Information Visualization I

School of Information, University of Michigan

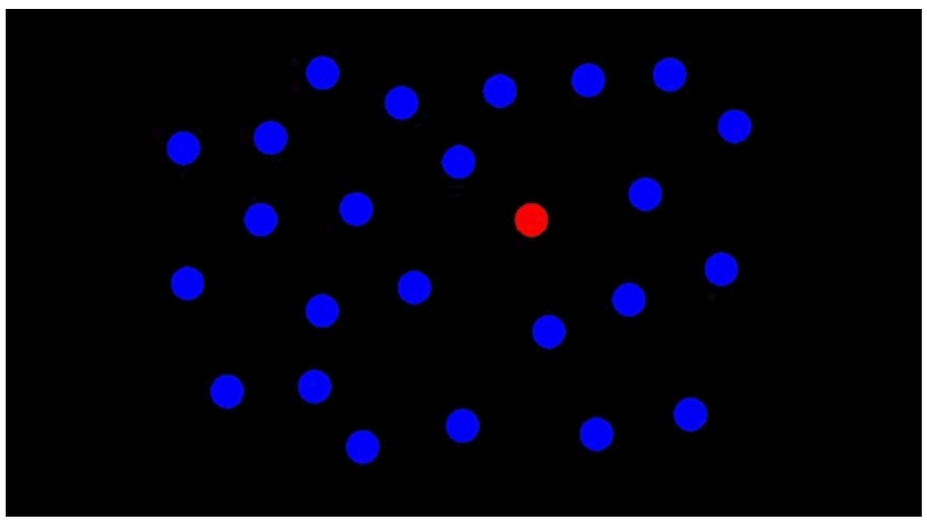
Week 3:

- Perception / Cognition

Assignment Overview

This assignment's objectives include:

- Review, refect, and apply the concepts of the perception pipeline. Justify how different encodings impact the effectiveness of a visualization depending on the human perception process.



Preattentive Processing

- Recreate visualizations and propose new and alternative visualizations using Altair (https://altair-viz.github.io/)

The total score of this assignment will be 100 points consisting of:

- Case study reflection: America's Favorite 'Star Wars' Movies (And Least Favorite Characters) (30 points)
- Altair programming exercise (70 points)

Resources:

- Article by FiveThirtyEight (https://fivethirtyeight.com) available online (https://fivethirtyeight.com/features/americas-favorite-star-wars-movies-and-least-favorite-characters/) (Hickey, 2014)
- Datasets from FiveThirtyEight, we have downloaded a subset of this data in the folder ./assets (assets)
  - The original dataset can be found at FiveThirtyEight Star Wars Survey (https://github.com/fivethirtyeight/data/tree/master/star-wars-survey)

Important notes:

1) Grading for this assignment is entirely done by manual inspection. Focus on getting the visualization to look like our example. It doesn't need to be pixel perfect (e.g., you may not always know what our example is scaled by), but it should be pretty close. Hint: go back to lab in week 2 on altair for some styling help. A *lot* of the look and feel can be done in one line of code.

2) There will be a couple of places where the numbers you get when you select rows may be a little different than 538, but the percents should still work (e.g., 828 instead of 834). You'll see this in our examples. If you can somehow get the data to match exactly, that's great too.

3) When turning in your PDF, please use the File -> Print -> Save as PDF option ***from your browser***. Do ***not*** use the File->Download as->PDF option. Complete instructions for this are under Resources in the Coursera page for this class.

Part 1. Perception and Cognition (30 points)

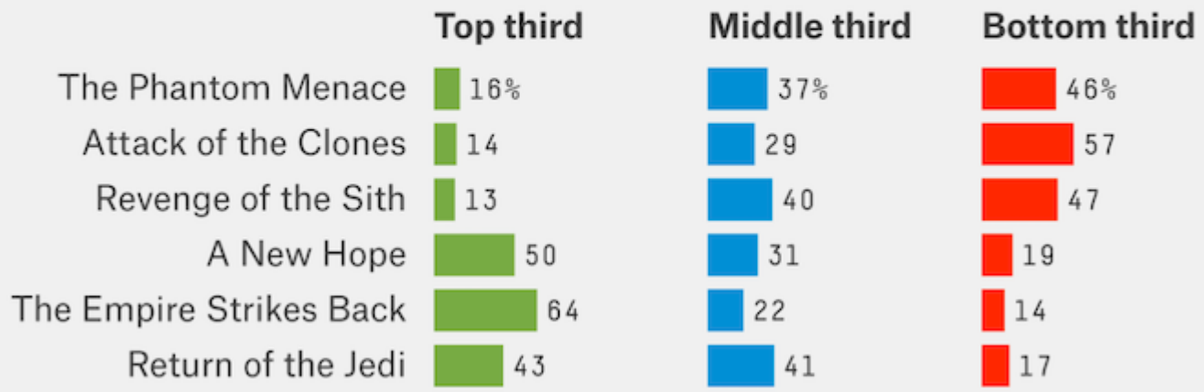
Read the article "America's Favorite 'Star Wars' Movies (And Least Favorite Characters)," (https://fivethirtyeight.com/features/americas-favorite-star-wars-movies-and-least-favorite-characters/) and answer the following questions:

1.1 List the different data types in the following visualizations and their encodings (10 points)

Look at the following visualizations. For each, list the variable, their type, and the encoding used (e.g., Weight, quantitative, color, ...)

How People Rate the 'Star Wars' Movies

How often each film was rated in the top, middle and bottom third (by 471 respondents who have seen all six films)



FIVETHIRTYEIGHT SOURCE: SURVEYMONKEY AUDIENCE

Which 'Star Wars' Movies Have You Seen?

Of 835 respondents who have seen any film

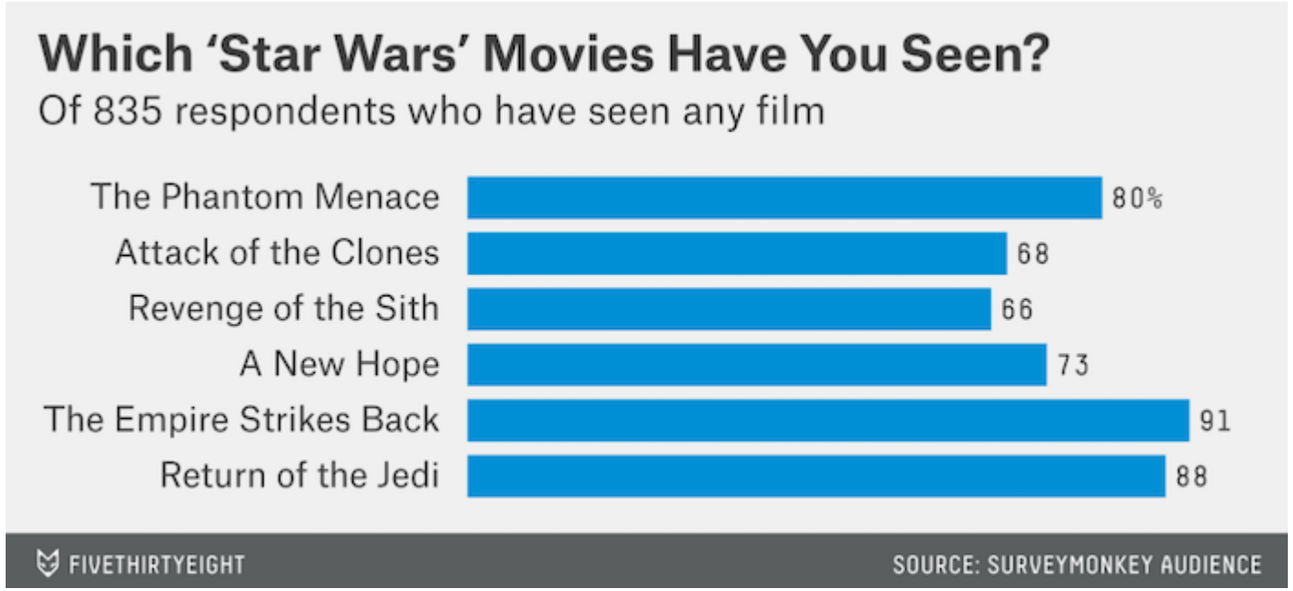
FIVETHIRTYEIGHT SOURCE: SURVEYMONKEY AUDIENCE

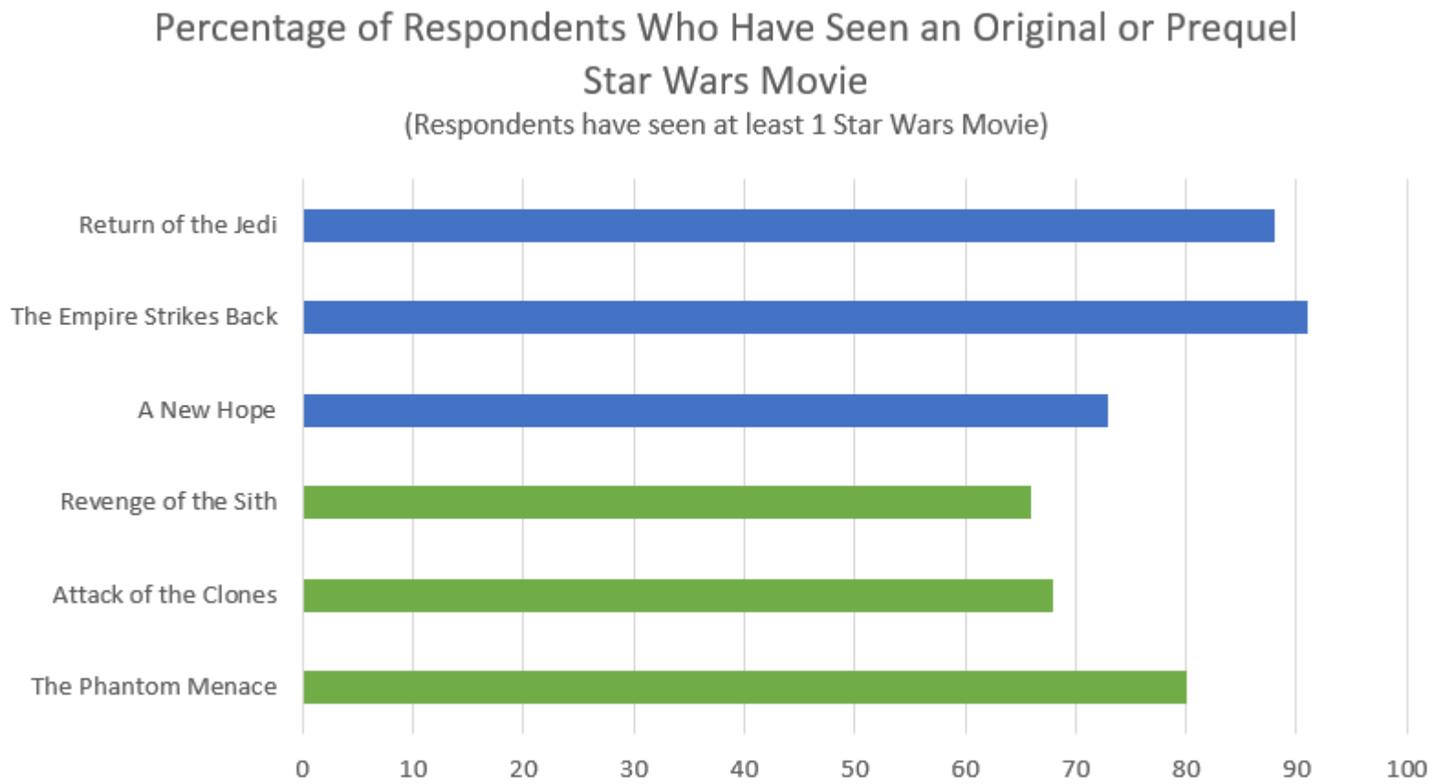
For the first figure above, the six Star Wars movies are nominal variables that utilized a position encoding and are displayed on the far left of the figure. The variable "percentage of time where the particular movie was in the top third of respondent choices" is a quantitative variable and it's encoded using not only the value itself but with length (bar graph length specifically) as well as its position on the x-axis. Additionally, stylistically, color is also used to denote that the bars correspond to this measurement of top third. Similarly, for the middle third and bottom third variables, these are also quantitative variables encoded using the value itself, length, and position, and color again denotes which category the bar corresponds to.

In the second figure above, the six movies are again nominal variables that have been encoded based on position in the far left. The percentage of respondents who have seen each film is a quantitative variable that was again encoded based on length of the bar and the value itself is presented alongside this length as well to assist in the effectiveness of the figure. This quantitative variable is also encoded based on position on the x-axis.

1.2 Propose an alternative encoding for the following visualization. Compare the visualizations based on perception. (10 points)

Either hand-draw or use an application to create a sketched solution. Upload an image and describe the differences between your solution and the FiveThirtyEight image in terms of perception (specifically for the task of comparing one movie to another).



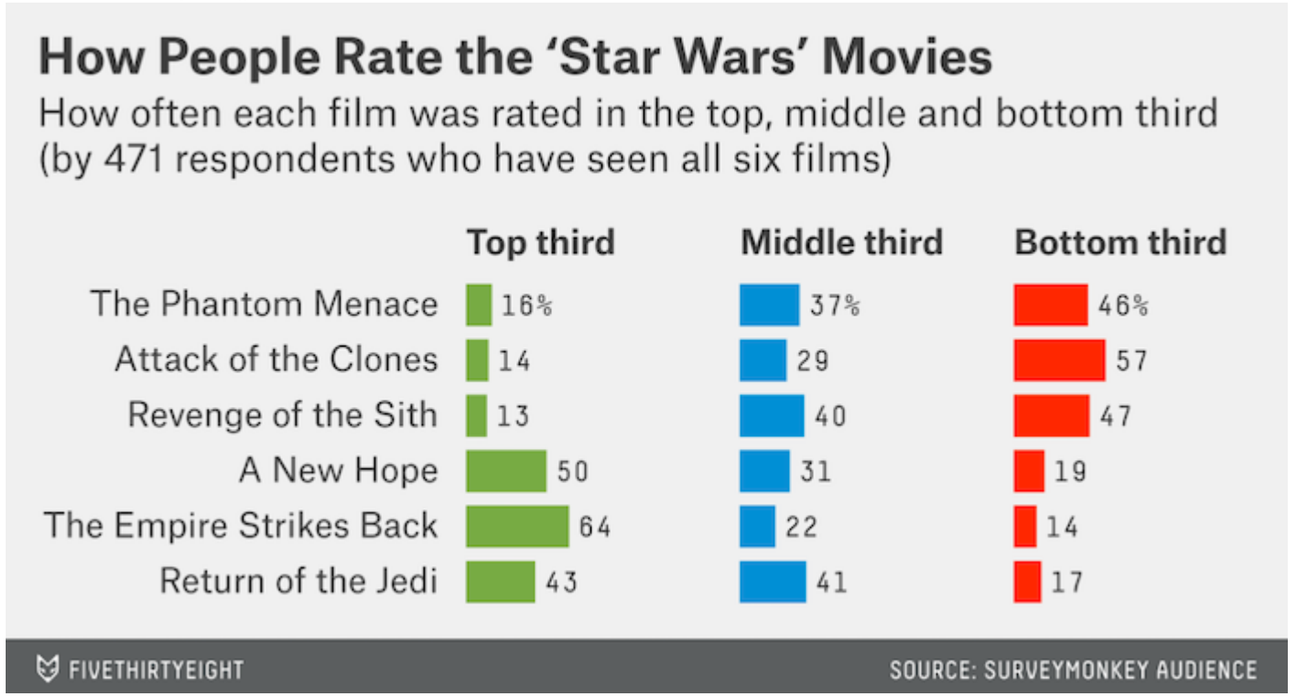


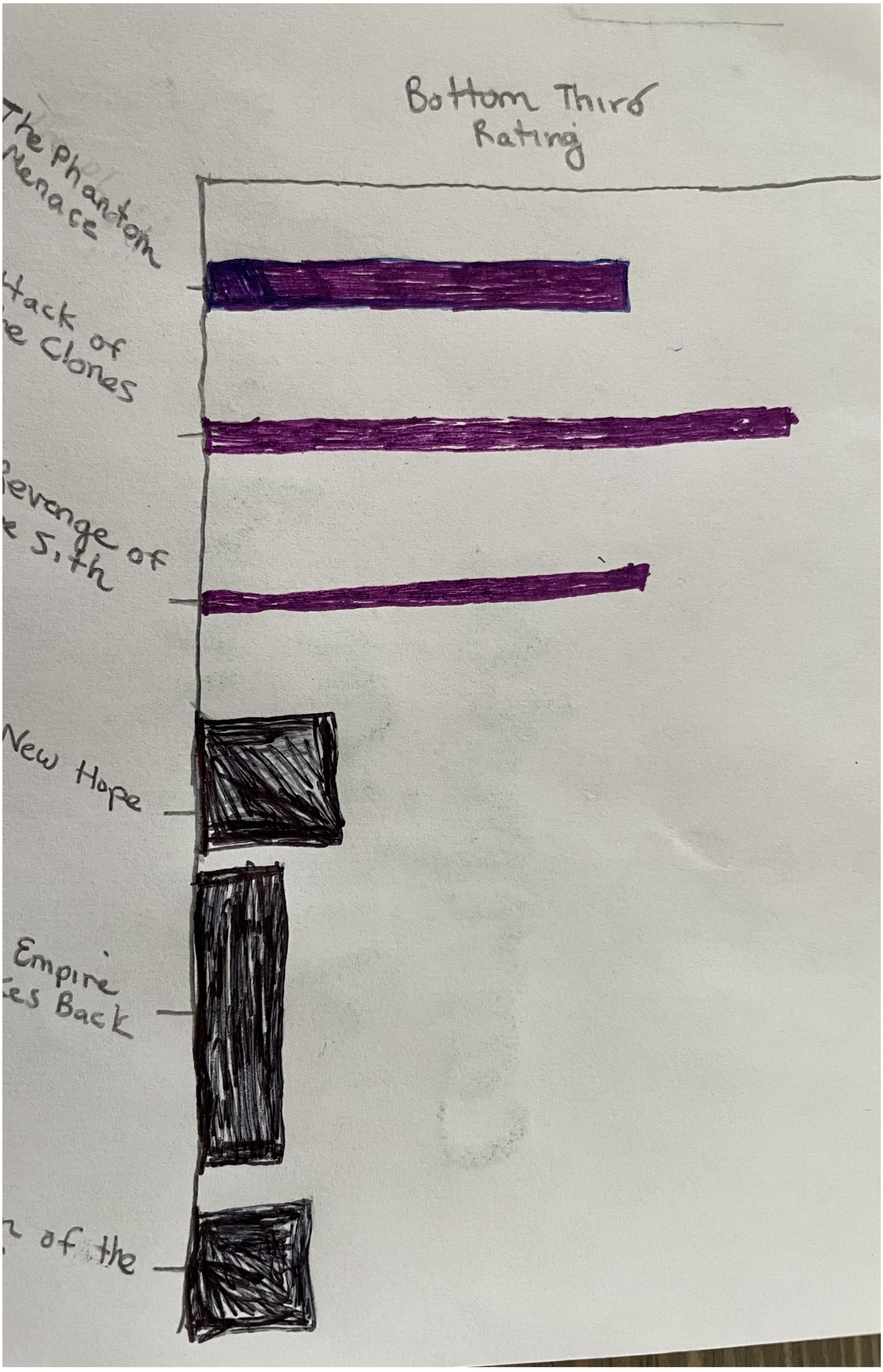
The above created image is very similar in format to the FiveThirtyEight figure in which the 6 Star Wars movies are nominal variables encoded on the far left and the percentage of respondents who have seen the movie is a quantitative variable encoded based on length. In terms of perception, however, the newly created image is utilizing the Gestalt principle of Similarity. Based on the use of color, the viewer should be able to discern that Revenge of the Sith, Attack of the Clones, and The Phantom Menace are grouped together (being the prequel trilogy), and A New Hope, The Empire Strikes Back, and Return of the Jedi are tied together as the original trilogy. This use of color is thus an easy way for the presenter to show the viewer that these movies can be considered a close pair. Preattentive processing is also utilized here, as the viewer can quickly look at this figure and know that the different colors will be related to each of the two trilogies under investigation, which is not possible with the original figure.

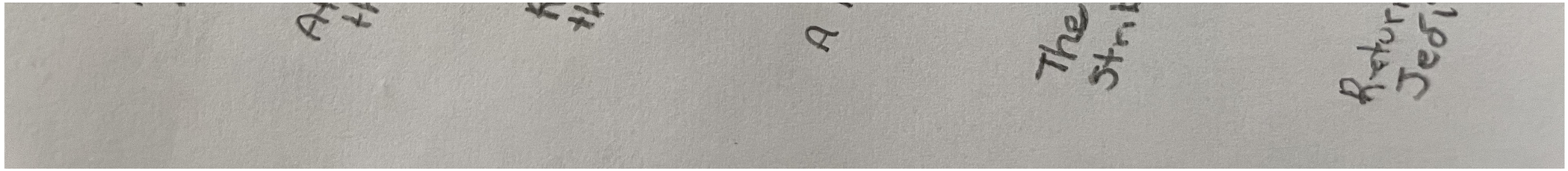
This breakout of the two trilogies also allows for a quick finding that the original trilogy movies are more watched relative to the prequel trilogy (with the exception of the Phantom Menace over a New Hope).

1.3 Propose an alternative encoding for the following visualization. Compare the visualizations based on perception. (10 points)

Again, either-hand draw or use an application to create a sketched solution. Upload an image and describe the differences between your solution and the FiveThirtyEight image in terms of perception (specifically for the task of comparing one movie to another).




Bottom Third Rating
The Phantom Menace
Attack of the Clones
Revenge of the Sith
New Hope
Empire Strikes Back
Return of the



The above visualization takes a different approach from the FiveThirtyEight figure and focuses on the Top Third and Bottom Third percentages. The six movies are again nominal variables and are available on the x-axis. The height of the figure is based on the percentage where the movie is rated in the bottom third. As can be seen, the prequel trilogy movies, when compared to the original trilogy, more often found themselves in the bottom third. Relatedly, the width of the bar is based on the percentage where the movie is rated in the top third. Again, the original trilogy is far more often found in the top third, and thus the bar is wider when compared to prequel trilogy movies. In terms of perception, this is actually an example of multiple encoding, specifically where x-size and y-size are both used to denote a particular quantitative variable. When utilizing x-size and y-size this typically isn't a preferred approach as these types of encodings are much more integral to each other as opposed to something like color and location, which are a bit more separable. However, despite this, this new figure still I believe accomplishes the goal in displaying that the original trilogy typically is the preferred choice of most movie watchers when compared to the prequel trilogy movies.

Part 2. Altair programming exercise (70 points)

We have provided you with some code and parts of the article America's Favorite 'Star Wars' Movies (And Least Favorite Characters) (https://fivethirtyeight.com/features/americas-favorite-star-wars-movies-and-least-favorite-characters/). This article is based on the dataset:

1. StarWars (data/StarWars.csv) Created by FiveThirtyEight based on a survey ran through SurveyMonkey Audience, surveying 1,186 respondents from June 3 to 6 2014. Available [online] (https://github.com/fivethirtyeight/data/tree/master/star-wars-survey (https://github.com/fivethirtyeight/data/tree/master/star-wars-survey))

To earn points for this assignment, you must:

- Recreate the visualizations in the article (replace the images in the article with a code cell that creates a visualization). We provide one example. Each visualization is worth 10 points (40 points/ 10 each x 4 total).
  - *Partial credit can be granted for each visualization (up to 5 points) if you provide the grammar of graphics description of the visualization without a functional Altair implementation*
- Propose one alternative visualization for one of the article visualizations. Add a short paragraph describing why your visualization is more *effective* based on principles of perception/cognition. (15 points/ 10 points plot + 5 justification)
- Propose a new visualization to complement a part of the article. Add a short paragraph justifying your decisions in terms of Perception/Cognition processes. (15 points/ 10 points plot + 5 justification)

In [1]:
```
import pandas as pd
import altair as alt
import numpy as np
import math
```

In [2]:
```
# enable correct rendering
alt.renderers.enable('default')
```

Out[2]:
```
RendererRegistry.enable('default')
```

In [3]:
```
# uses intermediate json files to speed things up
alt.data_transformers.enable('json')
```

Out[3]:
```
DataTransformerRegistry.enable('json')
```

In [4]:
```
sw = pd.read_csv('assets/StarWars.csv', encoding='latin1')
```

In [5]:
```
# Some format is needed for the survey dataframe, we provide the formatted dataset in a dataframe
sw = sw.rename(columns={'Have you seen any of the 6 films in the Star Wars franchise?':'seen_any_movie',
                        'Do you consider yourself to be a fan of the Star Wars film franchise?': 'fan',
                        'Which of the following Star Wars films have you seen? Please select all that apply.' : 'seen_EI',
                        'Unnamed: 4' : 'seen_EII',
                        'Unnamed: 5' : 'seen_EIII',
                        'Unnamed: 6' : 'seen_EIV',
                        'Unnamed: 7' : 'seen_EV',
                        'Unnamed: 8' : 'seen_EVI',
                        'Please rank the Star Wars films in order of preference with 1 being your favorite film in the franchise and 6 being your lea
                        'Unnamed: 10' : 'rank_EII',
                        'Unnamed: 11' : 'rank_EIII',
                        'Unnamed: 12' : 'rank_EIV',
                        'Unnamed: 13' : 'rank_EV',
                        'Unnamed: 14' : 'rank_EVI',
                        'Please state whether you view the following characters favorably, unfavorably, or are unfamiliar with him/her.' : 'Han Solo'
                        'Unnamed: 16' : 'Luke Skywalker',
                        'Unnamed: 17' : 'Princess Leia Organa',
                        'Unnamed: 18' : 'Anakin Skywalker',
                        'Unnamed: 19' : 'Obi Wan Kenobi',
                        'Unnamed: 20' : 'Emperor Palpatine',
                        'Unnamed: 21' : 'Darth Vader',
                        'Unnamed: 22' : 'Lando Calrissian',
                        'Unnamed: 23' : 'Boba Fett',
                        'Unnamed: 24' : 'C-3P0',
                        'Unnamed: 25' : 'R2 D2',
                        'Unnamed: 26' : 'Jar Jar Binks',
                        'Unnamed: 27' : 'Padme Amidala',
                        'Unnamed: 28' : 'Yoda',
                       })
sw = sw.drop([0])
```

In [6]:
```
# take a peak to look at the data
sw.sample(5)
```

Out[6]:

	RespondentID	seen_any_movie	fan	seen_EI	seen_EII	seen_EIII	seen_EIV	seen_EV	seen_EVI	rank_EI	...	Yoda	Which character shot first?	Are you familiar with the Expanded Universe?	Do you consider yourself to be a fan of the Expanded Universe?□æ	Do you consider yourself to be a fan of the Star Trek franchise?	Gender
1164	3.288417e+09	Yes	No	NaN	NaN	NaN	Star Wars: Episode IV A New Hope	Star Wars: Episode V The Empire Strikes Back	Star Wars: Episode VI Return of the Jedi	5	...	Very favorably	Han	Yes	No	No	Female
745	3.289812e+09	Yes	Yes	NaN	NaN	NaN	NaN	Star Wars: Episode V The Empire Strikes Back	Star Wars: Episode VI Return of the Jedi	1	...	Very favorably	Han	Yes	Yes	Yes	NaN
863	3.289526e+09	Yes	No	Star Wars: Episode I The Phantom Menace	NaN	NaN	Star Wars: Episode IV A New Hope	Star Wars: Episode V The Empire Strikes Back	Star Wars: Episode VI Return of the Jedi	4	...	Very favorably	I don't understand this question	No	NaN	No	Female
669	3.289972e+09	No	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	Yes	Female
628	3.290058e+09	Yes	Yes	Star Wars: Episode I The Phantom Menace	Star Wars: Episode II Attack of the Clones	Star Wars: Episode III Revenge of the Sith	Star Wars: Episode IV A New Hope	Star Wars: Episode V The Empire Strikes Back	Star Wars: Episode VI Return of the Jedi	2	...	Very favorably	I don't understand this question	No	NaN	No	Female

5 rows × 38 columns

America's Favorite 'Star Wars' Movies (And Least Favorite Characters)

Original article available at FiveThirtyEight (https://fivethirtyeight.com/features/americas-favorite-star-wars-movies-and-least-favorite-characters/)

By Walt Hickey (https://fivethirtyeight.com/contributors/walt-hickey/)

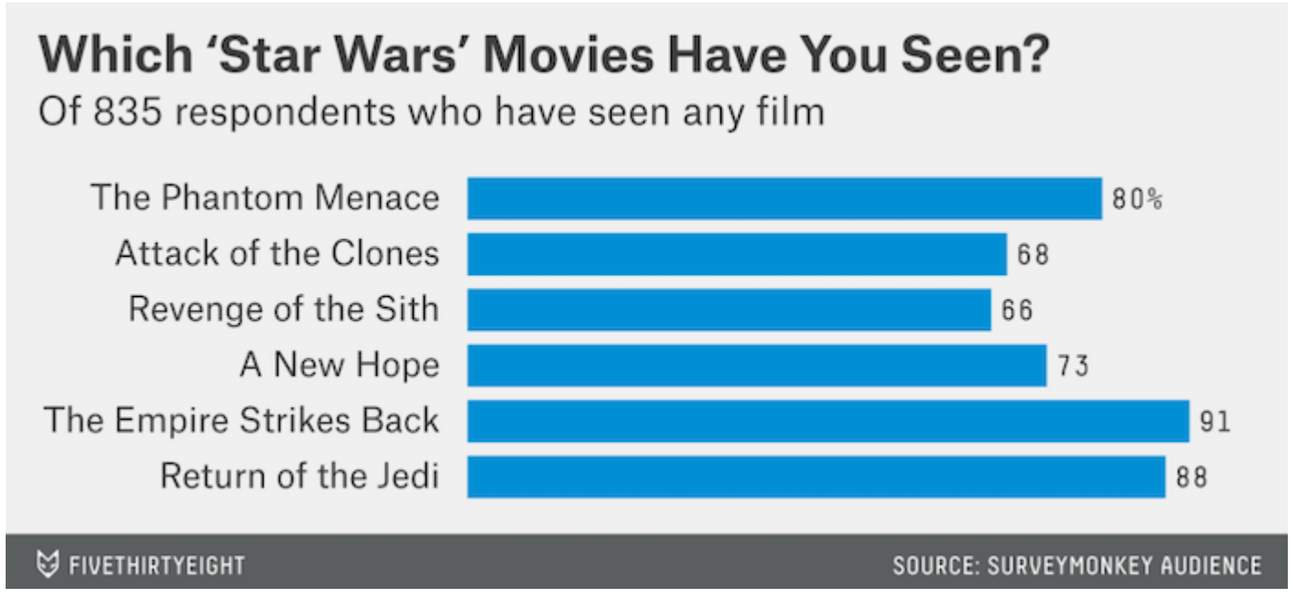
Filed under Movies (https://fivethirtyeight.com/tag/movies/)

Get the data on GitHub (https://github.com/fivethirtyeight/data/tree/master/star-wars-survey)

This week, I caught a sneak peek of the X-Wing fighter (http://www.wired.com/2014/07/star-wars-episode-vii-x-wing/) from the new "Star Wars" films in production. The forthcoming movies — and the middling response to the most recent trilogy — provide a perfect excuse to examine some questions I've long wanted answers to: How many people are "Star Wars" fans? Does the rest of America realize that "The Empire Strikes Back" is clearly the best of the bunch? Which characters are most well-liked and most hated? And who shot first, Han Solo or Greedo?

We ran a poll through SurveyMonkey Audience (https://www.surveymonkey.com/mp/audience/), surveying 1,186 respondents from June 3 to 6 (the data (https://github.com/fivethirtyeight/data/tree/master/star-wars-survey) is available on GitHub (https://github.com/fivethirtyeight/data)). Seventy-nine percent of those respondents said they had watched at least one of the "Star Wars" films. This question, incidentally, had a substantial difference by gender: 85 percent of men have seen at least one "Star Wars" film compared to 72 percent of women. Of people who have seen a film, men were also more likely to consider themselves a fan of the franchise: 72 percent of men compared to 60 percent of women.

We then asked respondents which of the films they had seen. With 835 people responding, here's the probability that someone has seen a given "Star Wars" film given that they have seen any Star Wars film:



In [7]:
```
# Sample visualization

# We're going to fix the labels a bit so will create a mapping to the full names
episodes = ['EI', 'EII', 'EIII', 'EIV', 'EV', 'EVI']
names = {
    'EI' : 'The Phantom Menance', 'EII' : 'Attack of the Clones', 'EIII' : 'Revenge of the Sith',
    'EIV': 'A New Hope', 'EV': 'The Empire Strikes Back', 'EVI' : 'The Return of the Jedi'
}

# we're also going to use this order to sort, so names_l will now have our sort order
names_l = [names[ep] for ep in episodes]

print("sort order: ",names_l)
```

```
sort order:  ['The Phantom Menance', 'Attack of the Clones', 'Revenge of the Sith', 'A New Hope', 'The Empire Strikes Back', 'The Return of the Jed
i']
```

In [8]:
```
# let's do some data pre-processing... sw (star wars) has everything

# We want to only use those people who have seen at least one movie, let's get the people, toss NAs
# and get the total count

# find people who have at least on of the columns (seen_*) not NaN
seen_at_least_one = sw.dropna(subset=['seen_' + ep for ep in episodes],how='all')
total = len(seen_at_least_one)

print("total who have seen at least one: ", total)
```

```
total who have seen at least one:  835
```

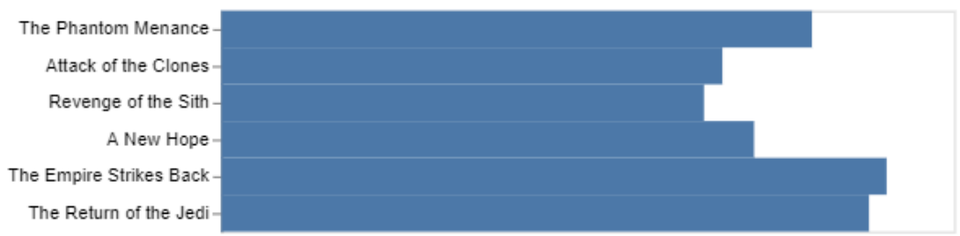
In [9]:
```
# for each movie, we're going to calculate the percents and generate a new data frame
percs = []

# loop over each column and calculate the number of people who have seen the movie
# specifically, filter out the people who are *NaN* for a specific episode (e.g., ep_EII), count them
# and divide by the percent
for seen_ep in ['seen_' + ep for ep in episodes]:
    perc = len(seen_at_least_one[~ pd.isna(seen_at_least_one[seen_ep])]) / total
    percs.append(perc)

# at this point percs is holding our percentages

# now we're going use a trick to make tuples--pairing names with percents--using "zip" and then make a dataframe
tuples = list(zip([names[ep] for ep in episodes],percs))
seen_per_df = pd.DataFrame(tuples, columns = ['Name', 'Percentage'])
seen_per_df
```

Out[9]:

	Name	Percentage
0	The Phantom Menance	0.805988
1	Attack of the Clones	0.683832
2	Revenge of the Sith	0.658683
3	A New Hope	0.726946
4	The Empire Strikes Back	0.907784
5	The Return of the Jedi	0.883832

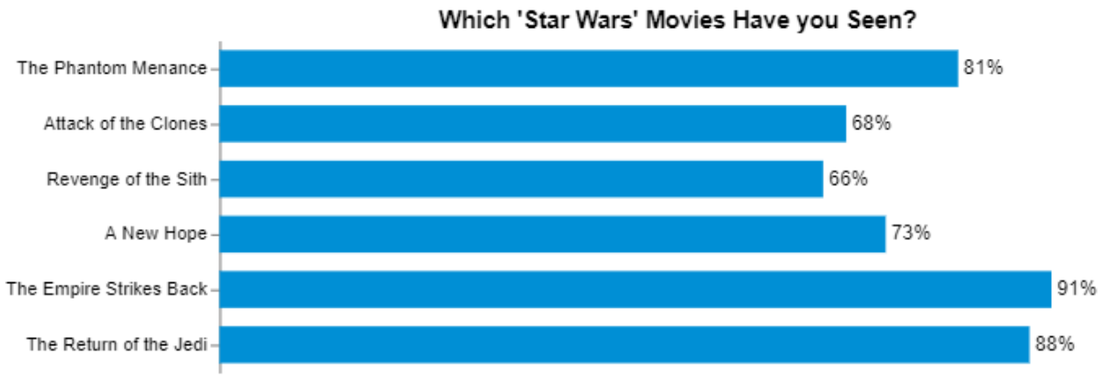
In [10]:
```
# ok, time to make the chart... let's make a bar chart (use mark_bar)
bars = alt.Chart(seen_per_df).mark_bar(size=20).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage',
        axis=None),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         # we give the sorting order to avoid alphabetical order
         sort=names_l
    )
)

# at this point we don't really have a great plot (it's missing the annotations, titles, etc.)
bars
```

Out[10]:



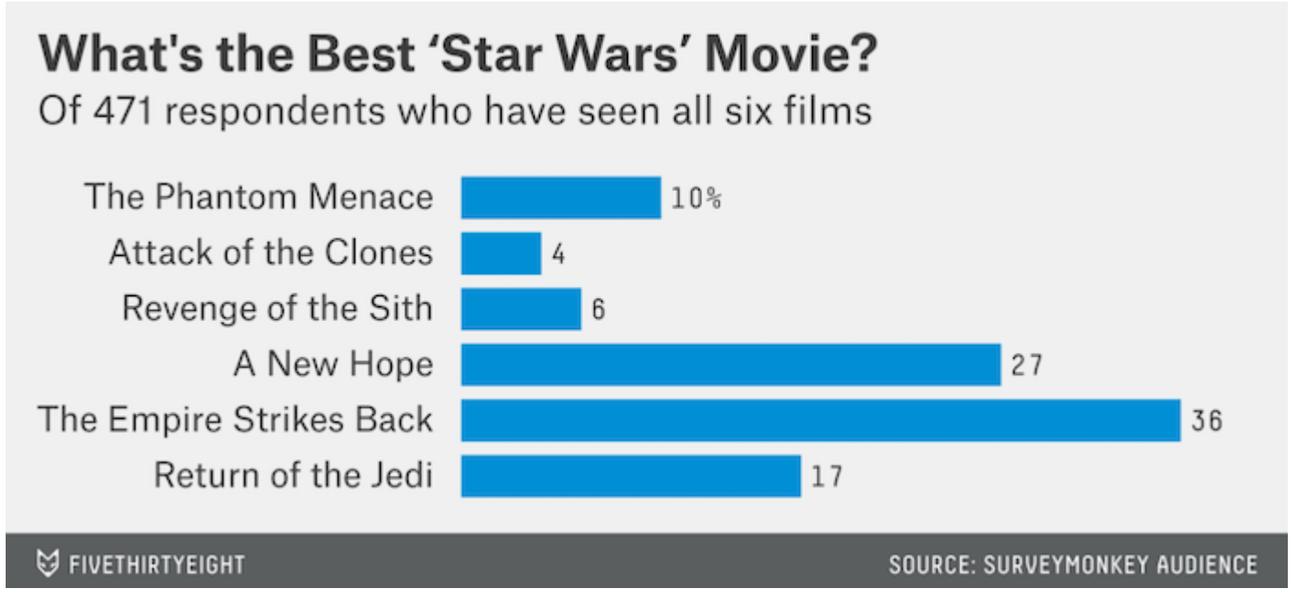
In [11]:
```
# we're going to overlay the text with the percentages, so let's make another visualization
# that's just text labels

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(
    # we'll use the percentage as the text
    text=alt.Text('Percentage:Q',format='.0%')
)

# finally, we're going to combine the bars and the text and do some styling
seen_movies = (text + bars).configure_mark(
    # we don't love the blue
    color='#008fd5'
).configure_view(
    # we don't want a stroke around the bars
    strokeWidth=0
).configure_scale(
    # add some padding
    bandPaddingInner=0.2
).properties(
    # set the dimensions of the visualization
    width=500,
    height=180
).properties(
    # add a title
    title="Which 'Star Wars' Movies Have you Seen?"
)

seen_movies

# note that we are NOT formatting this in the Five Thirty Eight Style yet... we'll leave that to you to figure out
```

Out[11]:



So we can see that “Star Wars: Episode V — The Empire Strikes Back” is the film seen by the most number of people, followed by “Star Wars: Episode VI — Return of the Jedi.” Appallingly, more people reported seeing “Star Wars: Episode I — The Phantom Menace” than the original “Star Wars” (renamed “Star Wars: Episode IV — A New Hope”).

So, which movie is the best? We asked the subset of 471 respondents who indicated they have seen every “Star Wars” film to rank them from best to worst. From that question, we calculated the share of respondents who rated each film as their favorite.



** Homework note: Click here (assets/best_movie.png) to see a version of this plot generated in Altair.

2.1 What's the best 'Star Wars' movie? Recreate the above image using altair (10 POINTS)

In [12]:
```
# Recreate this image using Altair
# try to match the "538 style" as best you can (hint: look at the altair lab at the start of the semester)

# find people who have none of the columns (seen_*) as NaN
seen_all = sw.dropna(subset=['seen_' + ep for ep in episodes], how='any')
total_21 = len(seen_all)

# for each movie, find the percentages where it's rated #1
percents_21 = []

for rank_ep in ['rank_' + ep for ep in episodes]:
    percent_21 = len(seen_all[seen_all[rank_ep] == '1']) / total_21
    percents_21.append(percent_21)

tuples_21 = list(zip([names[ep] for ep in episodes],percents_21))
seen_all_df = pd.DataFrame(tuples_21, columns = ['Name', 'Percentage'])
seen_all_df

#raise NotImplementedError()
```

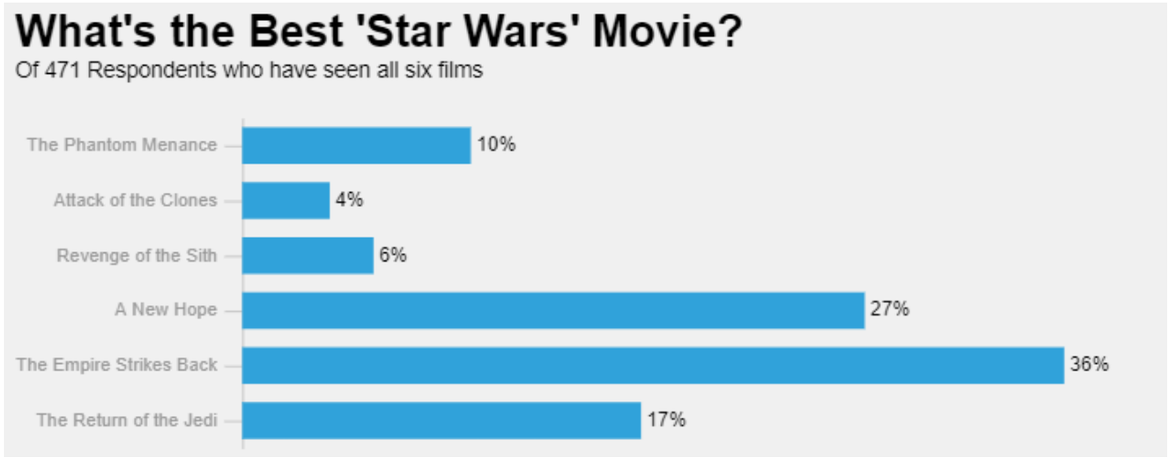
Out[12]:

	Name	Percentage
0	The Phantom Menance	0.099788
1	Attack of the Clones	0.038217
2	Revenge of the Sith	0.057325
3	A New Hope	0.271762
4	The Empire Strikes Back	0.358811
5	The Return of the Jedi	0.174098

In [13]:

```
alt.themes.enable('fivethirtyeight')

bars_21 = alt.Chart(seen_all_df).mark_bar(size=20).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage',
        axis=None),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         sort=names_l
    )
)


text_21 = bars_21.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('Percentage:Q',format='.0%')
)

seen_all_movies = (text_21 + bars_21).configure_mark(
    color='#008fd5'
).configure_view(
    strokeWidth=0
).configure_scale(
    bandPaddingInner=0.2
).properties(
    # set the dimensions of the visualization
    width=500,
    height=180
).properties(
      title={
      "text": ["What's the Best 'Star Wars' Movie?"],
      "subtitle": ["Of 471 Respondents who have seen all six films"]
      }
)

seen_all_movies
```

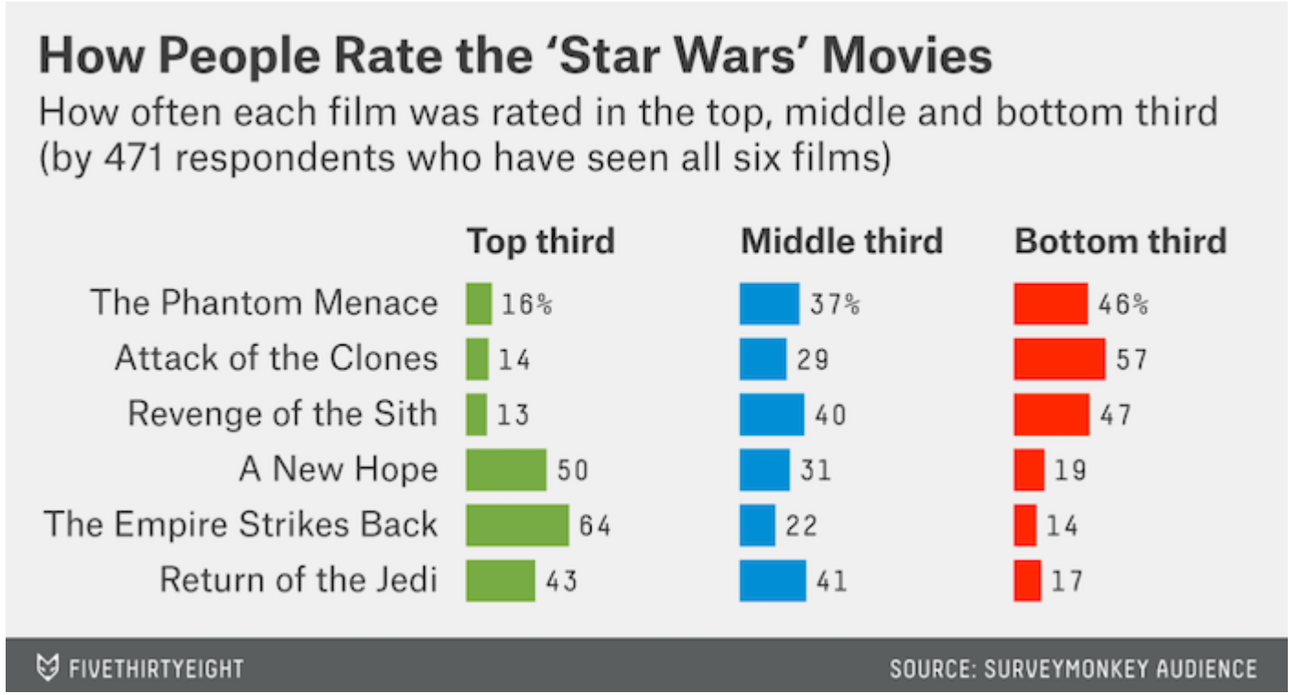
Out[13]:



Make sure to *style* your visualization to match the original the best you can

We can also drill down and find out, generally, how people rate the films. Overall, fans broke into two camps: those who preferred the original three movies and those who preferred the three prequels. People who said “The Empire Strikes Back” was their favorite were also likely to rate “A New Hope” and “Return of the Jedi” higher as well. Those who rated “The Phantom Menace” as the best film were more likely to rate prequels higher.

This chart shows how often each film was rated in the top third (best or second-best), the middle third (third or fourth) or the bottom third (second-worst or worst). It’s a more nuanced take on the series:



** Homework note: Click here (assets/people_rate.png) to see a version of this plot generated in Altair.

2.2 How people rate the 'Star Wars' movie? Recreate the above image using altair (10 POINTS)

In [14]:
```
# Recreate this image using altair here (10 POINTS)
total_22 = len(seen_all)

# for each movie, find the percentages where it's rated top third, middle third, or bottom third
top_22 = []
middle_22 = []
bottom_22 = []

for rank_ep in ['rank_' + ep for ep in episodes]:
    top = len(seen_all[seen_all[rank_ep].isin(['1', '2'])]) / total_22
    top_22.append(top)

    middle = len(seen_all[seen_all[rank_ep].isin(['3', '4'])]) / total_22
    middle_22.append(middle)

    bottom = len(seen_all[seen_all[rank_ep].isin(['5', '6'])]) / total_22
    bottom_22.append(bottom)

tuples_22 = list(zip([names[ep] for ep in episodes], top_22, middle_22, bottom_22))
rank_df = pd.DataFrame(tuples_22, columns = ['Name', 'Top third', 'Middle third', 'Bottom third'])

rank_df = pd.melt(rank_df, id_vars='Name', var_name='Bucket', value_name='Percentage')
rank_df

colors = []
for value in rank_df['Bucket']:
    if value == 'Top third':
        colors.append('green')
    elif value == 'Middle third':
        colors.append('blue')
    else:
        colors.append('red')

rank_df['Color'] = colors
rank_df
```

Out[14]:

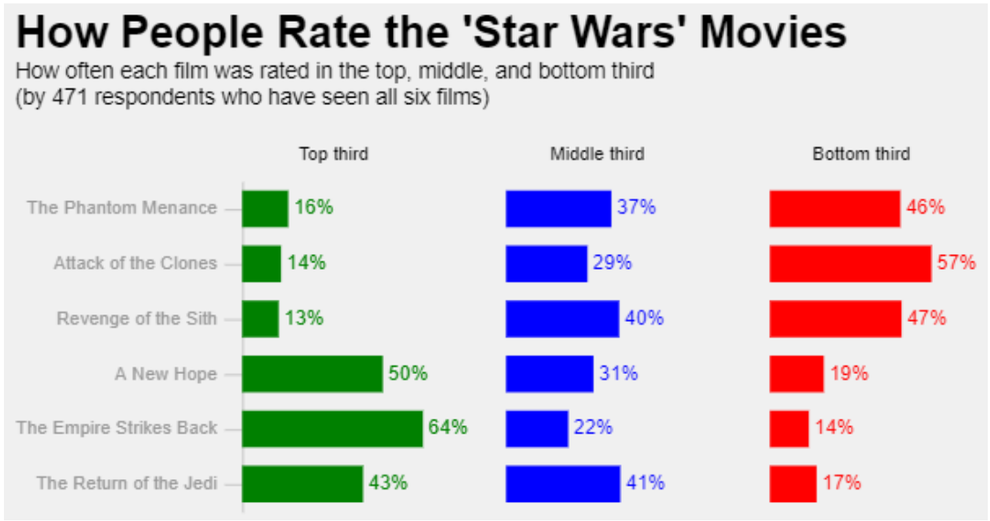
	Name	Bucket	Percentage	Color
0	The Phantom Menance	Top third	0.163482	green
1	Attack of the Clones	Top third	0.138004	green

	Name	Bucket	Percentage	Color
2	Revenge of the Sith	Top third	0.129512	green
3	A New Hope	Top third	0.498938	green
4	The Empire Strikes Back	Top third	0.641189	green
5	The Return of the Jedi	Top third	0.428875	green
6	The Phantom Menance	Middle third	0.373673	blue
7	Attack of the Clones	Middle third	0.288747	blue
8	Revenge of the Sith	Middle third	0.401274	blue
9	A New Hope	Middle third	0.309979	blue
10	The Empire Strikes Back	Middle third	0.220807	blue
11	The Return of the Jedi	Middle third	0.405520	blue
12	The Phantom Menance	Bottom third	0.462845	red
13	Attack of the Clones	Bottom third	0.573248	red
14	Revenge of the Sith	Bottom third	0.467091	red
15	A New Hope	Bottom third	0.191083	red
16	The Empire Strikes Back	Bottom third	0.138004	red
17	The Return of the Jedi	Bottom third	0.165605	red

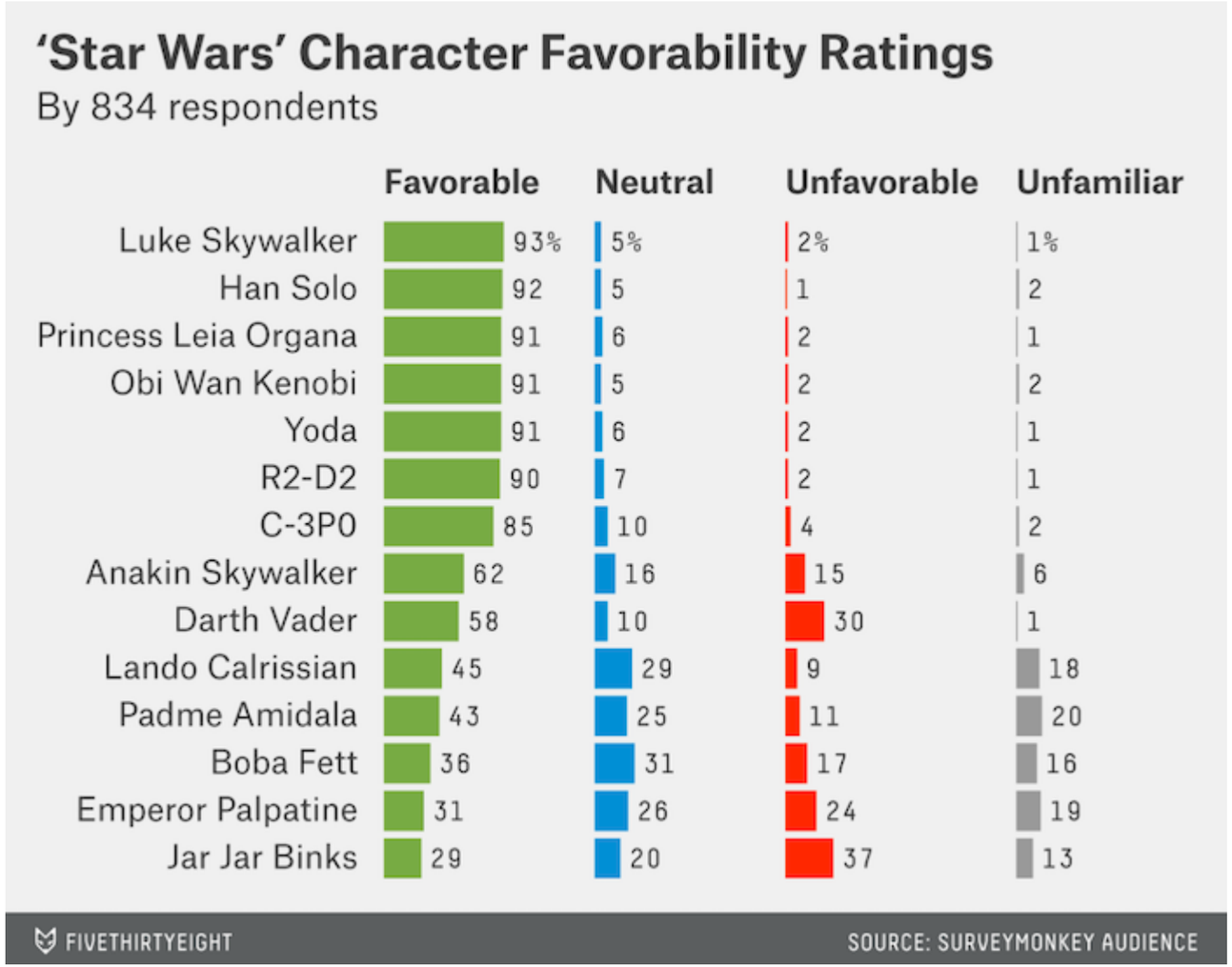
In [15]:

```
bars_22 = alt.Chart().mark_bar(size=20).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage',
        axis=None),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         sort=names_l
    ),
    color = alt.Color('Color:N', scale = None)
)


text_22 = bars_22.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('Percentage:Q',format='.0%')
)

rank_movies = alt.layer(bars_22, text_22, data=rank_df
    ).properties(width = 100, height = 180).facet(
    column= alt.Column('Bucket:N', title = '', sort = 'descending'),
    title={
      "text": ["How People Rate the 'Star Wars' Movies"],
      "subtitle": ["How often each film was rated in the top, middle, and bottom third",
                   "(by 471 respondents who have seen all six films)"]
      }
).configure_view(
    strokeWidth=0)

rank_movies
```

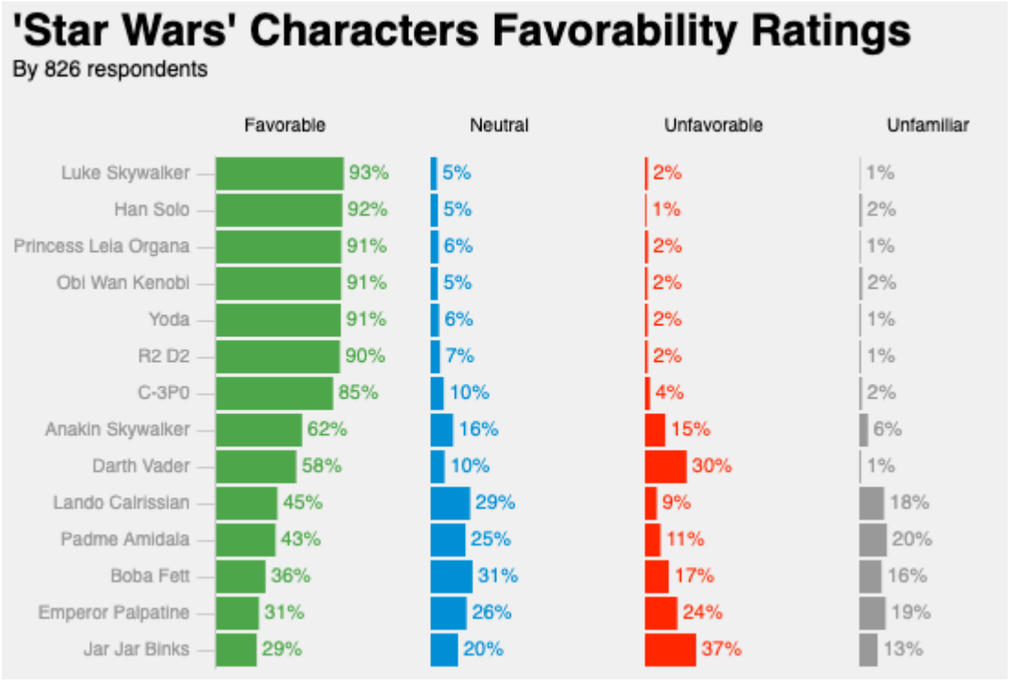
Out[15]:



Finally, we took a boilerplate format used by political favorability polls — “Please state whether you view the following characters favorably, unfavorably, or are unfamiliar with him/her” — and asked respondents to rate characters in the series.



** Homework note. Here's an example solution generated in Altair:



2.3 Star Wars' Characters Favorability Ratings. Recreate the above image using altair (10 POINTS)

In [16]:
```
# Recreate this image using altair here (10 POINTS)
characters = ['Han Solo', 'Luke Skywalker', 'Princess Leia Organa', 'Anakin Skywalker',
       'Obi Wan Kenobi', 'Emperor Palpatine', 'Darth Vader', 'Lando Calrissian', 'Boba Fett',
        'C-3P0', 'R2 D2', 'Jar Jar Binks', 'Padme Amidala', 'Yoda']

total_23 = len(seen_at_least_one)

# for each movie, find the percentages where it's rated top third, middle third, or bottom third
favs_23 = []
neutrals_23 = []
unfavs_23 = []
unfams_23 = []

for character in characters:
    fav_23 = len(seen_at_least_one[seen_at_least_one[character].isin(['Very favorably', 'Somewhat favorably'])]) / total_23
    favs_23.append(fav_23)

    neutral_23 = len(seen_at_least_one[seen_at_least_one[character].isin(['Neither favorably nor unfavorably (neutral)'])]) / total_23
    neutrals_23.append(neutral_23)

    unfav_23 = len(seen_at_least_one[seen_at_least_one[character].isin(['Very unfavorably', 'Somewhat unfavorably'])]) / total_23
    unfavs_23.append(unfav_23)

    unfam_23 = len(seen_at_least_one[seen_at_least_one[character].isin(['Unfamiliar (N/A)'])]) / total_23
    unfams_23.append(unfam_23)

tuples_23 = list(zip(characters, favs_23, neutrals_23, unfavs_23, unfams_23))
fav_df = pd.DataFrame(tuples_23, columns = ['Name', 'Favorable', 'Neutral', 'Unfavorable', 'Unfamiliar'])

fav_df = pd.melt(fav_df, id_vars='Name', var_name='View', value_name='Percentage')

colors_23 = []
for value in fav_df['View']:
    if value == 'Favorable':
        colors_23.append('green')
    elif value == 'Neutral':
        colors_23.append('blue')
    elif value == 'Unfavorable':
        colors_23.append('red')
    else:
        colors_23.append('grey')
```

```
fav_df['Color'] = colors_23

sort_fav = fav_df[fav_df['View'] == 'Favorable']
sort_fav.sort_values(by = ['Percentage'], ascending = False, inplace = True)

fav_list = list(sort_fav['Name'])
fav_list

fav_df.head()

#raise NotImplementedError()
```

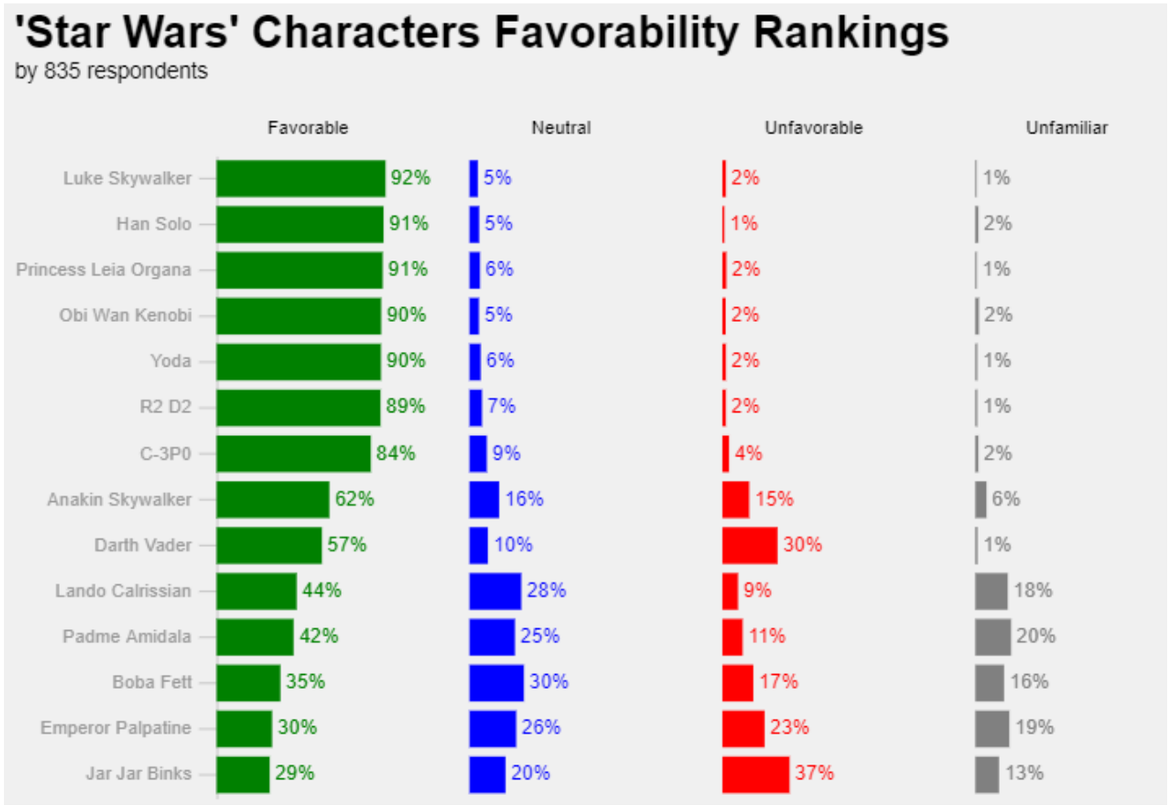
Out[16]:

	Name	View	Percentage	Color
0	Han Solo	Favorable	0.911377	green
1	Luke Skywalker	Favorable	0.922156	green
2	Princess Leia Organa	Favorable	0.906587	green
3	Anakin Skywalker	Favorable	0.615569	green
4	Obi Wan Kenobi	Favorable	0.898204	green

In [17]:

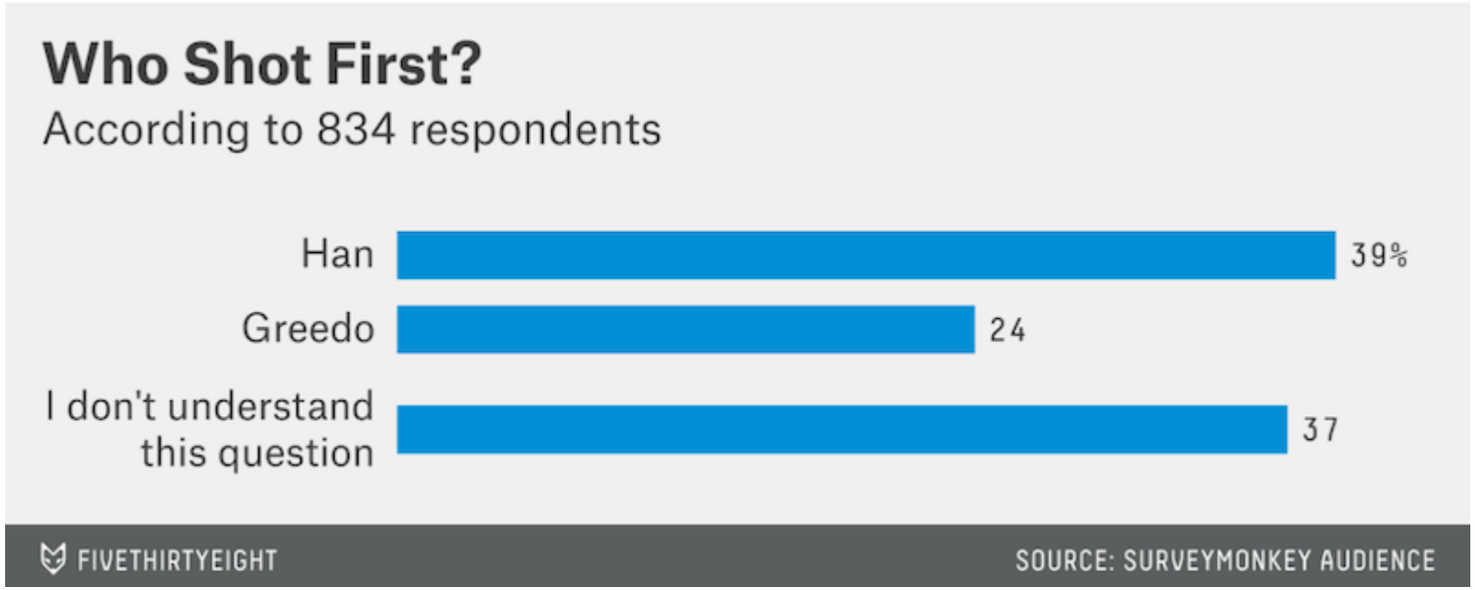
```
bars_23 = alt.Chart().mark_bar(size=20).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage',
        axis=None),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         sort= fav_list
    ),
    color = alt.Color('Color:N', scale = None)
)


text_23 = bars_23.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('Percentage:Q',format='.0%')
)

fav_movies = alt.layer(bars_23, text_23, data=fav_df
    ).properties(width = 100, height = 350).facet(
    column= alt.Column('View:N', title = '', sort = ['Favorable', 'Neutral', 'Unfavorable', 'Unfamiliar']),
    title={
      "text": ["'Star Wars' Characters Favorability Rankings"],
      "subtitle": ["by 835 respondents"]
      }
).configure_view(
    strokeWidth=0)

fav_movies
```

Out[17]:



You read that correctly. Jar Jar Binks has a lower favorability rating than the actual personification of evil in the galaxy.

And for those of you who want to know the impact that [historical revisionism (http://en.wikipedia.org/wiki/Han_shot_first)](http://en.wikipedia.org/wiki/Han_shot_first) can have on a society:



** Homework note: Click here (assets/shot_first.png) to see a version of this plot generated in Altair. You may find that you don't get 834 rows (as 538 did) but the percents should still work.

2.4 Who shot first? Recreate the above image using altair (10 POINTS)

In [18]:
```
# Recreate this image using altair here (10 POINTS)

total_24 = len(seen_at_least_one)
names_24 = ['Han', 'Greedo', "I don't understand this question"]

# for each movie, find the percentages where it's rated top third, middle third, or bottom third
wsf = []

hsf = len(seen_at_least_one[seen_at_least_one['Which character shot first?'] == 'Han']) / total_24
wsf.append(hsf)

gsf = len(seen_at_least_one[seen_at_least_one['Which character shot first?'] == 'Greedo']) / total_24
wsf.append(gsf)

idk = len(seen_at_least_one[seen_at_least_one['Which character shot first?'] == "I don't understand this question"]) / total_24
wsf.append(idk)

tuples_24 = list(zip(names_24, wsf))
wsf_df = pd.DataFrame(tuples_24, columns = ['Name', 'Shot First'])

wsf_df.head()

#raise NotImplementedError()
```

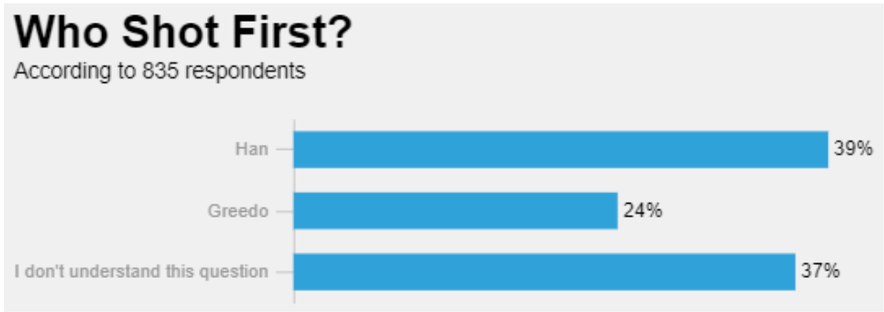
Out[18]:

	Name	Shot First
0	Han	0.389222
1	Greedo	0.235928
2	I don't understand this question	0.365269

In [19]:

```
bars_24 = alt.Chart(wsf_df).mark_bar(size=20).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Shot First',
        axis=None),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
        sort = names_24
    )
)


text_24 = bars_24.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('Shot First:Q',format='.0%')
)

shot_movies = (text_24 + bars_24).configure_mark(
    color='#008fd5'
).configure_view(
    strokeWidth=0
).configure_scale(
    bandPaddingInner=0.2
).properties(
    # set the dimensions of the visualization
    width=300,
    height=100
).properties(
      title={
      "text": ["Who Shot First?"],
      "subtitle": ["According to 835 respondents"]
      }
)

shot_movies
```

Out[19]:



2.5.1 Make your own (15 points/ 10 points plot + 5 justification)

Propose and code an alternative visualization for one of the visualizations *already in the article*. Add a short paragraph describing why your visualization is more (or less) *effective* based on principles of perception/cognition.

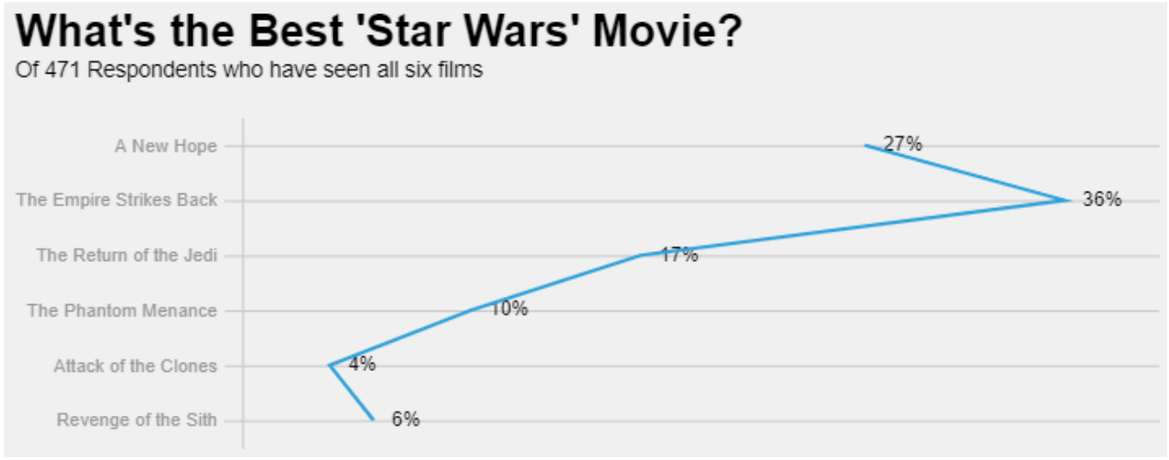
If you feel your visualization is worse, that's ok! Just tell us why.

In [20]:

```
episodes = ['EIV', 'EV', 'EVI', 'EI', 'EII', 'EIII', ]
names = {
    'EI' : 'The Phantom Menance', 'EII' : 'Attack of the Clones', 'EIII' : 'Revenge of the Sith',
    'EIV': 'A New Hope', 'EV': 'The Empire Strikes Back', 'EVI' : 'The Return of the Jedi'
}

# we're also going to use this order to sort, so names_l will now have our sort order
names_new = [names[ep] for ep in episodes]

line_25 = alt.Chart(seen_all_df).mark_line().encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage:Q',
        axis= None,
        title = ''
    ),
    y=alt.Y(
        # encode y using the name, use the movie name to label the axis, sort using the names_l
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         sort=names_new
    )
)


text_25 = line_25.mark_text(
    align='left',
    baseline='middle',
    dx=10  # Nudges text to right so it doesn't appear on top of the bar
).encode(
    text=alt.Text('Percentage:Q',format='.0%')
)

re_viz_movies = (text_25 + line_25).configure_mark(
    color='#008fd5'
).configure_view(
    strokeWidth=0
).configure_scale(
    bandPaddingInner=0.2
).properties(
    # set the dimensions of the visualization
    width=500,
```

```
    height=180
).properties(
      title={
      "text": ["What's the Best 'Star Wars' Movie?"],
      "subtitle": ["Of 471 Respondents who have seen all six films"]
      }
)

re_viz_movies
#raise NotImplementedError()
```

Out[20]:



This is a re-take on the 2.1 visualization created earlier and seen in the FiveThirtyEight article. While the same information is being encoded - the six Star Wars movies on the y-axis are nominal variables, and the percentage of respondents who believe the Star Wars movie is the best are quantitative variables on the x-axis, a line was chosen as the mark. In terms of perception/cognition, this was done in order to pull in the Gestalt Principle of Continuity. Viewers are very adept at following these types of lines and gathering a trend from it. As such, the movies were plotted in release order (4, 5, 6, 1, 2, 3) and this line graph I believe does an effective job of showing that the earlier movies are the "fan-favorites" whereas far fewer people choose prequel movies as their top option. Therefore, this graph and the 2.1 visualization have the same expressiveness, but I believe this chart is more effective in that it allows perhaps a non-Star Wars fan to discern that the original movies typically are the highest performers in these types of surveys.

2.5.2 Make your own (15 points/ 10 points plot + 5 justification)

Propose and code a *new visualization* to complement a part of the article. Add a short paragraph justifying your decisions in terms of Perception/Cognition processes.

If you feel your visualization is worse, that's ok! Just tell us why.

In [21]:
```
seen_all['Do you consider yourself to be a fan of the Expanded Universe?•æ'].unique()

chars_25 = ['Han Solo', 'Luke Skywalker', 'Princess Leia Organa', 'Anakin Skywalker',
       'Obi Wan Kenobi', 'Emperor Palpatine', 'Darth Vader', 'Lando Calrissian', 'Boba Fett',
        'C-3P0', 'R2 D2', 'Jar Jar Binks', 'Padme Amidala', 'Yoda']

yes_df = seen_all[seen_all['Do you consider yourself to be a fan of the Expanded Universe?•æ'] == 'Yes']
no_df = seen_all[seen_all['Do you consider yourself to be a fan of the Expanded Universe?•æ'] == 'No']

eu_yes = len(seen_all[seen_all['Do you consider yourself to be a fan of the Expanded Universe?•æ'] == 'Yes'])
eu_no = len(seen_all[seen_all['Do you consider yourself to be a fan of the Expanded Universe?•æ'] == 'No'])

favs_yes_25 = []
favs_no_25 = []

for character in chars_25:
    yes = len(yes_df[yes_df[character].isin(['Very favorably', 'Somewhat favorably'])]) / eu_yes
    favs_yes_25.append(yes)

    no = len(no_df[no_df[character].isin(['Very favorably', 'Somewhat favorably'])]) / eu_no
    favs_no_25.append(no)

tuples_25 = list(zip(chars_25, favs_yes_25, favs_no_25))
eu_df = pd.DataFrame(tuples_25, columns = ['Name', 'Expanded Universe Fan', 'Non-Expanded Universe Fan'])

sort_eu = eu_df.sort_values(by = ['Expanded Universe Fan'], ascending = False, inplace = True)
eu_list = list(sort_fav['Name'])

eu_df = pd.melt(eu_df, id_vars='Name', var_name='Favorability', value_name='Percentage')
eu_df.head()
```

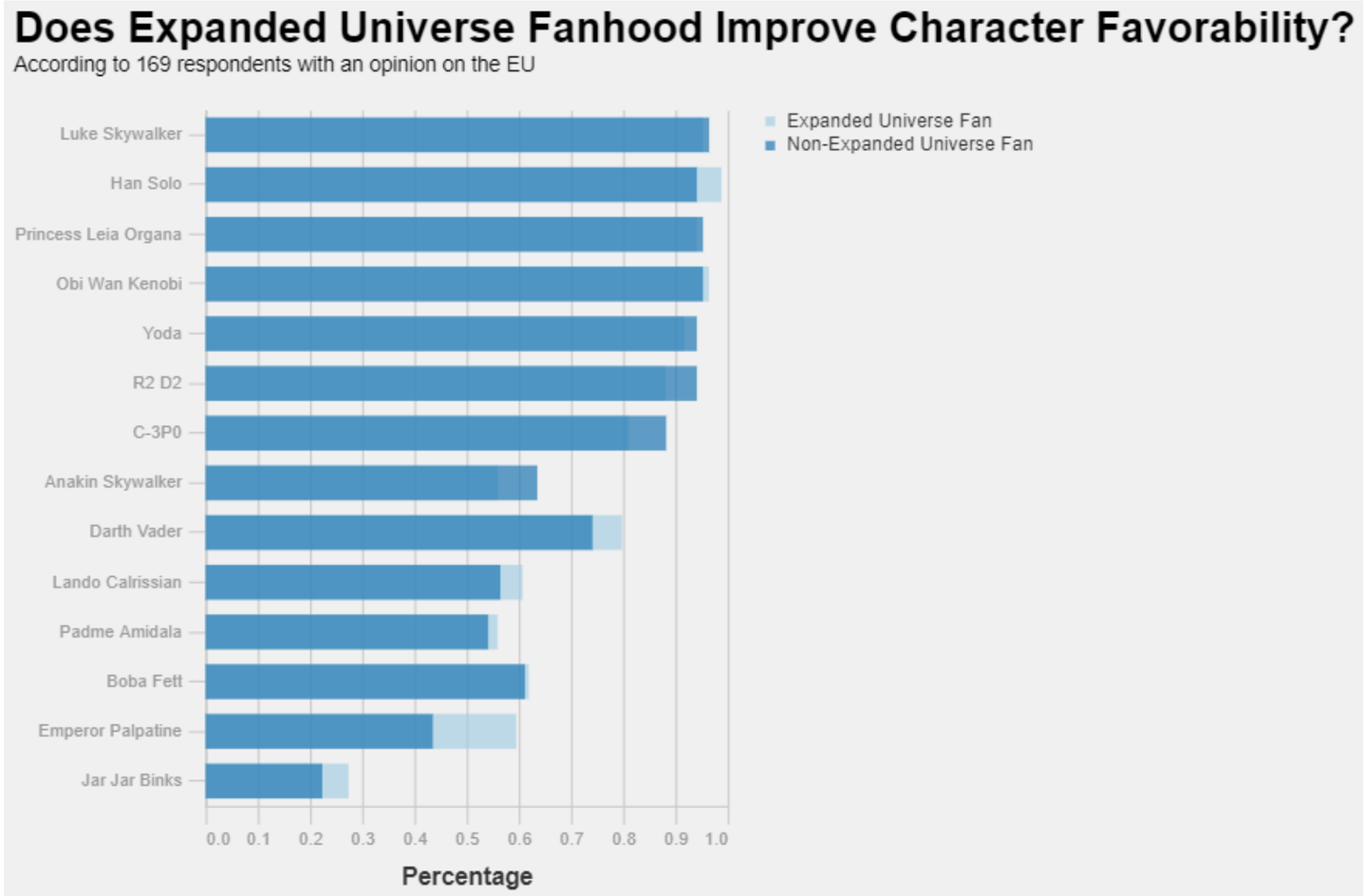
Out[21]:

	Name	Favorability	Percentage
0	Han Solo	Expanded Universe Fan	0.988095
1	Obi Wan Kenobi	Expanded Universe Fan	0.964286
2	Luke Skywalker	Expanded Universe Fan	0.952381
3	Princess Leia Organa	Expanded Universe Fan	0.940476
4	Yoda	Expanded Universe Fan	0.916667

In [22]:

```
bars_25 = alt.Chart(eu_df).mark_bar(size=20, opacity = 0.7).encode(
    # encode x as the percent, and hide the axis
    x=alt.X(
        'Percentage:Q',
        stack = None),
    y=alt.Y(
        'Name:N',
         axis=alt.Axis(tickCount=5, title=''),
         sort= eu_list
    ),
    color = alt.Color('Favorability:N', scale = alt.Scale(scheme = 'paired'), title = '')
)

eu_movies = (bars_25).configure_mark(
    color='#008fd5'
).configure_view(
    strokeWidth=0
).configure_scale(
    bandPaddingInner=0.2
).properties(
    # set the dimensions of the visualization
    width=300,
    height=400
).properties(
      title={
      "text": ["Does Expanded Universe Fanhood Improve Character Favorability?"],
      "subtitle": ["According to 169 respondents with an opinion on the EU"]
      }
)

eu_movies
```

Out[22]:



With the prequel movies often having the characters with the lowest level of favorability amongst fans based on the FiveThirtyEight article, I was interested in this complementary figure that is a layered bar chart that investigates whether Expanded Universe fanhood improves favorability. The Expanded Universe (EU) goes beyond just these six movies and has a lot of material surrounding characters seen in the prequel trilogy. My expectation is that these books, tv series, etc. would lead to greater development of these lower ranked prequel characters and lead to greater favorability. As seen with Jar Jar Binks in particular, this does look to be the case as fans of the EU have a more favorable opinion of the character relative to non-EU fans. In terms of perception, this visualization will utilize hue for the reader to quickly see favorability differences between fans or non-fans of the EU. This figure thus draws on the power of preattentive processing as a way of quickly scanning the file to find a meaningful result. This figure does have some limitations - it perhaps could be more effective to not layer but instead allow for side by side graphics and an additional encoding that actually specified whether the character was from the prequels or original trilogy was debated. However, the layering was chosen as a way to ensure a non-cluttered figure.

In [ ]: