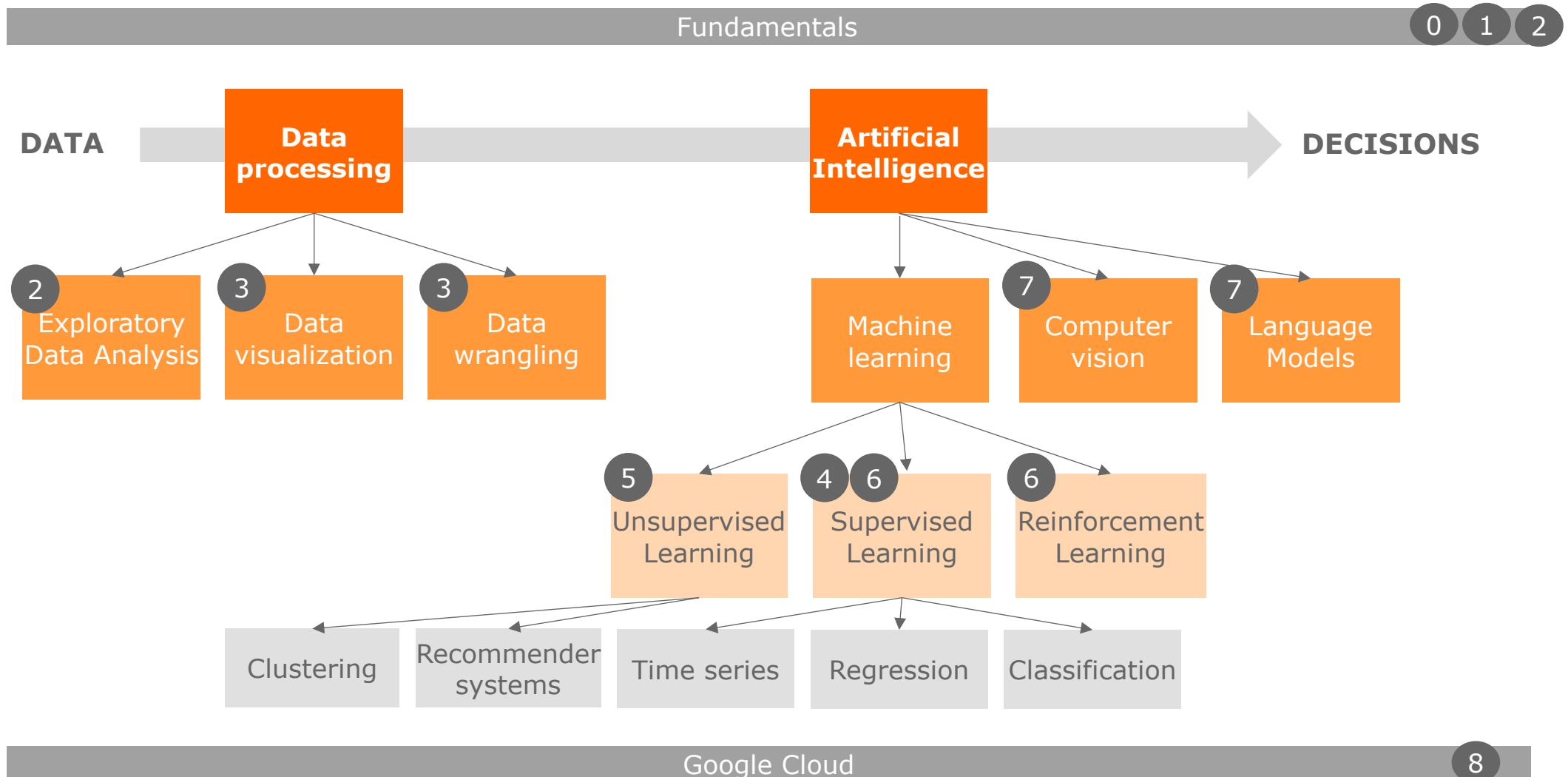
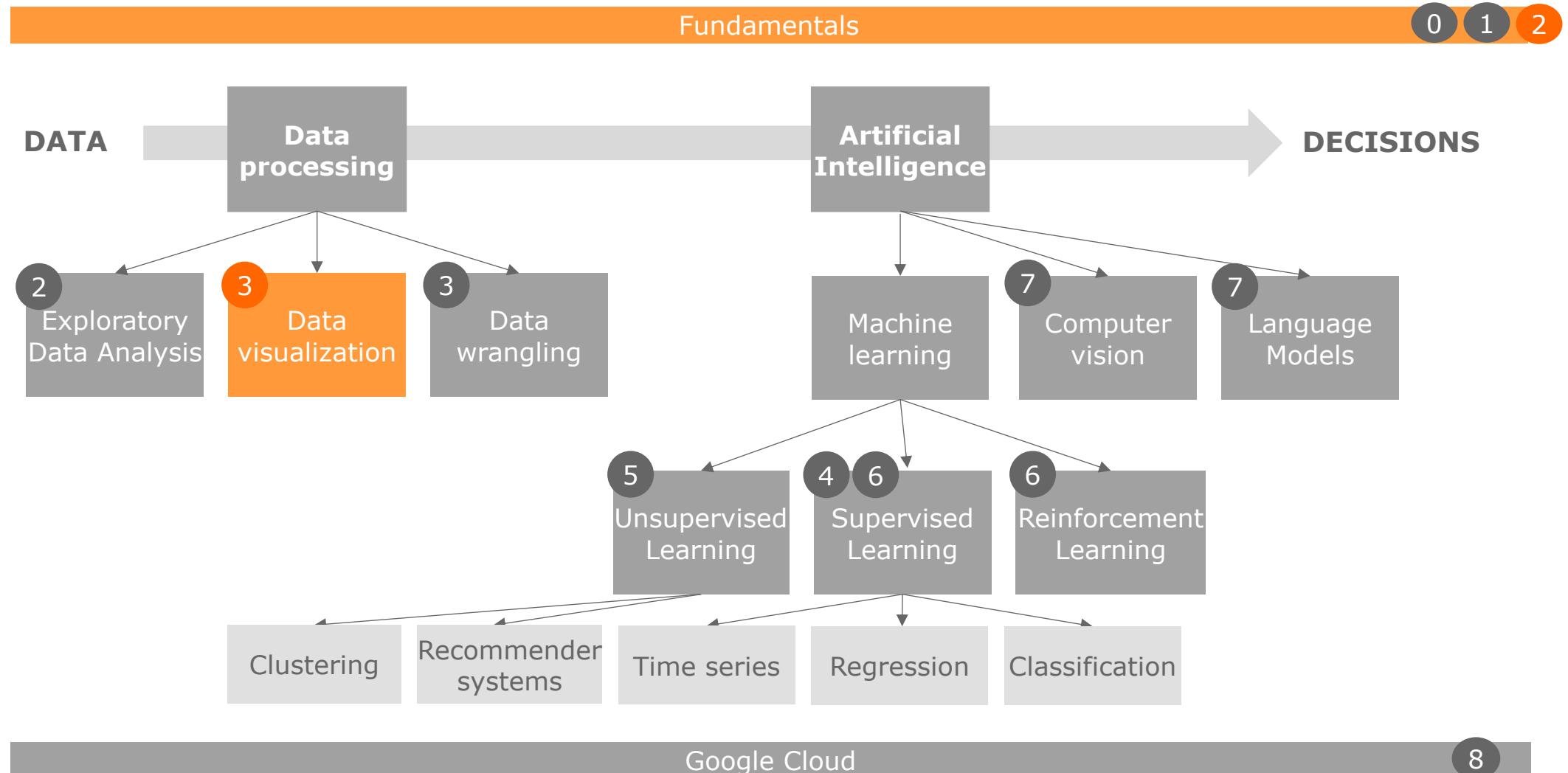


Advanced Topics in Dataviz

Sub-título

Mário Amorim Lopes/Fábio Neves Moreira





Data Wrangling and Data Visualization

Data Engineering

- Different types of data (unstructured, semi-structured and structured data)
- Data importing
- Data cleaning
- Data transformation (date parsing, character encodings, etc.)

Data Visualization

- Visualizing data in Python using Seaborn
- Line charts
- Bar charts and heat maps
- Scatter plots
- Histograms and density plots

EDA in Python

- Building a data processing pipeline (import, transform, visualize)
- Descriptive statistics of a dataset

Adv. Topics in Dataviz

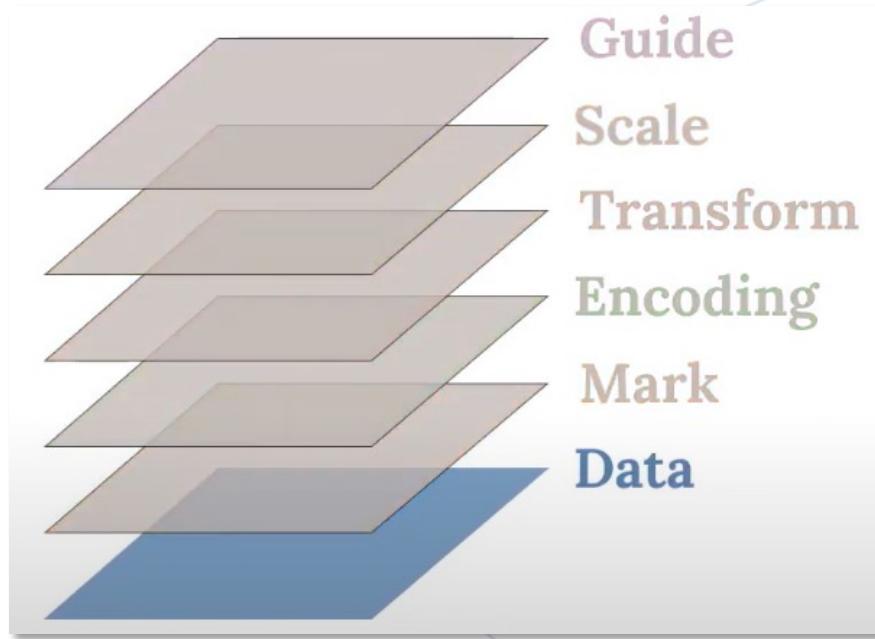
- Examples of advanced data visualization
- Univariate visualization
- Multivariate visualization
- Whole dataset visualisations

Data visualization with ALTAIR



- ✓ Altair uses a simple visualization grammar
- ✓ The grammar is broken down into several components
- ✓ A complete user guide can be found at https://altair-viz.github.io/user_guide/data.html

Data visualization with ALTAIR

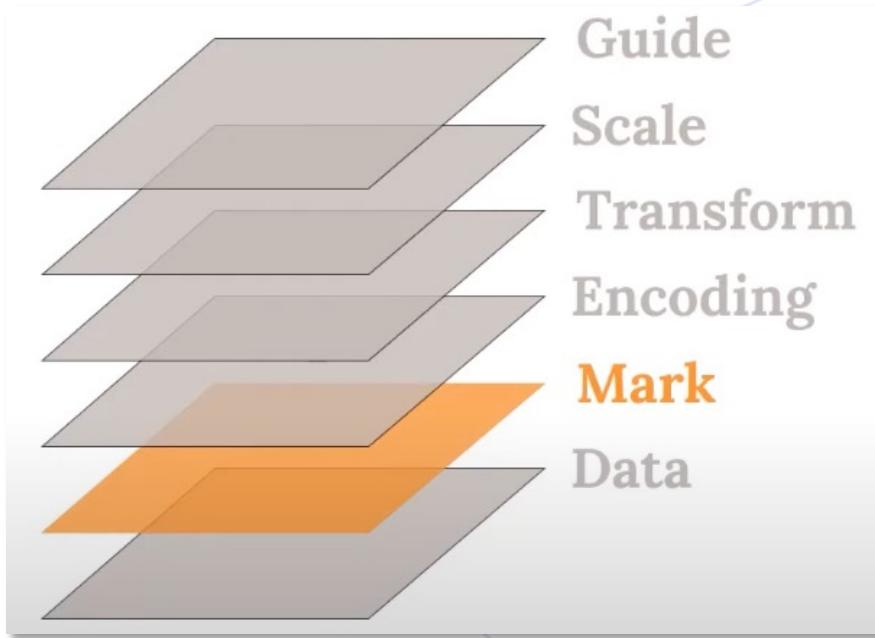


- ✓ Altair considers data in a tabular format
- ✓ The data can be a csv, json, or even a url pointing to the data

```
A,B,C,D,E
4,6,4,4,3
4,4,8,4,3
7,5,5,0,1
5,9,3,0,5
0,1,2,4,2
```

```
[{"A":4, "B":6, "C":4, "D":4, "E":3},
 {"A":4, "B":4, "C":8, "D":4, "E":3},
 {"A":7, "B":5, "C":5, "D":0, "E":1},
 {"A":5, "B":9, "C":3, "D":0, "E":5},
 {"A":0, "B":1, "C":2, "D":4, "E":2}]
```

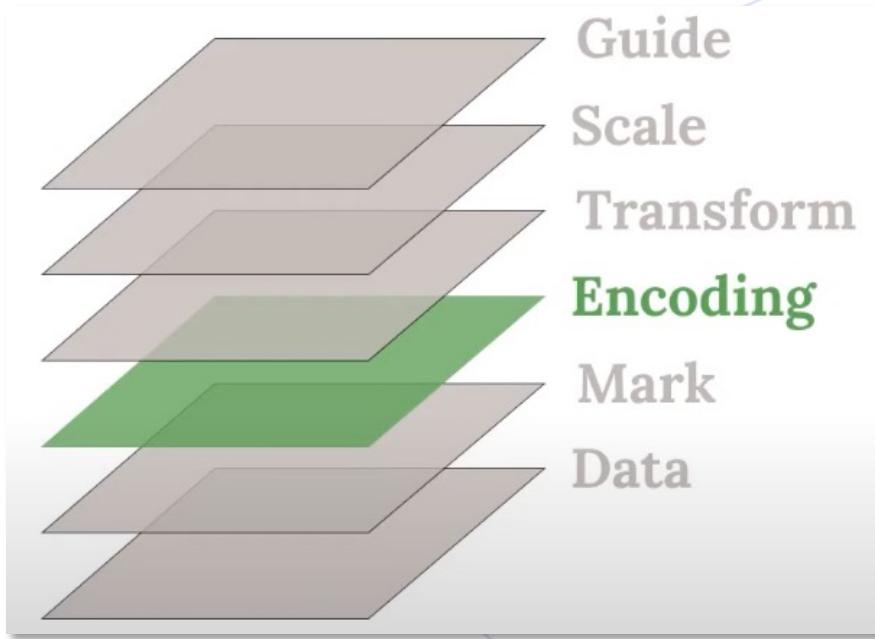
Data visualization with ALTAIR



- ✓ Marks are the geometric objects to be drawn on the screen
- ✓ Altair has many geometric primitives including Line, Point, Area, and Bar



Data visualization with ALTAIR



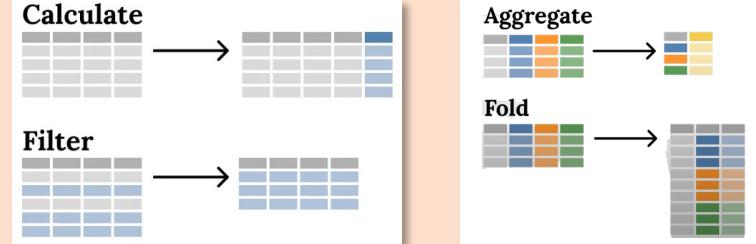
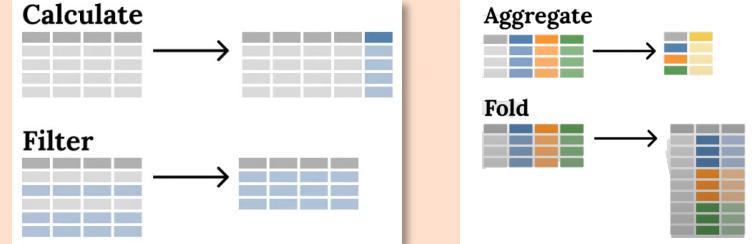
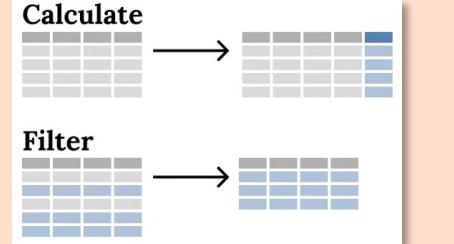
- ✓ Encodings are mappings between the visual properties of a mark and the variables of the data
- ✓ Visual properties or encoding channels include X position, Y position, Color, Size, (...)

Channel	Variable
X Position	A
Y Position	B
Color	C
Size	D

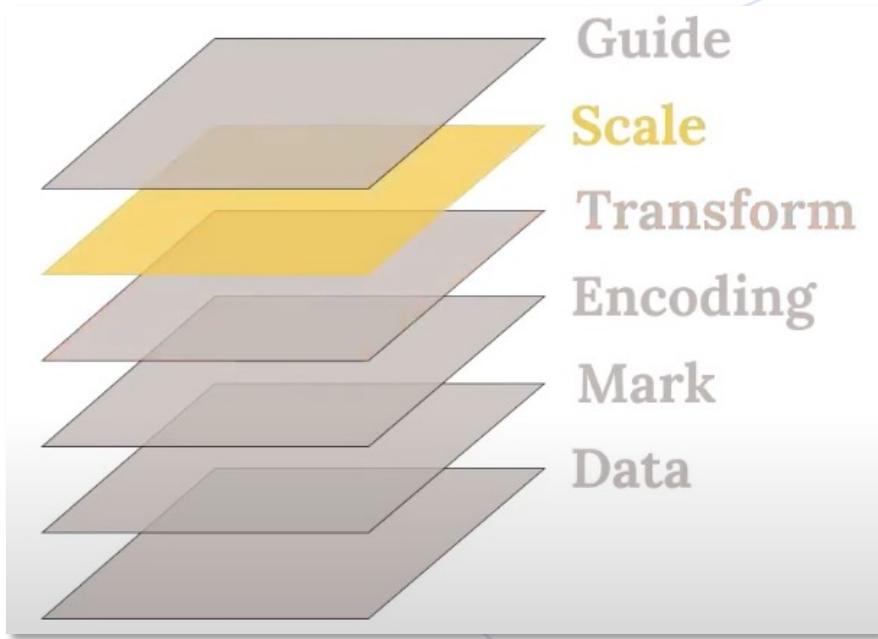
Data visualization with ALTAIR



- ✓ Altair can also perform several transformations on the data before offering the visualization
- ✓ Variables can be calculated, filters can be applied, and even advanced calculations such as aggregations and folds



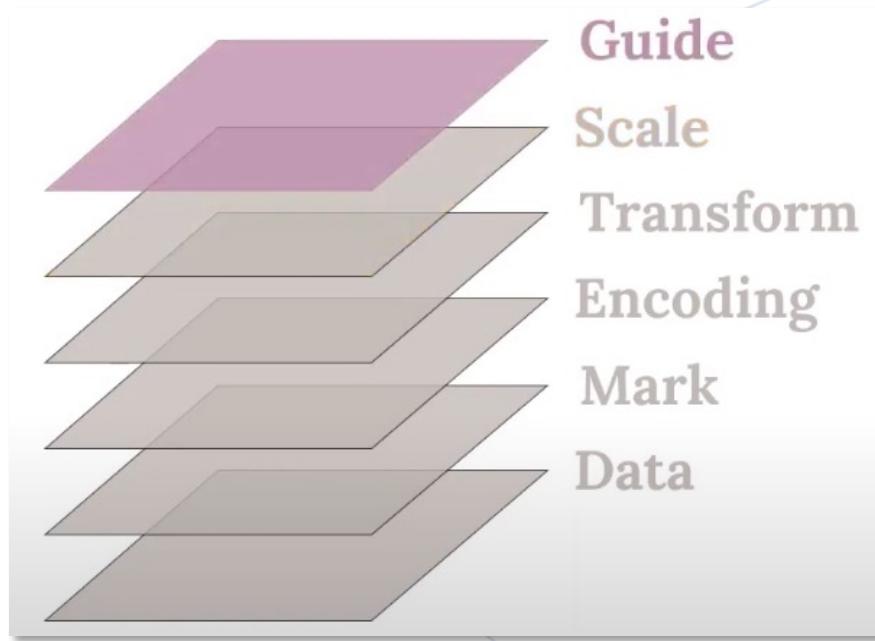
Data visualization with ALTAIR



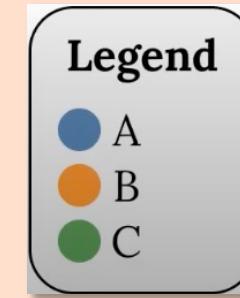
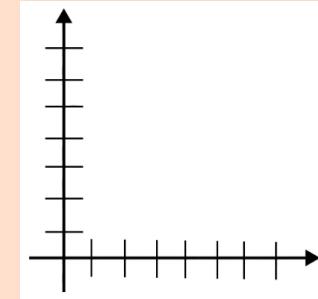
- ✓ Scales are nothing more than a function that uses data as an input and outputs pixels on the screen
- ✓ For instance, the scale determines where to draw the marks or what color to make them based on the encodings

$$f_{(\text{domain})} \longrightarrow \text{range}$$
$$f_{(\text{grid icon})} \longrightarrow \text{computer monitor icon}$$

Data visualization with ALTAIR



- ✓ Guides are useful visual aids often to visualize a scale
- ✓ For example, axis and legends are guides that typically added automatically by Altair



ALTAIR viz building examples

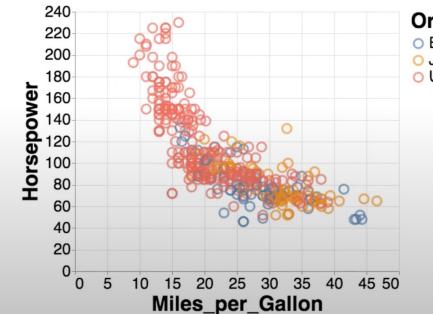


- ✓ Examples can be built by stacking these grammar layers from bottom up.

```
import altair as alt
from vega_datasets import data

cars = data.cars()

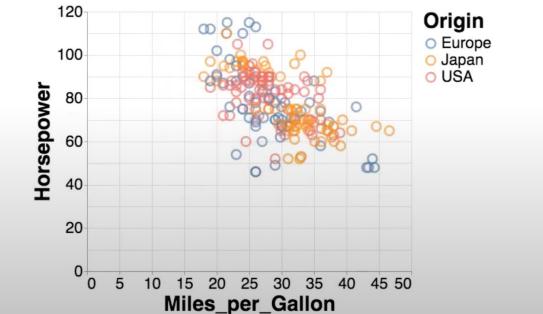
alt.Chart(cars).mark_point().encode(
    x ='Miles_per_Gallon',
    y = 'Horsepower',
    color = 'Origin'
)
```



```
import altair as alt
from vega_datasets import data

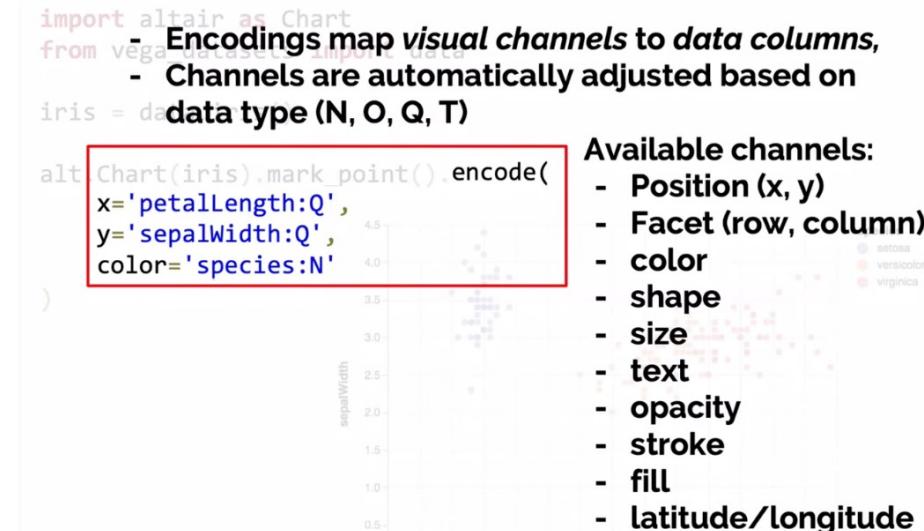
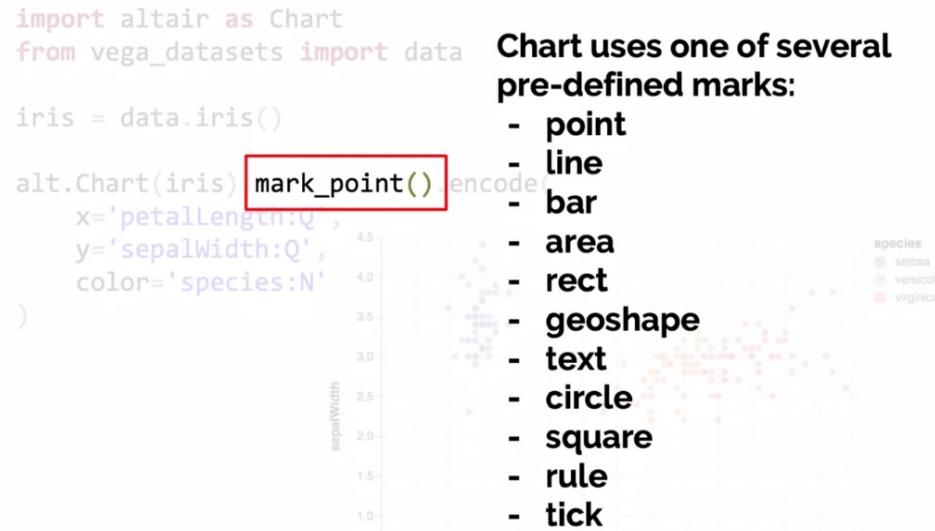
cars = data.cars()

alt.Chart(cars).mark_point().encode(
    x ='Miles_per_Gallon',
    y = 'Horsepower',
    color = 'Origin'
).transform_filter(alt.datum.Cylinders < 5)
```



ALTAIR viz building examples

- ✓ Several types of marks can be used in the chart
- ✓ Several channels are available to tweak your charts

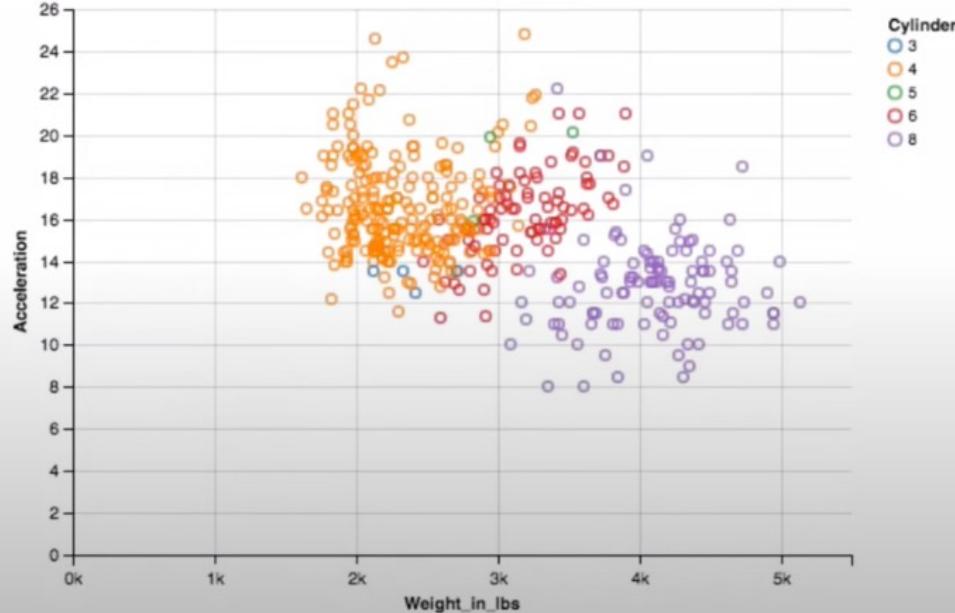


N (nominal), O (ordinal), Q (quantitative), T (temporal)

ALTAIR viz building examples

Example 1

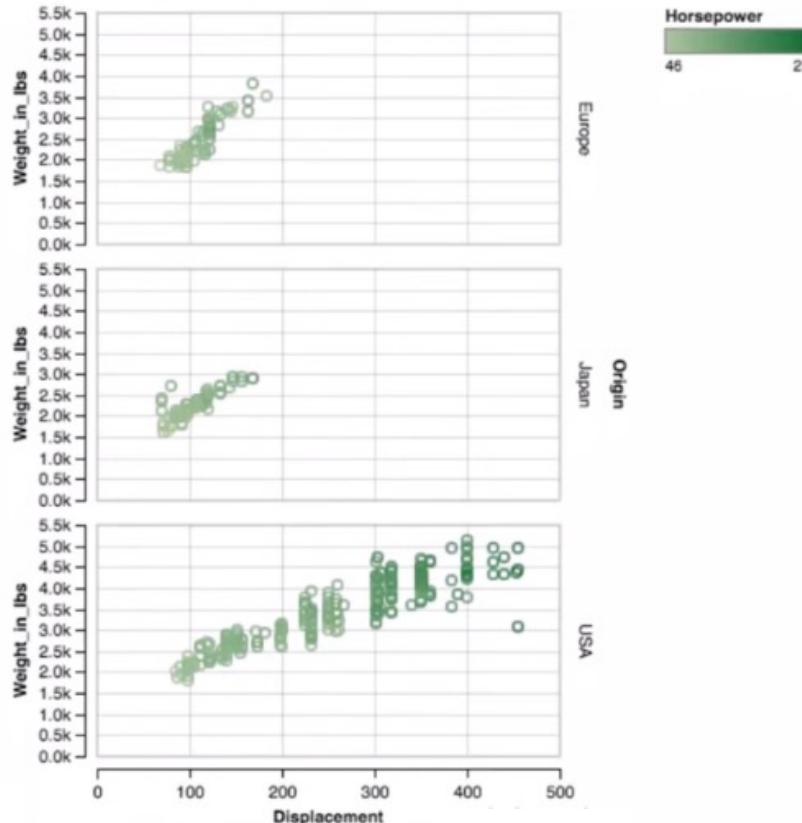
```
In [4]: Chart(cars).mark_point().encode(  
    x='Weight_in_lbs',  
    y='Acceleration',  
    color='Cylinders:N'  
)
```



ALTAIR viz building examples

Example 2

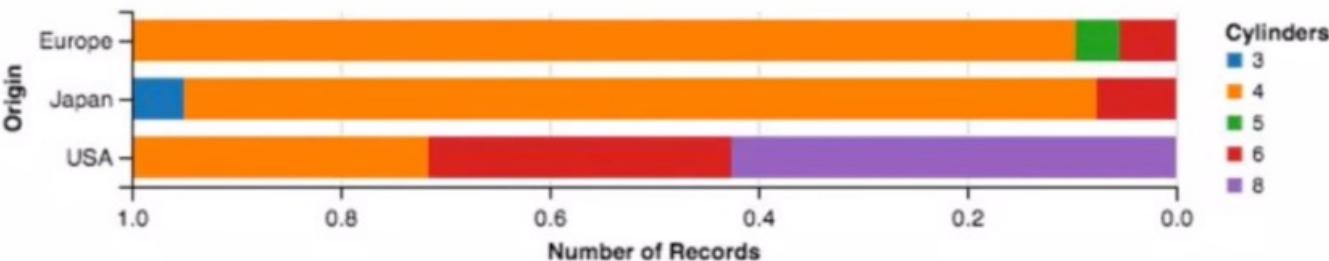
```
In [6]: Chart(cars).mark_point().encode(  
    x='Displacement',  
    y='Weight_in_lbs',  
    color='Horsepower',  
    row='Origin'  
) .configure_cell(width=300, height=150)
```



ALTAIR viz building examples

Example 3

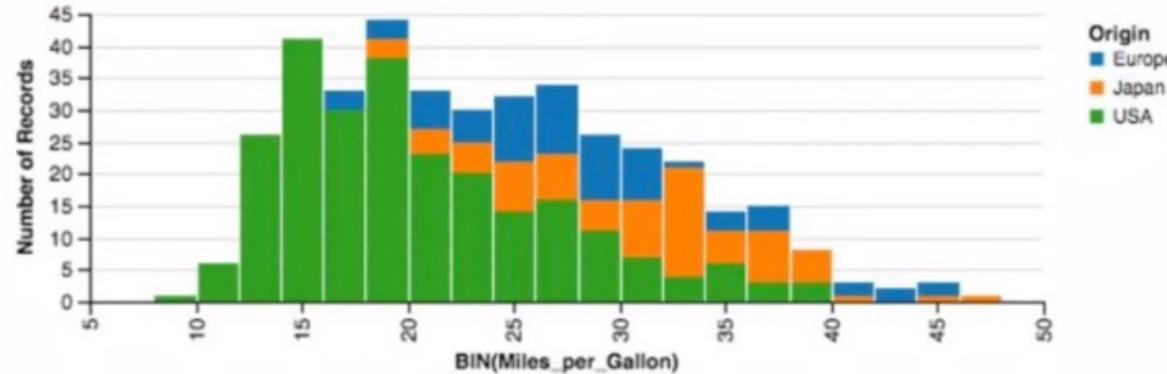
```
In [7]: Chart(cars).mark_bar(stacked='normalize').encode(  
    Y('Origin'),  
    X('*:Q', aggregate='count', sort='descending'),  
    Color('Cylinders:N')  
)
```



ALTAIR viz building examples

Example 4

```
In [8]: Chart(cars).mark_bar().encode(  
    X('Miles_per_Gallon', bin=Bin(maxbins=20)),  
    Y('*:Q', aggregate='count'),  
    Color('Origin')  
) .configure_cell(height=150)
```



ALTAIR viz building examples

Example 5

```
In [9]: Chart(cars).mark_text(applyColorToBackground=True).encode(  
    Row('Origin:O'),  
    Column('Cylinders:O'),  
    Color('mean(Miles_per_Gallon):Q', sort='descending'),  
    Text('mean(Miles_per_Gallon):Q')  
)
```

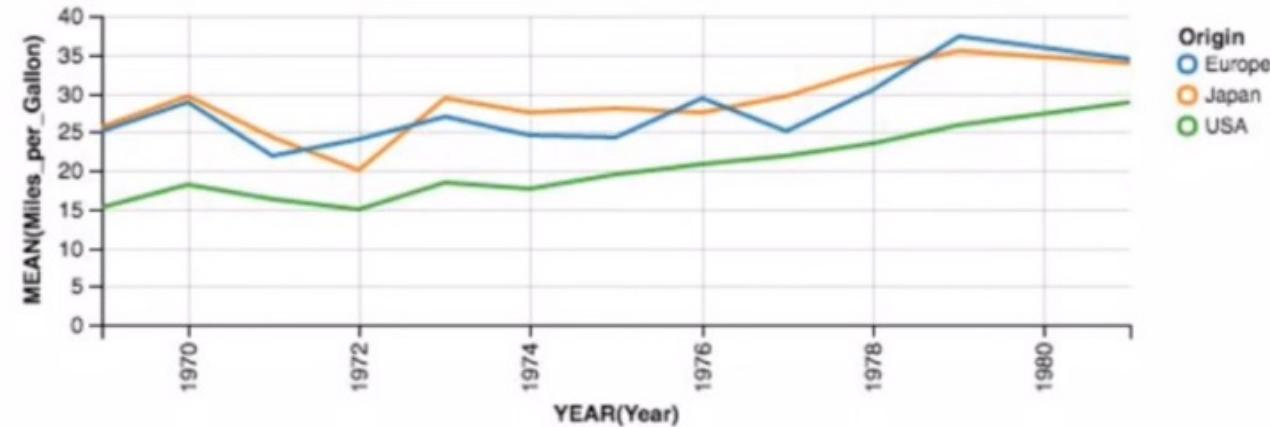


ALTAIR viz building examples

Example 7

In [10]:

```
Chart(cars).mark_line().encode(  
    X('Year:T', timeUnit='year'),  
    Y('Miles_per_Gallon:Q', aggregate='mean'),  
    Color('Origin:N')  
) .configure_cell(height=150)
```

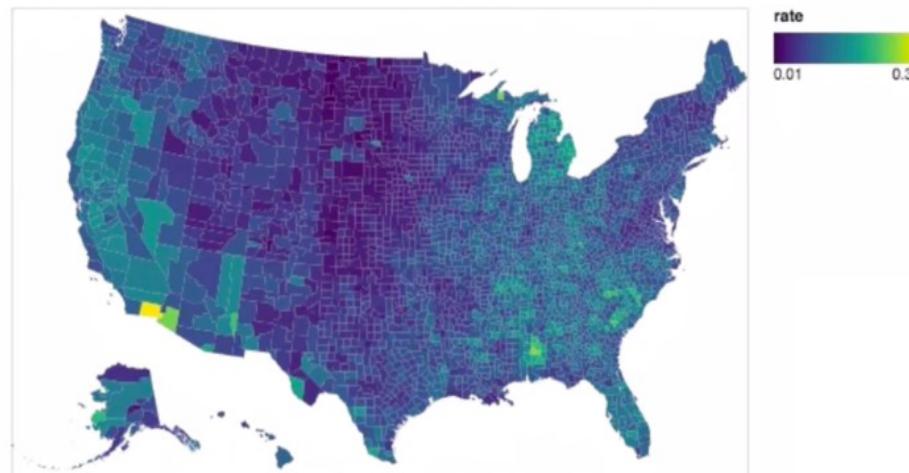


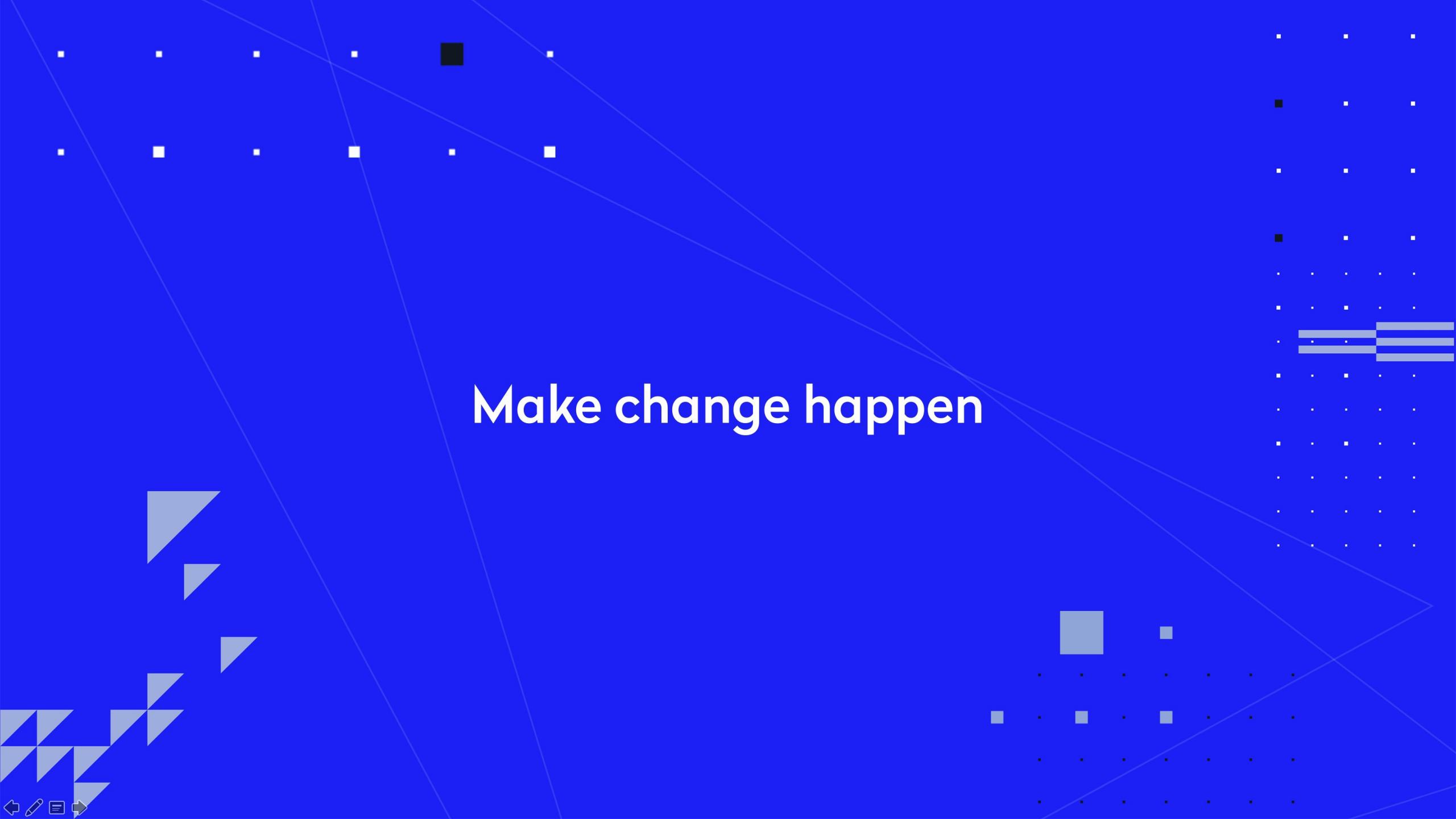
ALTAIR viz building examples

Example 8

```
In [11]: counties = alt.topo_feature(data.us_10m.url, 'counties')
unemp_data = data.unemployment.url

alt.Chart(counties).mark_geoshape().properties(
    projection={'type': 'albersUsa'},
    width=500, height=300
).encode(
    color='rate:Q'
).transform_lookup(
    lookup='id', from_=alt.LookupData(unemp_data, 'id', ['rate'])
)
```





Make change happen