SI649-F22 -> Altair II

Overview

We will continue working with the dataset from the article “The Dollar-And-Cents Case Against Hollywood’s Exclusion of Women” (https://fivethirtyeight.com/features/the-dollar-and-cents-case-against-hollywoods-exclusion-of-women/). We'll focus on **transformation** and build the following charts:

1. Line chart within a range
2. Heatmap with additional annotations
3. Bar chart with additional annotations
4. Bar chart with fold

After building these charts, you will make a website with these charts using streamlit.

Lab Instructions

- For this lab, please write Altair code to answer the questions. In many situations you could also solve the problem using Pandas. However, we want code that can be deployed without using Python so it's better practice to just do as much as we can in Altair. You can complete the entire lab without writing any pandas transformation.
- It's fine if your visualization looks slightly different from the example (e.g., getting 1.1 instead of 1.0, use orange instead of red)
- Save, rename, and submit the ipynb file (use your username in the name).
- Run every cell (do Runtime -> Restart and run all to make sure you have a clean working version), print to pdf, rename, submit the pdf file.
- For each visualization, we will ask you to write down a "Grammar of Graphics" plan first (basically a description of what you'll code).
- If you end up stuck, show us your work by including links (URLs) that you have searched for. You'll get partial credit for showing your work in progress.
- There are many bonus point opportunities in this lab.

In [1]:

```
# imports we will use
import altair as alt
import pandas as pd
datasetURL="https://raw.githubusercontent.com/eytanadar/si649public/master/lab5/assets/hw/movie_a
movies_test=pd.read_csv(datasetURL, encoding="latin-1")
```

Here's the processed Bechdel test dataset.

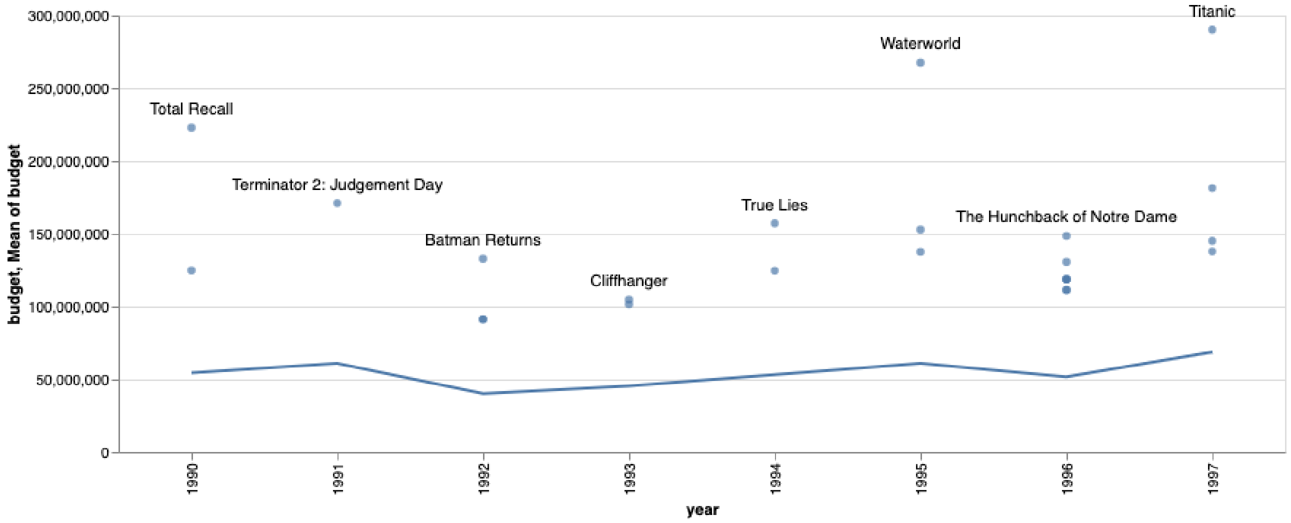
```
In [2]: movies_test
```

```
Out[2]:
```

	Unnamed: 0	year	title	binary	budget	dom_gross	int_gross	rating	country	language	test_result	country_bir
0	0	2013	21 &amp; Over	FAIL	13000000	25682380.0	42195766.0	5.9	United States	English	Women don't talk to each other	U.S. Can
1	1	2012	Dredd 3D	PASS	45658735	13611086.0	41467257.0	7.1	United Kingdom	English	Passes Bechdel Test	Internati
2	2	2013	12 Years a Slave	FAIL	20000000	53107035.0	158607035.0	8.1	United States	English	Women don't talk to each other	U.S. Can
3	3	2013	2 Guns	FAIL	61000000	75612460.0	132493015.0	6.7	United States	English	Women don't talk to each other	U.S. Can
4	4	2013	42	FAIL	40000000	95020213.0	95020213.0	7.5	United States	English	Women only talk about men	U.S. Can
...	...	...	...	...	...	...	...	...	...	...	...	
1610	1610	1990	Predator 2	FAIL	62404139	50489429.0	97650742.0	6.3	United States	English	Women only talk about men	U.S. Can
1611	1611	1990	Pretty Woman	PASS	24961656	318093987.0	771396947.0	7.0	United States	English	Passes Bechdel Test	U.S. Can
1612	1612	1990	The Hunt for Red October	FAIL	53489262	215222722.0	357486568.0	7.6	United States	English	Women don't talk to each other	U.S. Can
1613	1613	1990	Total Recall	FAIL	222871925	104977970.0	354127435.0	7.5	United States	English	Women don't talk to each other	U.S. Can
1614	1614	1990	Tremors	PASS	17829754	29717001.0	29717001.0	7.2	United States	English	Passes Bechdel Test	U.S. Can

1615 rows × 14 columns

Visualization 1: Line chart within a range



Description of the visualization:

We want to see a visualization of movies that were significantly above budget relative to the mean. We'd like to know the name of the top movie that year but also to have a sense of how many other outliers (we'll defined those as at least 2x the mean budget) were produced that year.

- Plot a line chart using year and average budget for movies between 1990 and 1997.
- Plot dots to represent movies with budgets that are at least 2 times bigger than the mean budget of that year. (e.g., if the mean budget of 1990 is 50M, plot movies made in 1990 whose budgets are at least 100M)
- For each year, annotate the title of the movie with the highest budget of that year.

Visualization 1 Plan:

TODO: edit this cell to write your visualization plan. You can write in altair syntax, in full sentence, or in bullet points, whichever way that helps you to plan your chart.

line chart

- Describe the encoding rules:
  - mark_line()
  - alt.X('year:O')
  - alt.Y('average(budget):Q')
- Describe the transformations:
  - transform_joinaggregate(groupby=['year'], averageBudget='average(budget)', MaxBudget='max(budget)')
  - transform_filter((alt.datum.budget > (2 * alt.datum.averageBudget)) & (alt.datum.year >= 1990) & (alt.datum.year <= 1997))

dot chart

- Describe the encoding rules:
  - mark_circle()
  - alt.X('year:O')
  - alt.Y('averageBudget:Q')
- Describe the transformations:
  - transform_joinaggregate(groupby=['year'], averageBudget='average(budget)', MaxBudget='max(budget)')
  - transform_filter((alt.datum.budget > (2 * alt.datum.averageBudget)) & (alt.datum.year >= 1990) & (alt.datum.year <= 1997))

text annotation

- Describe the encoding rules:
  - alt.Text('title:N')
- Describe the transformations:
  - dot_chart.transform_filter(alt.datum.budget == alt.datum.MaxBudget)

Replicate Vis 1

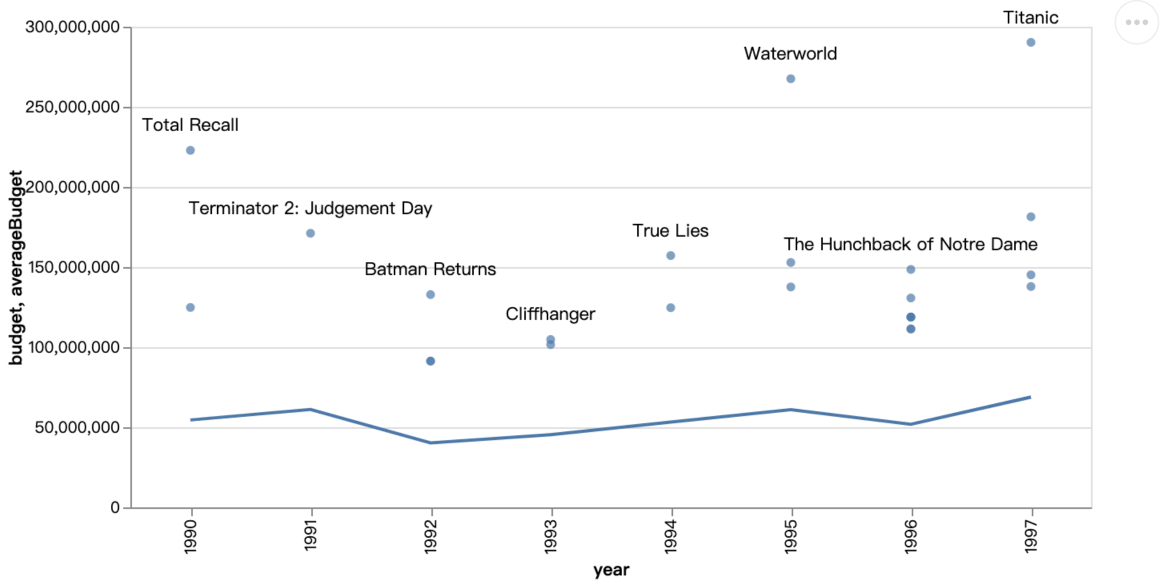
Hint:

- Line chart is the simplest one, you can start from there.
- Think about the difference between transform_aggregate and transform_joinaggregate

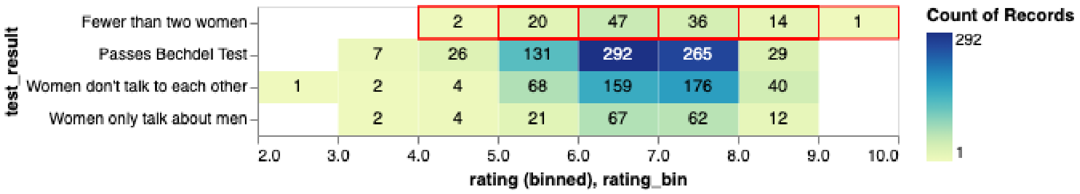
In [3]:

```
#TODO: Replicate visualization 1


base = alt.Chart(movies_test).transform_joinaggregate(
    groupby=['year'],
    averageBudget='average(budget)',
    MaxBudget='max(budget)'
).transform_filter(
    (alt.datum.budget > (2 * alt.datum.averageBudget)) & (alt.datum.year >= 1990 ) & (alt.datum.y
)

dot_chart = base.mark_circle().encode(
    alt.X('year:O'),
    alt.Y('budget:Q')
)

line_chart = base.mark_line().encode(
    alt.X('year:O'),
    alt.Y('averageBudget:Q')
)

text = dot_chart.transform_filter(
    alt.datum.budget == alt.datum.MaxBudget
).mark_text(
    align = 'center',
    dy= -15  # delta x = 7 pixel aside
).encode(alt.Text('title:N'))


(dot_chart + line_chart + text).properties(width=600, height=300)
```

Out[3]:



Visualization 2: heatmap with additional annotations



Description of this visualization:

We want to produce a heatmap contrasting different types of bechdel test results to the IMDB scores of those movies. Each cell in the heatmap will tell us how many movies (not normalized here, we want raw count) are in that category. The author of the article wants to point out some property of the really well-regarded movies so wants those highlighted.

- Plot a heatmap with rating (binned) and test result. Encode the count of movies as color. Remove the category "dubious".
- For each cell, add text to indicate the count of movies.
- If a test_result category has at least one movie whose rating is higher than 9, highlight that entire category in red. You can highlight by adding a layer of heatmap that is not filled.
- BONUS: For the darkest two cells (292 and 265) change the color of the annotation to white

Visualization2 Plan:

TODO: edit this cell to write your visualization plan. You can write in altair syntax, in full sentence, or in bullet points, whichever way that helps you to plan your chart.

heatmap

- Describe the encoding rules:
  - x='binned_rating:O'
  - y='test_result:N'
- Describe the transformations:
  - transform_bin('binned_rating', 'rating')
  - transform_aggregate(groupby=['binned_rating', 'test_result'], count='count()')
  - transform_filter(alt.datum.test_result != 'dubious')

text annotation

- Describe the encoding rules:
  - alt.X('binned_rating:O')
  - alt.Y('test_result:N')
  - alt.Text('count:Q')
- Describe the transformations:
  - transform_bin('binned_rating', 'rating')
  - transform_aggregate(groupby=['binned_rating', 'test_result'], count='count()')
  - transform_filter(alt.datum.test_result != 'dubious')

highlight

- Describe the encoding rules:
  - y='test_result:N'
  - x='binned_rating:O'
- Describe the transformations:
  - transform_joinaggregate(groupby=['test_result'],max_rating='max(binned_rating)')
  - transform_filter(alt.datum.max_rating >= 9)

Replicate Vis 2

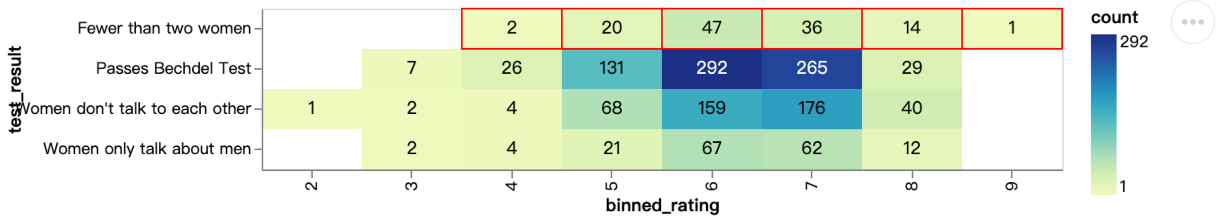
Hint:

- When you pass the heatmap to your text chart, you are passing all of the heatmap's encoding settings as well, which include the color setting.
- How do you translate "at least one movie with rating > 9 " into code?

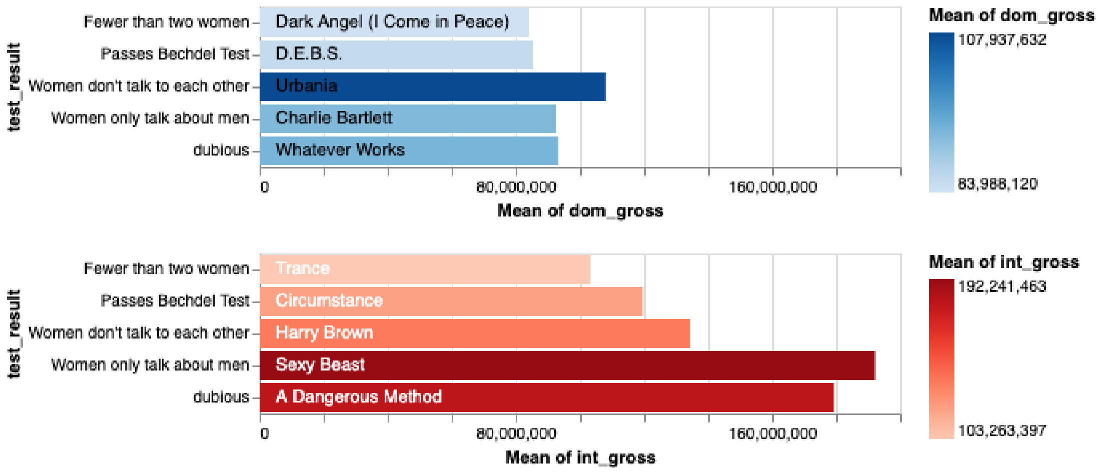
In [4]:

```
#TODO: remove NULL values



base = alt.Chart(movies_test).transform_bin(
    'binned_rating', 'rating'
).transform_aggregate(
    groupby=['binned_rating', 'test_result'],
    count='count()'
).transform_filter(
    (alt.datum.test_result != 'dubious') & (alt.datum.binned_rating >= 2)
)

heatmap = base.mark_rect().encode(
    y='test_result:N',
    x='binned_rating:O',
    color='count:Q'
)

text = base.mark_text().encode(
    alt.X('binned_rating:O'),
    alt.Y('test_result:N'),
    alt.Text('count:Q'))

text_white = base.transform_filter(
    alt.datum.count >= 265
).mark_text(color='white').encode(
    alt.X('binned_rating:O'),
    alt.Y('test_result:N'),
    text='count:Q')

highlight = base.transform_joinaggregate(
    groupby=['test_result'],
    max_rating='max(binned_rating)'
).transform_filter(
    alt.datum.max_rating >= 9
).mark_rect(filled=False, color='red').encode(
    y='test_result:N',
    x='binned_rating:O',
)


(heatmap + text + text_white + highlight).properties(width=500, height=100)
```

Out[4]:



Visualization 3: Bar chart with additional annotations



Description of this visualization: We want to contrast the domestic and international gross based on bedchel test passing category. We also want an example movie for each category so we're going to pick the 10th most popular one.

- Plot a bar chart for movies made in US and Canada, using the test result and mean of domestic gross.
- For test result category (i.e., each bar), find out the movie whose domestic gross ranks as the 10th lowest (sort by ascending order, the rank = 10). Annotate the title of this movie as text.
- Encode the mean domestic gross as color.
- BONUS: Align the text annotations with your bars.
- BONUS: Plot a similar bar chart for movies made internationally using the international gross. Concatenate the international and domestic charts charts. Make sure they share the same x axis but have independent color scales.

Visualization 3 Plan:

TODO: edit this cell to write your visualization plan. You can write in altair syntax, in full sentence, or in bullet points, whichever way that helps you to plan your chart.

bar chart

- Describe the encoding rules:
  - alt.X('average(dom_gross):Q', title='Mean of Domestic Gross'),
  - alt.Y('test_result:N'),
  - alt.Color('average(dom_gross):Q')
- Describe the transformations:
  - transform_filter(alt.datum.country_binary == 'U.S. and Canada')
  - transform_window(groupby=['test_result'],sort=[alt.SortField('dom_gross', order='ascending')],gross_rank='rank(*)')

text annotation

- Describe the encoding rules:
  - alt.X('gross_rank:Q', title='Mean of Domestic Gross') -> for text aligning purpose
  - alt.Y('test_result:N')
  - alt.Text('title:N')
- Describe the transformations:
  - transform_filter(alt.datum.country_binary == 'U.S. and Canada')
  - transform_window(groupby=['test_result'],sort=[alt.SortField('dom_gross', order='ascending')],gross_rank='rank(*)')
  - transform_filter(alt.datum.gross_rank == 10)

Replicate Vis 3

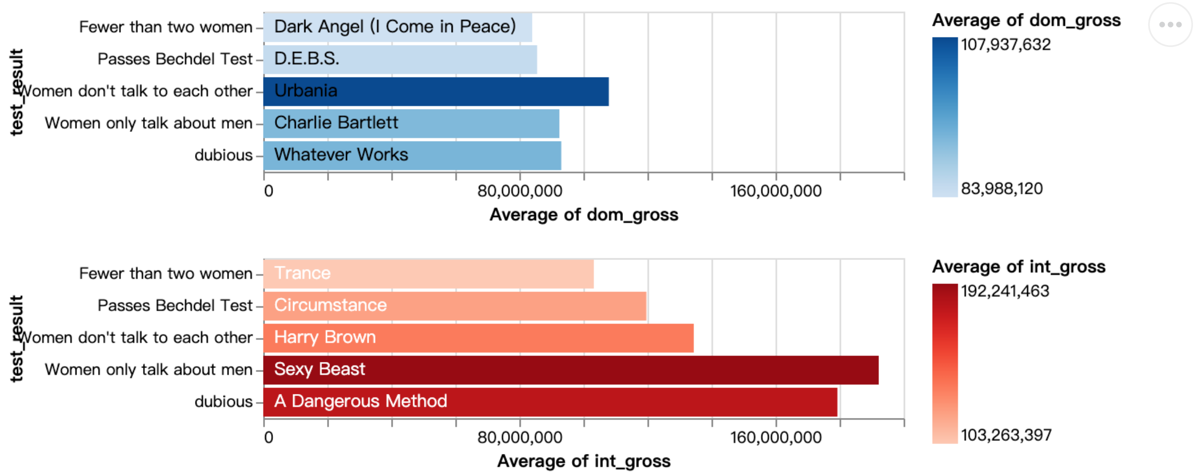
Hint:

- You want to generate the rank using a transformation method that we have covered in class.
- It's fine if text annotations don't align perfectly with your bars.

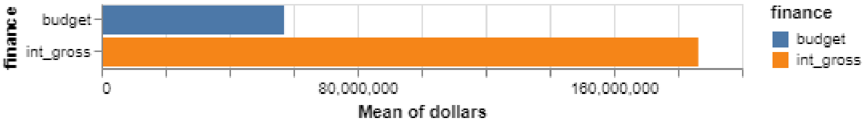
In [5]:

```
#TODO: Replicate Vis 3

base_dom = alt.Chart(movies_test).transform_filter(
    alt.datum.country_binary == 'U.S. and Canada'
).transform_window(
    groupby=['test_result'],
    sort=[alt.SortField('dom_gross', order='ascending')],
    gross_rank='rank(*)'
)

bar_dom = base_dom.mark_bar().encode(
    alt.X('average(dom_gross):Q', title='Average of dom_gross'),
    alt.Y('test_result:N'),
    alt.Color('average(dom_gross):Q')
)

text_dom = base_dom.transform_filter(
    alt.datum.gross_rank == 10
).mark_text(
    align = 'left',
    dx = 7,
    color='black'
).encode(
    alt.X('gross_rank:Q', title='Average of dom_gross'),
    alt.Y('test_result:N'),
    alt.Text('title:N')
)


base_int = alt.Chart(movies_test).transform_filter(
    alt.datum.country_binary != 'U.S. and Canada'
).transform_window(
    groupby=['test_result'],
    sort=[alt.SortField('int_gross', order='ascending')],
    gross_rank='rank(*)'
)


bar_int = base_int.mark_bar().encode(
    alt.X('average(int_gross):Q',  title='Average of int_gross'),
    alt.Y('test_result:N'),
    alt.Color('average(int_gross):Q', scale=alt.Scale(domain=[103263397, 192241463], scheme="reds
)

text_int = base_int.transform_filter(
    alt.datum.gross_rank == 10
).mark_text(
    align = 'left',
    dx = 7,
    color='white'
).encode(
    alt.X('gross_rank:Q',  title='Average of int_gross'),
    alt.Y('test_result:N'),
    alt.Text('title:N')
)


((bar_dom + text_dom)& (bar_int + text_int)).resolve_scale(
    color='independent',
    x='shared'
)
```

Out[5]:



Visualization 4: Bar chart



Description of this visualization:

- Plot a bar chart that enables the comparison between mean budget and mean international gross across the whole dataset.
- Two bars should be colored differently

Visualization4 Plan:

TODO: edit this cell to write your visualization plan. You can write in altair syntax, in full sentence, or in bullet points, whichever way that helps you to plan your chart.

- What kind of transformation do I need to plot this bar chart?
  - transform_fold(["budget", "int_gross"], as_=["Finance", "dollars"])
- What kind of encoding do I want to use??
  - alt.Y('Finance:N'),
  - alt.X('average(dollars):Q'),
  - alt.Color('Finance:N')

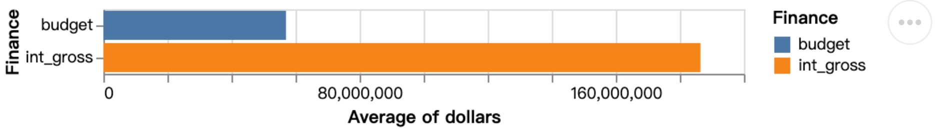
Replicate Vis 4

Hint:

- Do I need a long form or wide form data? Which type of data do I have? What transformation can I use?

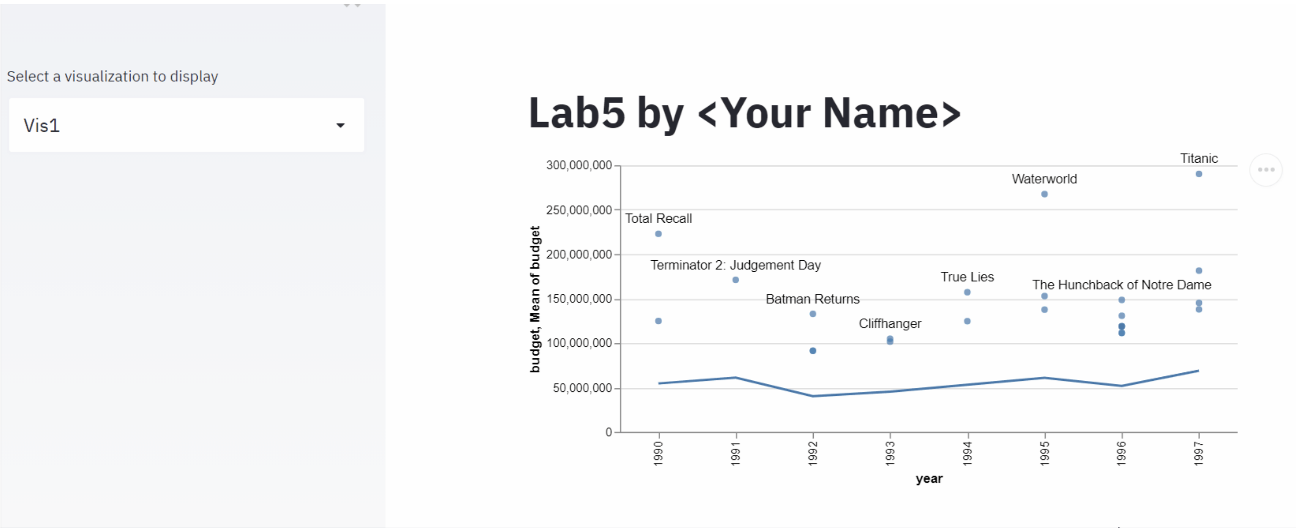
In [6]:
```
#TODO: replicate vis 4

base = alt.Chart(movies_test).transform_fold(
    ["budget", "int_gross"],
    as_=["Finance", "dollars"]
)
base.mark_bar().encode(
    alt.Y('Finance:N'),
    alt.X('average(dollars):Q'),
    alt.Color('Finance:N')
)
```

Out[6]:



Export Instruction

For this lab, we want you to build a website with streamlit. Your finished website will have a select widget on the sidebar. Depending on what the user selects, the website will display the corresponding visualization.



Select a visualization to display

Vis1

Lab5 by <Your Name>

To make this website, you will need to

1. Go over the material we provided on streamlit.
2. Make a new python script for streamlit. Make sure you import the streamlit package.
3. Create a title
4. Create a selectbox on the sidebar, with 4 options (e.g., vis1, vis2, vis3, and vis4).
5. Copy-paste the code used to generate visualizations. Make sure each chart is saved to a variable.
6. Connect charts with the selected option by adding an if-else-then statement.

This is the end of lab 5.

Please run all cells in your notebook, and

1. save to PDF (File->Print->Save PDF -> landscape, shrink to 80%)
2. save to ipynb (File -> Download .ipynb)
3. save the python file that contains your streamlit app.

Rename files with your uniqname: e.g. uniqname.pdf/ uniqname.ipynb/uniquename.py

Upload all three files to canvas.