L20 -- Outliers
[Jeff Phillips - Utah - Data Mining]

What is an outliers?
  ???

Build "model" of data. If data point is "way outside" model, it is an outlier.

Gaussian data:
  if data point is x standard deviations from mean.
  x=1 --- 1 out of 3 points is outlier
  x=2 --- 1 out of 20 points is outlier
  x=3 --- 1 out of 300 points is outlier
  x=4 --- 1 out of 16000 points is outlier

but if you have enough data, it will happen! So it is real data!

But should not influence building of model.
  -- but if you built model to find outlier, then model is wrong...

SOLUTION: remove outliers, rebuild model, and repeat...

does this converge?
  - what if we always take out 10 furthest points
  - don't take them out, but don't compute centers with them.

 + k-means clustering without t furthest points

---

density based:
 - regular points have dense neighborhoods
 - outlier points have non-dense neighborhoods

+ use distance to closest point (not ROBUST)
      distance to kth closest point   (what k?)
+ count points within fixed radius   (what radius?)

Some clusters have different distributional properties.
Model needs to be more complex to accurately detect outliers.

reverse nearest neighbors:
 - for each p, find kth nearest neighbor q.

```
      find kth nearest neighbor r to q.
      if ||p-q|| ~ ||q-r|| ok.  (otherwise, p outlier)

-------------

far points un-reliable.  So down-weight them in model
     -->  don't care about outliers
k-kernel cluster
  = each cluster center of P_c maximizes
    c = arg max_x sum_{p in P_c} K(x,p)

how to find c?
    can view phi(c) = (1/|P_c|) sum_{p in P_c} phi(p)
        in Reproducing Kernel Hilbert Space (RKHS)
    phi^{-1}(phi(c)) not in R^d, (not necessarily), but ok for Lloyd's
approach


Many of the techniques are very expensive (and annoying).
So they are often left undone unless some fishy things happen.


-------------------------------------
-------------------------------------

Heavy Tails.

Zipf Law:  frequency of data is inversely proportional to its rank
  multiset X with x=i in [u]
  f_i = |{x in X | x=i}|/|X|
  Sort f_i so f_i > f_{i+1}
  f_i ~ constant * (1/i)

  "the" 7% of all words  (Brown Corpus)
  "of" 3.5% of all words
  "and" 2.8% of all words
   ...

Very common in "internet-scale" data.
 - Finding largest components may miss 30% of customers
 - Cannot be dismissed as outliers
 - Learn main components (easy part)
 - run specialized analysis on remainder
   + repeat

-------------------------------------
-------------------------------------
```

Uncertain Data.

Often assumed P as input is correct.
But P is sensed somehow - and thus has noise.

model each p in P as being from some distribution mu_p

imprecise: mu_p is fixed region
 - p could be anywhere in region
   often used for rounding error
 + much work on worst case error on f(P)

indecisive: mu_p = {p_1, p_2, ... p_k}
  - one of k positions
    for instance, different probes of a distribution
  + databases geometry. explodes in complexity of not careful

stochastic: mu_p has p fixed, but a probability it exists.
  - often points always exist, but edges between them might not.