In [1]:

```python
# imports
import panel as pn
pn.extension('plotly')
import plotly.express as px
import pandas as pd
import hvplot.pandas
import matplotlib.pyplot as plt
import os
from pathlib import Path
from dotenv import load_dotenv
```

In [2]:

```python
# Initialize the Panel Extensions (for Plotly)
import panel as pn
pn.extension("plotly")
```

In [3]:

```python
# Read the Mapbox API key
load_dotenv()
map_box_api = os.getenv("mapbox")
px.set_mapbox_access_token(map_box_api)
```

In [4]:

```python
# Import the CSVs to Pandas DataFrames
file_path = Path("toronto_neighbourhoods_census_data.csv")
to_data = pd.read_csv(file_path, index_col="year")

file_path = Path("toronto_neighbourhoods_coordinates.csv")
df_neighbourhood_locations = pd.read_csv(file_path)
```

In [5]:

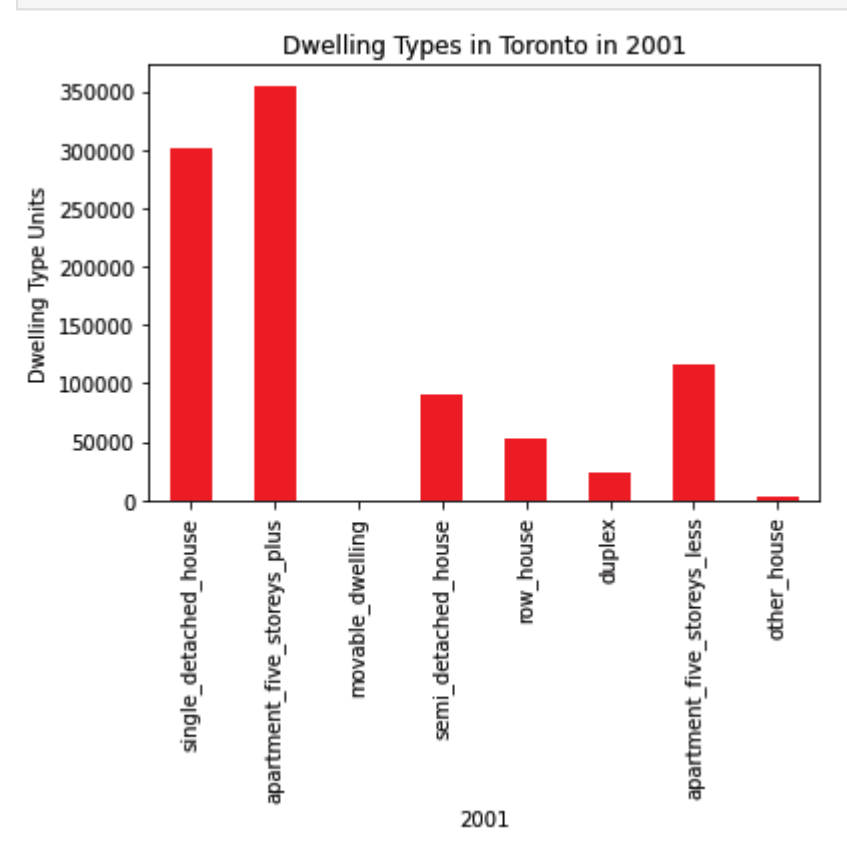
```python
# Getting the data from the top 10 expensive neighbourhoods
dwelling_types = [
    "single_detached_house",
    "apartment_five_storeys_plus",
    "movable_dwelling",
    "semi_detached_house",
    "row_house",
    "duplex",
    "apartment_five_storeys_less",
    "other_house"
]

# Getting the data from the top 10 expensive neighbourhoods
df_expensive_neighbourhoods = to_data.groupby(by="neighbourhood").mean()
df_expensive_neighbourhoods = df_expensive_neighbourhoods.sort_values(
    by="average_house_value", ascending=False
).head(10)
df_expensive_neighbourhoods = df_expensive_neighbourhoods.reset_index()

# Calculate the mean number of dwelling types units per year
df_dwelling_units = to_data[dwelling_types].groupby(to_data.index).sum()

# Calculate the average monthly shelter costs for owned and rented dwellings
df_avg_costs = (
    to_data[["shelter_costs_owned", "shelter_costs_rented"]]
    .groupby([to_data.index])
    .mean()
)
```

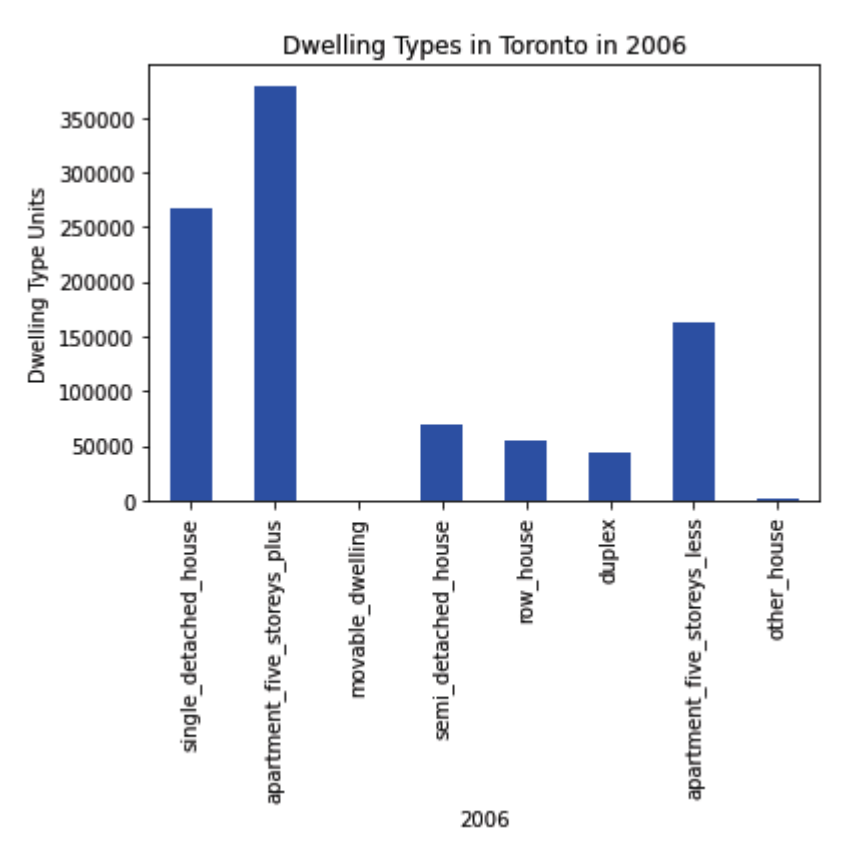
In [6]:

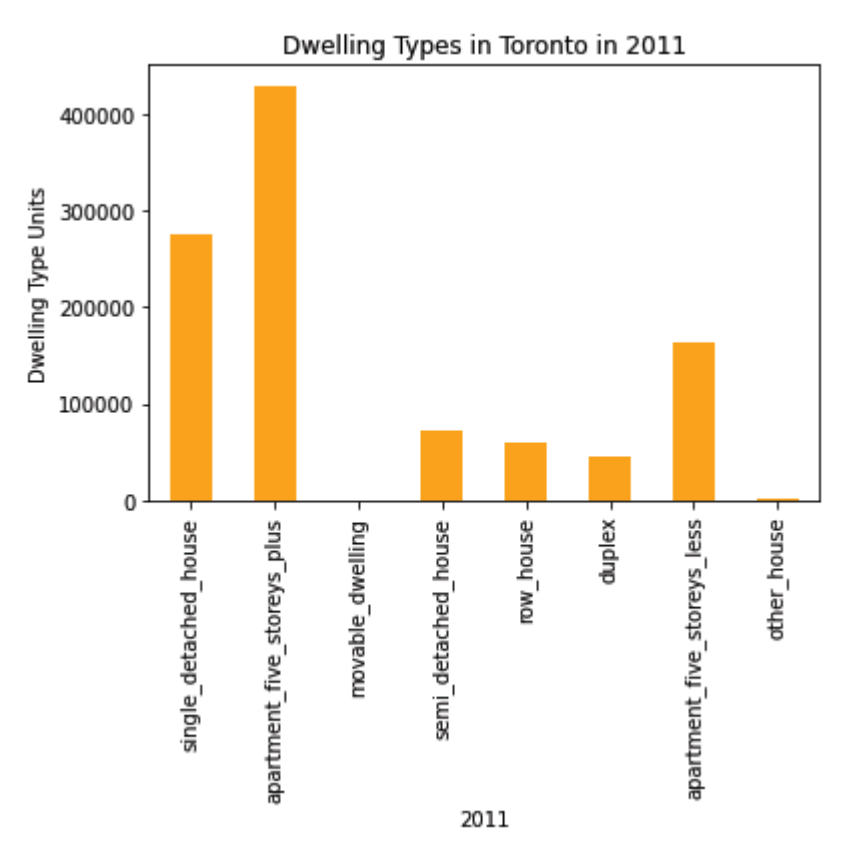
```python
# Define Panel visualization functions
def neighbourhood_map():
    """Neighbourhood Map"""

    df_all_neighbourhoods = to_data.groupby(by="neighbourhood").mean()
    df_all_neighbourhoods = df_all_neighbourhoods.reset_index()

    df_all_neighbourhoods = pd.merge(
        df_neighbourhood_locations, df_all_neighbourhoods, on="neighbourhood"
    )

    map_plot = px.scatter_mapbox(
                                df_all_neighbourhoods,
                                lat='lat',
                                lon='lon',
                                size="average_house_value",
                                color="average_house_value",
                                color_continuous_scale=px.colors.cyclical.IceFire,
                                size_max=15,
                                zoom=9,
                                hover_name="neighbourhood",
                                hover_data= ["shelter_costs_owned",
                                             "shelter_costs_rented",
                                             "single_detached_house",
                                             "apartment_five_storeys_plus",
                                             "movable_dwelling",
                                             "semi_detached_house",
                                             "row_house",
                                             "duplex",
                                             "apartment_five_storeys_less",
                                             "other_house"],
                                height=600
                                )
    ploty_panel=pn.pane.Plotly(map_plot)
    ploty_panel._updates=True
    return ploty_panel


def create_bar_chart(data, title, xlabel, ylabel, color):
    """
    Create a barplot based in the data argument.
    """
    fig = plt.figure()
    bar_chart = data.plot.bar(color=color)
    bar_chart.set_xlabel(xlabel)
    bar_chart.set_ylabel(ylabel)
    bar_chart.set_title(title)
    plt.show()
    plt.close(fig)

    return pn.pane.Matplotlib(fig,tight=True)


def create_line_chart(data, title, xlabel, ylabel, color):
    """
    Create a line chart based in the data argument.
    """

    fig = plt.figure()
    line_chart = data.plot.line(color=color)
    line_chart.set_xlabel(xlabel)
    line_chart.set_ylabel(ylabel)
    line_chart.set_title(title)
    plt.show()
    plt.close(fig)

    return pn.pane.Matplotlib(fig,tight=True)

def average_house_value():
    """Average house values per year."""

    average_house_value = to_data['average_house_value'].groupby('year').mean()
    fig_avg_sale_price = plt.figure(figsize=(8, 4))
    plot_avg_sale_price = average_house_value.plot()
    plot_avg_sale_price.set_xlabel("Year", fontsize=12)
    plot_avg_sale_price.set_ylabel("Avg. House Value", fontsize=12)
    plot_avg_sale_price.set_title(
        "Average House Value in Toronto",
        fontsize=14,
        fontweight="bold",
    )

    # Close the plot
    plt.close(fig_avg_sale_price)

    return pn.pane.Matplotlib(fig_avg_sale_price, tight=True)

def average_value_by_neighbourhood():
    """Average house values by neighbourhood."""

    avg_value_by_neighbourhood = to_data[['neighbourhood','average_house_value']]
    avg_value_by_neighbourhood.reset_index(inplace=True)
    avg_value_by_neighbourhood.rename(columns = {'year':'Year', 'average_house_value':'Avg. House Value'}, inpl
    avg_value_plot = avg_value_by_neighbourhood.hvplot.line(x='Year', y='Avg. House Value', groupby='neighbourh

    return avg_value_plot

def number_dwelling_types():
    """Number of dwelling types per year"""

    all_dwelling_types = to_data.reset_index()
    all_dwelling_types.rename(columns = {'year':'Year'}, inplace = True)
    all_dwelling_types_plot = all_dwelling_types.hvplot.bar(x='Year', ylabel= "Dwelling Type Units", groupby='n

    return all_dwelling_types_plot

def top_most_expensive_neighbourhoods():
    """Top 10 most expensive neighbourhoods."""
    expensive_neighbourhoods = df_expensive_neighbourhoods.hvplot.bar(
                                                "neighbourhood",
                                                 "average_house_value",
                                                 title="Top 10 Expensive Neighbourhoods
                                                 xlabel="Neighbourhood",
                                                 ylabel="Avg. House Value",
                                                 height=500,
                                                 rot=90,
                                                 ylim=(0, 1600000)
                                                 )






    return expensive_neighbourhoods
```

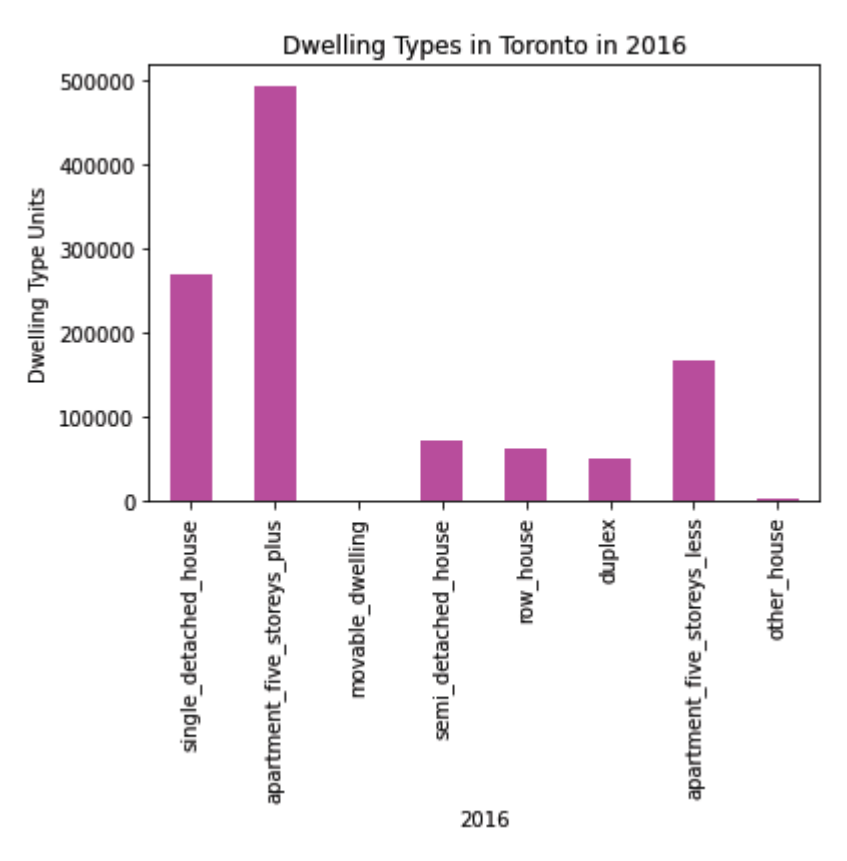
In [7]:

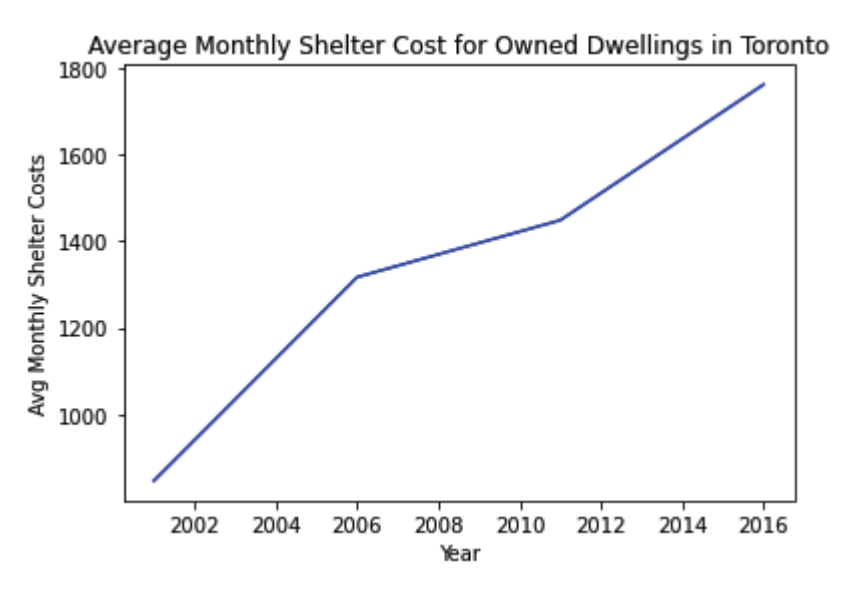
```python
# Create a Title for the Dashboard
title = pn.pane.Markdown(
    """
# Toronto Real Estate Analysis from 2001 through 2016
""",
    width=800,
)

# Define a welcome text
welcome = pn.pane.Markdown(
    """
This dashboard displays a visual analysis of historical house values,
dwelling types per neighbourhood and dwelling costs in Toronto, Ontario
using quinquennial census data from 2001 to 2016. Users can navigate through the tabs above
to explore more details about the evolution of the real estate market in The Six Boroughs through the years.
""",
    width=1024,
)

# Create a tab layout for the dashboard
tabs = pn.Tabs(
    # First tab
    (
        "Welcome",
        pn.Column(
            welcome,
            neighbourhood_map()
        )
    ),
    # Second tab
    (
        "Periodic Market Analysis",
        pn.Column(
                pn.Row(
                    create_bar_chart(
                        df_dwelling_units.loc[2001],
                        "Dwelling Types in Toronto in 2001",
                        "2001",
                        "Dwelling Type Units",
                        "red"
                    ),
                    create_bar_chart(
                        df_dwelling_units.loc[2006],
                        "Dwelling Types in Toronto in 2006",
                        "2006",
                        "Dwelling Type Units",
                        "blue"
                    ),
                ),
                pn.Row(
                    create_bar_chart(
                        df_dwelling_units.loc[2011],
                        "Dwelling Types in Toronto in 2011",
                        "2011",
                        "Dwelling Type Units",
                        "orange"
                    ),
                    create_bar_chart(
                        df_dwelling_units.loc[2016],
                        "Dwelling Types in Toronto in 2016",
                        "2016",
                        "Dwelling Type Units",
                        "magenta"
                    ),
                )
        )
    ),
    # Third tab
    (
        "Shelter Costs Trends",
        pn.Column(
            create_line_chart(
                df_avg_costs["shelter_costs_owned"],
                "Average Monthly Shelter Cost for Owned Dwellings in Toronto",
                "Year",
                "Avg Monthly Shelter Costs",
                "blue"
            ),
            create_line_chart(
                df_avg_costs["shelter_costs_rented"],
                "Average Monthly Shelter Cost for Rented Dwellings in Toronto",
                "Year",
                "Avg Monthly Shelter Costs",
                "orange"
            ),
            average_house_value(),
        ),
    ),
    # Fourth Tab
    (
        "Neighbourhood Analysis",
        pn.Row(
            pn.Column(
                average_value_by_neighbourhood(),
                number_dwelling_types(),
            )
        )
    ),
    # Fifth tab
    (
        "Top 10 Expensive Neighbourhoods",
        pn.Row(
            top_most_expensive_neighbourhoods(),

        )
    )

)


# Create the main dashboard
dashboard = pn.Column(
    pn.Row(title),
    tabs,
    width=1024,
    height=768
    )
```

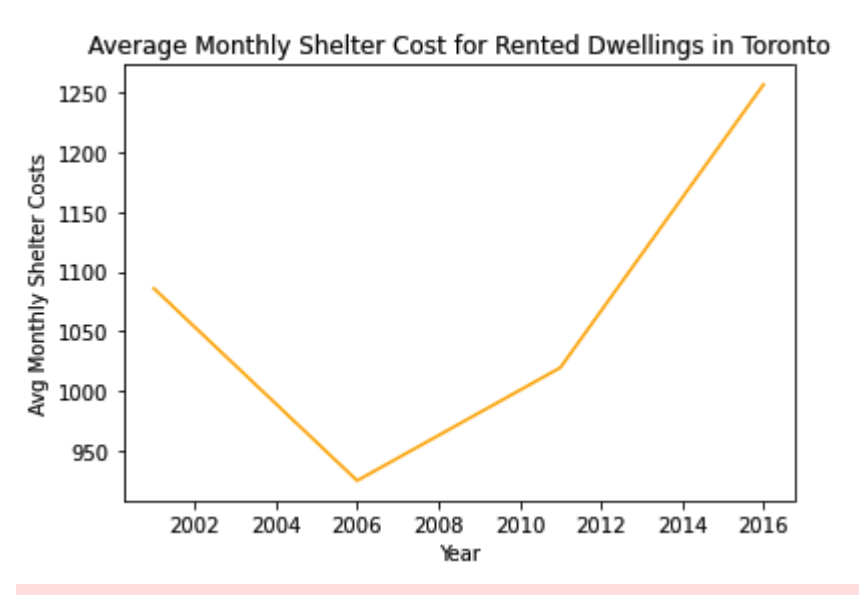













```
C:\Users\Owner\AppData\Local\Temp\ipykernel_8784\1549317382.py:93: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret
urning-a-view-versus-a-copy
```

In [8]:

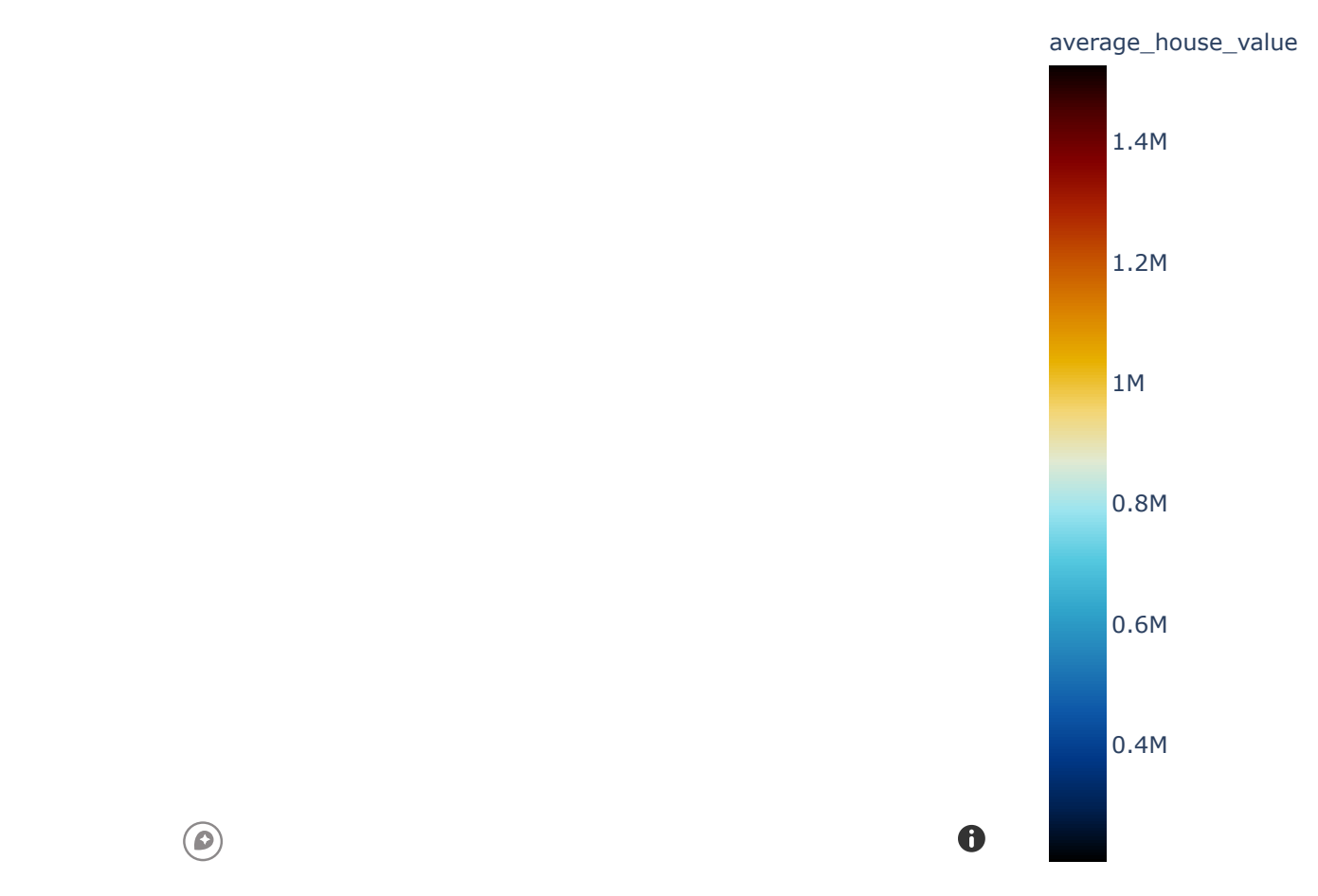
```python
dashboard.servable()
```

Out[8]:

Toronto Real Estate Analysis from 2001 through 2016

Welcome | Periodic Market Analysis | Shelter Costs Trends | Neighbourhood Analysis | Top 10 Expensive Neighbourhoods

This dashboard displays a visual analysis of historical house values, dwelling types per neighbourhood and dwelling costs in Toronto, Ontario using quir
from 2001 to 2016. Users can navigate through the tabs above to explore more details about the evolution of the real estate market in The Six Borough:



In [ ]: