Lecture #20: Missingness
(& a bit of Visualization)
aka STAT109A, AC209A, CSCIE-109A

CS109A Introduction to Data Science
Pavlos Protopapas, Kevin Rader and Chris Gumb

Lecture Outline

- **Dealing with Missingness**
  - Types of Missingness
- Imputation Methods
- Visualizations – Revisited
- Interpreting Black Box Models
- Ineffective Visualizations
- Effective Visualizations
  - Historical Interlude
  - Principles of Effective Visualizations

What is missing data?

Often times when data is collected, there are some missing values apparent in the dataset. This leads to a few questions to consider:

- How does this show up in pandas?
- How does sklearn handle these NaNs?
- How does this effect our modeling?

What are the simplest ways to handle missing data?

Naively handling missingness

- Drop the observations that have any missing values.
  - Use pd.DataFrame.dropna(axis=0)
- Impute the mean/median (if quantitative) or most common class (if categorical) for all missing values.
  - Use pd.DataFrame.fillna(value=x.mean())

How do statsmodels and sklearn handle these NaNs?

What are some consequences in handling missingness in these fashions?

Missingness Indicator Variable

One simple way to handle missingness in a variable, X_j, is to impute a value (like 0 or $\overline{X_j}$), then create a new variable, $X_{j,miss}$, that indicates this observation had a missing value. If X_j is categorical then just impute 0.

Then include both $X_{j,miss}$ and X_j as predictors in any model.

Illustration is to the right.

X_1	X_2	X_1^*	X_2^*	$X_{1,miss}$	$X_{2,miss}$
10	.	10	0	0	1
5	1	5	1	0	0
21	0	21	0	0	0
15	0	15	0	0	0
16	.	16	0	0	1
.	.	0	0	1	1
21	1	21	1	0	0
12	0	12	0	0	0
.	1	0	1	1	0

Why use a Missingness Indicator Variable?

How does this missingness indicator variable improve the model?

Because the group of individuals with a missing entry may be systematically different than those with that variable measured. Treating them equivalently could lead to bias in quantifying relationships (the β's) and underperform in prediction.

For example: imagine a survey that asks whether or not someone has ever recreationally used opioids, and some people chose not to respond. Does the fact that they did not respond provide extra information? Should we treat them equivalently as never-users?

This approach essentially creates a third group for this predictor: the "did not respond" group.

Lecture Outline

- Dealing with Missingness
    - **Types of Missingness**
- Imputation Methods
- Visualizations – Revisited
- Interpreting Black Box Models
- Ineffective Visualizations
- Effective Visualizations
    - Historical Interlude
    - Principles of Effective Visualizations

Sources of Missingness

Missing data can arise from various places in data:

- A survey was conducted and values were just randomly missed when being entered in the computer.
- A respondent chooses not to respond to a question like `Have you ever recreationally used opioids?'.
- You decide to start collecting a new variable (due to new actions: like a pandemic) partway through the data collection of a study.
- You want to measure the speed of meteors, and some observations are just 'too quick' to be measured properly.

The source of missing values in data can lead to the major types of missingness:

Types of Missingness

There are 3 major types of missingness to be concerned about:

1. **Missing Completely at Random (MCAR)** - the probability of missingness in a variable is the same for all units. Like randomly poking holes in a data set.
2. **Missing at Random (MAR)** - the probability of missingness in a variable depends only on available information (in other predictors).
3. **Missing Not at Random (MNAR)** - the probability of missingness depends on information that has not been recorded and this information also predicts the missing values.

What are examples of each these 3 types?

Missing completely at random (MCAR)

Missing Completely at Random is the best case scenario, and the easiest to handle:

- Examples: a coin is flipped to determine whether an entry is removed. Or when values were just randomly missed when being entered in the computer.
- Effect if you ignore: there is no effect on inferences (estimates of β).
- How to handle: lots of options, but best to impute (more on next slide).

Missing at random (MAR)

Missing at Random is still a case that can be handled.

- Example(s): men and women respond to the question "have you ever felt harassed at work?" at different rates (and may be harassed at different rates).
- Effect if you ignore: inferences are biased (estimates of β's) and predictions are usually worsened.
- How to handle: use the information in the other predictors to build a model and **impute** a value for the missing entry.

Key: we can fix any biases by modeling and imputing the missing values based on what is observed!

Missing Not at Random (MNAR)

Missing Not at Random is the worst-case scenario, and impossible to handle properly:

- Example(s): patients drop out of a study because they experience some really bad side effect that was not measured. Or cheaters are less likely to respond when asked if you've ever cheated.
- Effect if you ignore: there are major effects on inferences (estimates of beta) or predictions.
- How to handle: you can 'improve' things by dealing with it like it is MAR, but you [likely] may never completely fix the bias. And incorporating a **missingness indicator variable** may actually be the best approach (if it is in a predictor).

What type of missingness is present?

Can you ever tell based on your data what type of missingness is actually present?

Since we asked the question, the answer must be no.

It generally cannot be determined whether data really are missing at random, or whether the missingness depends on unobserved predictors or the missing data themselves. The problem is that these potential "**lurking variables**" are unobserved (by definition) and so can never be completely ruled out.

In practice, a model with as many predictors as possible is used so that the *missing at random* assumption is reasonable.

Lecture Outline

- Dealing with Missingness
    - Types of Missingness
- **Imputation Methods**
- Visualizations – Revisited
- Interpreting Black Box Models
- Ineffective Visualizations
- Effective Visualizations
    - Historical Interlude
    - Principles of Effective Visualizations

Handling Missing Data

When encountering missing data, the approach to handling it depends on…

1. whether the missing values are in the **response** or in the **predictors**. Generally speaking, it is much easier to handle missingness in predictors.
2. the **type of variable**: whether the variable is quantitative or categorical.
3. the **amount of missingness** present in the variable. If there is too much missingness, you may be doing more harm than good.

Generally speaking, it is a good idea to attempt to **impute** (or 'fill in') entries for missing values in a variable (assuming your method of imputation is a good one).

Imputation Methods

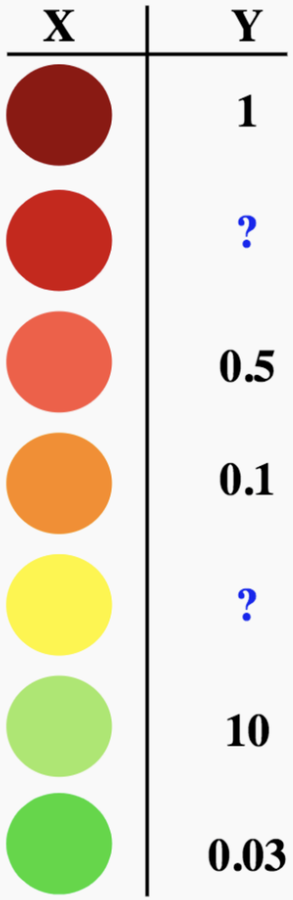
There are several different approaches to imputing missing values:

1. **Impute the mean or median** (quantitative) or most common class (categorical) for all missing values in a variable.
2. Create a new variable that is an **indicator of missingness**, and include it in any model to predict the response (also plug in zero or the mean/median (or reference class/group) in the actual variable).
3. **Hot deck imputation**: for each missing entry, randomly select an observed entry in the variable and plug it in.
4. **Model the imputation**: plug in predicted values ($\hat{y}$) from a model based on the other observed predictors.
5. **Model the imputation with uncertainty**: plug in predicted values plus randomness ($\hat{y} + \epsilon$) from a model based on the other observed predictors.

What are the advantages and disadvantages of each approach?

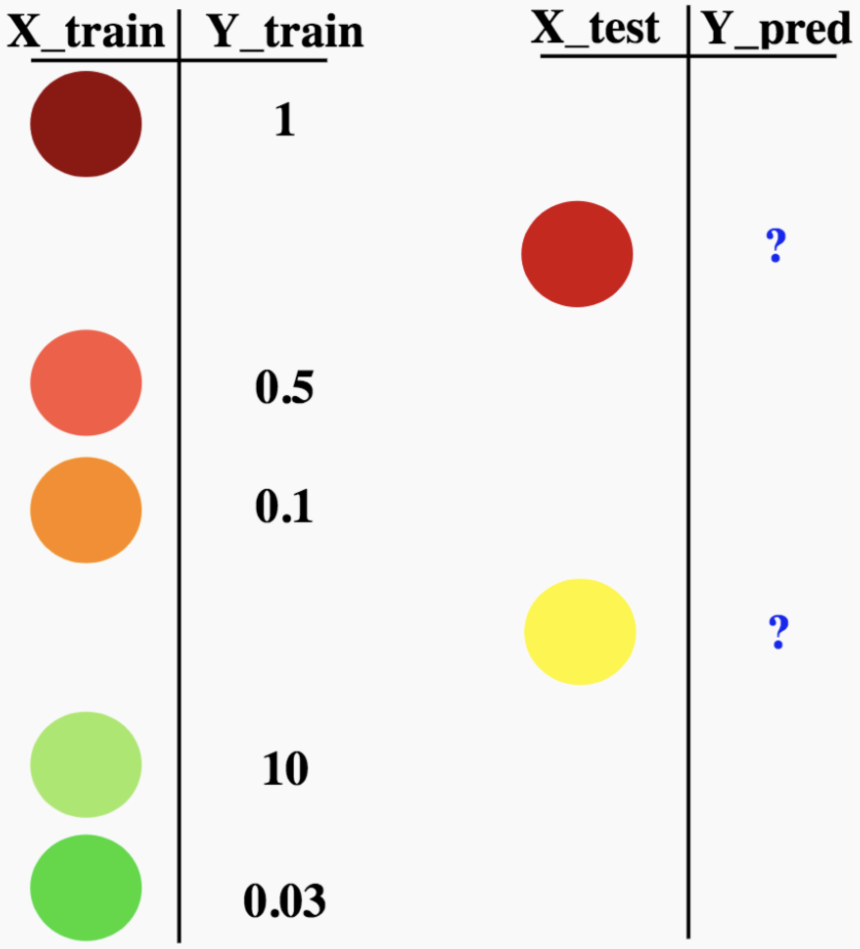
Schematic: imputation through modeling

How do we use models to fill in missing data?



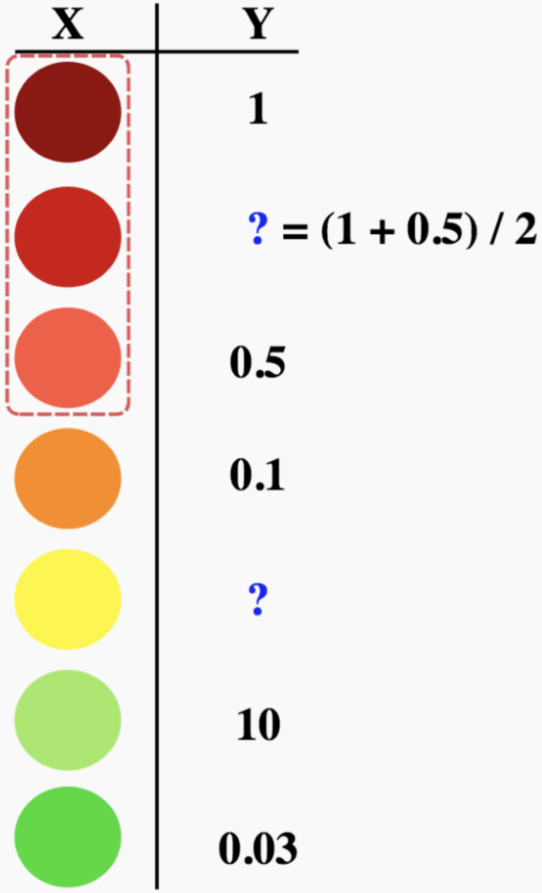
Schematic: imputation through modeling

How do we use models to fill in missing data?



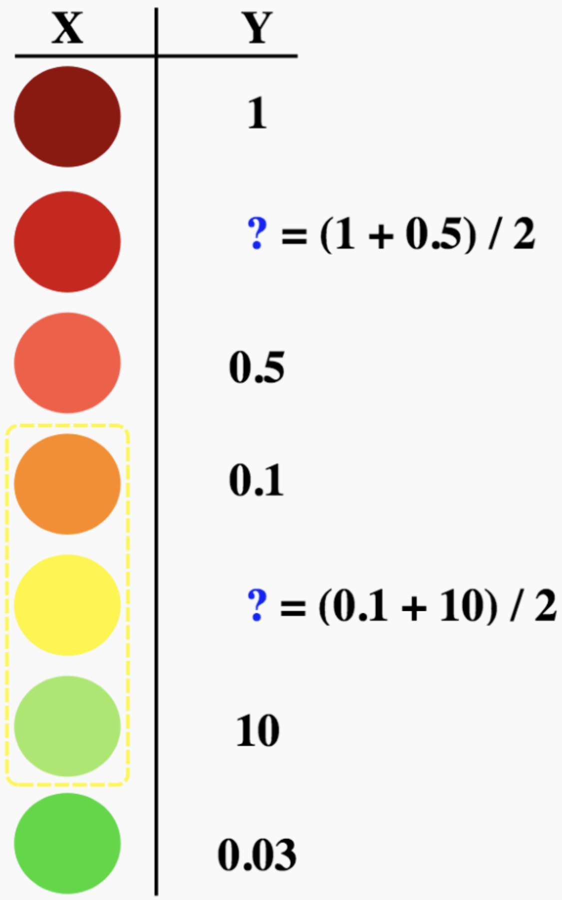
Schematic: imputation through modeling

How do we use models to fill in missing data? Using *k*-NN for *k* = 2?



Schematic: imputation through modeling

How do we use models to fill in missing data? Using k-NN for k = 2?



Schematic: imputation through modeling

How do we use models to fill in missing data? Using linear regression?

Where m and b are computed from the observations (rows) that do not have missingness (we should call them $b = \beta_0$ and $m = \beta_1$).

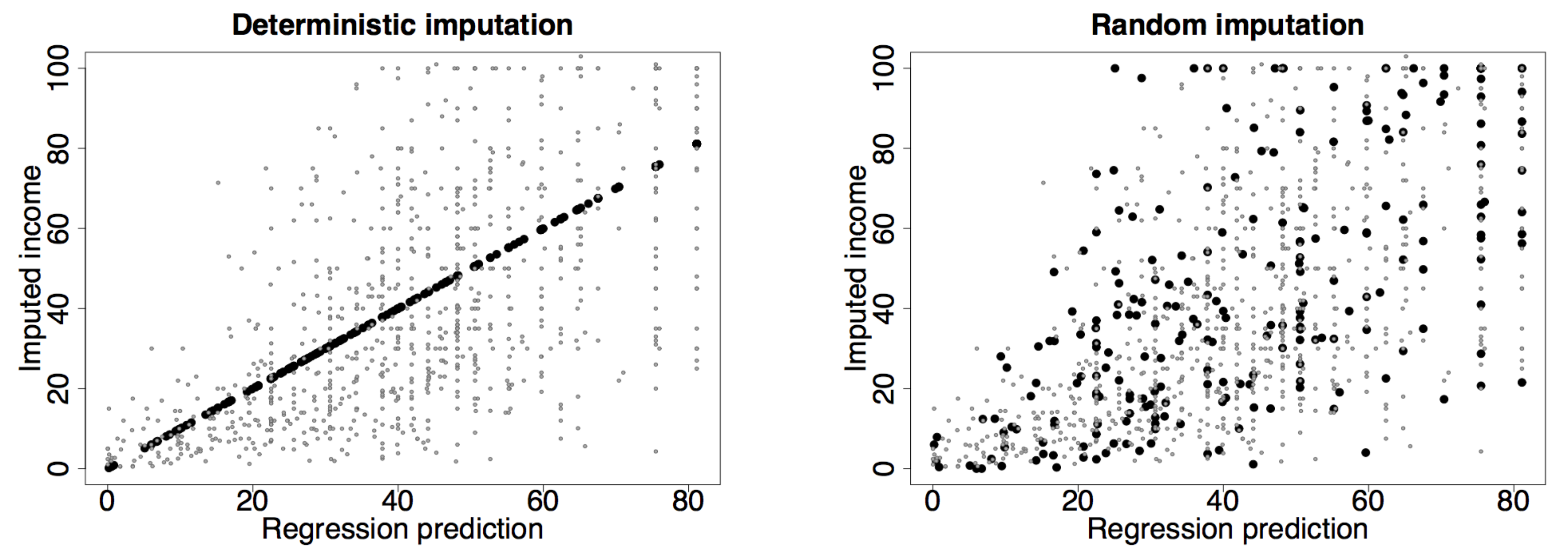
Imputation through modeling with uncertainty

The schematic in the last few slides ignores the approach of imputing with uncertainty. What happens if you ignore this fact and just use the 'best' model to impute values solely on $\hat{y}$?

The distribution of the imputed values will be too narrow and not represent real data (see next slide for illustration). The goal is to impute values that include the uncertainty of the model.

How can this be done in practice in k-NN? In linear regression? In logistic regression?

Imputation: modeling with uncertainty (an illustration)



Imputation: modeling with uncertainty (an algorithm)

Recall the probabilistic model in linear regression:

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_P + \epsilon \quad \text{where } \epsilon \sim N(0, \sigma^2)$$

How can we take advantage of this model to impute with uncertainty?

It's a 3-step process:

1. Fit a model to predict the predictor variable with missingness from all the other predictors.
2. Predict the missing values from the model in the previous part.
3. Add in a measure of uncertainty to this prediction by randomly sampling from a $N(0, \sigma^2)$ distribution*, where σ^2 is the mean square error (MSE) from the model.

*Instead of sampling from a $N(0, \sigma^2)$ distribution, you can bootstrap sample from the observed residuals!

Imputation: modeling with uncertainty (k-NN regression)

How can we use k-NN regression to impute values that mimic the error in our observations?

Two ways:

- Use $k = 1$.
- Use any other k, but randomly select from the nearest neighbors in $\mathcal{N}_0$. This can be done with equal probability or with some weighting (inverse to the distance measure used).

Imputation through modeling with uncertainty: classifiers

For classifiers, this imputation with uncertainty/randomness is a little easier process. How can it be implemented?

If a classification model (logistic, *k*-NN, etc…) is used to predict the variable with missingness on the observed predictors, then all you need to do is flip a ‘biased coin’ (or multi-sided die) with the probabilities of coming up for each class equal to the predicted probabilities from the model.

Warning: do not just classify blindly using the predict command in sklearn!

Imputation across multiple variables

If only one variable has missing entries, life is easy. But what if all the predictor variables have a little bit of missingness (with some observations having multiple entries missing)? How can we handle that?

It's an iterative process. Impute X_1 based on $X_2, ..., X_p$. Then impute X_2 based on X_1 and $X_3, ..., X_p$. And continue down the line.

Any issues?

Yes, not all of the missing values may be imputed with just one 'run' through the data set. So you will have to repeat these 'runs' until you have a completely filled in data set.

Multiple imputation: beyond this class

What is an issue with treating your now ‘complete’ data set (a mixture of actually observed values and imputed values) as simply all observed values?

Any inferences or predictions carried out will be tuned and potentially overfit to the random entries imputed for the missing entries. How can we prevent this phenomenon?

By performing **multiple imputation**: rerun the imputation algorithm many times, refit the model on the response many times (one time each), and then ’average’ the predictions or estimates of β coefficients to perform inferences (also incorporating the uncertainty involved).

Note: this is beyond what we would expect in this class. But it generally a good thing to be aware of.

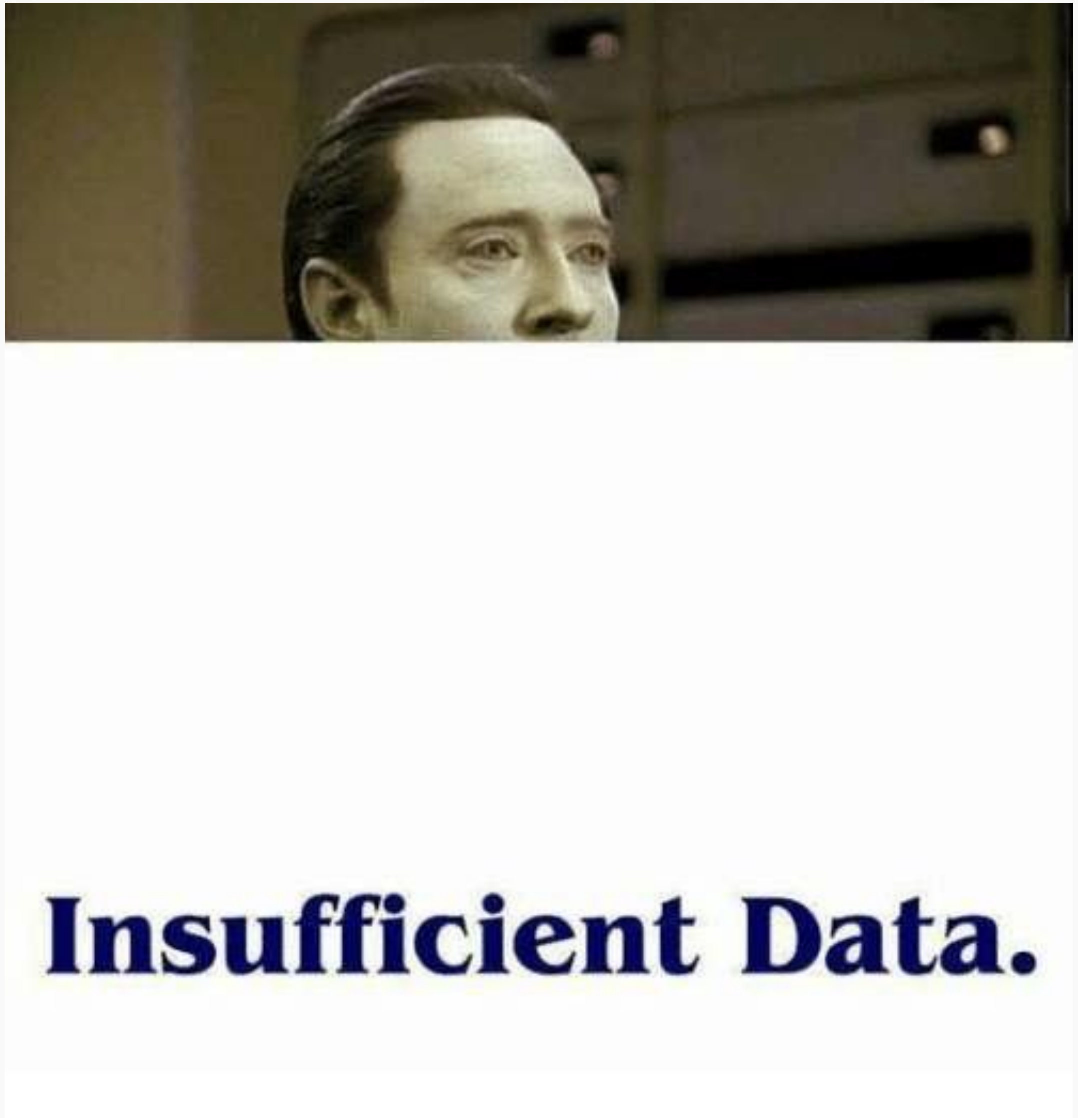
sklearn's impute module

Want to do imputation in sklearn? Of course there are functions for that!

The 4 main functions in this module are:

1. **`sklearn.impute.SimpleImputer`**: perform mean, median, mode, or any specific value to impute
2. **`sklearn.impute.IterativeImputer`**: perform multivariate imputation that estimates each predictor from all the others (user-supplied model/estimator).
3. **`sklearn.impute.KNNImputer`**: perform imputation automatically from a *k*-NN model from all the other predictors.
4. **`sklearn.impute.MissingIndicator`**: to create binary indicator(s) for where the missing values occur.

What we are trying to avoid…



Lecture Outline

- Dealing with Missingness
    - Types of Missingness
- Imputation Methods
- **Visualizations – Revisited**
- Interpreting Black Box Models
- Ineffective Visualizations
- Effective Visualizations
    - Historical Interlude
    - Principles of Effective Visualizations

CS109A
GAME Time

What is the goal of visualization?

Options

A. To explore the distributions of variables.

B. To explore the data to build hypotheses.

C. To communicate results of your models.

D. To trick and manipulate your audience.

What is the goal of visualization?

Options

A. To explore the distributions of variables.

B. To explore the data to build hypotheses.

C. To communicate results of your models.

D. To trick and manipulate your audience.

Lecture Outline

- Dealing with Missingness
    - Types of Missingness
- Imputation Methods
- Visualizations – Revisited
- **Interpreting Black Box Models**
- Ineffective Visualizations
- Effective Visualizations
    - Historical Interlude
    - Principles of Effective Visualizations

Parametric vs. Nonparametric

What is the difference between a parametric vs. non-parametric model?

What are examples of each that we have learned?

Regression Problems

Parametric: Linear Regression

Non-parametric: *k*-NN Regression

Classification Problems

Parametric: Logistic Regression

Non-parametric: *k*-NN Classification

Throughout the rest of the course, we will be learning/using tree-based model, which are difficult to interpret relationships from.

Parametric vs. Nonparametric

How can we understand the relationship between Y and X_j in a complex model?

In linear/logistic regression, we often can just interpret the appropriate β_j.

What about if there are many interaction terms involved?

What about for a k-NN model? Or a decision tree model?

¯_(ツ)_/¯

Partial Dependence Plots

We can interpret relationships between a response and predictor(s) in a complex model by plotting the model's predictions!

There are often called **Partial dependence plots** (PDPs).

A Partial dependency plot depicts the relationship between the response (Y) and a {set} of input predictor(s) of interest (X_j), *marginalizing* over the values of all other predictors in the model.

English please!

English: *k*-NN Classification

Recall the *k*-NN approach for a regression problem:

1. Determine an observation's *k* nearest neighbors (based on some [Euclidean] distance measure)
   - Likely makes sense to standardize predictors
2. Use these *k* neighbors to predict that observation's response: $\hat{Y}$.

What changes when applying the *k*-NN approach in a classification problem?

1. Determining the neighbors is unchanged.
2. Predicting the response from those neighbors can be turned into a proportion in each group or a pure classification (majority/plurality wins). Just like in a decision tree!!!

Simple *k*-NN Classification

```
df_heart.head()
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	1	63	1	typical	145	233	1	2	150	0	2.3	3	0.0	fixed	No
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5	2	3.0	normal	Yes
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6	2	2.0	reversable	Yes
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5	3	0.0	normal	No
4	5	41	0	nontypical	130	204	0	2	172	0	1.4	1	0.0	normal	No

```
x = df_heart[["MaxHR"]]
y = df_heart.AHD.map(lambda x: 0 if x=='No' else 1)
knn50 = sk.neighbors.KNeighborsClassifier(50)
knn50.fit(x, y)

xsynth = np.linspace(np.min(x),np.max(x))
phat =
knn50.predict_proba(pd.DataFrame(xsynth,columns=["MaxHR"]))
yhat_class =
knn50.predict(pd.DataFrame(xsynth,columns=["MaxHR"]))

pd.DataFrame({'x':xsynth,
              'yhat':yhat[:,1],
              'yhat_class':yhat_class}).tail(30)
```

	x	yhat	yhat_class
20	124.469388	0.68	1
21	127.142857	0.68	1
22	129.816327	0.66	1
23	132.489796	0.72	1
24	135.163265	0.66	1
25	137.836735	0.64	1
26	140.510204	0.64	1
27	143.183673	0.60	1
28	145.857143	0.58	1
29	148.530612	0.48	0
30	151.204082	0.40	0
31	153.877551	0.38	0
32	156.551020	0.40	0
33	159.224490	0.38	0
34	161.897959	0.36	0
35	164.571429	0.34	0
36	167.244898	0.26	0
37	169.918367	0.26	0
38	172.591837	0.22	0
39	175.265306	0.18	0
40	177.938776	0.16	0
41	180.612245	0.16	0

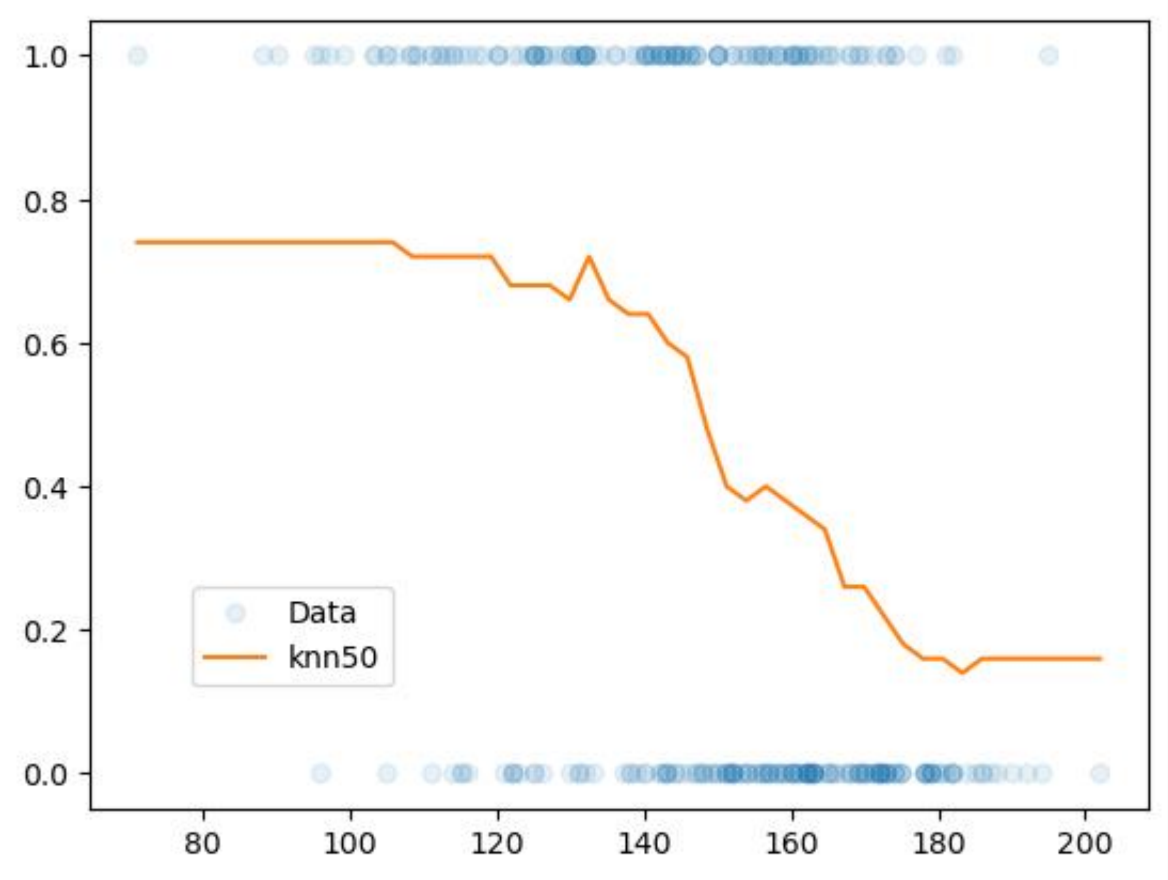
Plotting Predictions

```
x = df_heart[["MaxHR"]]
y = df_heart.AHD.map(lambda x: 0 if x=='No' else 1)

knn50 = sk.neighbors.KNeighborsClassifier(50)
knn50.fit(x, y)

xsynth = np.linspace(np.min(x),np.max(x))
yhat = knn50.predict_proba(xsynth.reshape(-1,1))

plt.plot(x, y, 'o' ,alpha=0.1, label='Data')
plt.plot(xsynth, yhat[:,1], label='knn50')
plt.legend(loc='best', bbox_to_anchor=(0.3, 0.3))
```



What does this plot suggest about the relationship between Y = AHD and X = MaxHR?

k-NN Classification (multiple predictors)

```
df_heart.head()
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	1	63	1	typical	145	233	1	2	150	0	2.3	3	0.0	fixed	No
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5	2	3.0	normal	Yes
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6	2	2.0	reversable	Yes
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5	3	0.0	normal	No
4	5	41	0	nontypical	130	204	0	2	172	0	1.4	1	0.0	normal	No

```
X = df_heart[['MaxHR','Age','Sex','RestBP','Chol',
              'Fbs', 'RestECG', 'ExAng', 'Oldpeak',
              'Slope', 'Ca']]
knn50.fit(X, y);
```

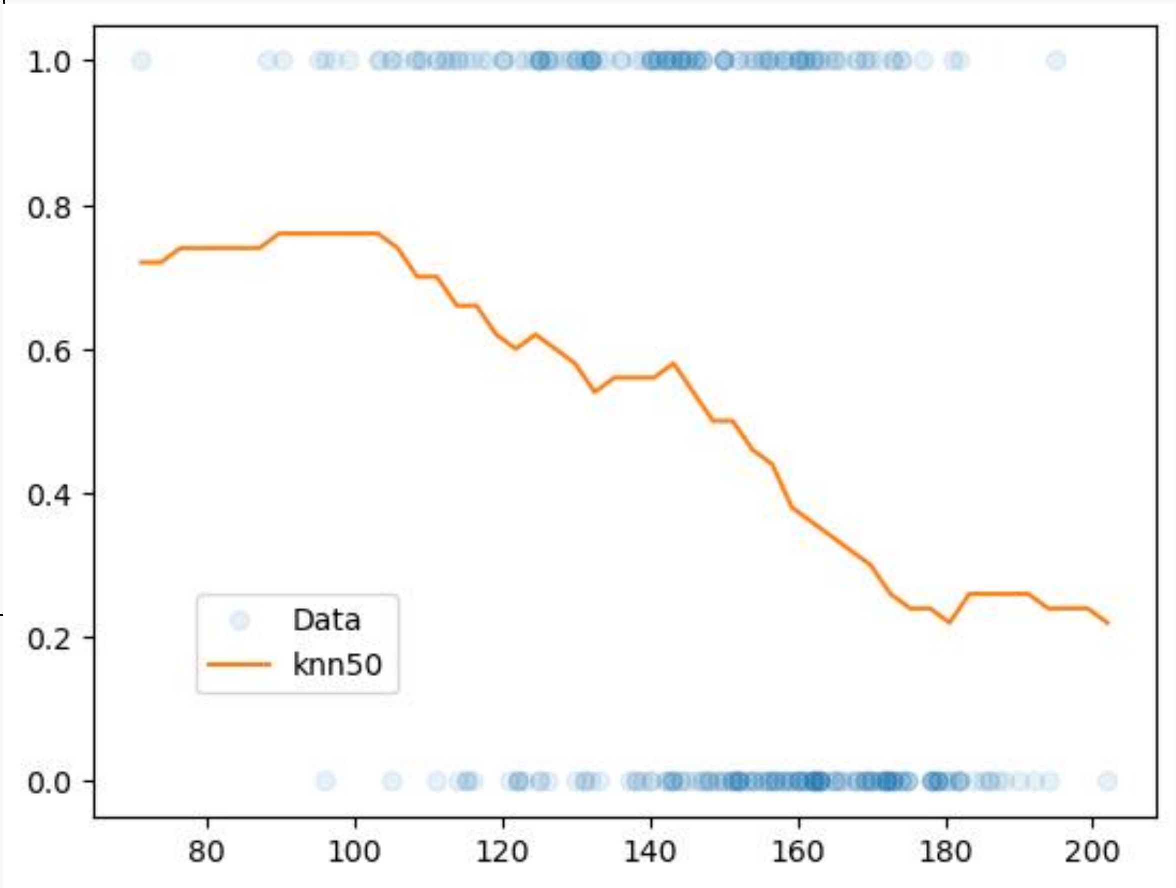
- How do we interpret the results of this model?
- How do we predict from this model?
- What sort of feature *engineering* should Kevin have considered doing?

Plotting Predictions

```
X = df_heart[['MaxHR','Age','Sex','RestBP','Chol',
              'Fbs', 'RestECG', 'ExAng', 'Oldpeak',
              'Slope', 'Ca']]
knn50.fit(X, y);


xsynth = np.linspace(np.min(x),np.max(x))
Xsynth = pd.concat([pd.DataFrame(X.median()).T] *
     len(xsynth), ignore_index=True)
Xsynth["MaxHR"] = xsynth
yhat = knn50.predict_proba(Xsynth)

plt.plot(x, y, 'o' ,alpha=0.1, label='Data')
plt.plot(xsynth, yhat[:,1], label='knn50')
plt.legend(loc='best', bbox_to_anchor=(0.3, 0.3));
```



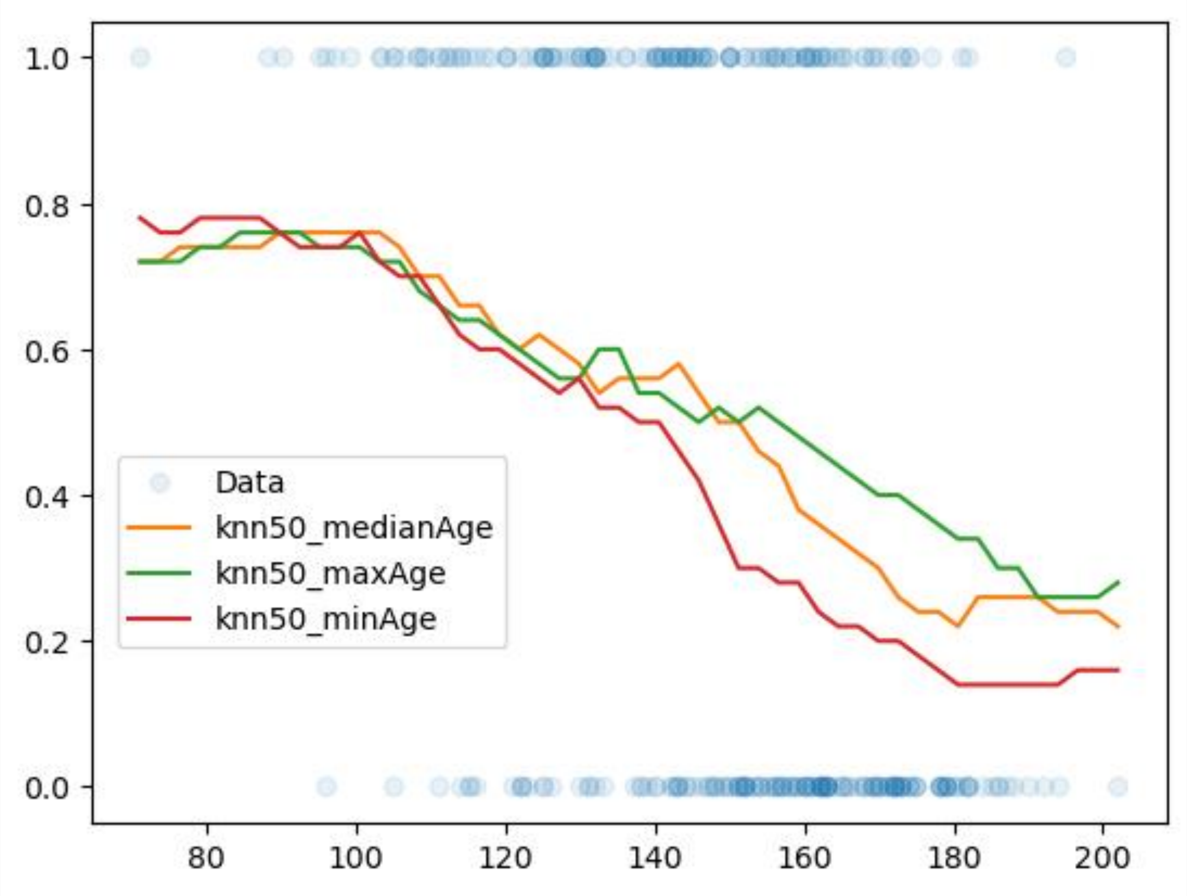
What does this plot suggest about the relationship between Y = AHD and X = MaxHR?

Plotting Predictions

```
Xsynth["Age"] = X["Age"].median()
yhat = knn50.predict_proba(Xsynth)

Xsynth["Age"] = X["Age"].max()
yhat_maxAge = knn50.predict_proba(Xsynth)
Xsynth["Age"] = X["Age"].min()
yhat_minAge = knn50.predict_proba(Xsynth)


plt.plot(x, y, 'o' ,alpha=0.1, label='Data')
plt.plot(xsynth, yhat[:,1], label='knn50_medianAge')
plt.plot(xsynth, yhat_maxAge[:,1],
label='knn50_maxAge')
plt.plot(xsynth, yhat_minAge[:,1],
label='knn50_minAge')
plt.legend(loc='best', bbox_to_anchor=(0.4, 0.2));
```

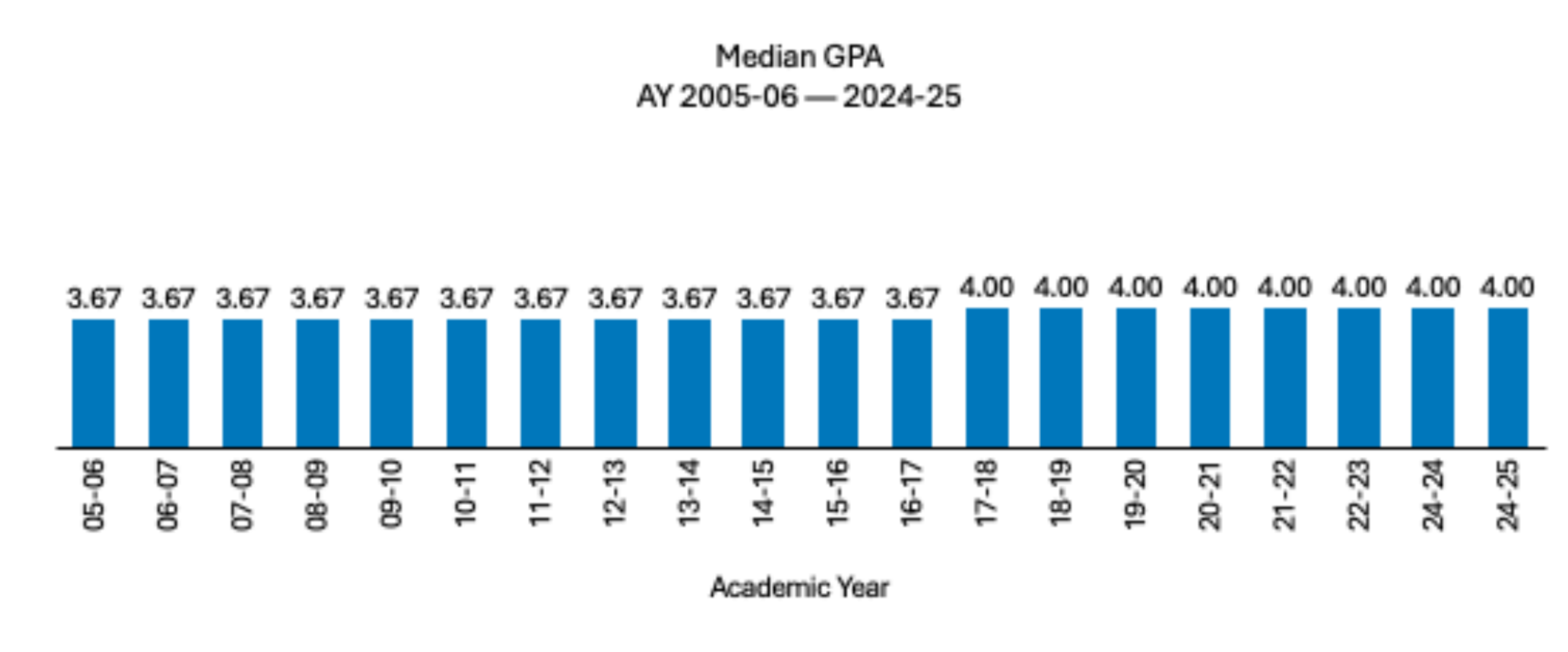


What does this plot suggest about the relationship between Y = AHD and X = MaxHR (and the other predictors involved)?

Lecture Outline

- Dealing with Missingness
  - Types of Missingness
- Imputation Methods
- Visualizations – Revisited
- Interpreting Black Box Models
- **Ineffective Visualizations**
- Effective Visualizations
  - Historical Interlude
  - Principles of Effective Visualizations

Motivation

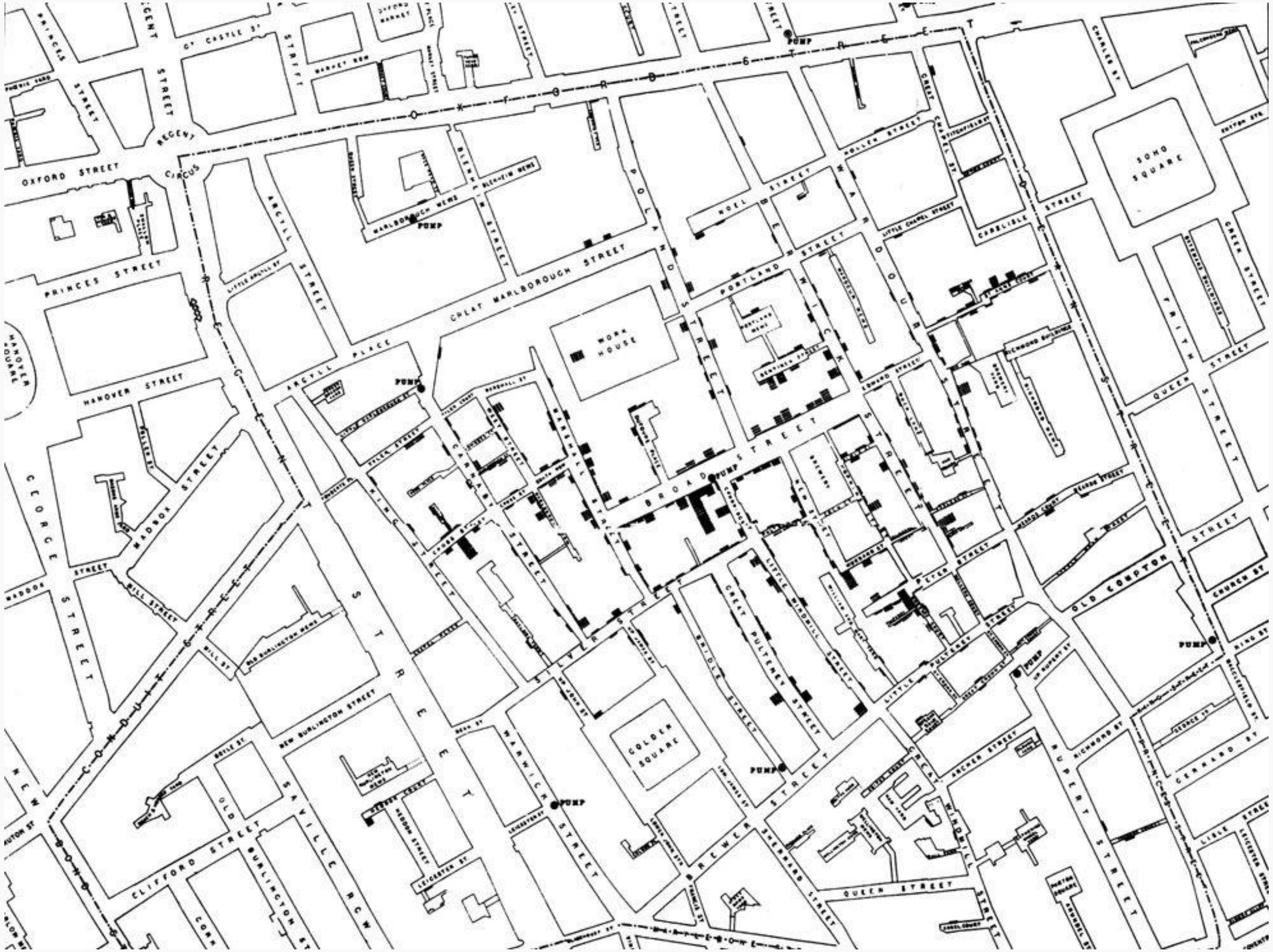


Median GPA
AY 2005-06 — 2024-25
3.67
3.67
3.67
3.67
3.67
3.67
3.67
3.67
3.67
3.67
3.67
3.67
4.00
4.00
4.00
4.00
4.00
4.00
4.00
4.00
05-06
06-07
07-08
08-09
09-10
10-11
11-12
12-13
13-14
14-15
15-16
16-17
17-18
18-19
19-20
20-21
21-22
22-23
24-24
24-25
Academic Year

Lecture Outline

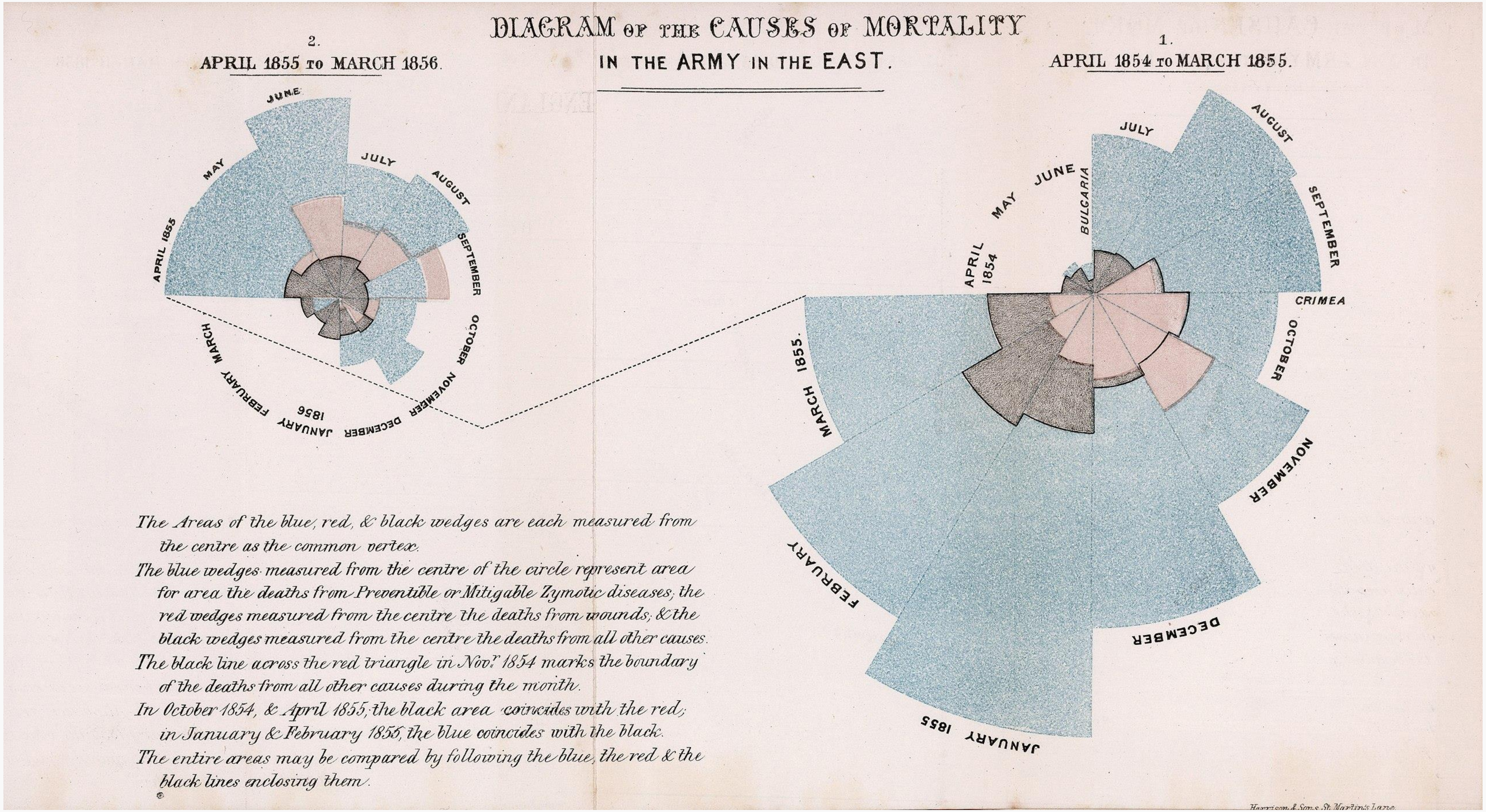
- Dealing with Missingness
  - Types of Missingness
- Imputation Methods
- Visualizations – Revisited
- Interpreting Black Box Models
- Ineffective Visualizations
- **Effective Visualizations**
  - **Historical Interlude**
  - Principles of Effective Visualizations

John Snow’s Cholera Outbreak (1854)

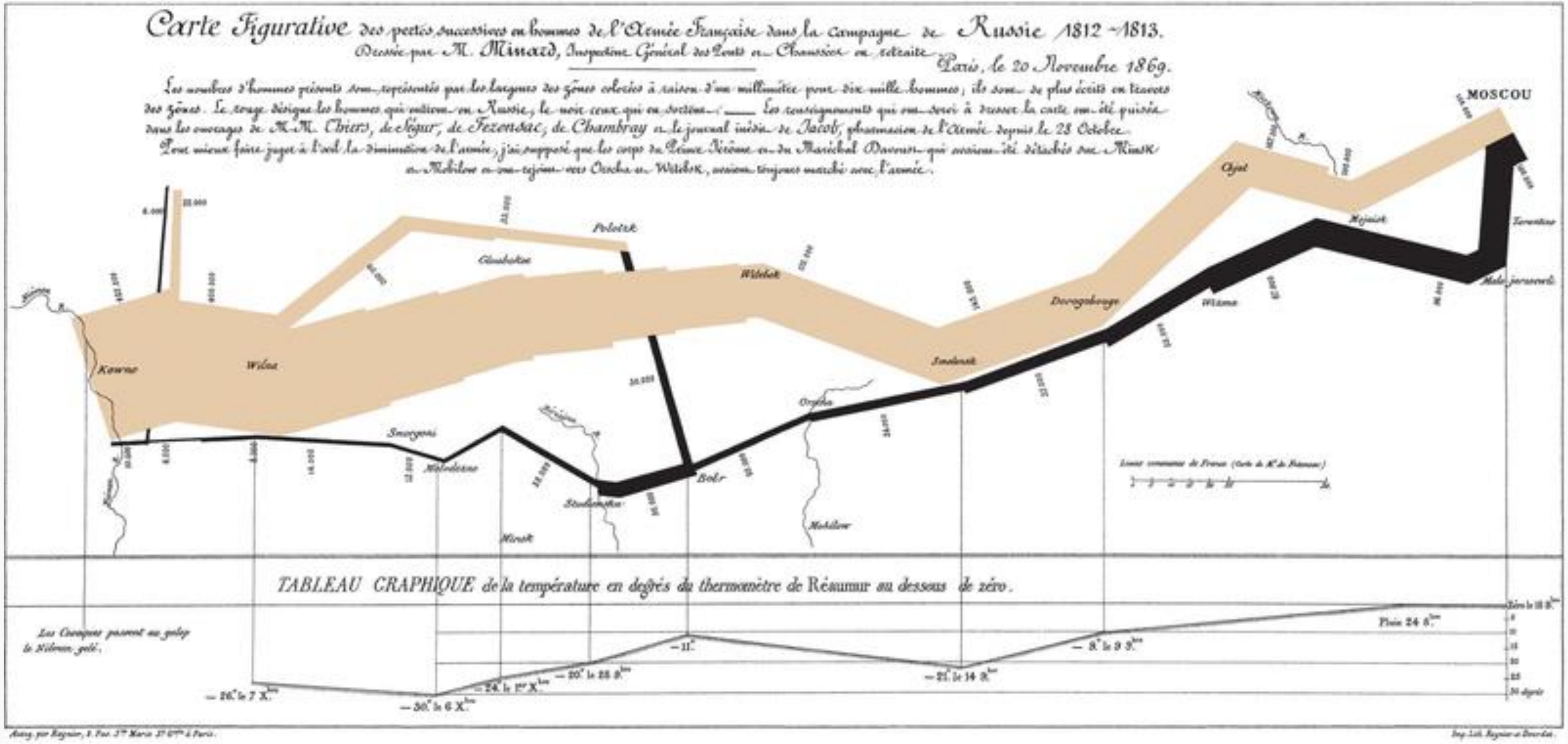


https://en.wikipedia.org/wiki/1854_Broad_Street_cholera_outbreak

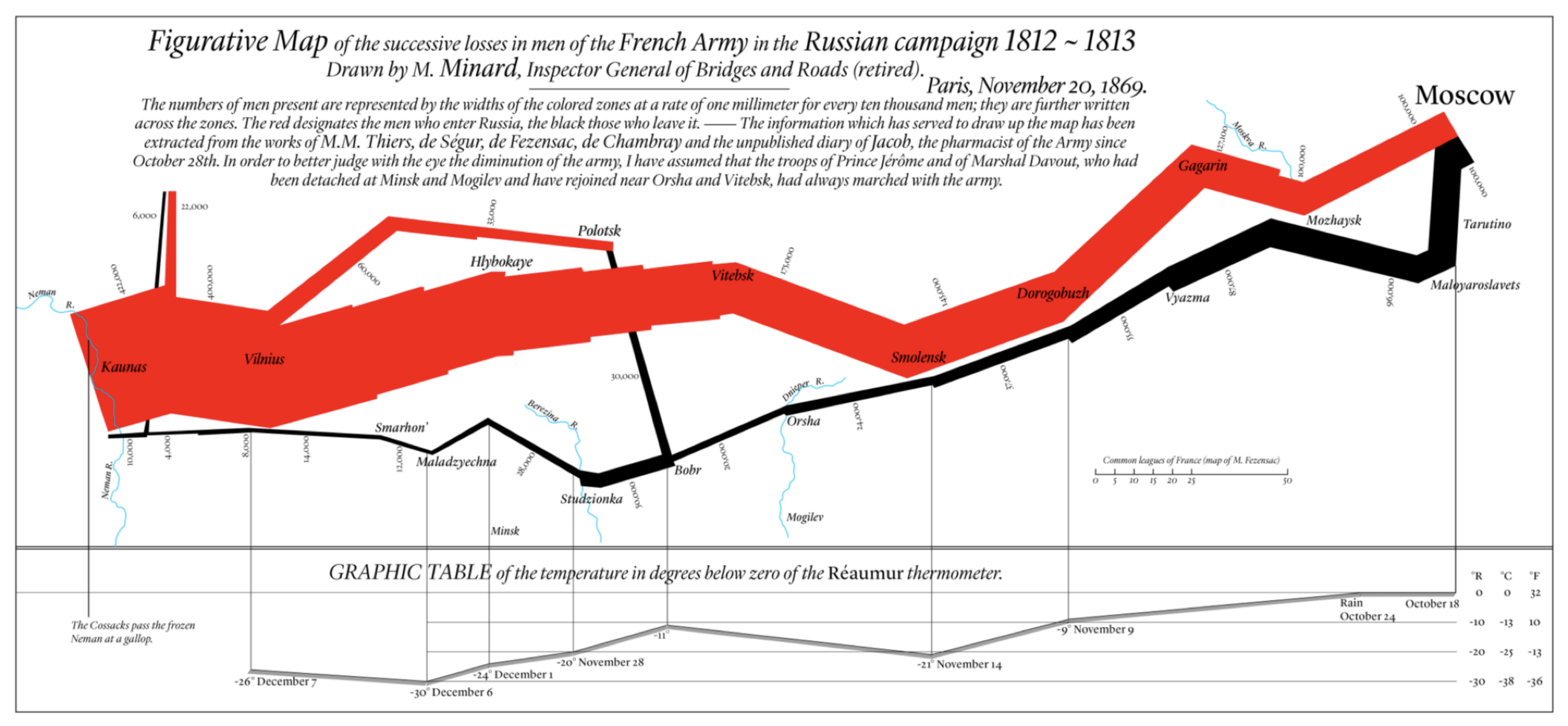
Florence Nightingale's Rose Chart of the Crimean War (1858)



Minard’s Visual of Napoleon’s March through Russia (1869)



Minard’s Visual of Napoleon’s March through Russia (1869)

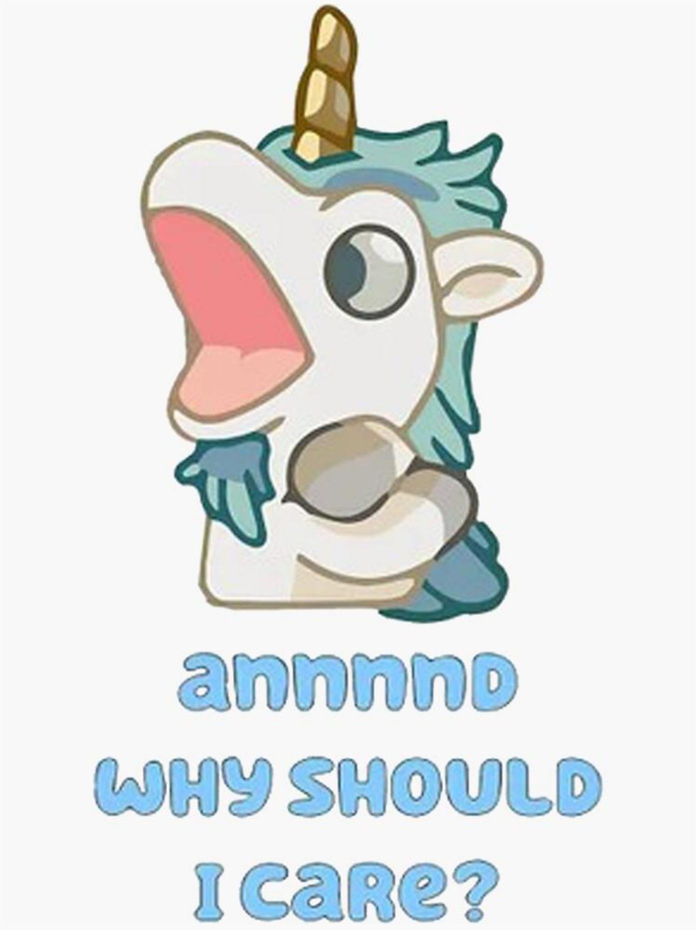


Lecture Outline

- Dealing with Missingness
  - Types of Missingness
- Imputation Methods
- Visualizations – Revisited
- Interpreting Black Box Models
- Ineffective Visualizations
- Effective Visualizations
  - Historical Interlude
  - **Principles of Effective Visualizations**

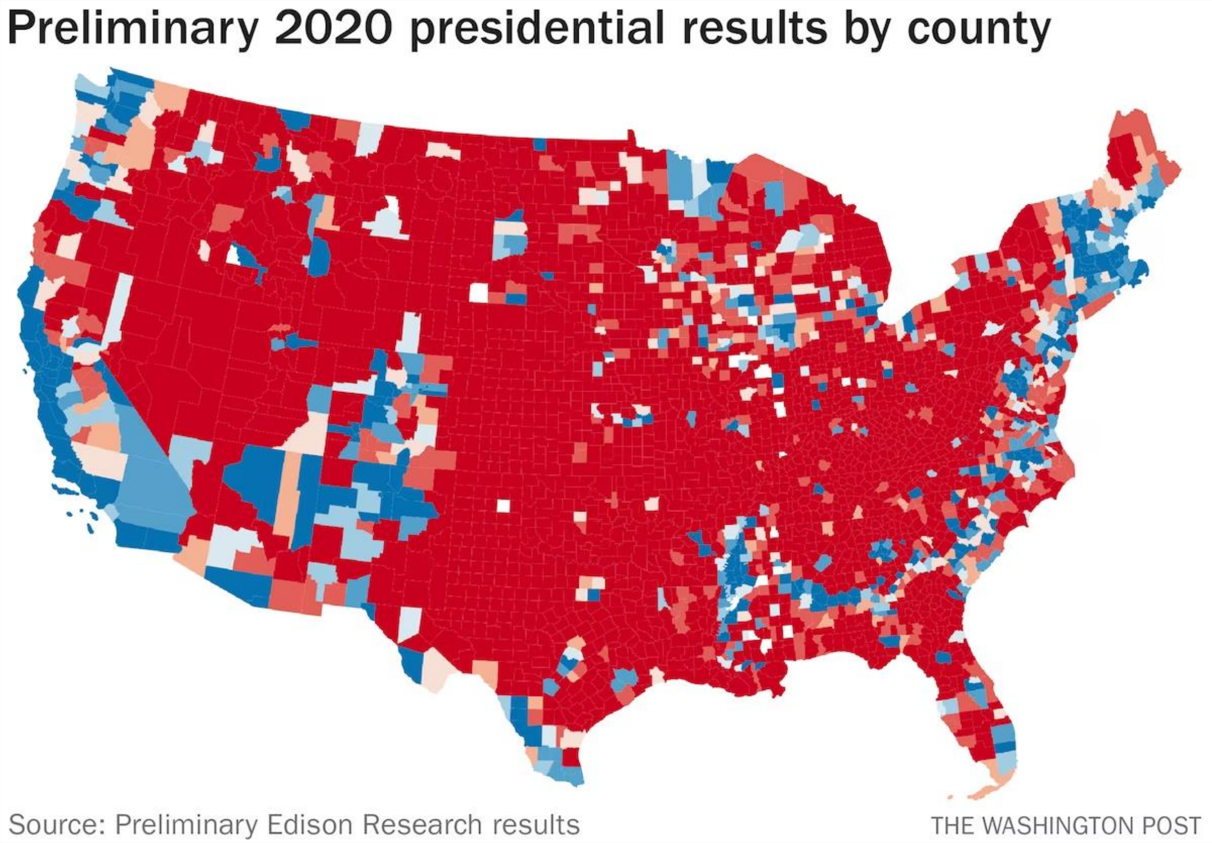
Pointers for Effective Visualizations

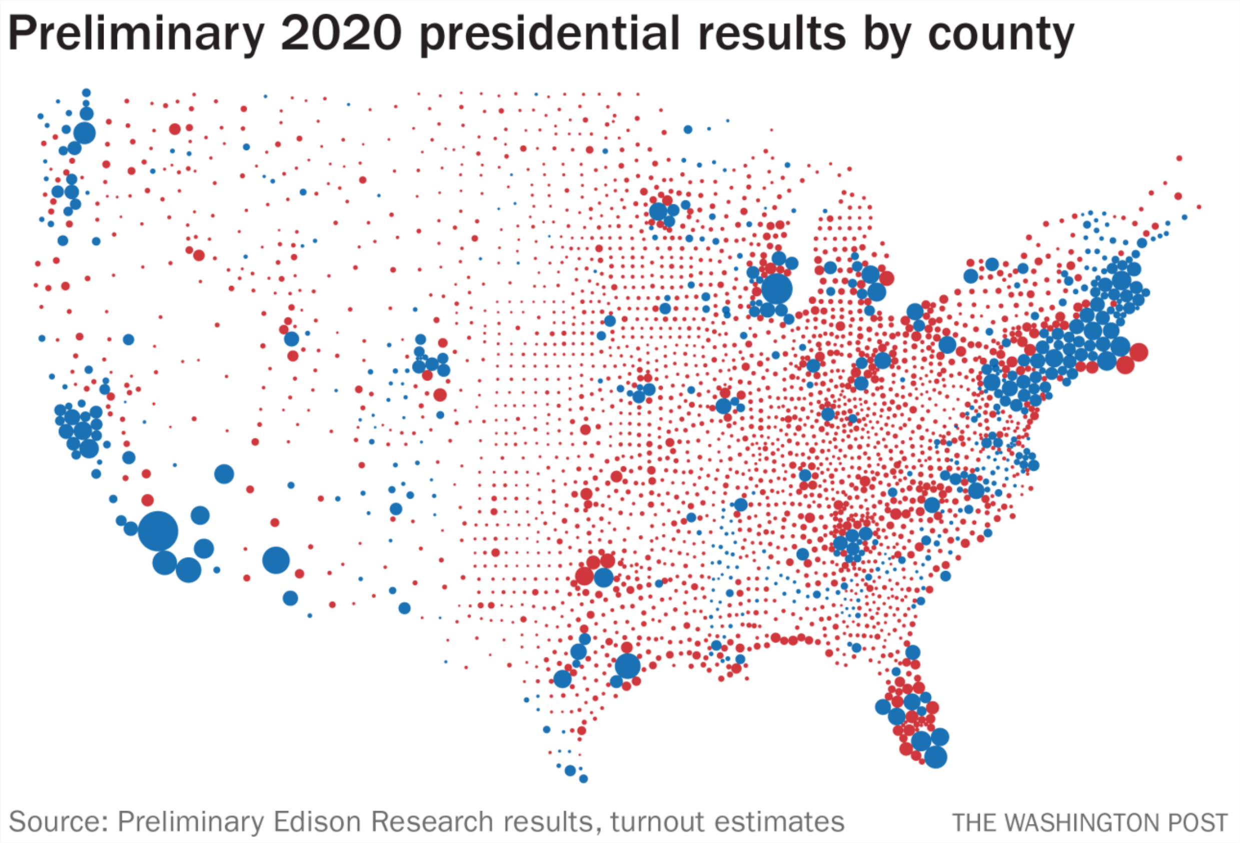
- Have graphical integrity
- Keep it simple
- Use the right display
- Use color strategically
- Know your audience



Graphical Integrity

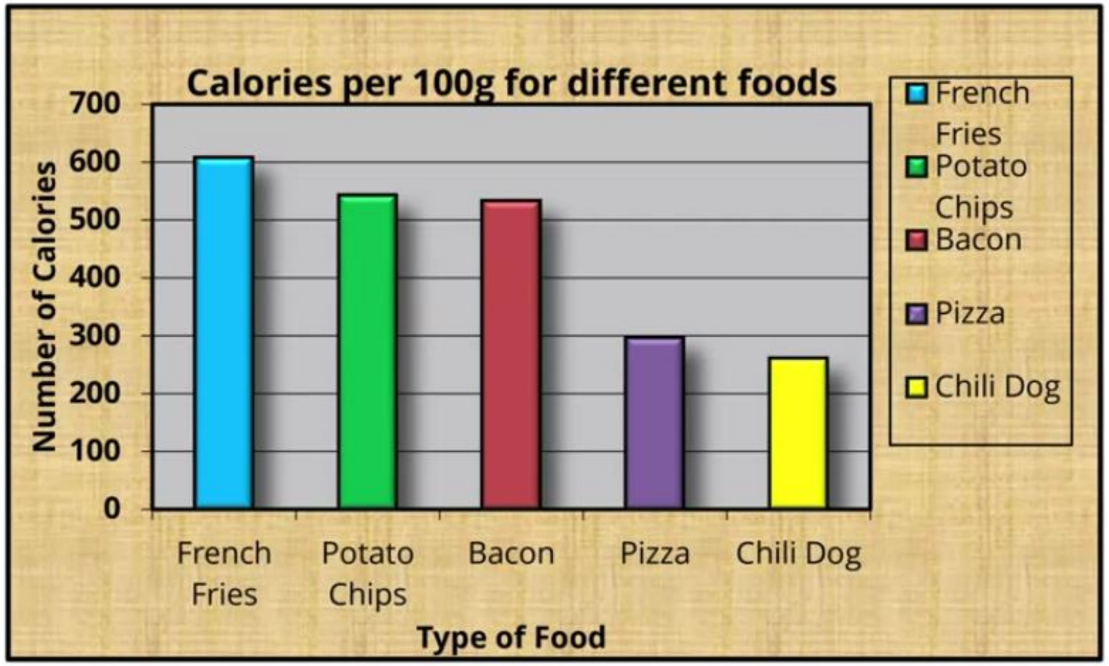
What does this map suggest about the 2020 election?

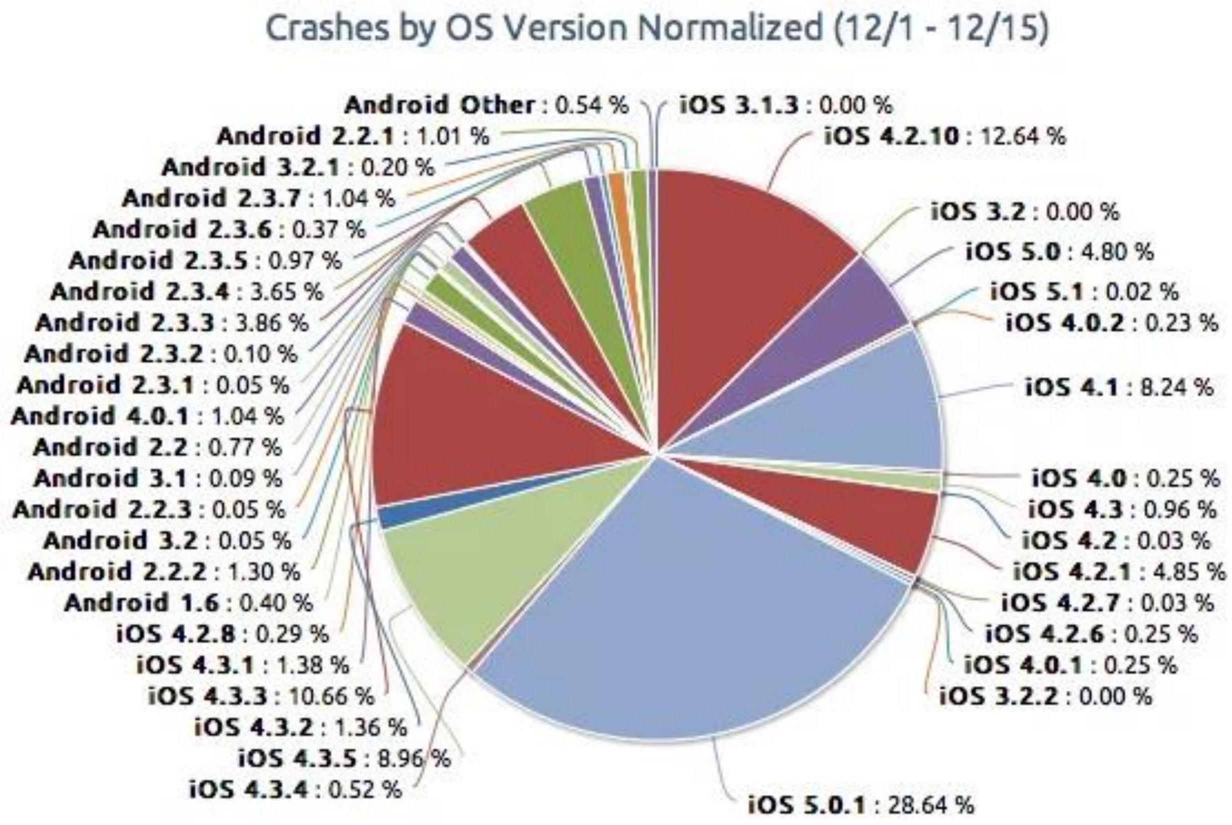




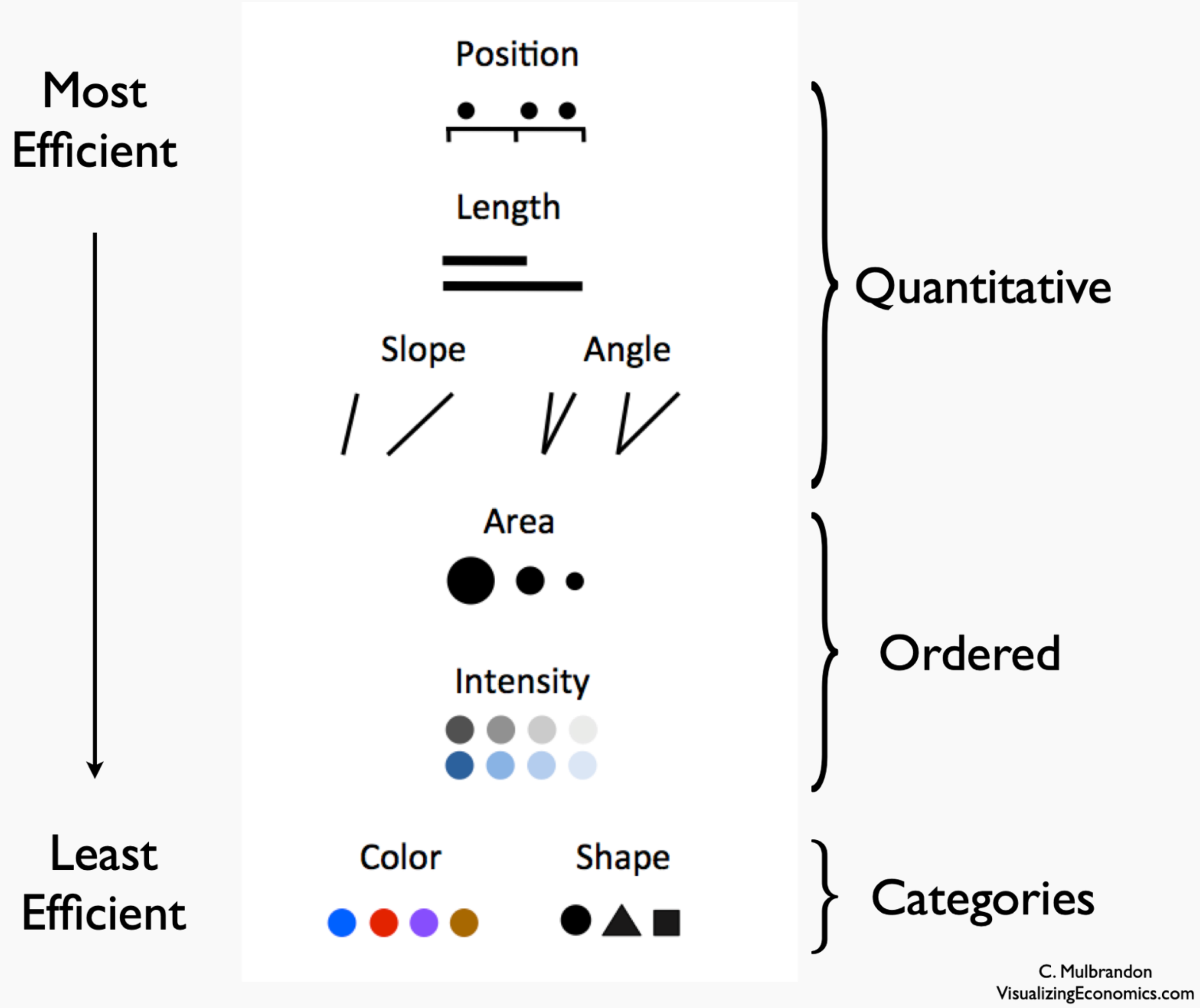
Keep it simple

Avoid Chart Junk



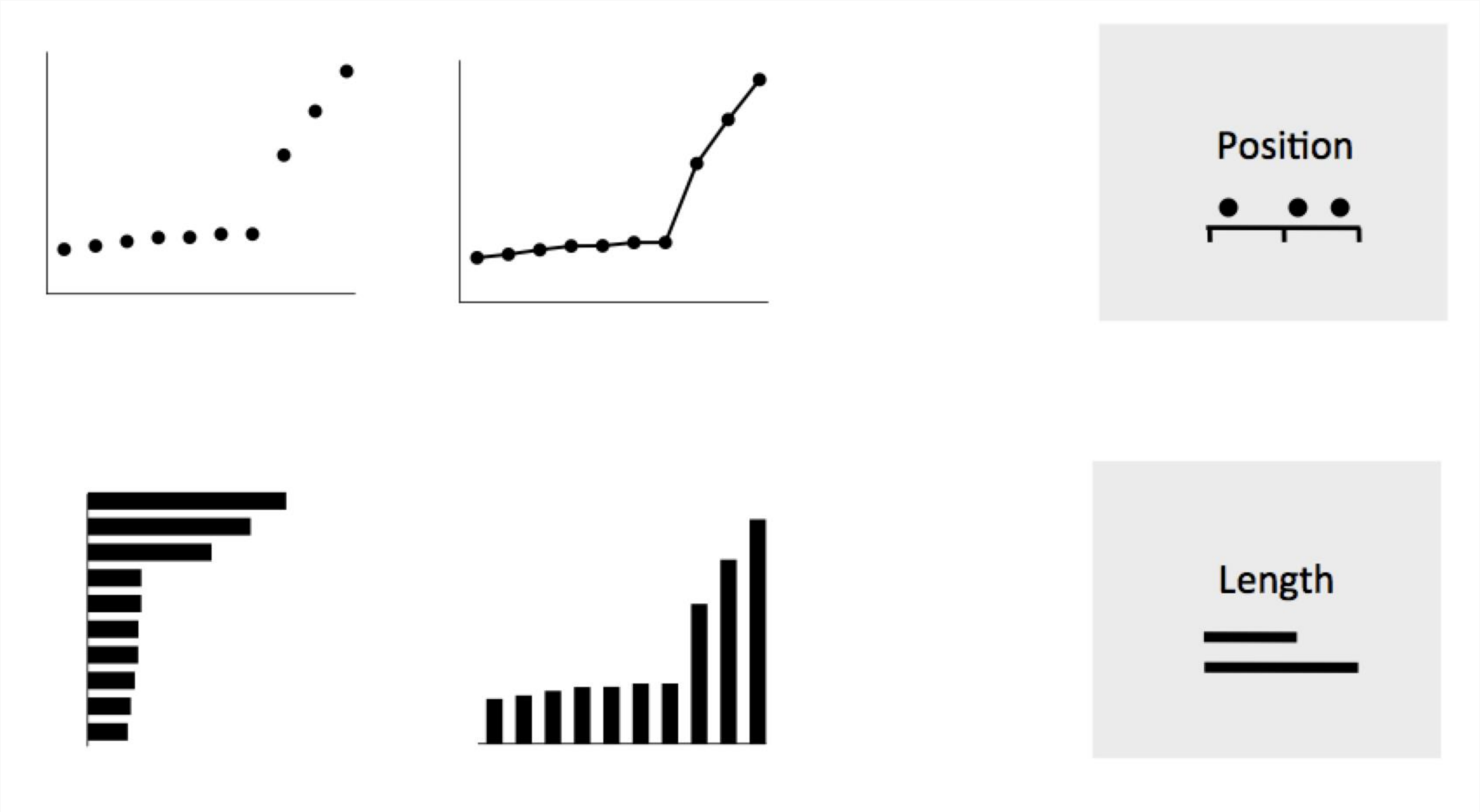


Use the right display



Use the right display

More effective (numeric scale)

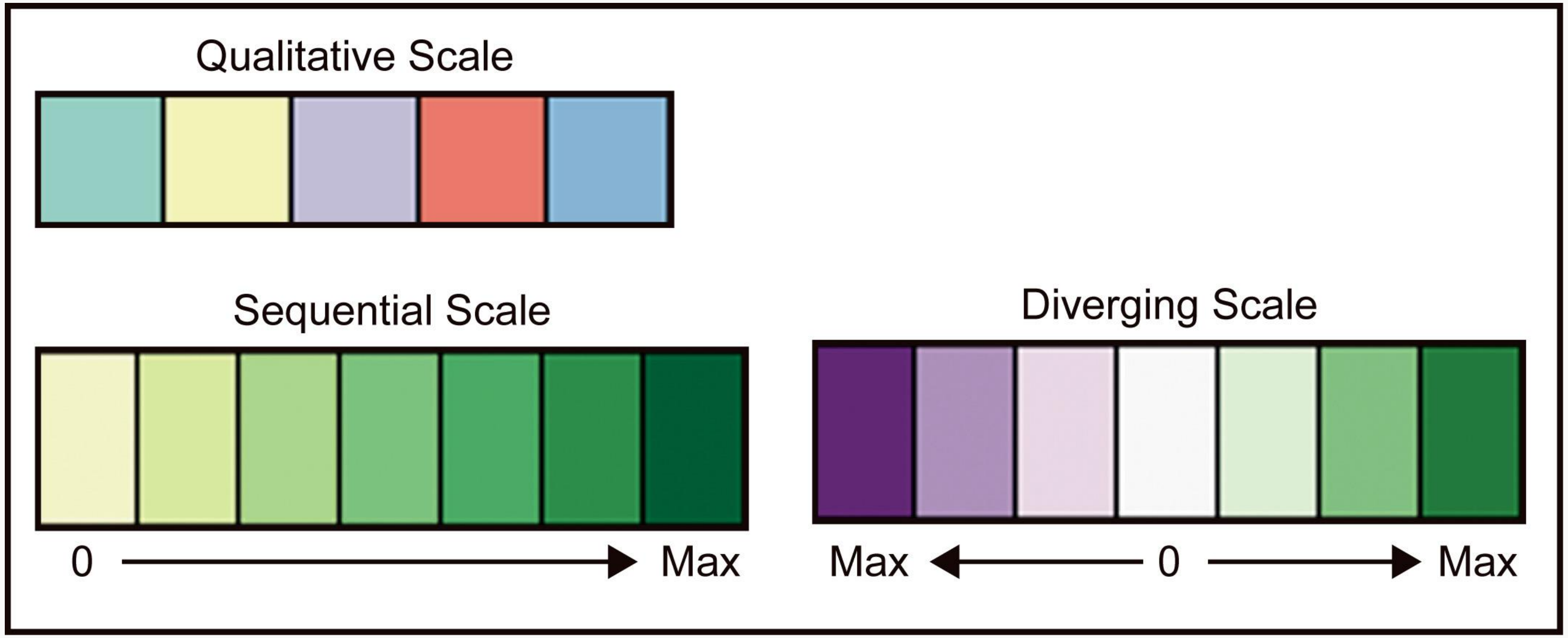


Use the right display

Less effective (numeric scale)

Use color strategically

Type of scale matters:



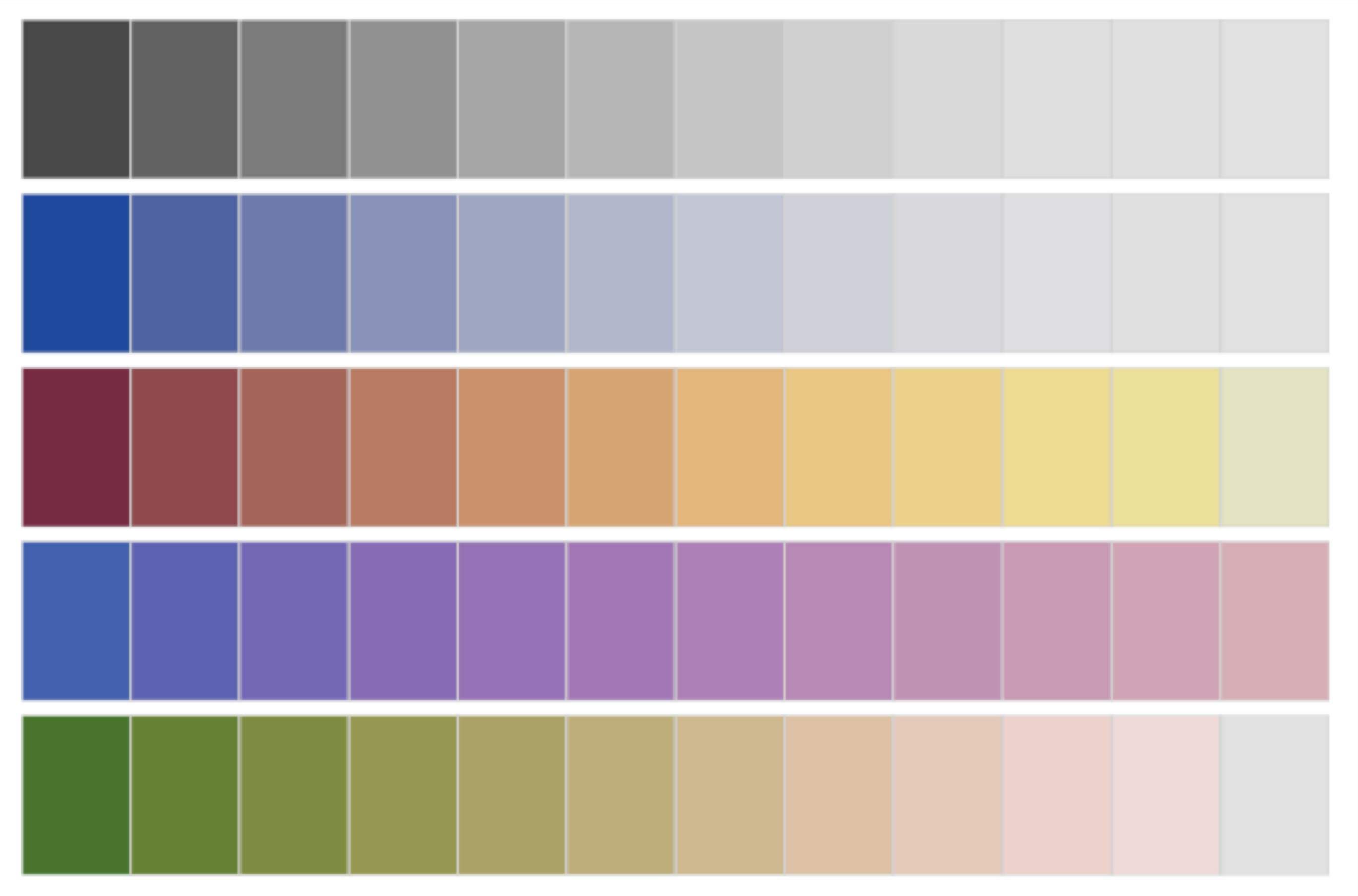
What are example variables for each?

Use color strategically

Colors for categories (Do not use more than 5-8 colors at once)

Use color strategically

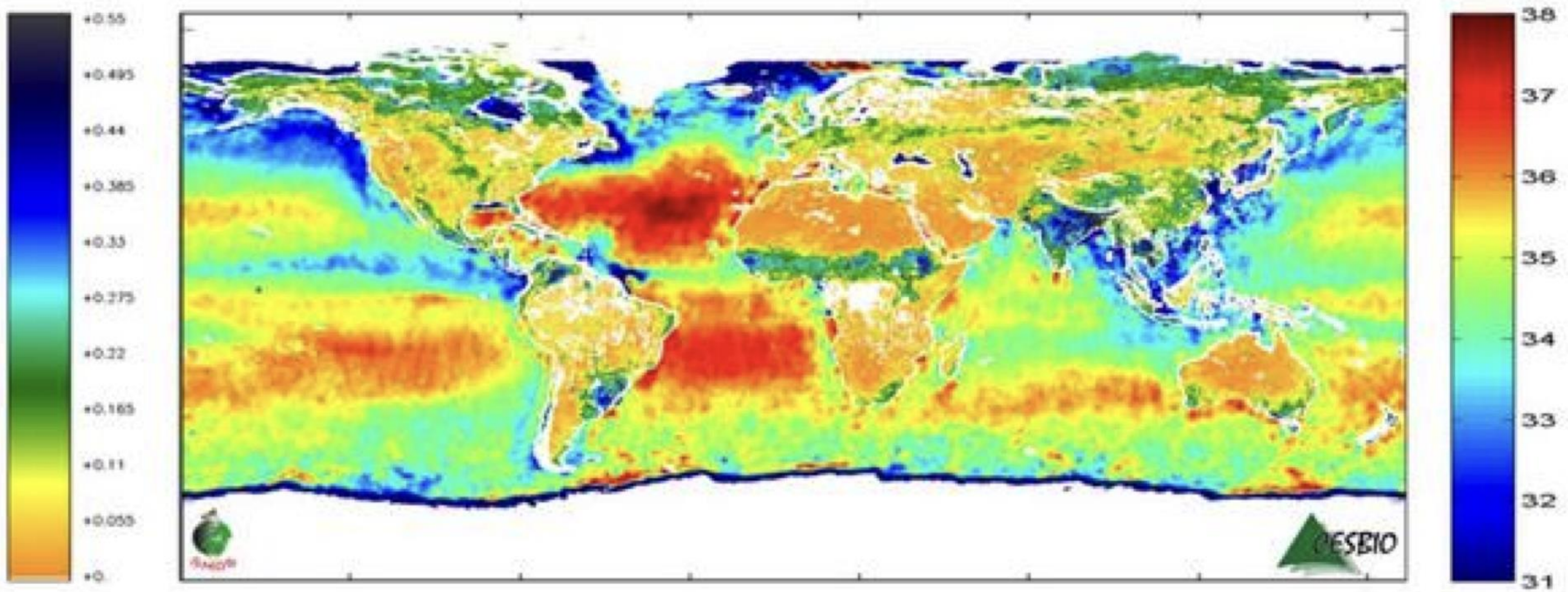
Colors for Ordinal Data -- Vary luminance and saturation



Zeilis et al, 2009, "Escaping RGBland: Selecting Colors for Statistical Graphics"

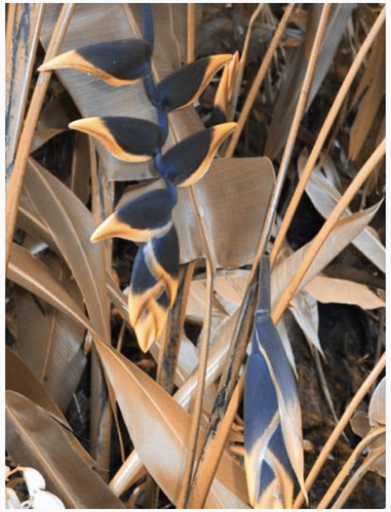
Use color strategically

Beware of the Rainbow Colormap (Keep it simple)

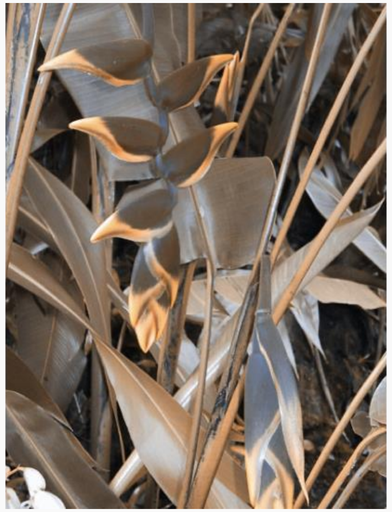


Use color strategically

Be cognizant of color blindness



Protanope



Deuteranope

Tritanope

Red / green deficiencies

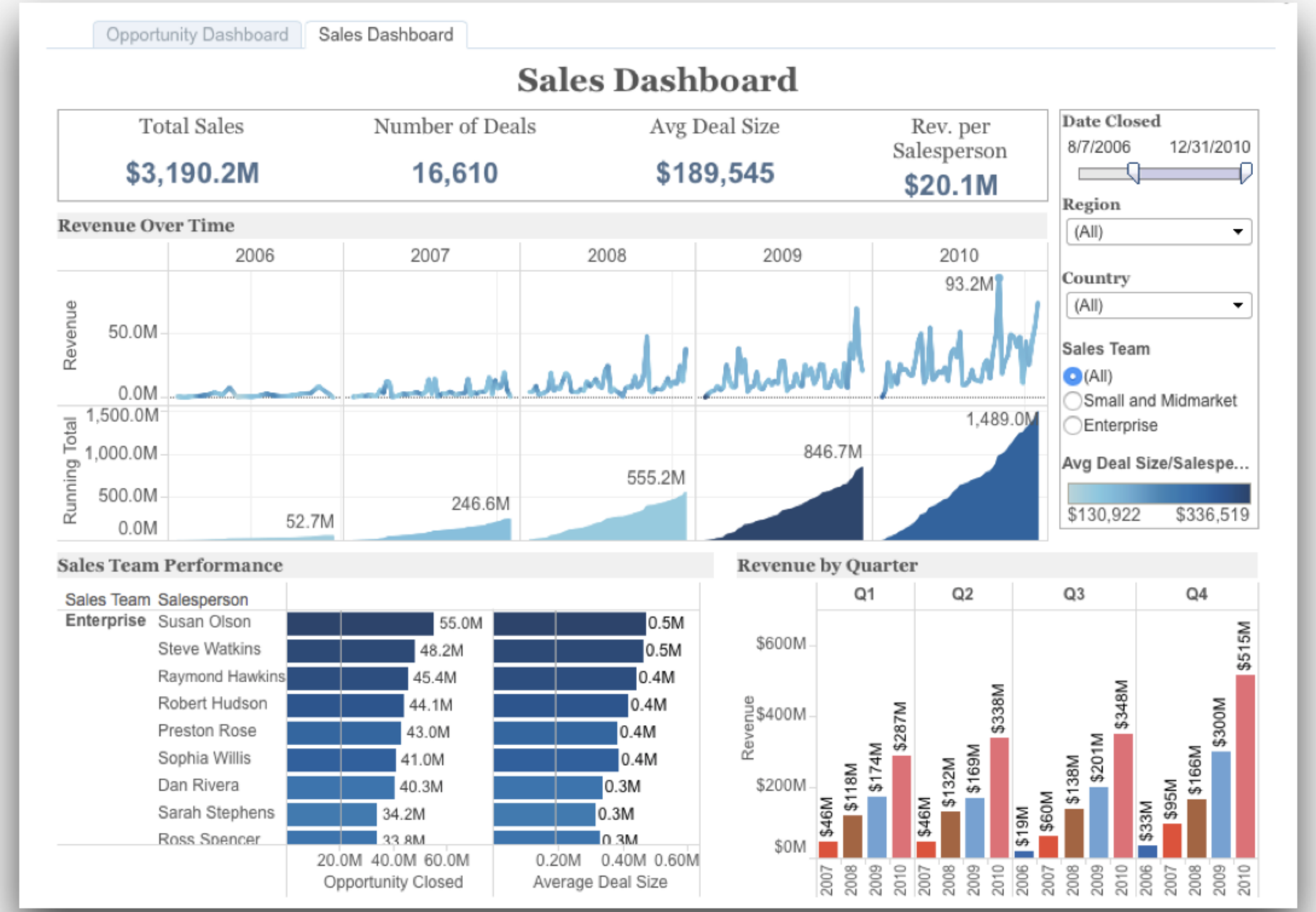
Blue / Yellow deficiency

Know your audience

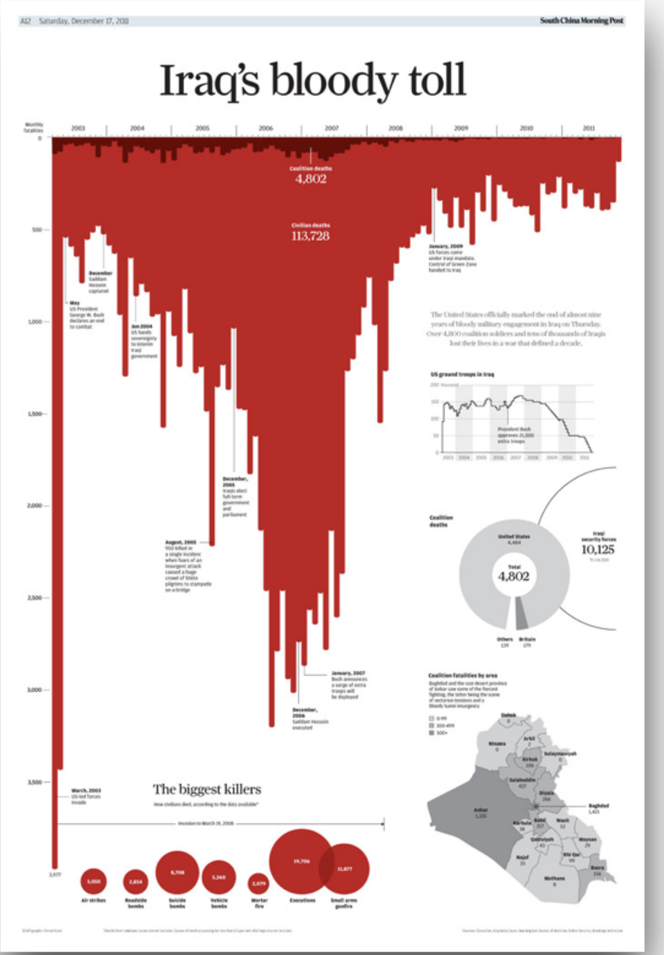
- What do they know?
- What motivates them? What do they desire?
- What experiences do you share? What are common goals?
- What insights can you provide? What tools and “magical gifts”do you have access to?

Know your audience

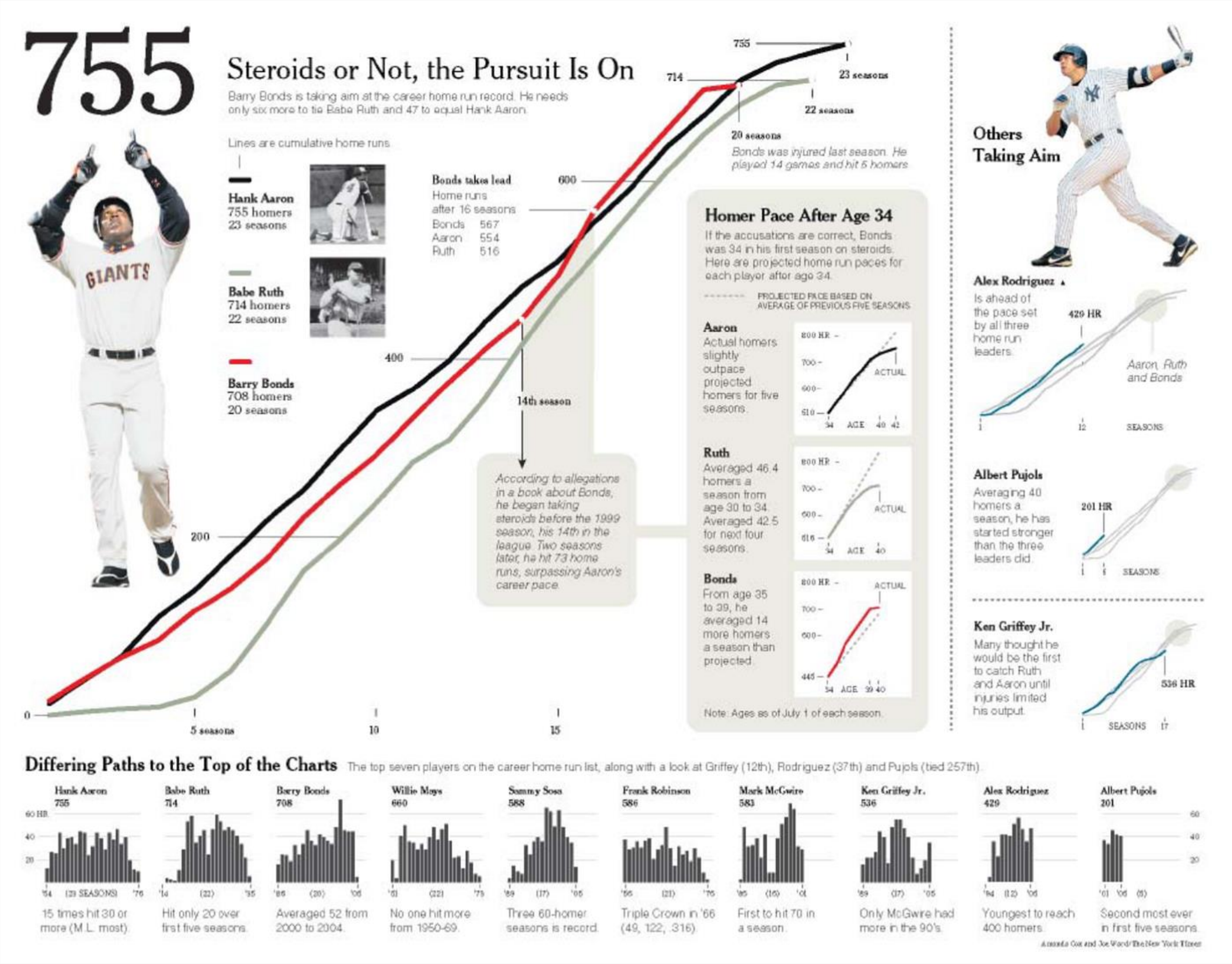
Explanatory Neutral



Explanatory Opinionated

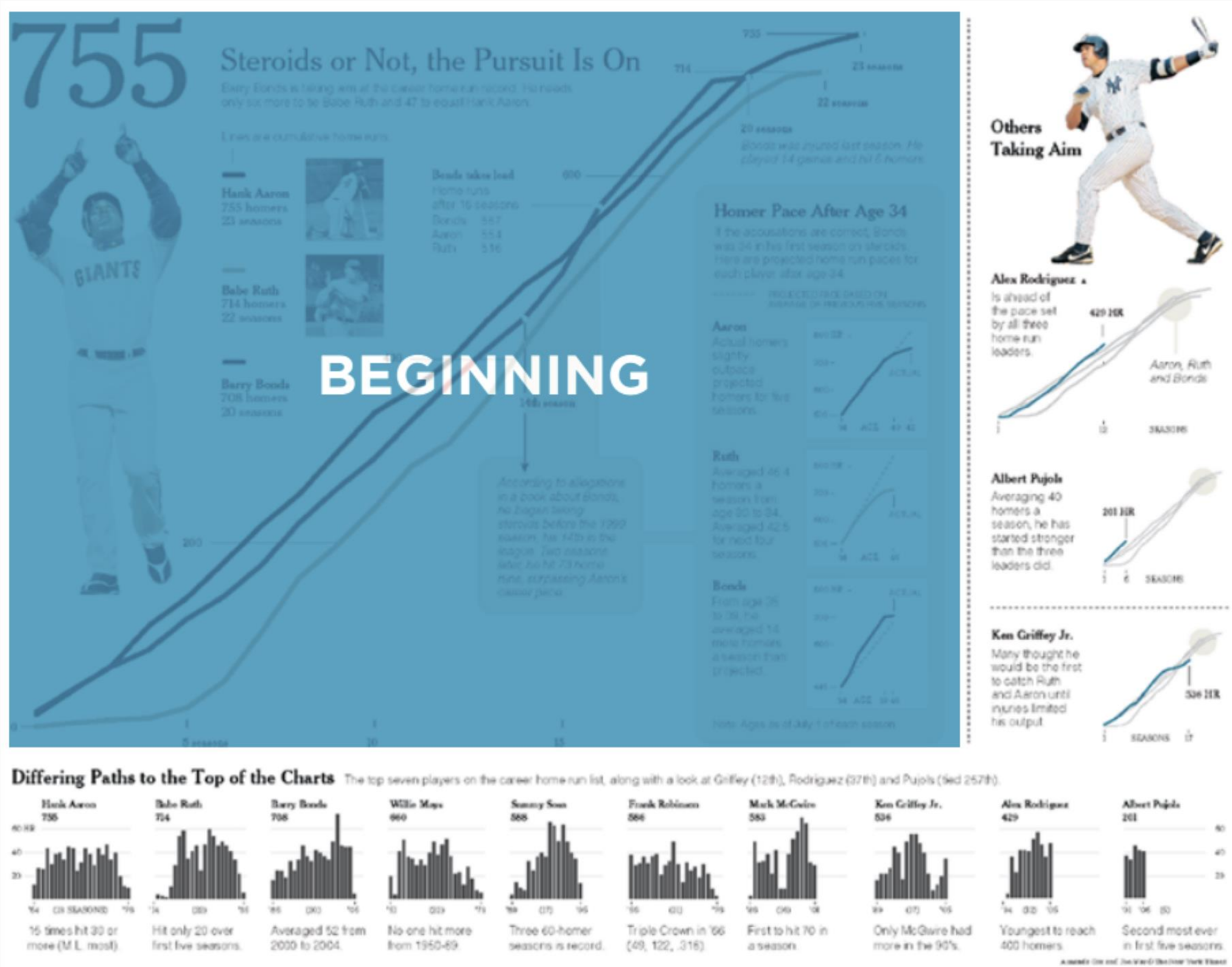


Tell a story



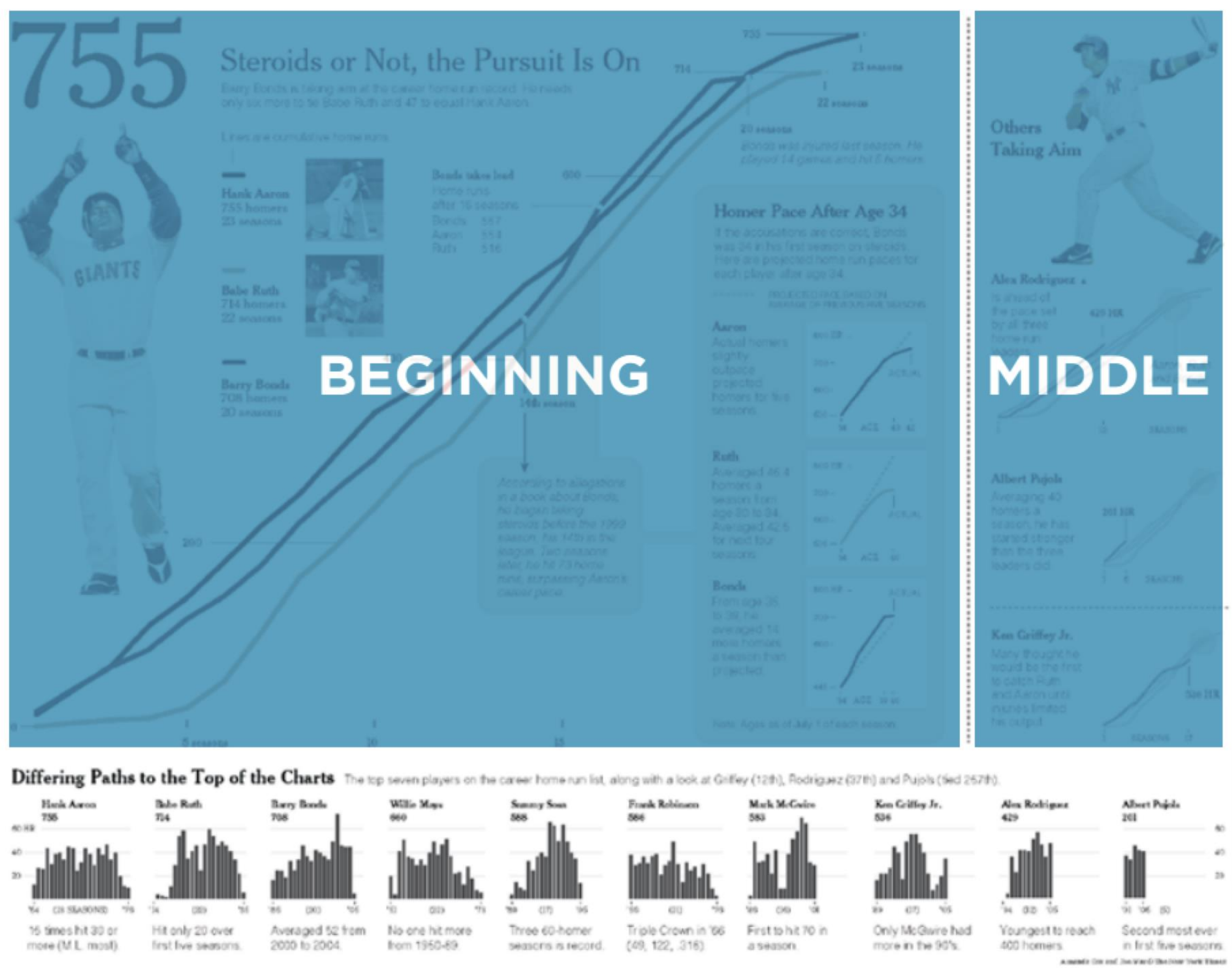
755
Steroids or Not, the Pursuit Is On
Barry Bonds is taking aim at the career home run record. He needs only six more to tie Babe Ruth and 47 to equal Hank Aaron.
Lines are cumulative home runs.
Hank Aaron
755 homers
23 seasons
Babe Ruth
714 homers
22 seasons
Barry Bonds
708 homers
20 seasons
Bonds takes lead
Home runs after 16 seasons
Bonds 567
Aaron 554
Ruth 516
14th season
According to allegations in a book about Bonds, he began taking steroids before the 1999 season, his 14th in the league. Two seasons later, he hit 73 home runs, surpassing Aaron's career pace.
23 seasons
22 seasons
20 seasons
Bonds was injured last season. He played 14 games and hit 5 homers.
5 seasons
10
15
Homer Pace After Age 34
If the accusations are correct, Bonds was 34 in his first season on steroids. Here are projected home run paces for each player after age 34.
PROJECTED PACE BASED ON AVERAGE OF PREVIOUS FIVE SEASONS
Aaron
Actual homers slightly outpace projected homers for five seasons.
Ruth
Averaged 46.4 homers a season from age 30 to 34. Averaged 42.5 for next four seasons.
Bonds
From age 35 to 39, he averaged 14 more homers a season than projected.
ACTUAL
AGE
Note: Ages as of July 1 of each season.
Others Taking Aim
Alex Rodriguez
Is ahead of the pace set by all three home run leaders.
429 HR
Aaron, Ruth and Bonds
SEASONS
Albert Pujols
Averaging 40 homers a season, he has started stronger than the three leaders did.
201 HR
Ken Griffey Jr.
Many thought he would be the first to catch Ruth and Aaron until injuries limited his output.
536 HR
Differing Paths to the Top of the Charts
The top seven players on the career home run list, along with a look at Griffey (12th), Rodriguez (37th) and Pujols (tied 257th).
Hank Aaron 755
Babe Ruth 714
Barry Bonds 708
Willie Mays 660
Sammy Sosa 588
Frank Robinson 586
Mark McGwire 583
Ken Griffey Jr. 536
Alex Rodriguez 429
Albert Pujols 201
15 times hit 30 or more (M.L. most).
Hit only 20 over first five seasons.
Averaged 52 from 2000 to 2004.
No one hit more from 1950-69.
Three 60-homer seasons is record.
Triple Crown in '66 (49, 122, .316).
First to hit 70 in a season.
Only McGwire had more in the 90's.
Youngest to reach 400 homers.
Second most ever in first five seasons.
Amanda Cox and Joe Ward/The New York Times

Tell a story

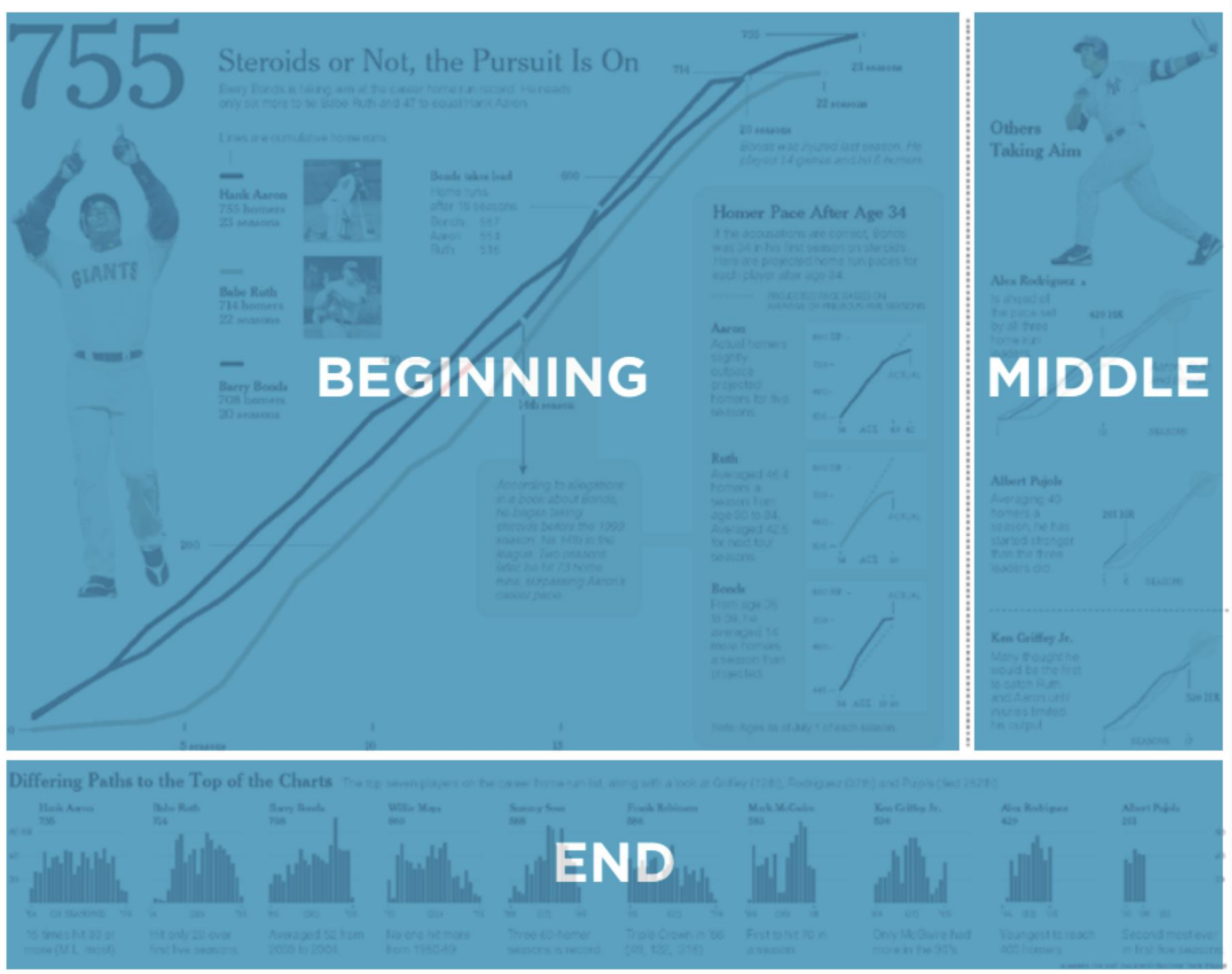


755
Steroids or Not, the Pursuit Is On
Hank Aaron
755 homers
23 seasons
Babe Ruth
714 homers
22 seasons
Barry Bonds
708 homers
20 seasons
BEGINNING
Homer Pace After Age 34
Aaron
Ruth
Bonds
Others Taking Aim
Alex Rodriguez
Is ahead of the pace set by all three home run leaders.
Aaron, Ruth and Bonds
Albert Pujols
Averaging 40 homers a season, he has started stronger than the three leaders did.
Ken Griffey Jr.
Many thought he would be the first to catch Ruth and Aaron until injuries limited his output.
Differing Paths to the Top of the Charts
The top seven players on the career home run list, along with a look at Griffey (12th), Rodriguez (37th) and Pujols (tied 257th).
Hank Aaron 755
Babe Ruth 714
Barry Bonds 708
Willie Mays 660
Sammy Sosa 588
Frank Robinson 586
Mark McGwire 583
Ken Griffey Jr. 536
Alex Rodriguez 429
Albert Pujols 201
15 times hit 30 or more (M.L. most).
Hit only 20 over first five seasons.
Averaged 52 from 2000 to 2004.
No one hit more from 1950-69.
Three 60-homer seasons is record.
Triple Crown in '66 (49, 122, .316).
First to hit 70 in a season.
Only McGwire had more in the 90's.
Youngest to reach 400 homers.
Second most ever in first five seasons.

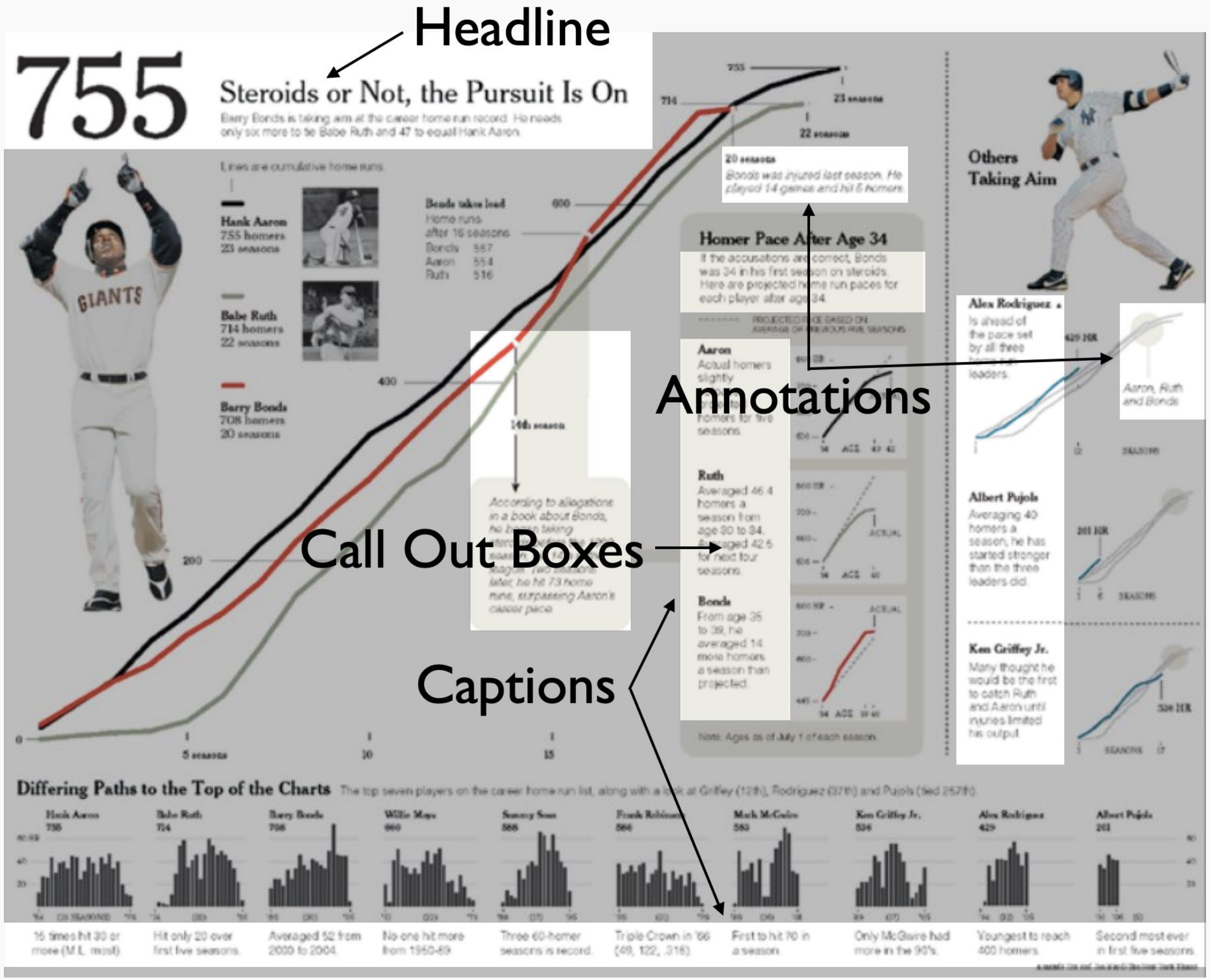
Tell a story


755
Steroids or Not, the Pursuit Is On
Hank Aaron
755 homers
23 seasons
Babe Ruth
714 homers
22 seasons
Barry Bonds
708 homers
20 seasons
BEGINNING
Homer Pace After Age 34
Aaron
Ruth
Bonds
Others Taking Aim
Alex Rodriguez
Albert Pujols
Ken Griffey Jr.
MIDDLE
Differing Paths to the Top of the Charts
The top seven players on the career home run list, along with a look at Griffey (12th), Rodriguez (37th) and Pujols (tied 257th).
Hank Aaron 755
Babe Ruth 714
Barry Bonds 708
Willie Mays 660
Sammy Sosa 588
Frank Robinson 586
Mark McGwire 583
Ken Griffey Jr. 536
Alex Rodriguez 429
Albert Pujols 201
15 times hit 30 or more (M.L. most).
Hit only 20 over first five seasons.
Averaged 52 from 2000 to 2004.
No one hit more from 1950-69.
Three 60-homer seasons is record.
Triple Crown in '66 (49, 122, .316).
First to hit 70 in a season.
Only McGwire had more in the 90's.
Youngest to reach 400 homers.
Second most ever in first five seasons.

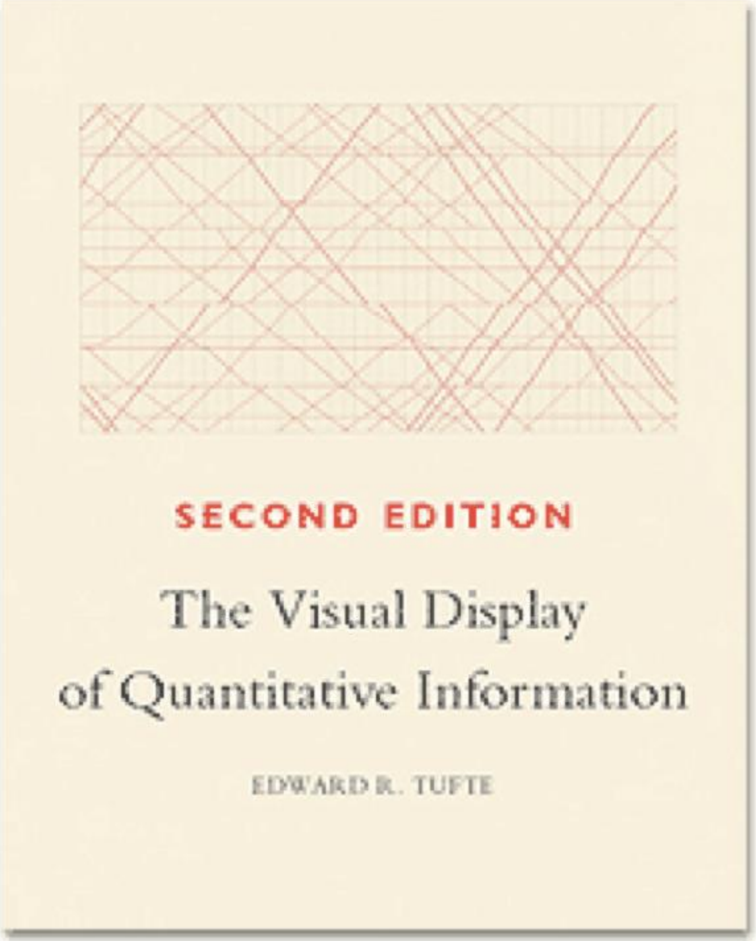
Tell a story

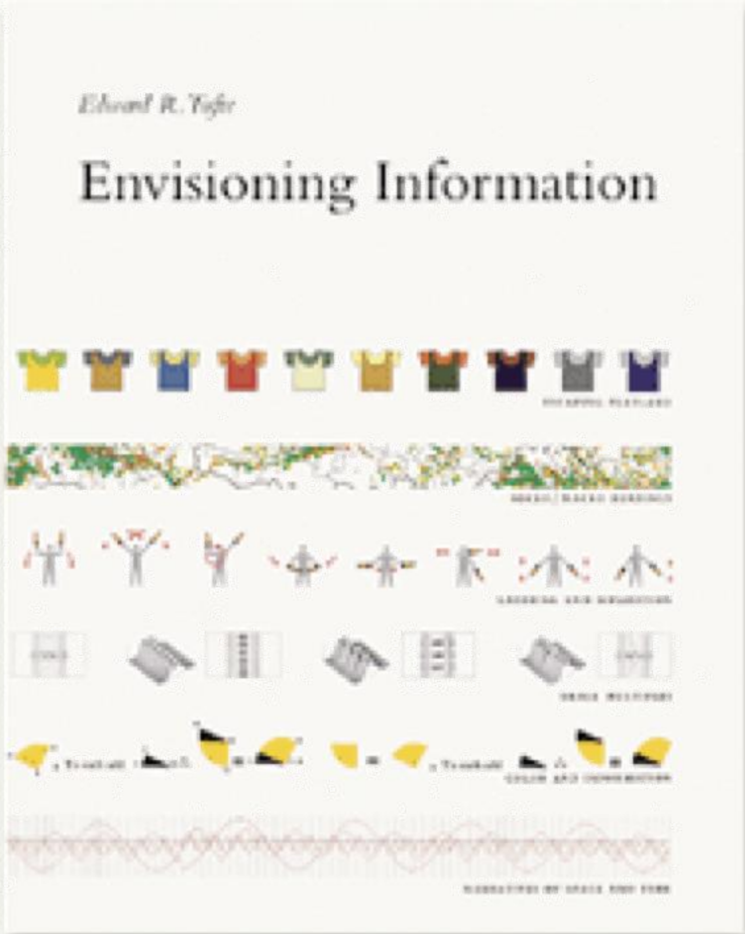

755
Steroids or Not, the Pursuit Is On
BEGINNING
Homer Pace After Age 34
Others Taking Aim
MIDDLE
Differing Paths to the Top of the Charts
END

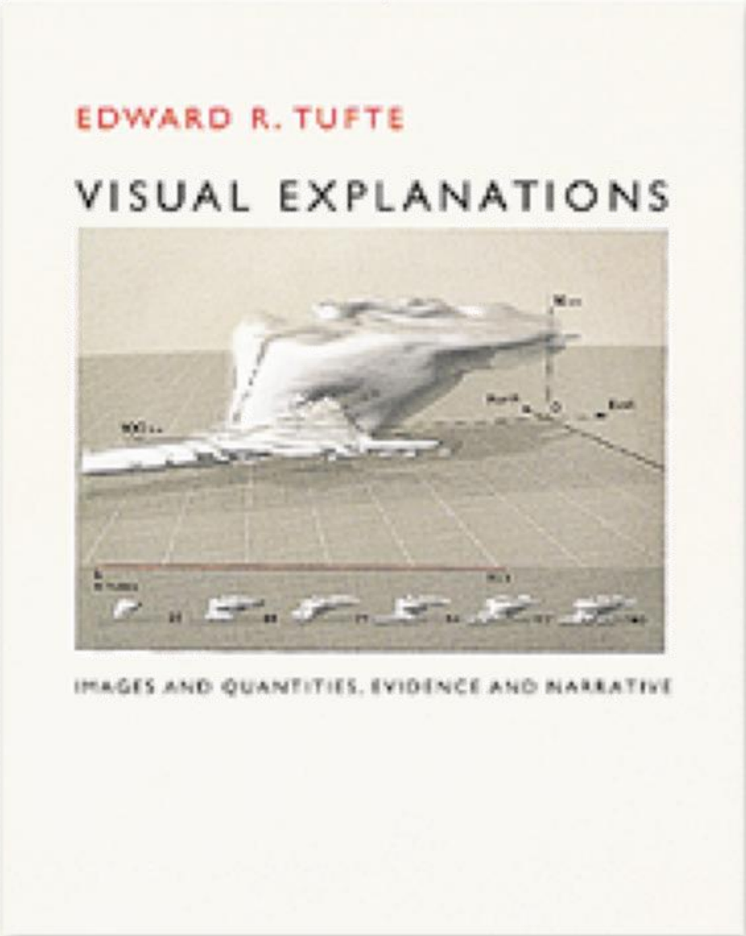
Tell a story


Headline
755
Steroids or Not, the Pursuit Is On
Annotations
Call Out Boxes
Captions
Homer Pace After Age 34
Others Taking Aim
Differing Paths to the Top of the Charts

Edward Tufte







Edward Tufte's Principles of Graphical Excellence

Graphical excellence …

- is the well-designed presentation of interesting data—a matter of substance, of statistics, and of design.
- consists of complex ideas communicated with clarity, precision and efficiency.
- is that which gives to the viewer the greatest number of ideas in the shortest time with the least ink in the smallest space.
- is nearly always multivariate.
- requires telling the truth.

XKCD for the win