Quantitative Cleaning

Data Cleaning & Integration

CompSci 590.01 Spring 2017

DUKE
COMPUTER SCIENCE

Some contents were based on:
Hellerstein's *QDB* 2009 keynote slides

J. Hellerstein. "Quantitative Data Cleaning for Large Databases." Technical Report, UC Berkeley, 2008

- Not published elsewhere, but nonetheless an excellent, well-cited survey
- Focuses on cleaning quantities (numbers)
  - Though some techniques are still applicable to non-numeric data; e.g. "counts" are quantities that can be derived from them

Biggest take-away points?

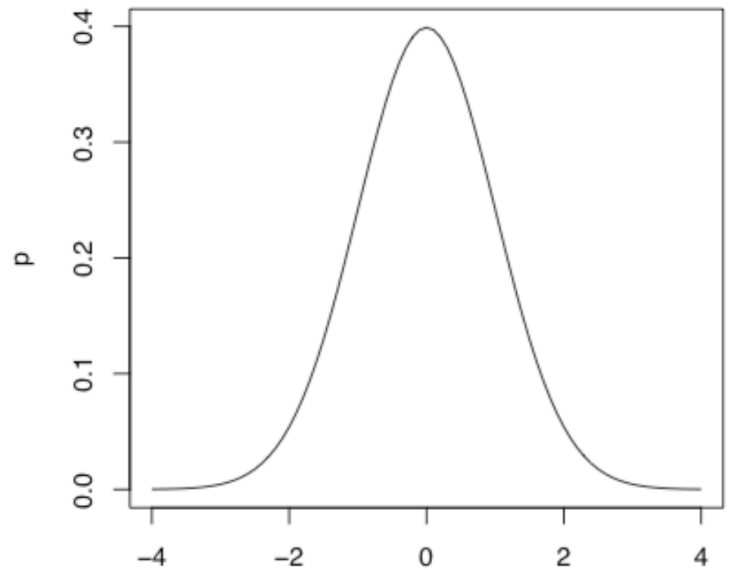
(For Jun:)

- Before you can talk about what's dirty, you need to define what's clean or "typical"
  - E.g., center/dispersion
- You need "robust" statistics
  - Or else your measures of what's "typical" can be easily tainted by dirty data in the first place
- Friction works better than constraint enforcement, which often invites "spurious integrity"

Atypical = “far from the center”

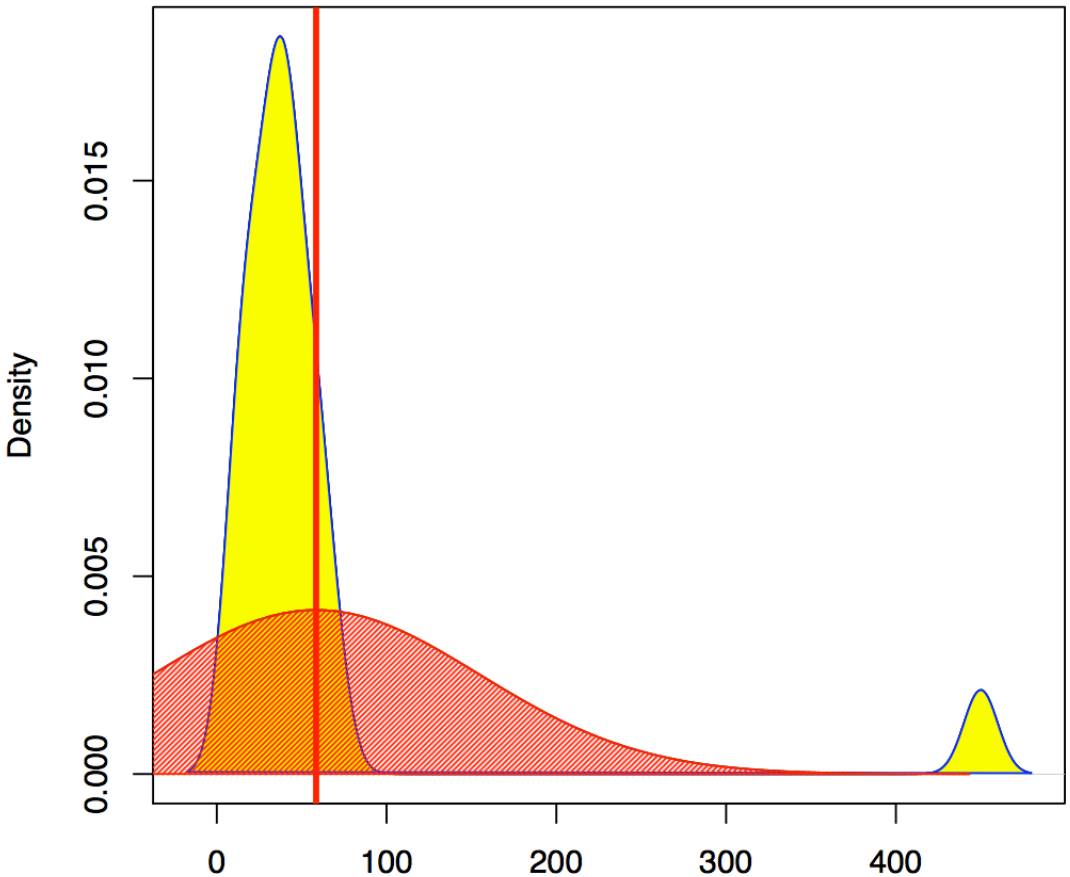
- Center: mean
- Dispersion: variance (or standard deviation)

- Implicit assumption: normal distribution (a.k.a. Gaussian, bell curve)
  - Knowing just mean and variance, Gaussian is your “maximum entropy distribution”



Traditional center/dispersion

- Ages of employees: [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 68, 450]

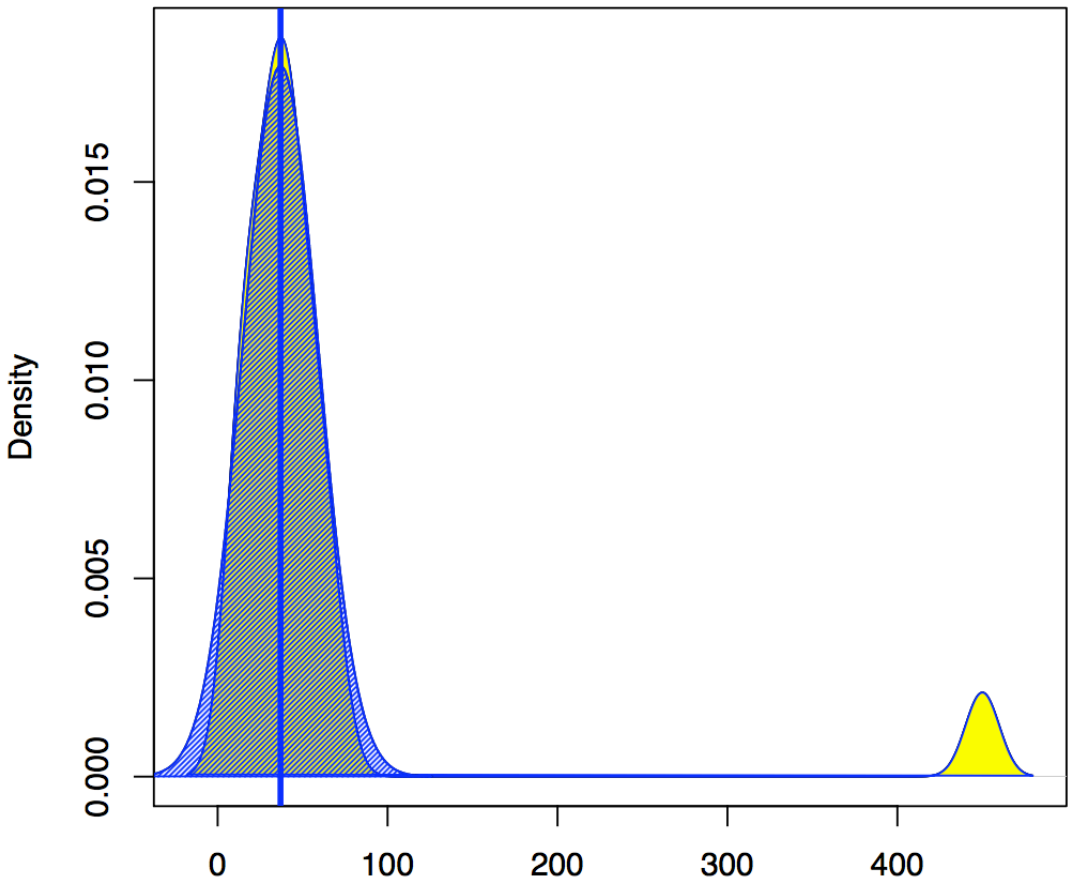


Mean: 58.52632

Variance: 9252.041

Robust center/dispersion

- Ages of employees: [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 68, 450]



Median: 37

MAD: 22.239

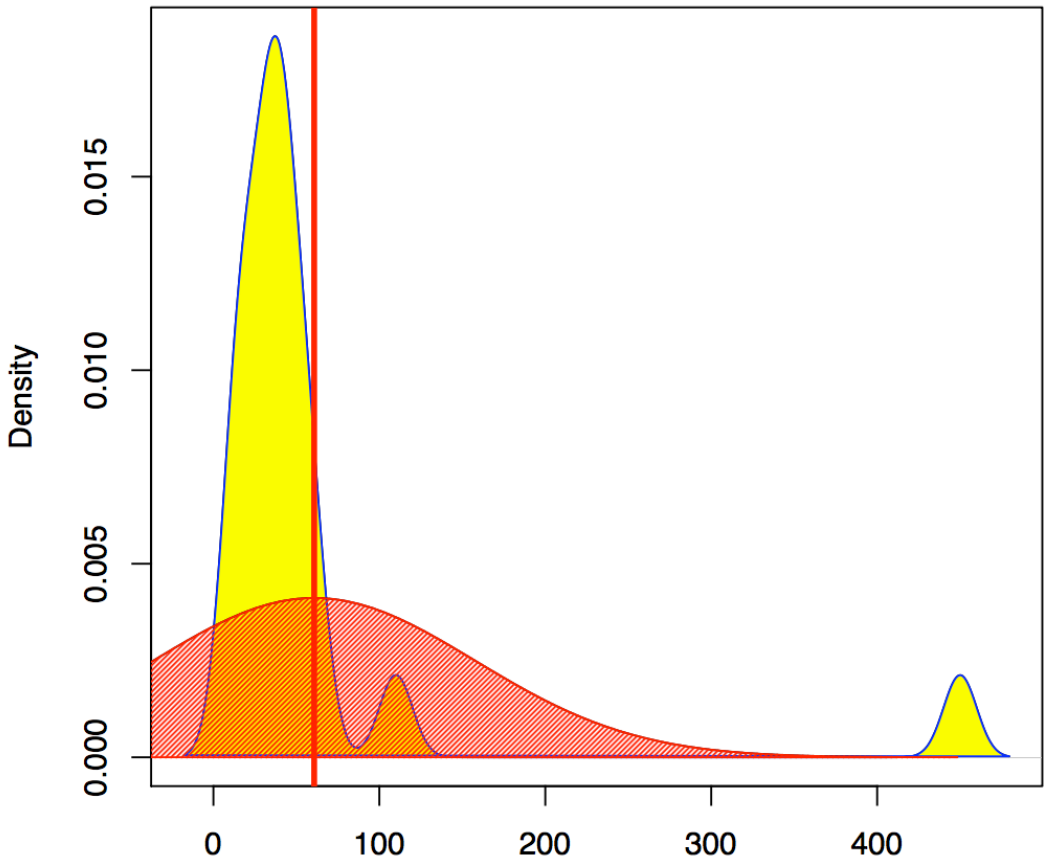
Two measures compared

- Standard deviation (square root of variance):
  $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(k_i - \mu)^2}$, where $\mu = \frac{1}{n}\sum_{i=1}^{n} k_i$

- Median absolute deviation (MAD):
  $$\mathrm{median}_i\{\, |k_i - \mathrm{median}_i\{k_i\}|\, \}$$

Subtler problems

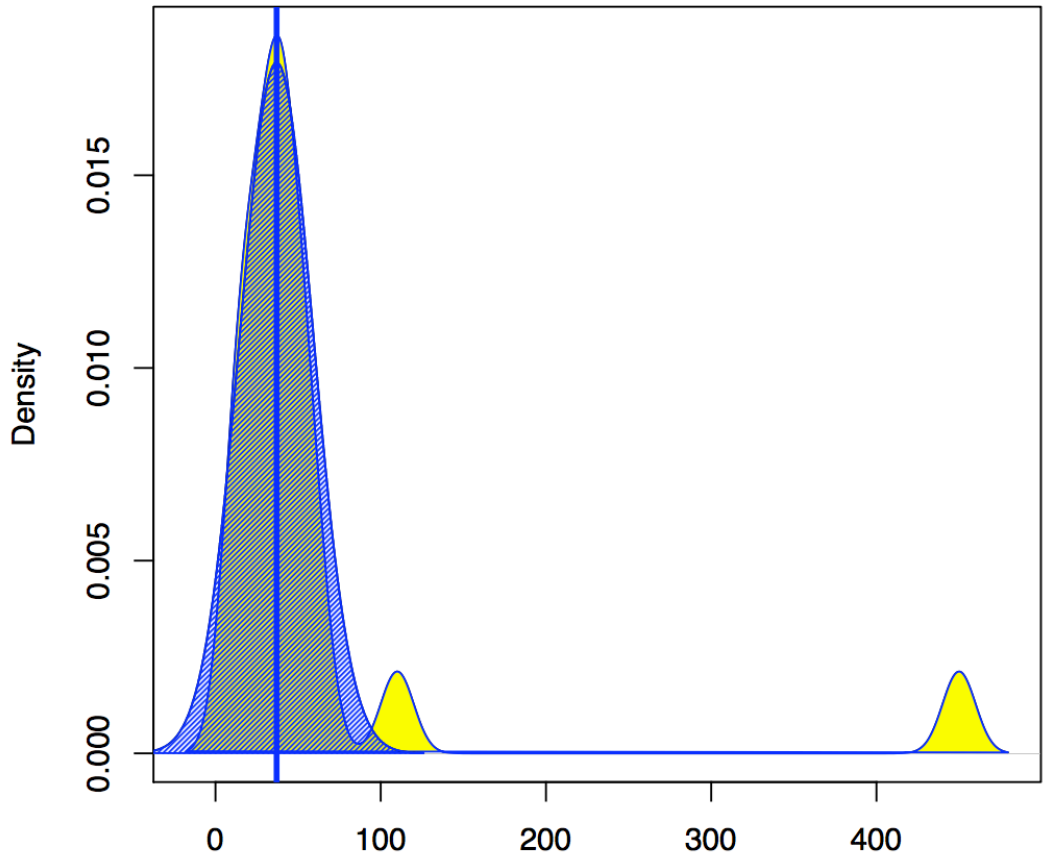
- Ages of employees: [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 110, 450]



Masking: magnitude of some outlier masks others, making them harder to identify

How robust statistics help

- Ages of employees: [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 110, 450]



Cope with multiple outliers and large-magnitude outliers

How robust is robust?

- "Breakdown point" measures the robustness of an estimator
  - Proportion of "dirty" data the estimator can handle before giving an arbitrarily erroneous result
  - Think adversarially
- Best possible breakdown point is 50%
  - Beyond 50% "noise," what's the "signal"?

Some breakdown points

- Mean?
  - $1/n$: just one outlier is enough!
- Standard deviation?
  - Also $1/n$
- Mode?
  - Depends on the frequency of the mode
- Median?
  - 50%, the best you can hope for
    - It does NOT mean you need to tweak 50% of the data to change the median

Robust centers

- Ages of employees: [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 110, 450]
- Median: 37
- *k*% trimmed mean
  - Remove lowest/highest *k*% values (12, 13, 110, 450)
  - Compute mean on the remainder (37.933)
- *k*% winsorized mean
  - Remove lowest/highest *k*% values (12, 13, 110, 450)
  - Replace low removed with lowest remaining (14, 14)
  - Replace high removed with highest remaining (61, 61)
  - Compute mean on the result (37.842)

Breakdown points for these?

Robust dispersion

- MAD
- IQR (interquartile range)
  - Difference between 25% and 75% quartiles
  - [12, 13, 14, 21, 22, 26, 33, 35, 36, 37, 39, 42, 45, 47, 54, 57, 61, 110, 450]
  - Note for symmetric distributions, MAD is IQR/2
- Or just do *k*% trimming or winsorization and then compute standard deviation

Fit using robust center/dispersion

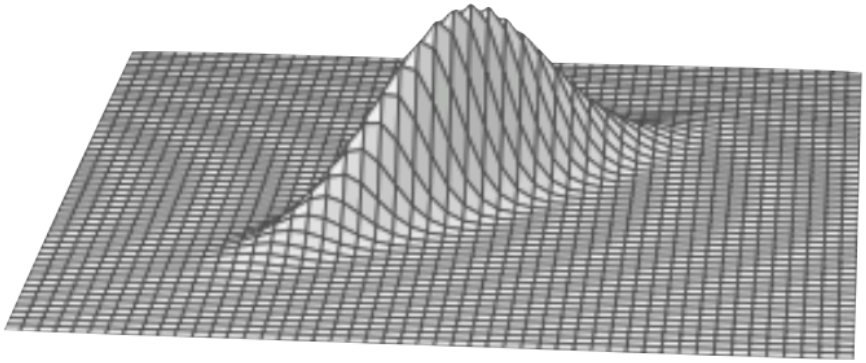
- Gaussian
  - MAD: 75 percentile, so re-scale standard deviation in terms of MAD
    - $\hat{\sigma} = 1.4826 \cdot \mathrm{MAD}$
  - Re-center at median and off you go
- What if data is not Gaussian?
  - Transform so the remaining error (or "residual") is

Computing order-based stats

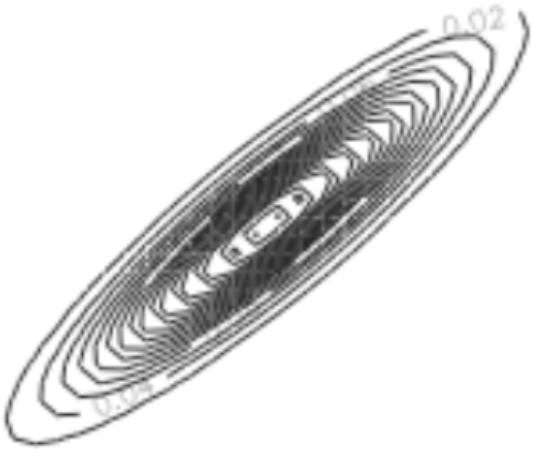
- SQL sucks; use user-defined functions
- Sorting is always an option
- Linear-time median (or selection in general)
  - Pivot, with median of medians
- One-pass algorithms?
  - Randomized algorithms that use sampling or sketches (e.g., range counting with CountMin by Cormode & Muthukrishnan, *J. Algo.* 2005)
  - Deterministic algorithms that produce approximate answers (e.g., Greenwald & Khanna, *SIGMOD* 2001; Shrivastava et al., *SenSys* 2004)

Moving to multiple dimensions

- Intuition: multivariate Gaussian
  - Center: multi-dimensional mean
  - Dispersion?



Probability density



Equi-probable contours

Multivariate dispersion

- Sample covariance matrix Σ: $d \times d$ for N d-dimensional points
  $$\Sigma_{ij} = \frac{1}{N-1} \sum_{k=1}^{N} (x_{ik} - \overline{x_i}) \left(x_{jk} - \overline{x_j}\right)$$
  - Symmetric
  - Diagonal stores variance per dimension
  - Off-diagonal reflects covariance between dimensions
  - Captures the scale and orientation of the Gaussian
- Mahalanobis distance between point x and mean μ:
  $$\sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}$$
  - Works in the space scaled and rotated by Σ

Robust multivariate outliers

- Derive order-based statistics based on Mahalanobis distance
- Apply robust stats to individual dimensions, rescale, and then trim using Euclidean distance
- Iteratively delete max-Mahalanobis points, recompute center/dispersion, and repeat
- Define new robust estimators for mean and covariance
  - E.g., MCD (minimum covariance determinant): pick a subset of points that minimize the determinant of Σ

Resampling techniques

- Start with the dataset $\mathcal{D}$ with outliers
- Repeatedly take samples in a controlled fashion, and compute summary statistics
- Combine computed statistics from the samples carefully to produce a better estimate (than computing it directly from $\mathcal{D}$)
- Standard techniques:
  - Bootstrap: draw $|\mathcal{D}|$ items with replacement; compute statistics; rinse and repeat
  - Jackknife: leave out one item at a time; compute statistics; rinse and repeat

Frequency outliers

- Scenario 1: find potential keys (attributes that uniquely identify rows)
  - Compute *unique row ratio*: # distinct values vs. # rows
    - Not robust to a few values with high frequency
  - More robust: *unique value ratio*: # "unique" values vs. # distinct values
- Scenario 2: find attribute values with unusually high frequencies

Computational techniques

- # of distinct values in one pass
  - FM Sketch (Flajolet & Martin, *JCSS* 1985)
- Heavy hitters in one pass
  - Extension of streaming majority (Misra & Gries, 1982)
  - Or just CountMin Sketch (Cormode & Muthukrishnan, *J. Algo.* 2005)

FM (Flajolet-Martin) sketch

Let $\text{Tail}_0(h(x))$ = # of trailing consecutive 0's

- $\text{Tail}_0(101001) = 0$
- $\text{Tail}_0(101010) = 1$
- $\text{Tail}_0(001100) = 2$
- $\text{Tail}_0(101000) = 3$
- $\text{Tail}_0(000000) = 6$

FM sketch

- Maintain a value K (max 0-tail length)
- Initialize K to 0
- For each new value
  - Compute $\text{Tail}_0(h(x))$
  - Replace K with this value it is greater than K
- $F' = 2^K$ is an estimate of F, the true number of distinct elements

- K require very little space to store

Rough intuition

If we have F distinct elements, we'd expect

- $F/2$ of them to have $\text{Tail}_0(x) = 0$
- $F/4$ of them to have $\text{Tail}_0(x) = 1$
- …
- $F/2^i$ of them to have $\text{Tail}_0(x) = i$
- …

So $F' = 2^K$ is pretty good guess of F

But this is not very accurate!

- Use lots of independent sketches and let them vote

Summary

- We didn't discuss lots of topics in this survey
  - Other distributions like Zipfian
  - Other non-parametric methods for distance/density-based outlier detection
  - Time series data
  - Interface design
- Stats and computer science both have a lot to offer to the problem of data quality
  - Challenge is to learn stuff outside our comfort zone!
- On Thursday, we will look at one specific case of quantitative data "cleaning" in a fairly complex application setting