

▼ SI649-20-Winter Lab 2 -> Altair I

Overview

We're going to re-create some of the visualizations we did in Tableau but this time using Altair for the article: ["The Dollar-And-Cents Case Against Hollywood's Exclusion of Women"](#). We'll be teaching you different pieces of Altair over the next few weeks so we'll focus on just a few visualizations this time:

1. Replicate 2 visualizations in the original article
2. Implementing 2 new visualizations according to our specifications

For this lab, we have done all of the necessary data transformation for you. You do not need to modify any dataframe. You only need to write Altair code. It's fine if your visualization looks slightly different from the example (e.g., getting 1.1 instead of 1.0)

Lab Instructions (read the full version on the handout of the previous lab)

- 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, 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.

We encourage you to go through the Altair tutorials before next week:

- [UW Course](#)
- [Altair tutorial](#)

Resources

- [Altair Documentation](#)

- [Colab Overview](#)
- [Markdown Cheatsheet](#)
- [Pandas DataFrame Introduction](#)
- [Vega-Lite documentation](#)
- [Vega/Vega-Lite editor](#)

```
# imports we will use
import altair as alt
import pandas as pd
from collections import defaultdict

# load data and perform basic data processing
# get the CSV
datasetURL="https://raw.githubusercontent.com/LiciaHe/SI649/master/week2/movies_individual_task.csv"
movieDF=pd.read_csv(datasetURL, encoding="latin-1")

# fix the result column, rename the values
movieDF['test_result'] = movieDF['clean_test'].map({
    "ok": "Passes Bechdel Test",
    "men": "Women only talk about men",
    "notalk": "Women don't talk to each other",
    "nowomen": "Fewer than two women",
    "dubious": "dubious"
})

# fix the location column for later use
locationDict = defaultdict(lambda: 'International')
locationDict["United States"]="U.S. and Canada"
locationDict["Canada"]="U.S. and Canada"
movieDF["country_binary"]=movieDF["country"].map(locationDict)

##calculate ROI for 2nd chart
movieDF["roi_dom"]=movieDF["domgross_2013$"]/movieDF["budget_2013$"]
movieDF["int_only_gross"]=movieDF["intgross_2013$"]-movieDF["domgross_2013$"]
movieDF["roi_int"]=movieDF["int_only_gross"]/movieDF["budget_2013$"]
```

```
movieDF=movieDF.drop(columns=["Unnamed: 0", "test", "budget", "domgross", "intgross", "code", "period code", "decade code", "director", "genre", "directo
movieDF_since_1990=movieDF[movieDF.year>1989]
```

```
#take a look at the new dataset
movieDF.sample(20)
# movieDF_since_1990.sample(3)
```



	year	title	clean_test	binary	budget_2013\$	domgross_2013\$	intgross_2013\$	rating	country	language	test_re
1460	1996	Independence Day	ok	PASS	111387369	454711839.0	1.213975e+09	7.0	United States	English	P. Bechde
343	2010	Frozen	notalk	FAIL	160234776	419868647.0	1.073092e+09	6.2	United States	English	Women talk to
1623	1989	Road House	notalk	FAIL	18795888	56481697.0	5.648170e+07	6.5	United States	English	Women talk to
76	2013	The Heat	ok	PASS	43000000	159581587.0	2.307816e+08	6.6	United States	English	P. Bechde
1653	1987	Lethal Weapon	notalk	FAIL	30755961	133670224.0	2.464421e+08	7.6	United States	English	Women talk to
1777	1974	The Texas Chain Saw Massacre	notalk	FAIL	661322	125520924.0	1.255209e+08	7.5	United States	English	Women talk to
351	2010	How to Train Your Dragon	ok	PASS	176258254	232427199.0	5.286370e+08	8.1	United States	English	P. Bechde
295	2011	The Twilight Saga: Breaking Dawn - Part 1	ok	PASS	132050575	291326492.0	7.352769e+08	4.9	United States	English	P. Bechde
1738	1980	The Fog	ok	PASS	50899822	83450574.0	1.047646e+08	6.8	United States	English	P. Bechde
285	2011	The Help	ok	PASS	25892270	175762513.0	2.207264e+08	8.1	United States	English	P. Bechde
1089	2002	Die Another Day	men	FAIL	183915449	208448913.0	5.594425e+08	6.1	United Kingdom	English	Women talk
484	2000	Harry Potter and the Philosopher's Stone	ok	PASS	87140000	827046641.0	1.014506e+09	7.6	United Kingdom	English	P. Bechde

484	2009	and the Hair-Blood Prince	OK	PASS	271432899	327846641.0	1.014526e+09	7.0	Kingdom	English	Bechde
655	2008	Vicky Cristina Barcelona	ok	PASS	17316101	25126430.0	1.131010e+08	7.1	Spain	English	P. Bechde
613	2008	Pontypool	nowomen	FAIL	1623384	4183.0	3.454100e+04	6.6	Canada	English	Fewe two w
133	2012	ParaNorman	notalk	FAIL	60878314	56822855.0	1.097423e+08	7.0	United States	English	Wome talk tc
680	2007	Disturbia	ok	PASS	22470870	90119077.0	1.320984e+08	6.9	United States	English	P. Bechde
174	2012	The Words	notalk	FAIL	6087831	11663106.0	1.231018e+07	7.1	United States	English	Wome talk tc
227	2011	I Am Number Four	notalk	FAIL	51784539	57067015.0	1.511930e+08	6.1	United States	English	Wome talk tc
584	2008	Fly Me to the Moon	nowomen	FAIL	27056408	15740274.0	6.228033e+07	4.5	Belgium	English	Fewe two w
989	2004	The Chronicles of Riddick	men	FAIL	147983145	71170953.0	1.322140e+08	6.7	United States	English	Wome talk

▼ Visualization 1: Recreate this visualization

Median Budget For Films Since 1990

2013 dollars



FIVETHIRTYEIGHT

SOURCE: BECHDELTEST.COM, THE-NUMBERS.COM

Step 1: Write down your plan for the visualization (edit this cell)

- Data Name: *movieDF_since_1990*
- mark type: bar
- Encoding Specification:
 - x: median(budget_2013\$):qualitative
 - y: test_result:nominal

Example encoding, if we had the nominal variable 'movietype' and we wanted to use color, it would be:

color: movietype:nominal

▼ Step 2: Create your chart.

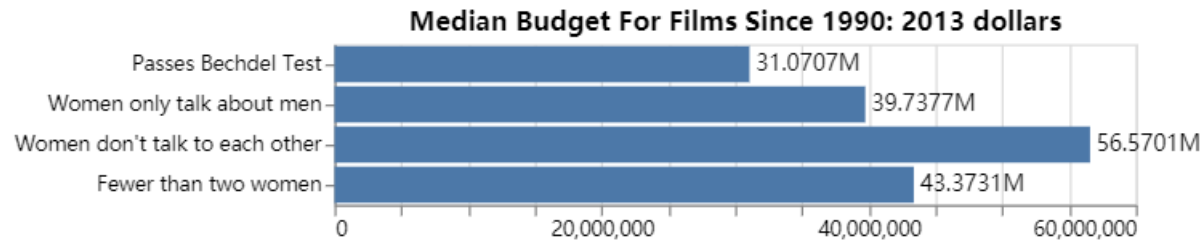
Please take a look at the checkpoints below. You can follow the checkpoint to work through the problem step-by-step. Don't forget to paste your FINAL answer to the cell immediately below this block (it will allow us to grade). You can search for the keyword "TODO" to locate cells that need your edits

```
bars = alt.Chart(movieDF_since_1990, title="Median Budget For Films Since 1990: 2013 dollars").mark_bar().encode(
    x=alt.X(
        'median(budget_2013$)',
        title=""
    ),
    y=alt.Y(
        'test_result:N',
        sort=["Passes Bechdel Test", "Women only talk about men", "Women don't talk to each other", "Fewer than two women"],
        title="",
    ),
).transform_filter(
    alt.datum.test_result != "dubious"
)

text = bars.mark_text(
    align='left',
    baseline='middle',
    dx=3 # Nudges text to right so it doesn't appear on top of the bar
).encode(
    text=alt.Text("median(budget_2013$):Q", format=".6s")
)

bars + text
```

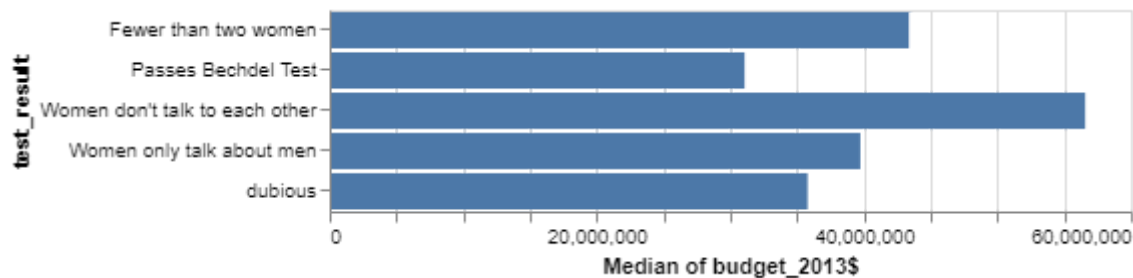




checkpoint 1: basic bar chart: you get full points if you

- Specify the correct mark
- Use the correct x and y encoding
- Plotting the right data (hint: make sure you examine the data frame and use the correct columns)

You chart should look like:

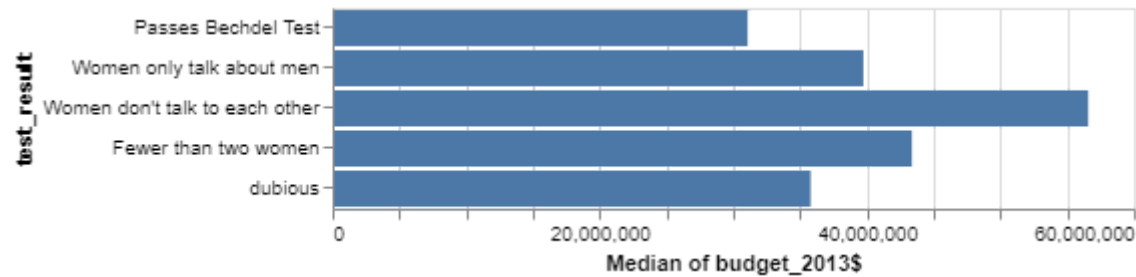


checkpoint 2: basic bar chart with sorted order: you get full points if you

- Completed checkpoint1
- Align the order of your y-axis values with the provided example.
- *i.e., from top to bottom, the order of the bars is "Passes Bechdel Test","Women only talk about men","Women don't talk to each other","Fewer than two women","dubious".*

Hint: [Sort](#)

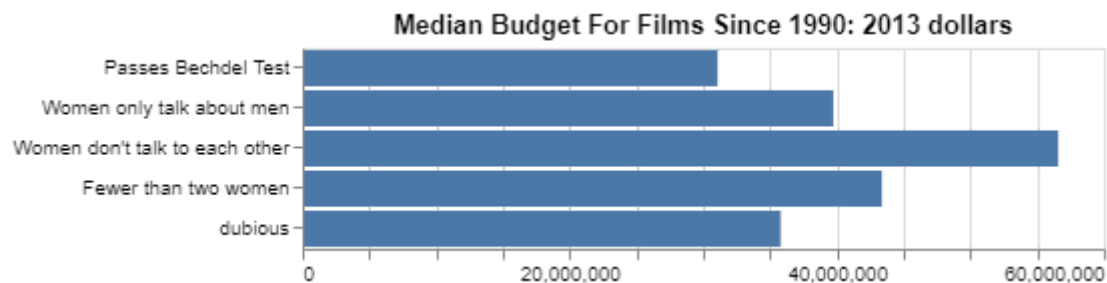
You chart should look like:



checkpoint 3: basic bar chart with title: you get full points if you

- Completed checkpoint2
- Remove labels on x-axis and y-axis
- Add a chart title

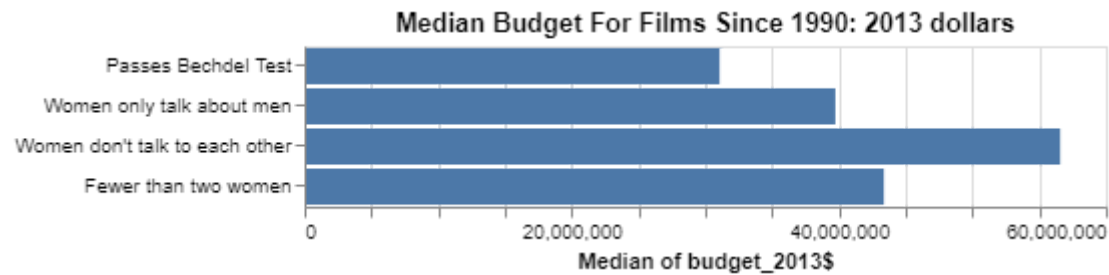
You chart should look like:



checkpoint 4: BONUS: remove dubious. You will get full point if you

- Complete checkpoint 3
- Remove the bar for "dubious" (using Altair, no Pandas)

You chart will look like:

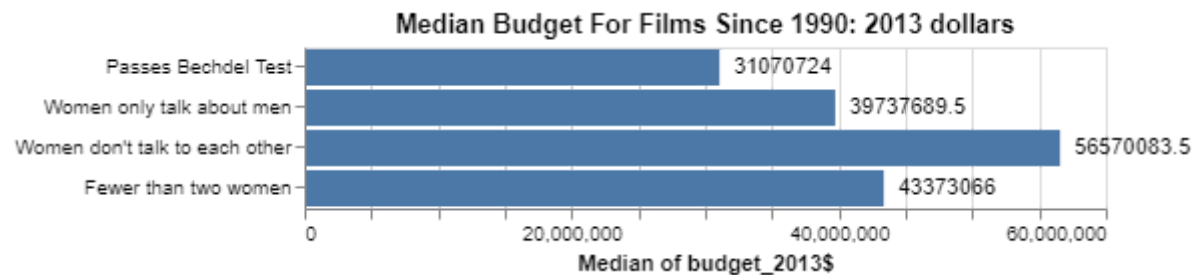


checkpoint 5: BONUS: add number labels.

You will get full point if you

- Complete checkpoint 4
- Add number as labels of your bars

You chart will look like:

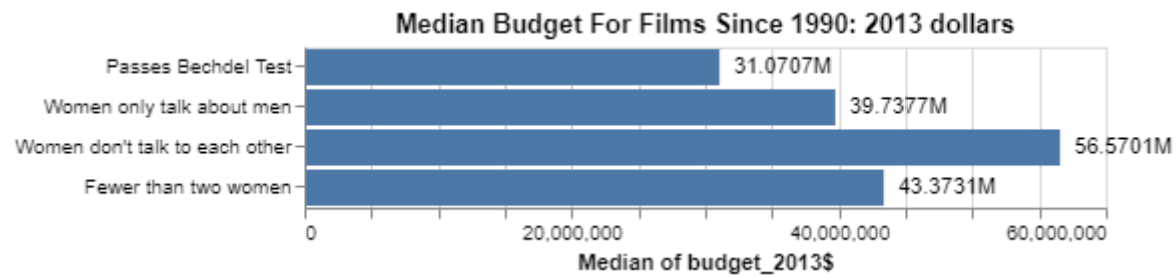


checkpoint 6: BONUS: format numbers.

You will get full points if you

- Complete checkpoint 5
- Adjust number labels to display millions. e.g. (31.4592 M instead of 31459218). You might want to read about [format](#), and [D3's format specification](#).



You chart will look like:





▼ Visualization 2 Replicate this visualization

Step 1: Write down your plan for the visualization (edit this cell)

Left chart:

- Data Name: *movieDF*
- mark type: *bars*
- Encoding Specification:
-  *x: median(roi_dom): qualitative*
-  *y: test_result: nominal*

Right chart:

- Data Name: *movieDF*
- mark type: *bars*
- Encoding Specification:
-  *x: median(roi_int): qualitative*
-  *y: test_result: nominal*

Compound Method (how to join these charts together?): *horizontally concat*

Example encoding, if we had the nominal variable 'movietype' and we wanted to use color, it would be:

color: movietype: nominal

▼ Step 2: Create your chart.

Please take a look at the checkpoints below. You can follow the checkpoint to work through the problem step-by-step. Don't forget to paste your FINAL answer to the cell immediately below this block (it will allow us to grade). You can search for the keyword "TODO" to locate cells that need

your edits

```
bars1 = alt.Chart(movieDF, title="U.S. and Canada").mark_bar().encode(
    x=alt.X(
        'median(roi_dom)',
        title=""
    ),
    y=alt.Y(
        'test_result:N',
        sort=["Passes Bechdel Test", "Women only talk about men", "Women don't talk to each other", "Fewer than two women", "dubious"]
    ),
)

text1 = bars1.mark_text(
    align='left',
    baseline='middle',
    dx=3 # Nudges text to right so it doesn't appear on top of the bar
).encode(
    text=alt.Text("median(roi_dom):Q", format=".5f")
)

bars2 = alt.Chart(movieDF, title="International").mark_bar(color="orange").encode(
    x=alt.X(
        'median(roi_int)',
        title=""
    ),
    y=alt.Y(
        'test_result:N',
        sort=["Passes Bechdel Test", "Women only talk about men", "Women don't talk to each other", "Fewer than two women", "dubious"]
        axis=None,
        title=""
    ),
)

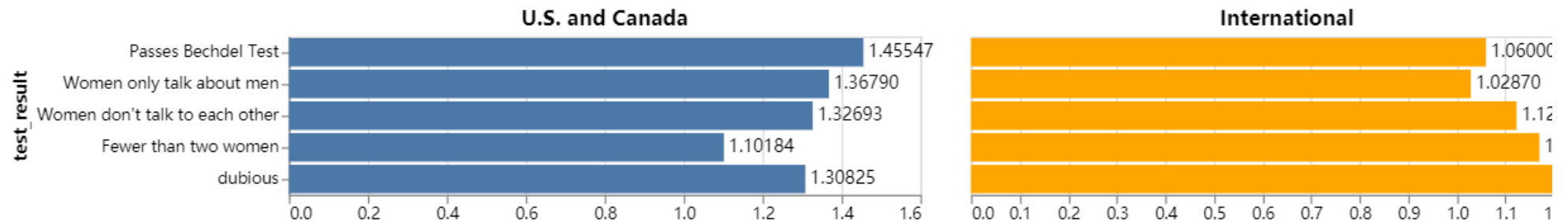
text2 = bars2.mark_text(
    align='left',
    baseline='middle',
    dx=3 # Nudges text to right so it doesn't appear on top of the bar
```

```

).encode(
    text=alt.Text("median(roi_int):Q", format=".5f")
)

((bars1 + text1) | (bars2 + text2)).resolve_scale(y="shared")

```

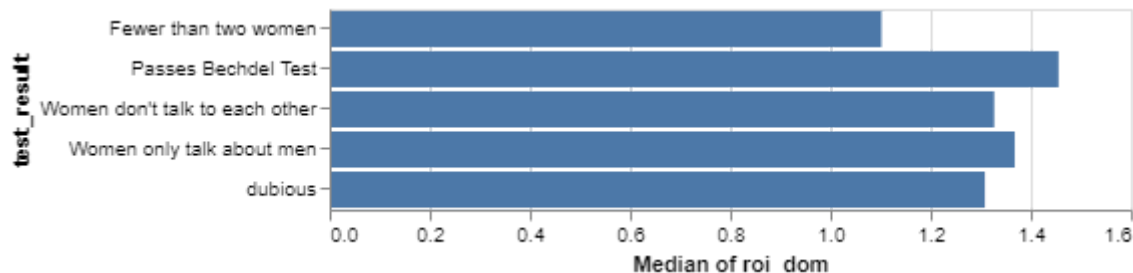


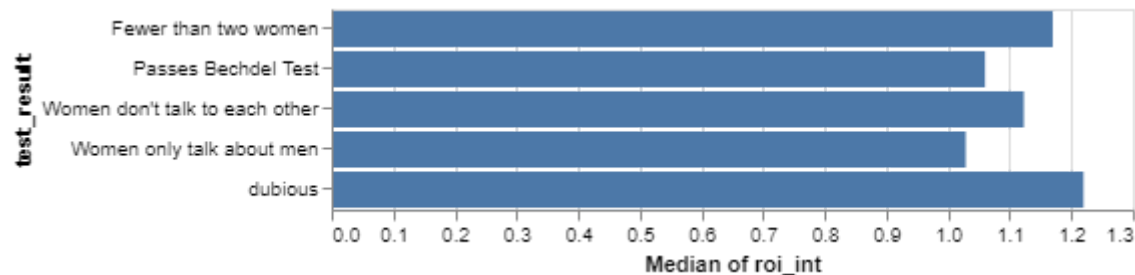
▼ Visualization 2 Checkpoints

checkpoint 1: basic bar charts

- Specify the correct mark
- Use the correct x and y encoding
- Plotting the right data (hint: make sure you examine the data frame and use the correct columns)
- You will have 2 charts, one for U.S.&Canada, one for International

You chart will look like:



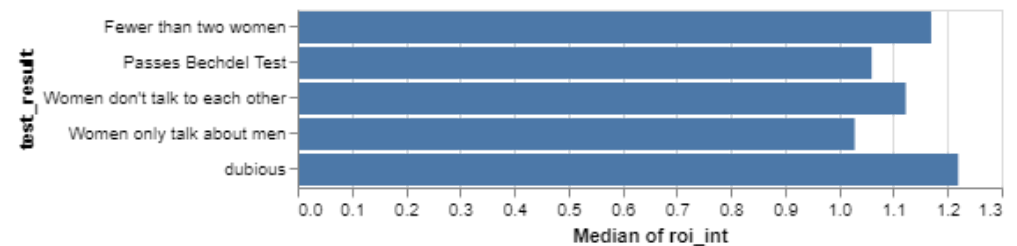
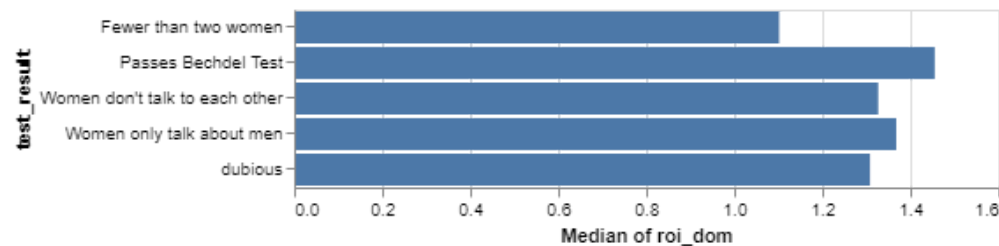


and

checkpoint 2: joining two charts

- completed checkpoint1
- joined two charts

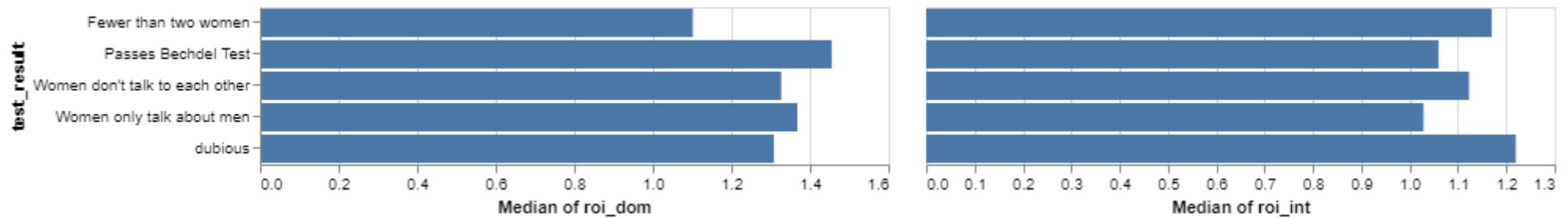
You chart will look like:



checkpoint 3: resolve y scale and hide the second y-axis

- completed checkpoint2
- ensure that two charts are sharing the same y-axis
- remove the second y-axis

You chart will look like:

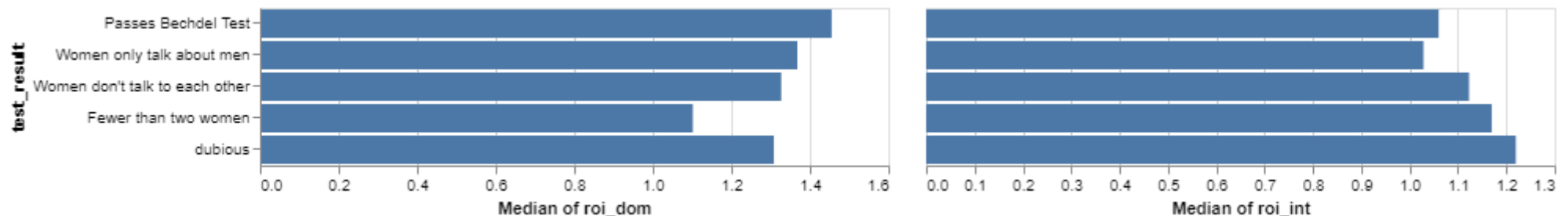


checkpoint 4: sort y-axis

- completed checkpoint 3
- Sort y-axis so that the order of the bars is (from top to bottom):

"Passes Bechdel Test","Women only talk about men","Women don't talk to each other","Fewer than two women","dubious"

You chart will look like:

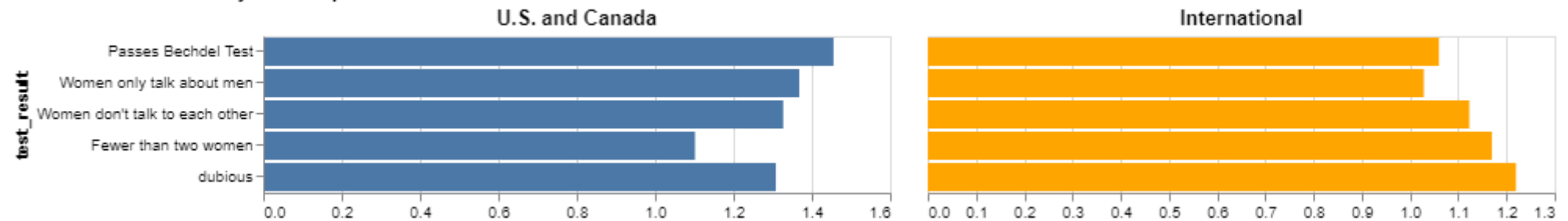


checkpoint 5: Change color and titles

- completed checkpoint 4
- color bars of these two charts with different colors
- add title to the compound chart
- edit axis labels (you can also remove axis label and add chart title to individual chart)

You chart will look like:

Dollars Earned for Every Dollar Spent

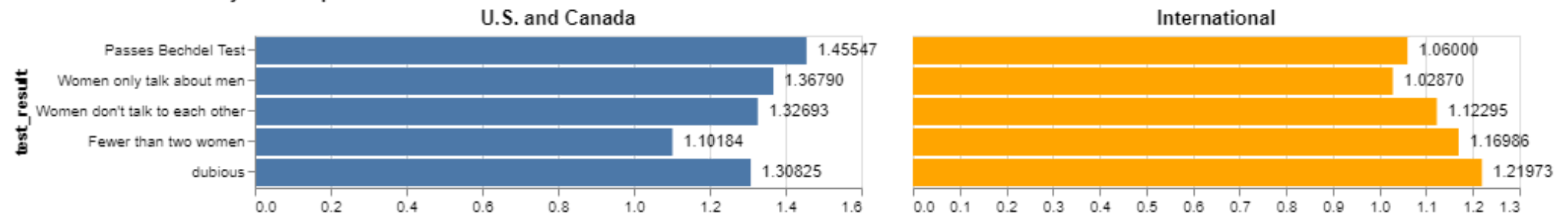


checkpoint 6: BONUS: Add number layer

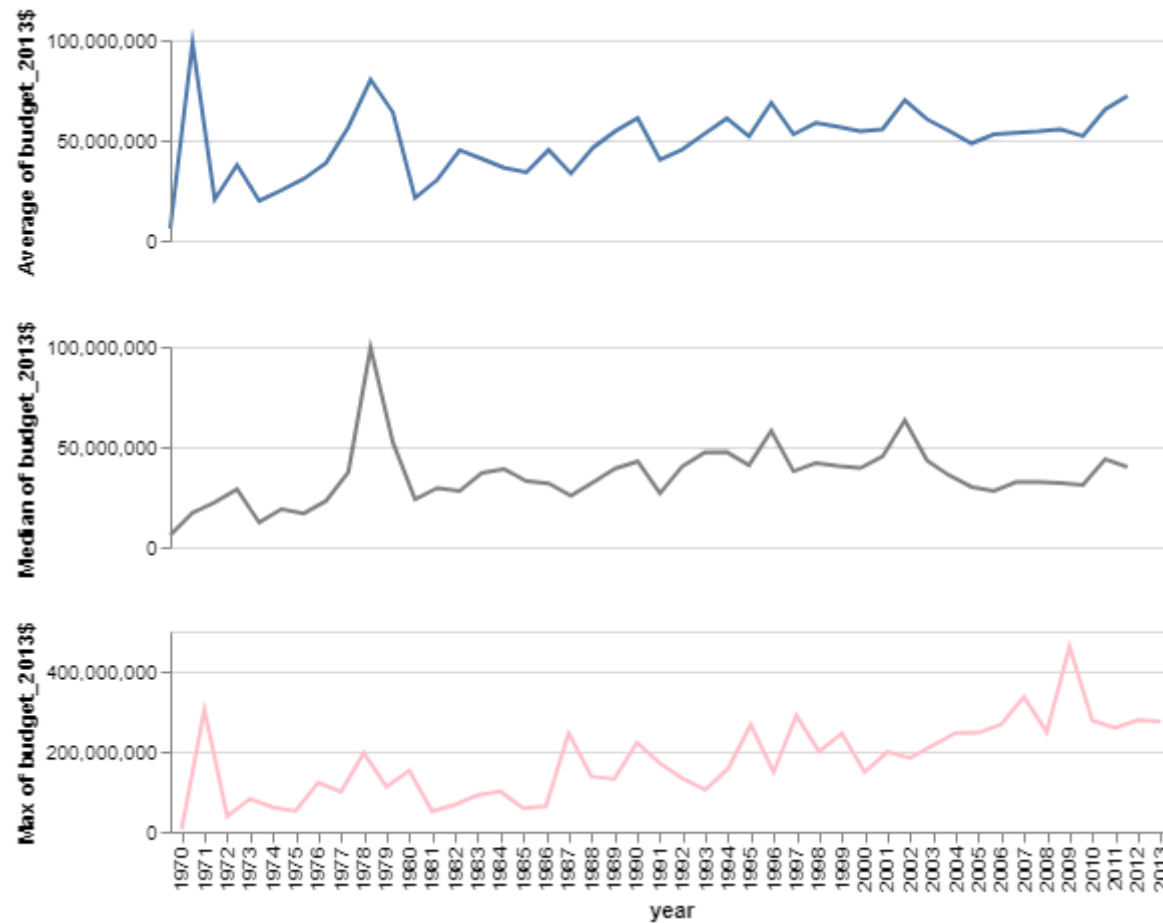
- completed checkpoint 5
- add number annotations

You chart will look like:

Dollars Earned for Every Dollar Spent



▼ Visualization 3: Replicate this visualization



Step 1: Write down your plan for the visualization (edit this cell)

- Data Name: *movieDF*
- mark type: *line*
- Encoding Specification (1st chart):
- x:year:ordinary*

- `y: average(budget_2013$):qualitative`
- Encoding Specification (2nd chart):
- `x:year:ordinary`
- `y: median(budget_2013$):qualitative`
- Encoding Specification (3rd chart):
- `x:year:ordinary`
- `y: max(budget_2013$):qualitative`

▼ Step 2: Create your chart.

Please take a look at the checkpoints below. You can follow the checkpoint to work through the problem step-by-step. Don't forget to paste your FINAL answer to the cell immediately below this block (it will allow us to grade). You can search for the keyword "TODO" to locate cells that need your edits

```
line1 = alt.Chart(movieDF, height=100, width=500).mark_line().encode(
    x=alt.X(
        'year:O',
        axis=None,
        title=""
    ),
    y=alt.Y(
        'average(budget_2013$)',
    ),
)

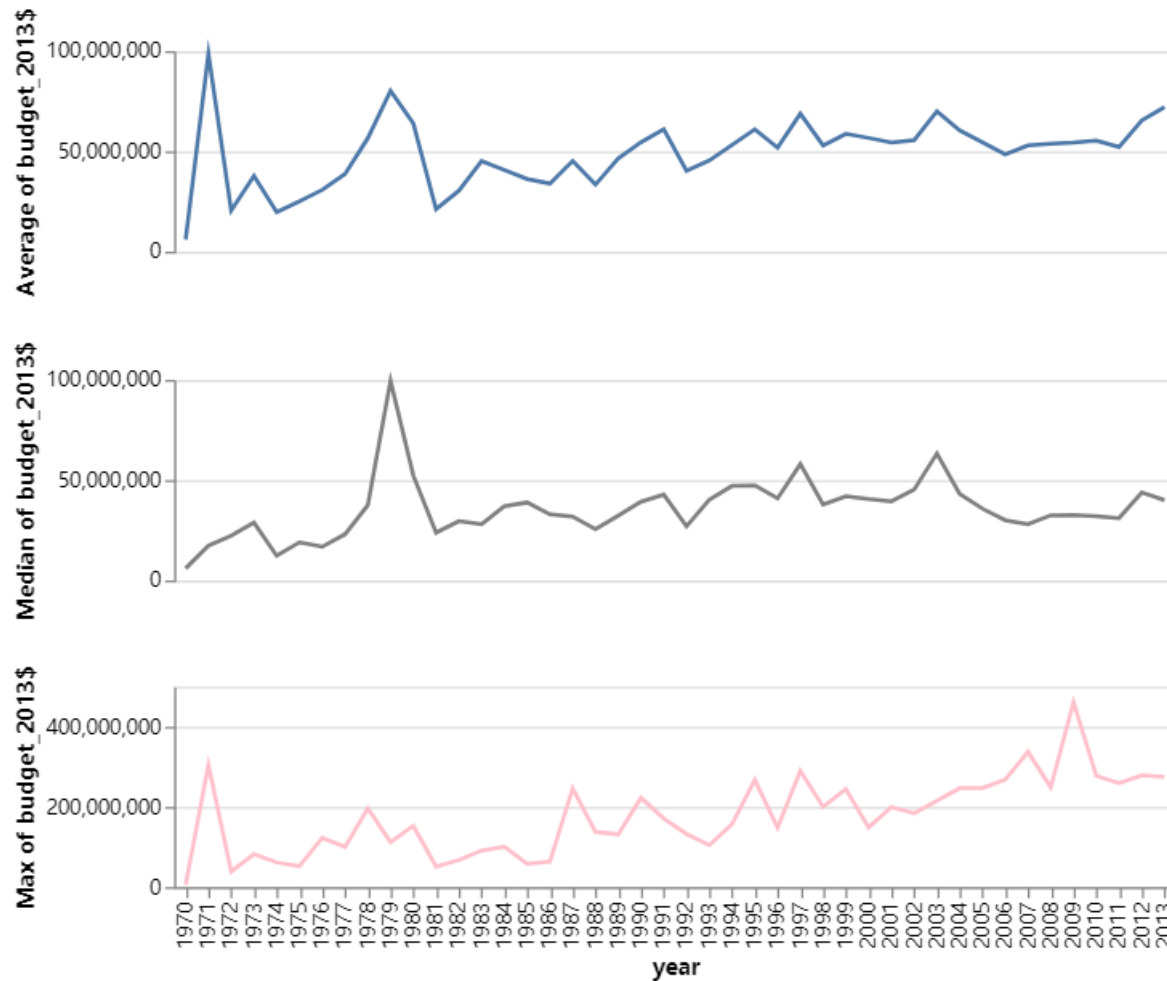
line2 = alt.Chart(movieDF, height=100, width=500).mark_line(color='grey').encode(
    x=alt.X(
        'year:O',
        axis=None,
```

```
        title=""
    ),
    y=alt.Y(
        'median(budget_2013$)',
    ),
)

line3 = alt.Chart(movieDF, height=100, width=500).mark_line(color='pink').encode(
    x=alt.X(
        'year:O',
    ),
    y=alt.Y(
        'max(budget_2013$)',
    ),
)

(line1 & line2 & line3).resolve_scale(x="shared")
```





▼ Visualization3 Checkpoints

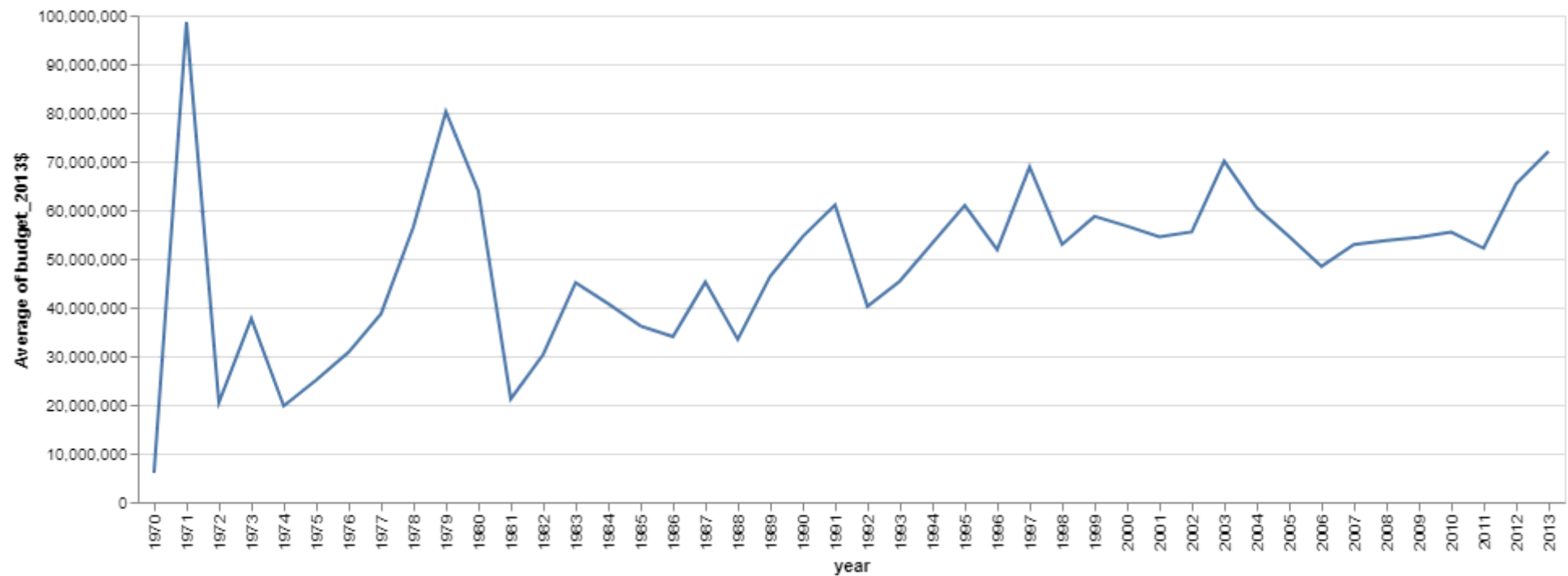
checkpoint 1: line chart for average, median, and max of budget

You will get full points if you

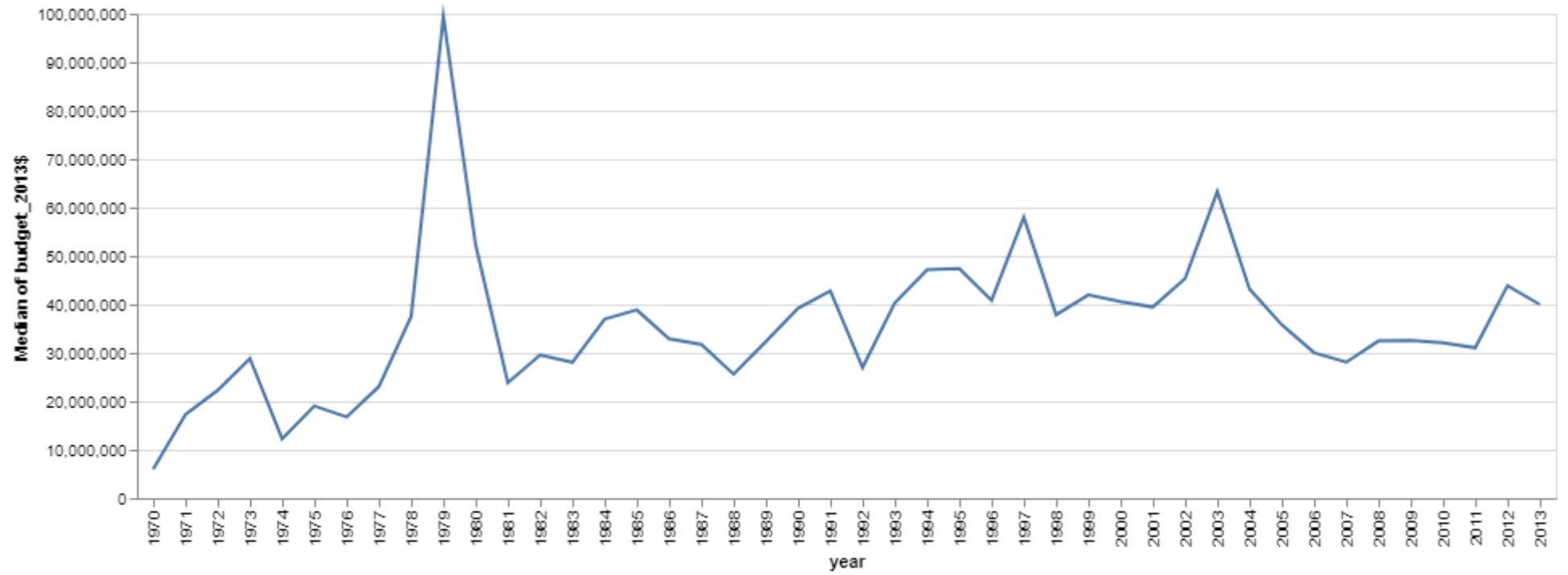
- Specify the correct mark

- Use the correct x and y encoding
- Plotting the right data
- Produce 3 line charts

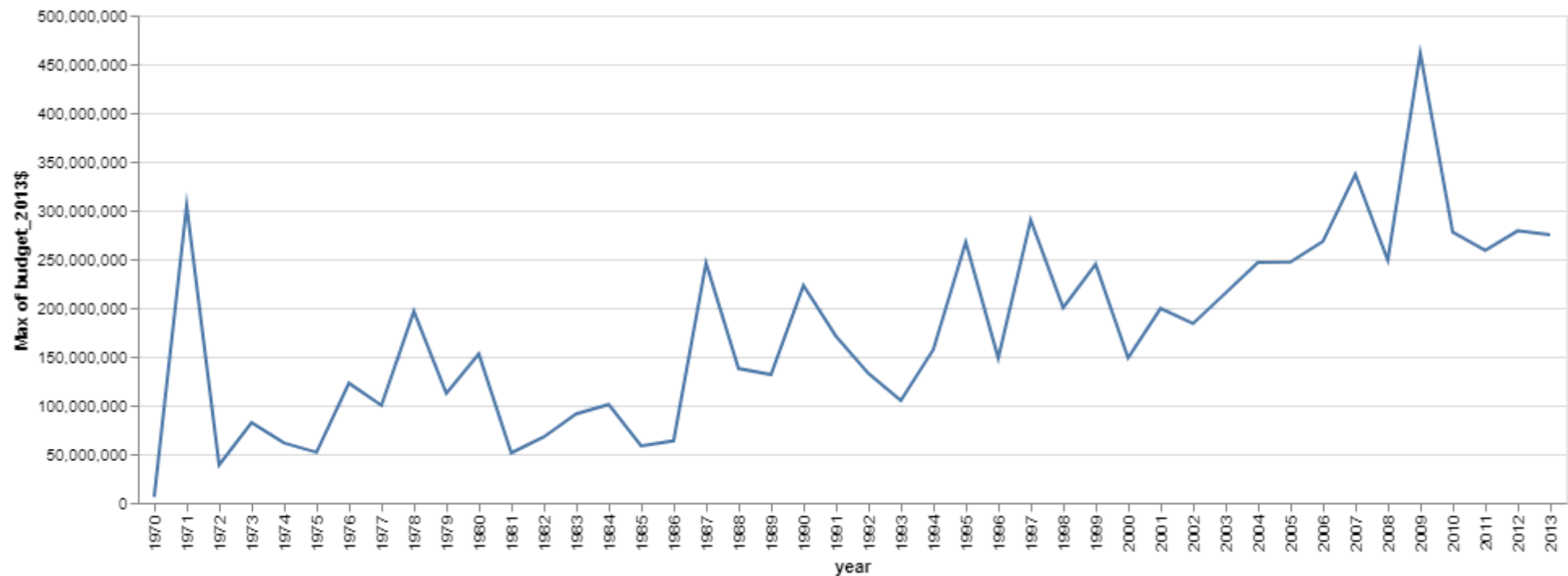
You chart will look like:



and



and



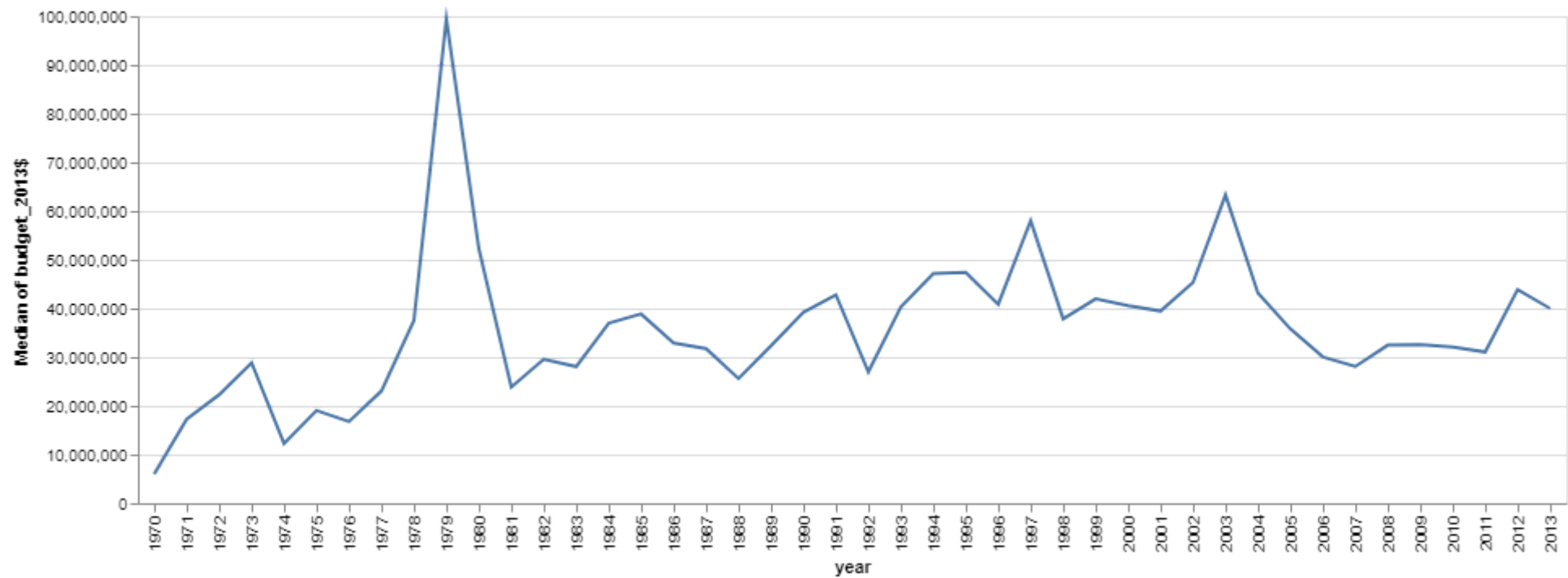
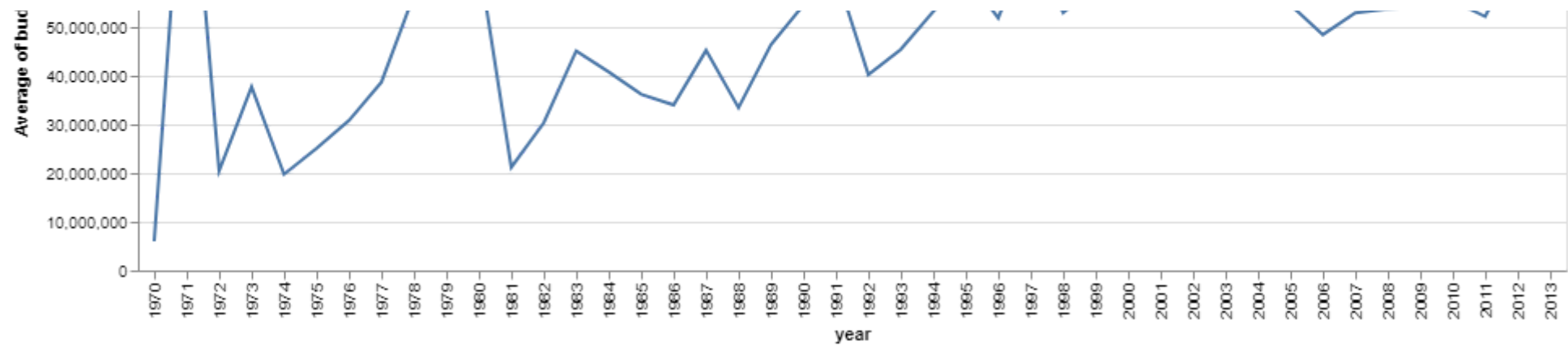
checkpoint 2: concat 3 line charts

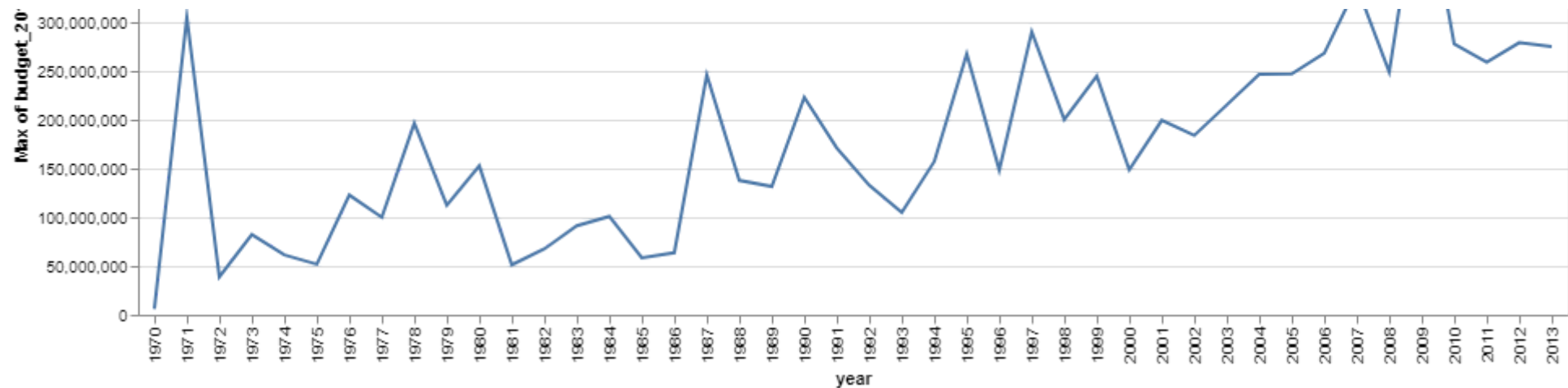
You will get full points if you

- Complete checkpoint 1
- Concat 3 charts vertically

You chart will look like:





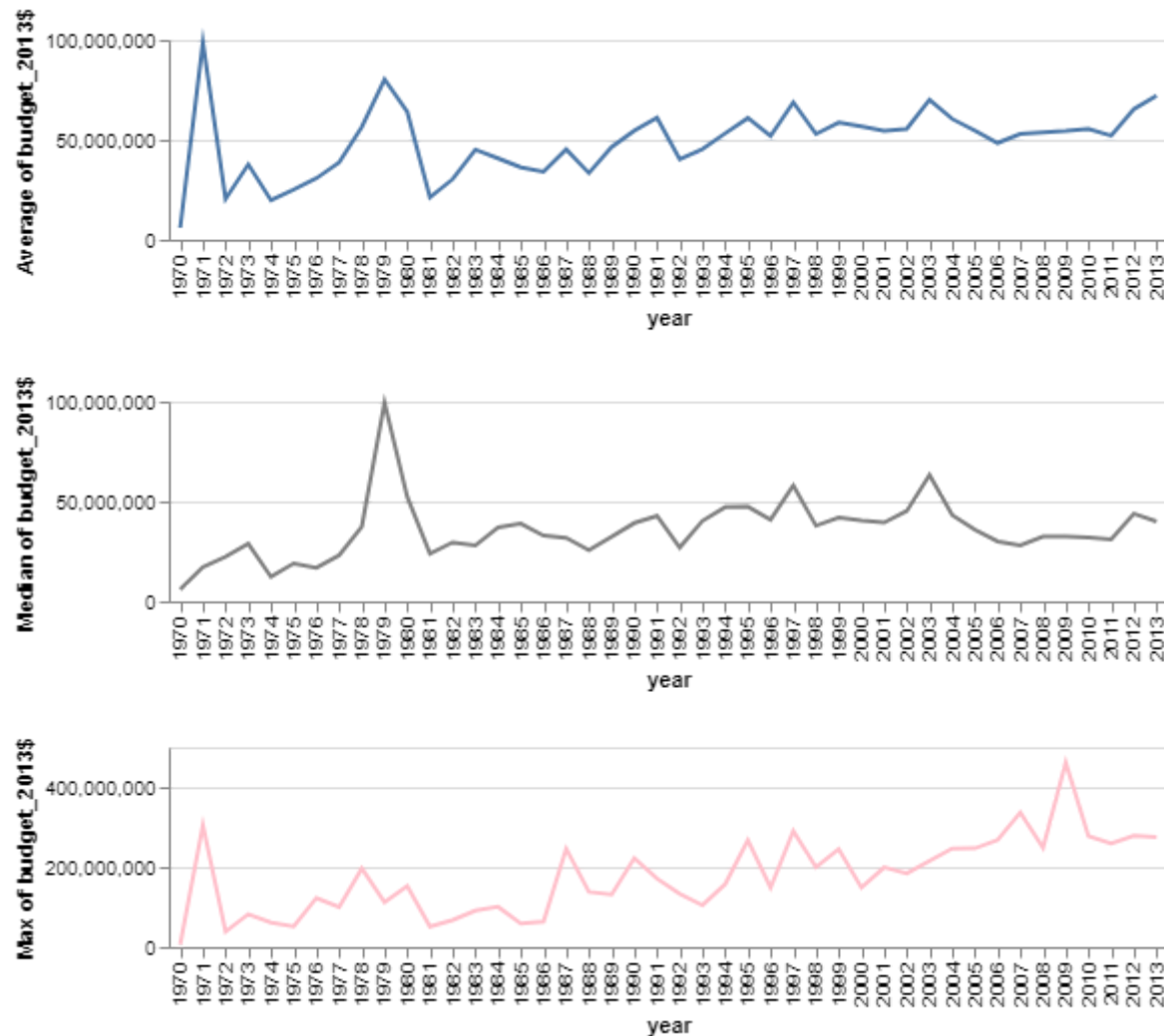


checkpoint 3: adjust width, height and color

Each chart should be 500x100, plotted with different colors

You will get full points if you

- Complete checkpoint 2
- Adjust chart width and height
- Plot charts with different colors



You chart will look like:

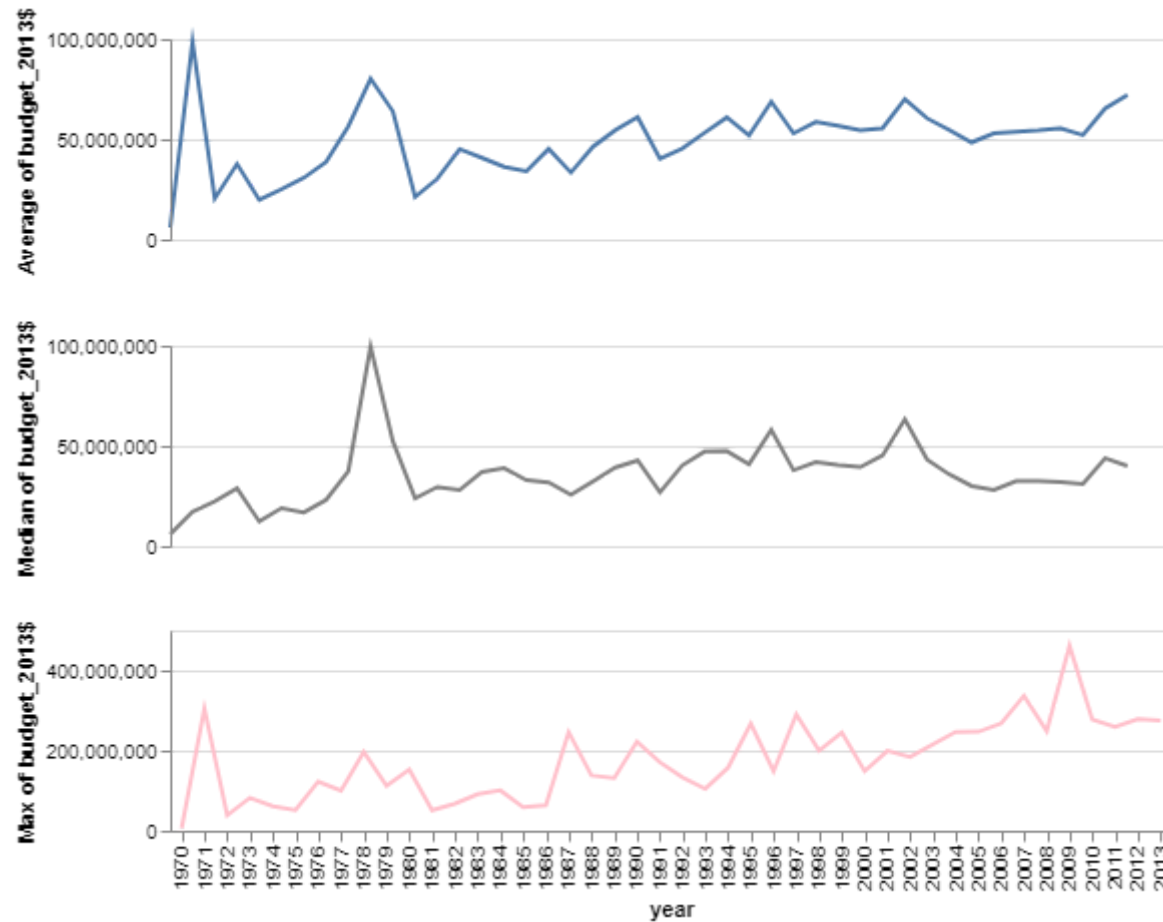
checkpoint 4: resolve axis and remove duplicated x-axis

You will get full points if you

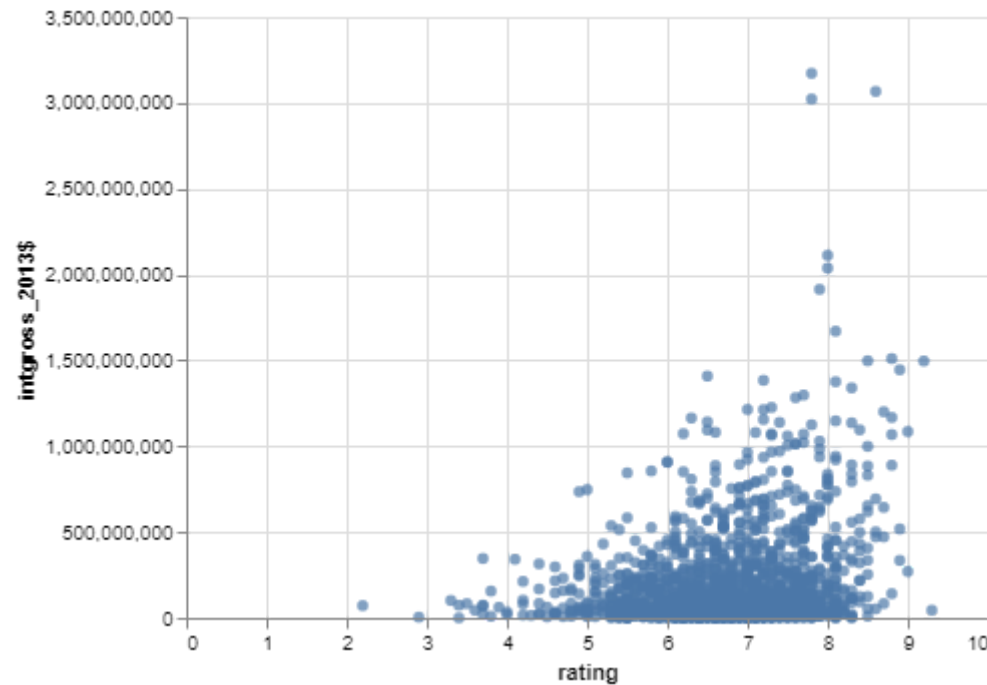
- Complete checkpoint 3
- Ensure that 3 charts are sharing the same x-axis

- Remove duplicate axis ticks.

You chart will look like:



▼ Visualization 4: Replicate this visualization



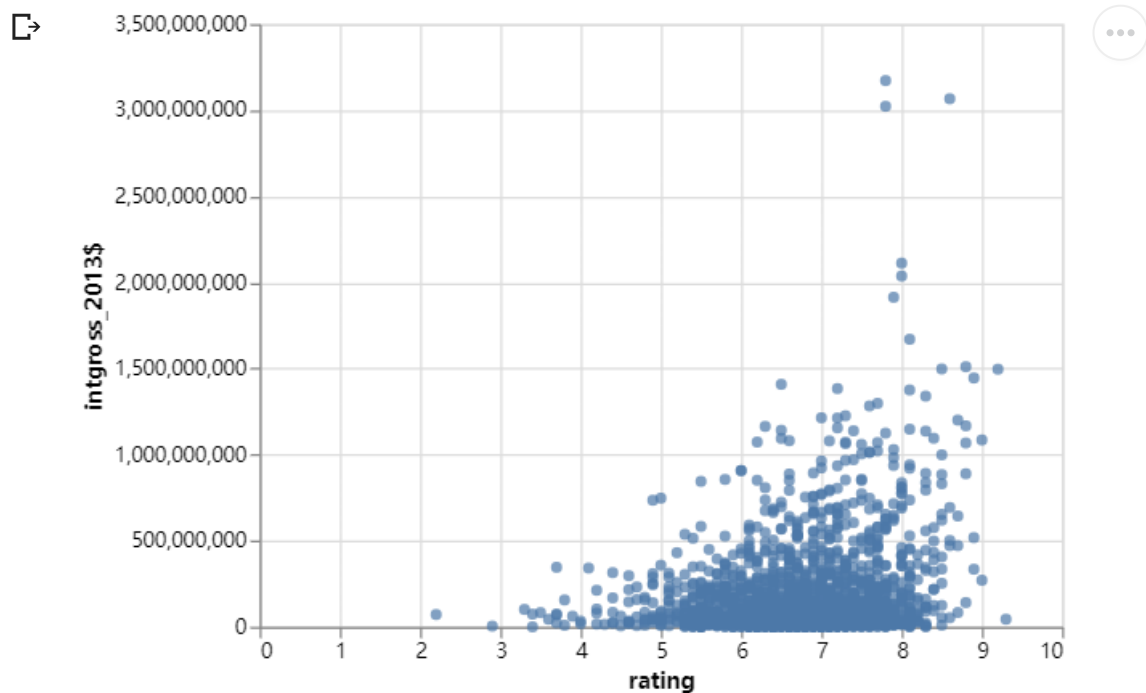
Step 1: Write down your plan for the visualization (edit this cell)

- Data Name: *movieDF*
- mark type: *point*
- Encoding Specification:
 - *x:rating:qualitative*
 - *y:intgross_2013\$:qualitative*

▼ Step 2: Create your chart.

This chart is relatively simple so there's no checkpoint.

```
points = alt.Chart(movieDF).mark_point(filled=True).encode(  
    x=alt.X(  
        'rating:Q',  
    ),  
    y=alt.Y(  
        'intgross_2013$:Q',  
    ),  
)  
points
```



End of LAB2

Please run all cells (Runtime->Run all), and

1. save to PDF (File->Print->Save PDF)
2. save to ipynb (File -> Download .ipynb)

Rename both files with your unqiename: e.g. unqiename.pdf/ unqiename.ipynb Upload both files to canvas.