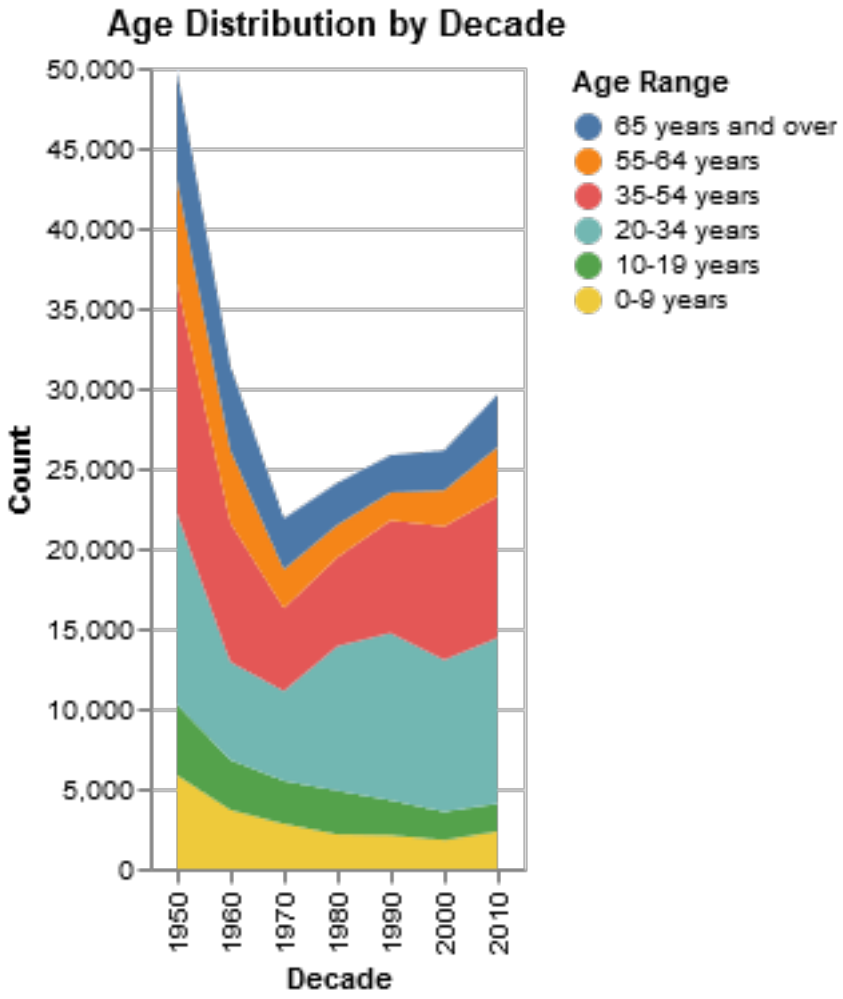
[10]:
```
alt.Chart(df).mark_line().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title='Age Range', sort='descending'),
    order=alt.Order(
        'Subcategory:N',
        sort='ascending'
    )
).transform_filter(
    (datum.Category == 'Age')
).properties(
    title='Age Distribution by Decade'
)
```

[10]:



[11]:
```
alt.Chart(df).mark_area().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title='Age Range', sort='descending'),
    order=alt.Order(
        'Subcategory:N',
        sort='ascending'
    )
).transform_filter(
    (datum.Category == 'Age')
).properties(
    title='Age Distribution by Decade'
)
```

[11]:

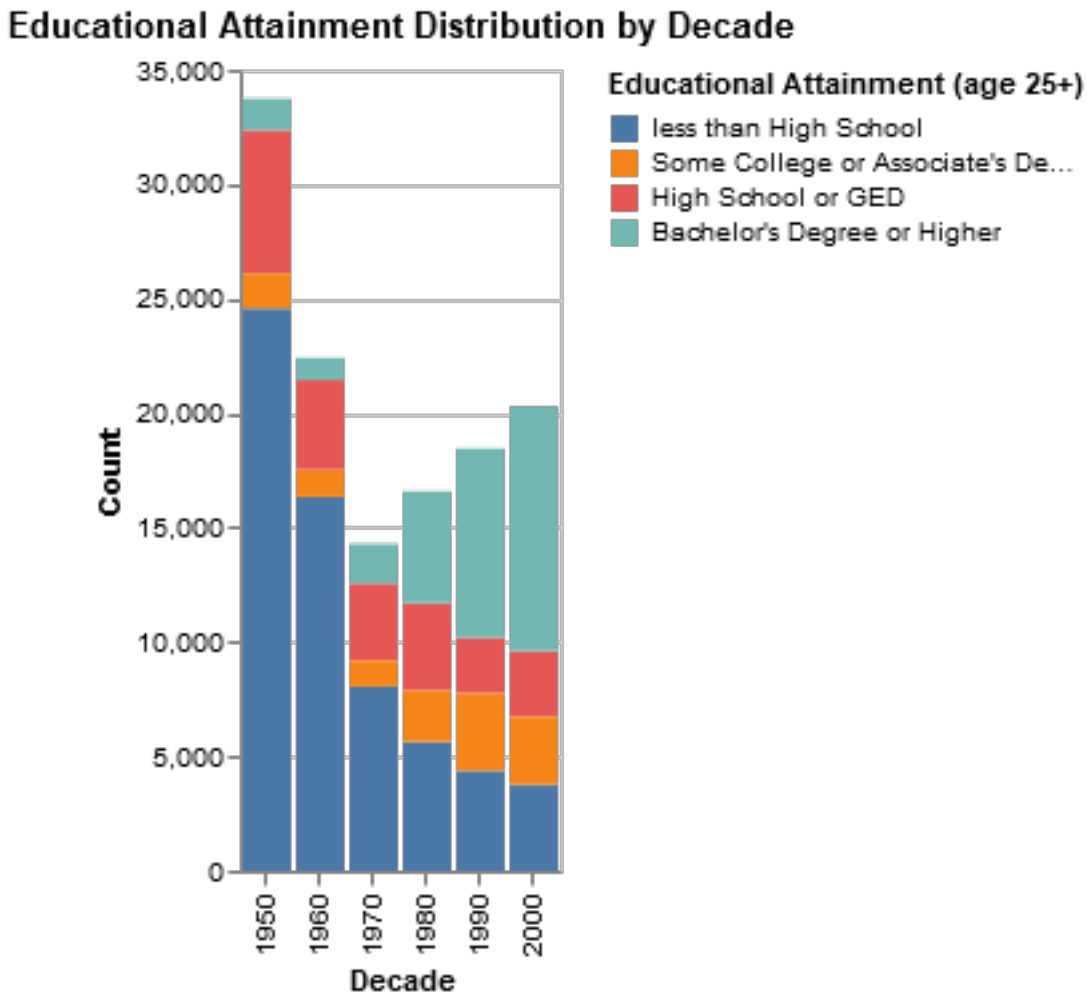


4.2 Educational Attainment by Decade

[12]:
```
eduField = 'Educational Attainment (age 25+)'
```

[13]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title=eduField, sort='descending'),
).transform_filter(
    (datum.Category == eduField)
).properties(
    title='Educational Attainment Distribution by Decade'
)
```

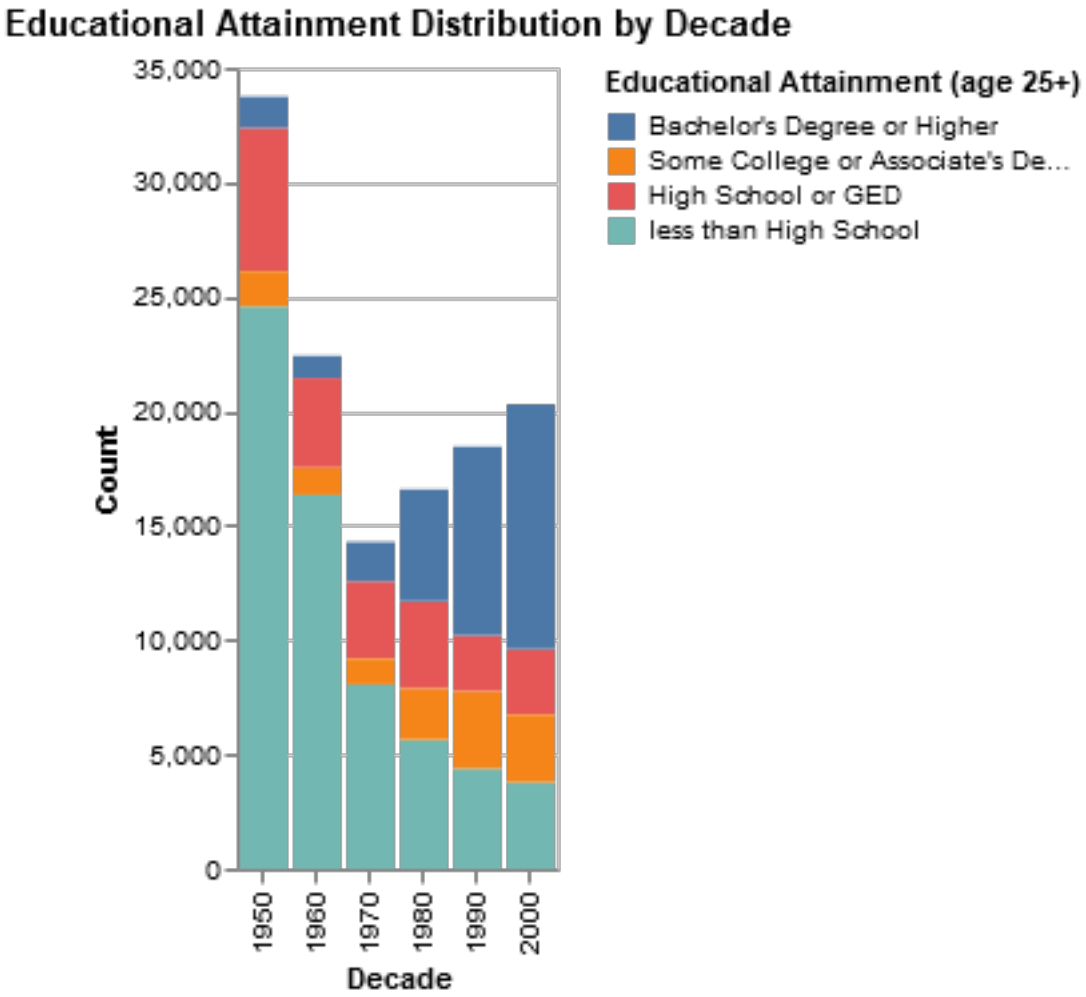
[13]:



[14]:
```
eduSortOrder = [
        "Bachelor's Degree or Higher",
        "Some College or Associate's Degree",
        'High School or GED',
        'less than High School'
        ]
```

[15]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title=eduField, sort=eduSortOrder),
).transform_filter(
    (datum.Category == eduField)
).properties(
    title='Educational Attainment Distribution by Decade'
)
```

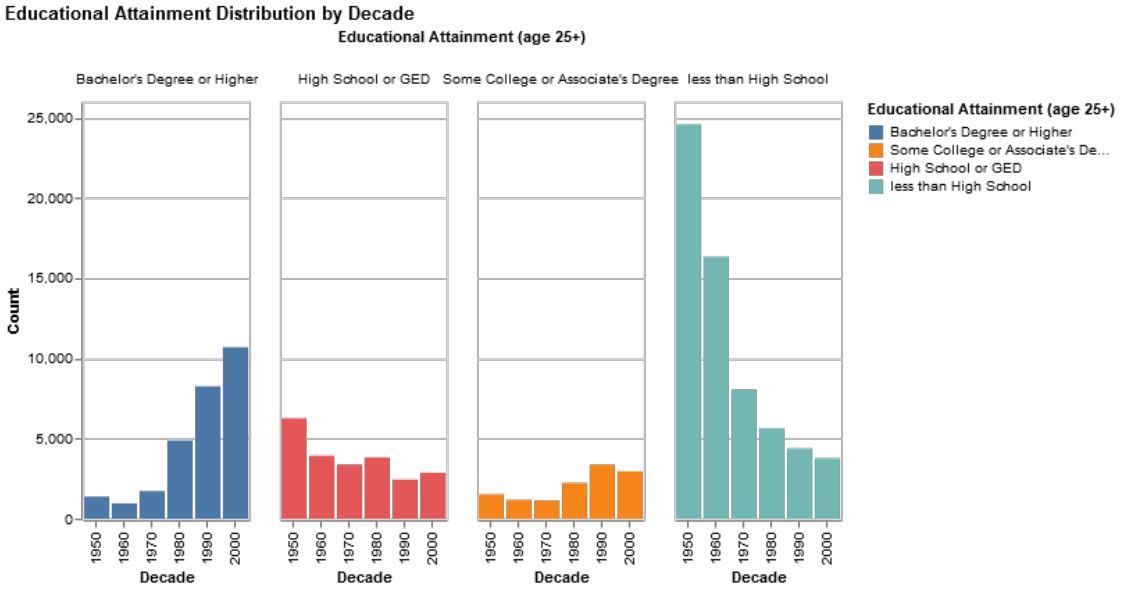
[15]:



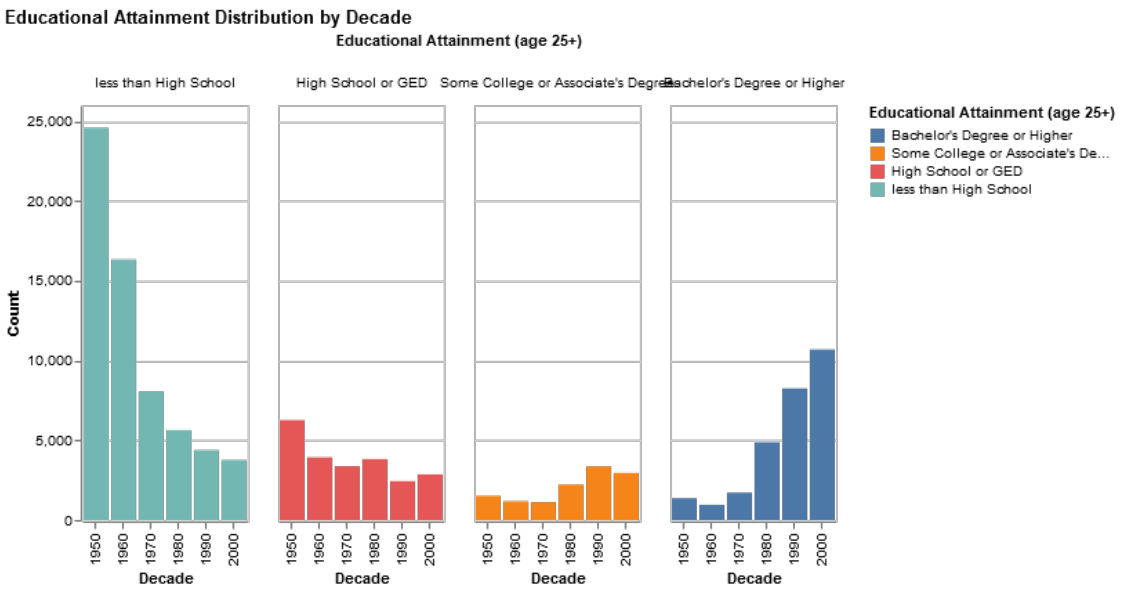
[16]:

```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title=eduField, sort=eduSortOrder),
    column=alt.Column('Subcategory:N', title=eduField)
).transform_filter(
    (datum.Category == eduField)
).properties(
    title='Educational Attainment Distribution by Decade'
)
```

[16]:



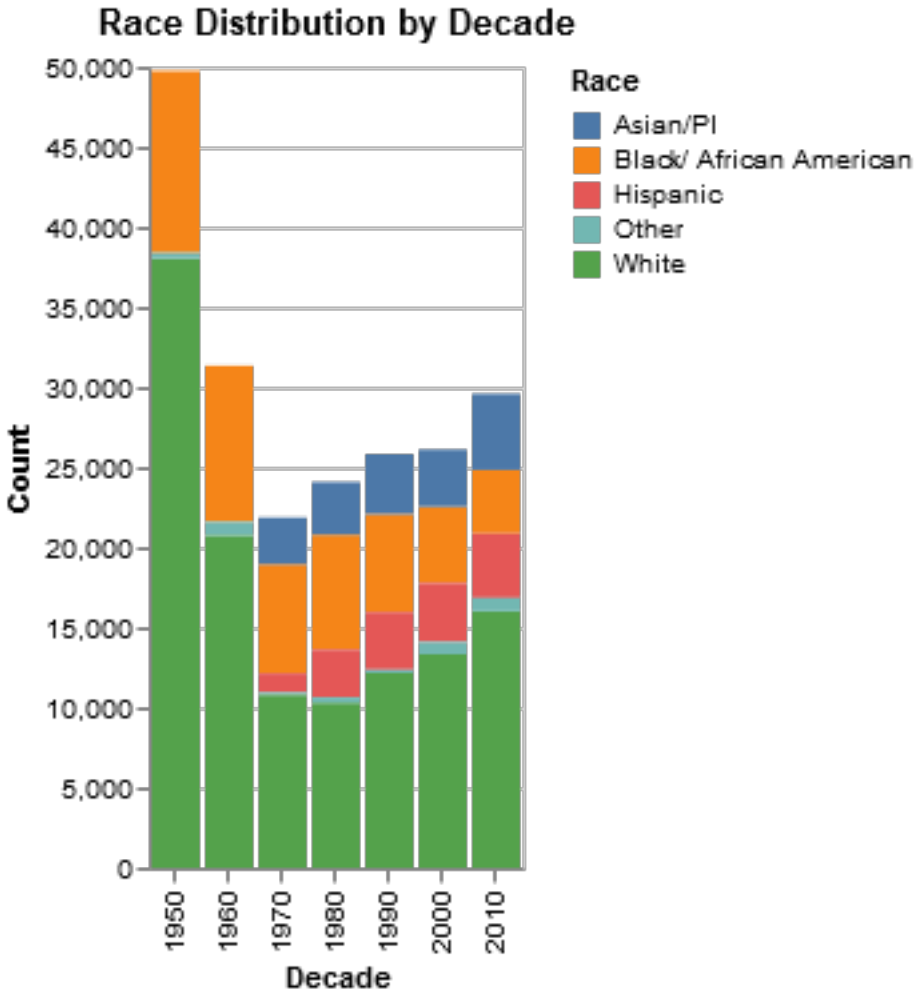
[17]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title=eduField, sort=eduSortOrder),
    column=alt.Column('Subcategory:N', title=eduField, sort=eduSortOrder[::-1]),
).transform_filter(
    (datum.Category == eduField)
).properties(
    title='Educational Attainment Distribution by Decade'
)
```

[17]:


4.3 Race by Decade

[18]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', title='Race')
).transform_filter(
    (datum.Category == 'Race/ Ethnicity')
).properties(
    title='Race Distribution by Decade'
)
```

[18]:

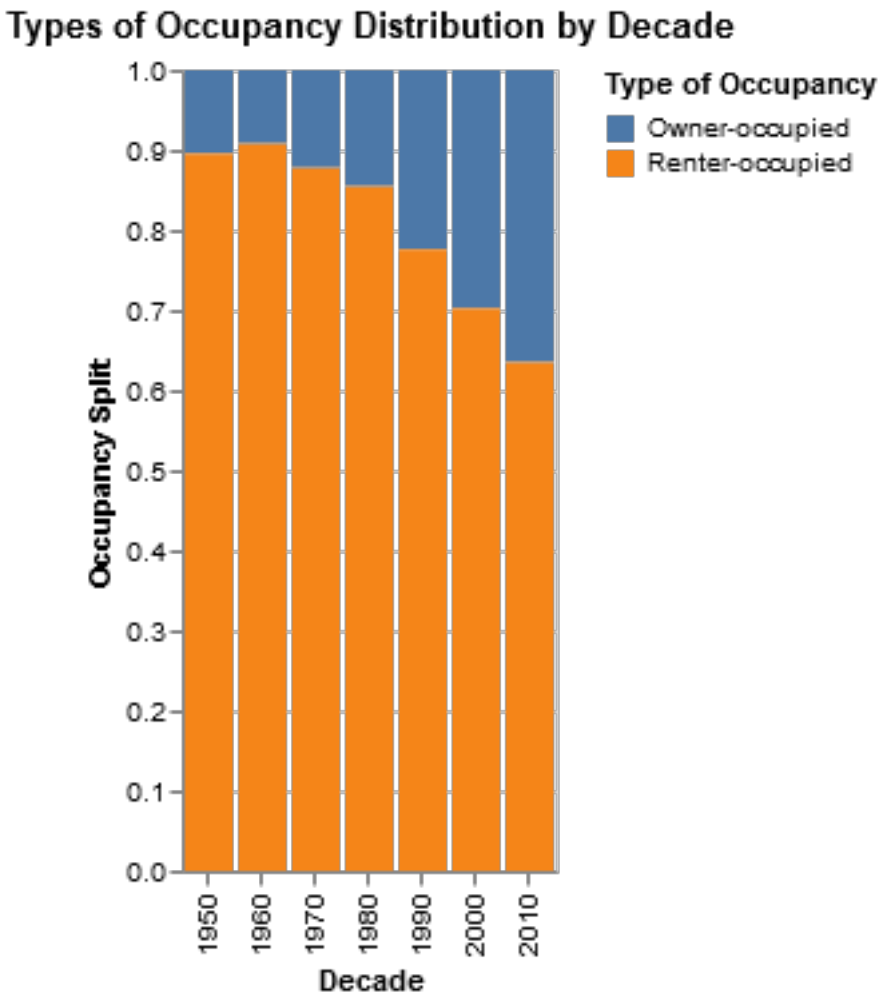


[19]:
```
chart_race = alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y=alt.Y('Count:Q', stack='normalize', title='Demographic Split'),
    color=alt.Color('Subcategory:N', title='Race')
).transform_filter(
    (datum.Category == 'Race/ Ethnicity')
).properties(
    title='Race Distribution by Decade'
)
```

4.4 Types of Occupancy by Decade

[20]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y=alt.Y('Count:Q', stack='normalize', title='Occupancy Split'),
    color=alt.Color('Subcategory:N', title='Type of Occupancy')
).transform_filter(
    (datum.Category == 'Housing Tenure') &
    (datum.Subcategory != 'Occupied Housing Units')
).properties(
    title='Types of Occupancy Distribution by Decade'
)
```

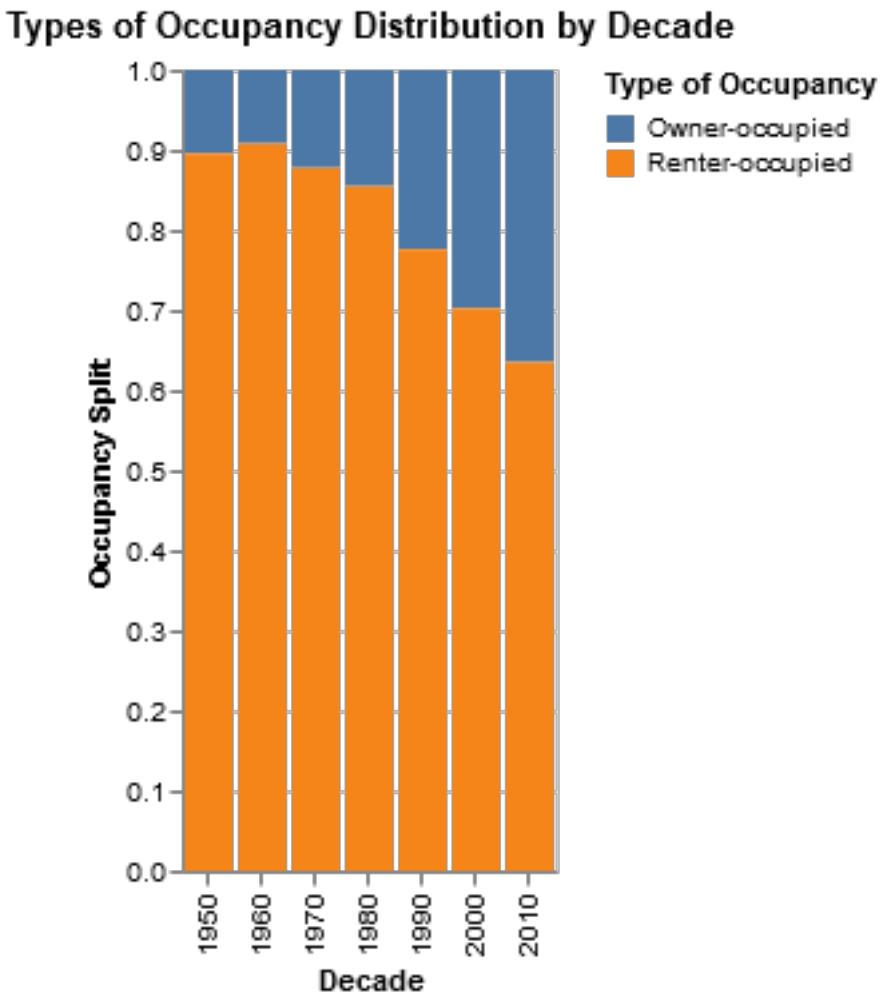
[20]:



5 Saving Your Results

[21]:
```
chart_occupancy = alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y=alt.Y('Count:Q', stack='normalize', title='Occupancy Split'),
    color=alt.Color('Subcategory:N', title='Type of Occupancy')
).transform_filter(
    (datum.Category == 'Housing Tenure') &
    (datum.Subcategory != 'Occupied Housing Units')
).properties(
    title='Types of Occupancy Distribution by Decade'
)
chart_occupancy
```

[21]:

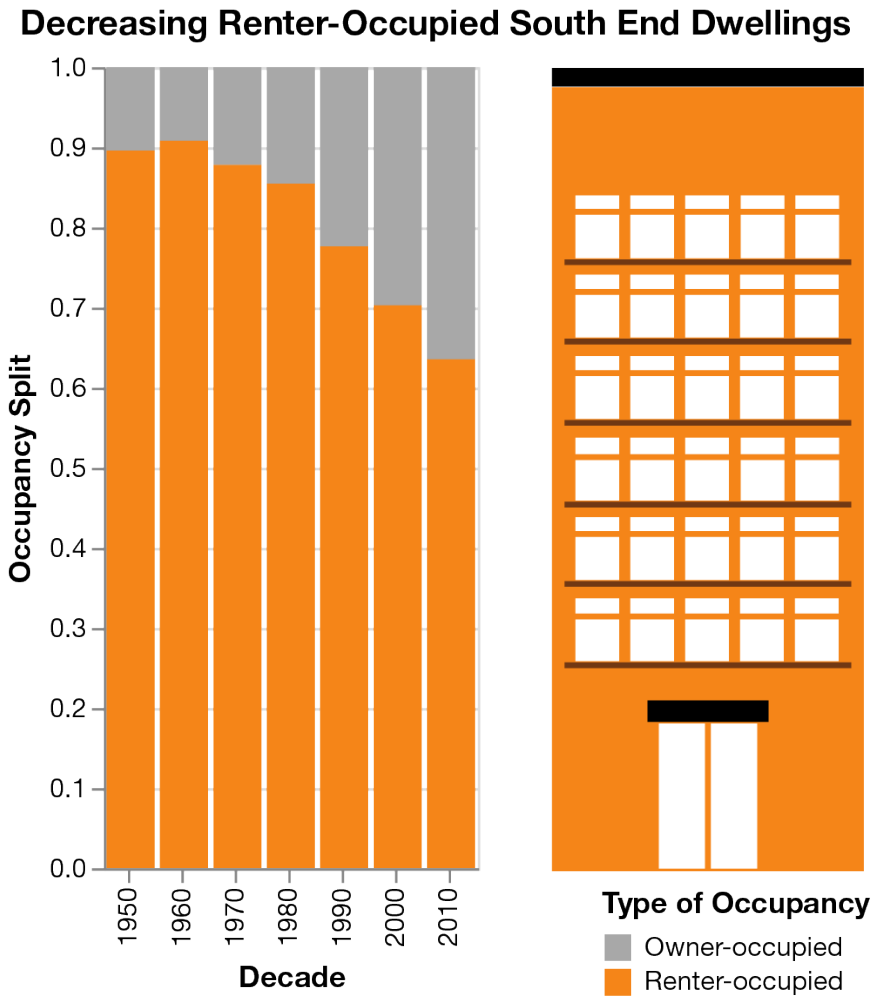


5.1 Saving to a web page...

[22]:
```
chart_occupancy.save('chart_occupancy.html', embed_options={'renderer':'svg'})
```

5.2 Saving an SVG, PNG, Vega Editor...

Here is an example infographic you could create by exporting an SVG and editing in Illustrator:



In Illustrator: * Open the SVG. * Use the magic wand tool to select the owner series & the legend. * Edit->Edit Colors->Saturate and reduce saturation. * Add the apartment_building.svg and scale to ~20%