Introduction to Data Science

DATA PREP FOR MODELING: SAMPLING & LABELING

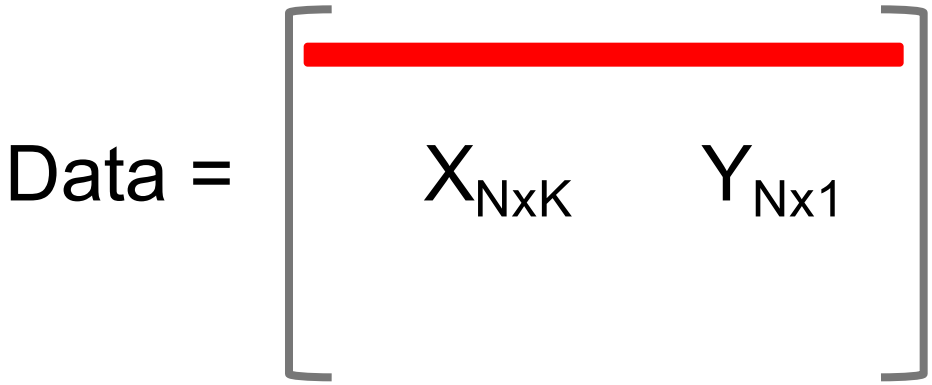
BRIAN D'ALESSANDRO

Fine Print: these slides are, and always will be a work in progress. The material presented herein is original, inspired, or borrowed from others' work. Where possible, attribution and acknowledgement will be made to content's original source. Do not distribute, except for as needed as a pedagogical tool in the subject of Data Science.

DATA MUNGING

Before any analysis we need to carefully craft our matrix of instances (rows), features and a label.

Set of considerations = {Instances, Features, Labels}



DS Practice problems usually lead us straight into modeling. But real problems require creating data. The hardest part is often defining instances, crafting features, and assuring you have the correct label definition.

Copyright: Brian d'Alessandro, all rights reserved

DEFINING AN INSTANCE OF THE DATA

What is the instance space for your problem?
This sounds like it should be trivial, but it does require careful thinking in certain problems!

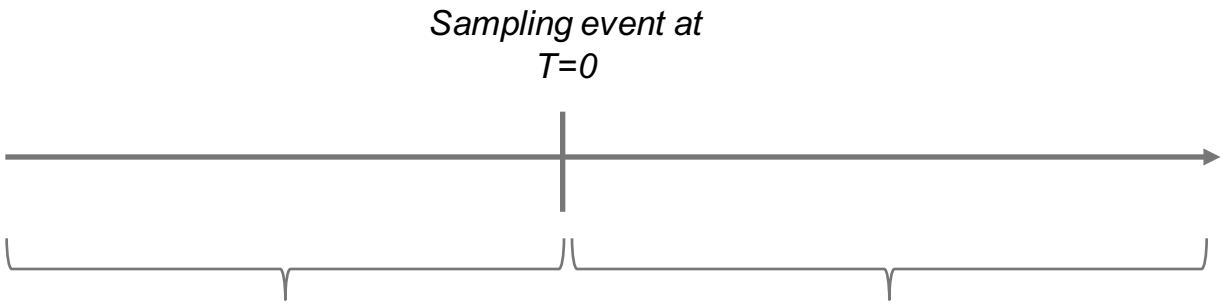
1. **Who or what should be sampled? Are positives and negatives drawn from the same population or process? Some times we observe only positives and have to find appropriate negatives.**

    - *Examples of same process/population:*
        - Kickstarter projects: funded/not funded
        - Ads served: clicked/not clicked
    - *Examples of not same process/population:*
        - People with a disease at a clinic (pos) + random hospital patients (neg)
        - People visiting your site (pos) + random internet browsers (neg)

2. **Are the instances independent of each other?**
    - Geo-spatial data
    - Time series data
    - Pairwise instances (social networks, search)

Copyright: Brian d'Alessandro, all rights reserved

THE MOST STRAIGHTFORWARD CASE



Your features can come from any data occurring before T=0.

You observe your label after T=0. In some cases the delta could be seconds (advertising), but it could also be years (life insurance)

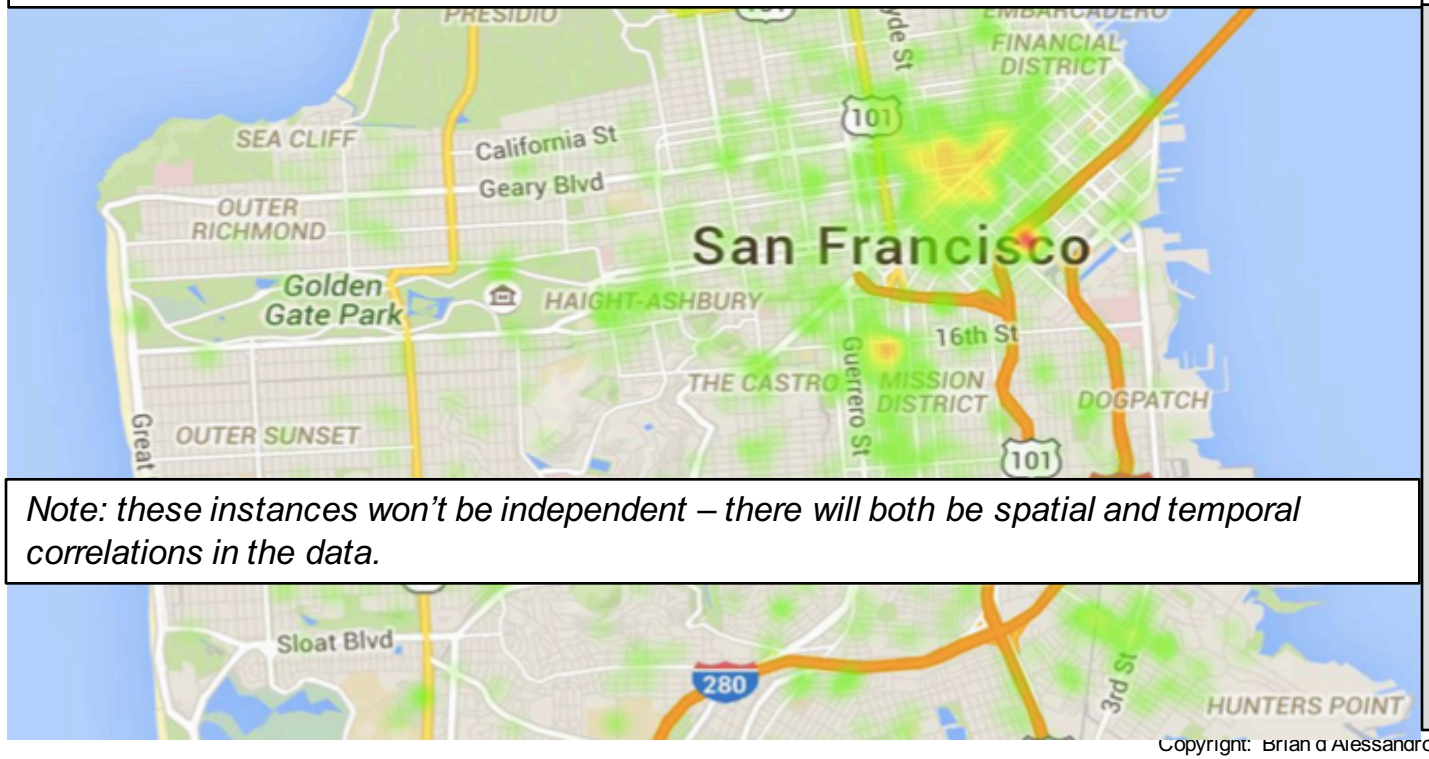
In many cases the sampling event is straightforward (i.e., serving an ad, applying for a loan, making a recommendation).

But even if the event is straightforward, a lot of care should go into defining which set of events matches the needs of the problem.

Copyright: Brian d'Alessandro, all rights reserved

DEFINING AN INSTANCE OF THE DATA

Example 1: SF Crime data used to predict when (and what type of) crimes will happen. There is no automatically well defined instance. We have crime data on a map. We need to discretize across time and space to make geo-time instances.



Note: these instances won't be independent – there will both be spatial and temporal correlations in the data.

As the analyst you have to decide what level of granularity to choose between time and spatial dimensions.

Too fine grained and most of your observations will have no data, and you'll create computational burdens.

Too course and your predictions won't be useful.

The right answer depends really on how you might use a model

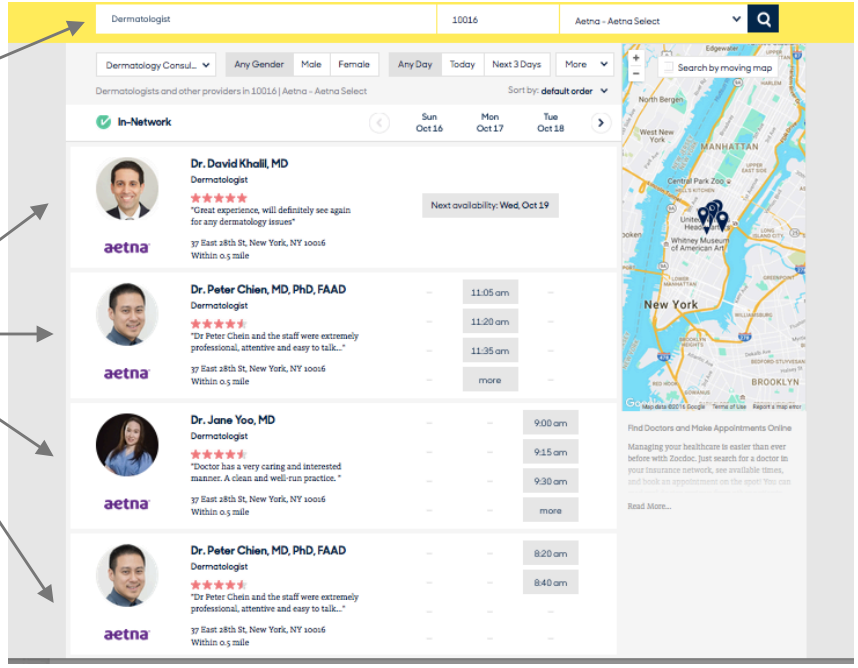
Copyright: Brian d Alessandro, all rights reserved

DEFINING AN INSTANCE OF THE DATA

Example 2: Search. Is a single search the instance? No, we need to define each search result – search query as an instance.

The user query defines a discrete search.

Each search result is an independent exposure, on which we observe a label of click or not.



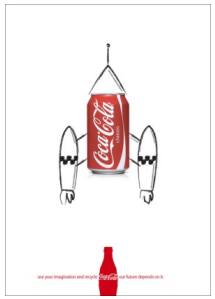
In this case a single search yields multiple results. In ranking problems for recommenders, each result yields an instance in a training data set.

Copyright: Brian d'Alessandro, all rights reserved

CHOOSE THE TARGET VARIABLE

This should be directly determined by the problem. I.e., what are you trying to predict?

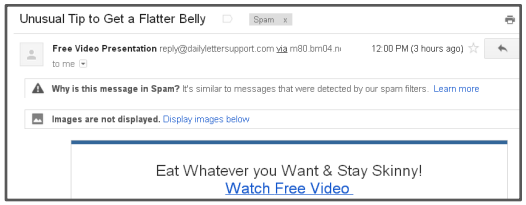
Will someone click on an ad?:
C=[No,Yes]



Is this pill good for headaches?:
C=[No,Yes]

What number is this?:
C=[0,1,2,3,4,5,6,7,8,9]

Is this e-mail spam?: *C=[No,Yes]*



What is this news article about?:
C=[Politics, Sports, Finance …]

Copyright: Brian d'Alessandro, all rights reserved

SOMETIMES YOU HAVE TO BE CREATIVE TO DEFINE A LABEL

Target variables aren't always obviously specified, or sometimes we transform the logical target variable to create a simpler problem.

- **What do we want to model?**
  - Search & Recommendation: Clicks on a recommendation vs purchases?
  - Newsfeeds: likes, comments, hovers?

- **Do we want to convert this to a binary problem?**
  - Online reviews: Y in {1, 2, 3, 4, 5} => Y' = I(Y >= 4)
  - Time till event: Y' = I(T < k)

- **Proxy modeling (we don't observe someone buying a car online, but we observe them visiting the car dealer's web site)**
  - Y = I(Buys Car = True) => Y' = I(Visits Web Site = True)
  - Predict "good" employee. What is definition of "good?"
  - Assumption: Y and Y' are highly correlated, but Y' more abundant

Remember: Even if the outcome is easily observable, the target variable definition may still need creative thinking. Some things to consider:

1. What makes the most sense to the business problem?
2. What makes the problem more analytically tractable?
3. What has the least chance of introducing negative biases?

Copyright: Brian d'Alessandro, all rights reserved

SAMPLING

- Data is often big, and whether or not you need all of it is a case by case decision. Despite this, there are trends to guide you.

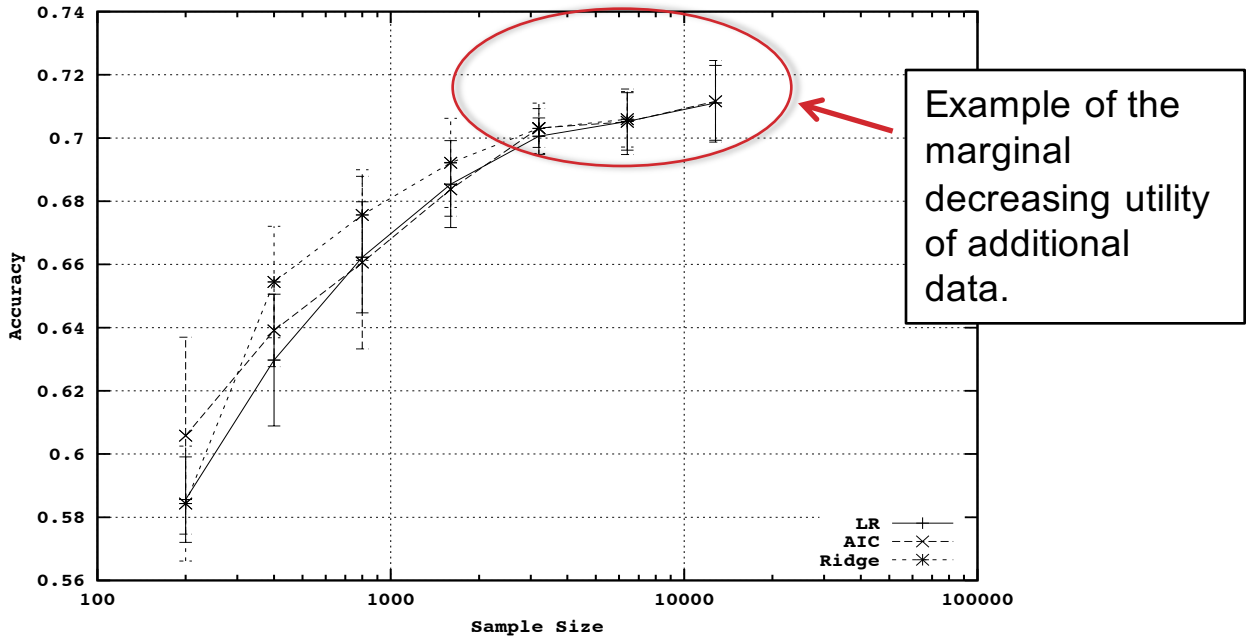
- **Two common sampling strategies used in practice:**
  - 'down-sampling.'
    - Take 100% of the minority class,
    - Take K% of the dominant class
  - 'up – sampling'
    - Take 100% of the dominant class
    - Sample with replacement the minority class until equal

- **Two main reasons to sample:**
  - Reduce computational burden of training (down-sample)
  - Rebalance classes (up or down-sample)

Copyright: Brian d'Alessandro, all rights reserved

SHOULD YOU DOWN SAMPLE?

This is largely an empirical question. Rule of thumb – less complex algorithms & models with information rich features require less data. Learning curves are a good way to visualize and measure the sample size – performance tradeoff



Learning curves: 1) choose a range of sample sizes to test, 2) for each size sample w/ replacement from full training data K times, 3) build a model on each iteration and evaluate on holdout, 4) avg holdout results and plot by sample size.

This is a good method for measuring empirically the effect of down-sampling and/or overall sample size.

Source: Tree Induction vs. Logistic Regression, a Learning Curve Analysis
http://pages.stern.nyu.edu/~fprovost/Papers/logtree.pdf

Copyright: Brian d'Alessandro, all rights reserved

IF YOU DOWN SAMPLE.

1. Use Learning Curve analysis to justify down-sampling rate

2. Be aware of the effect on probability estimates.

- P(Y) and P(Y|X) changes when you down sample based on Y
- Weighting the down-sampled class by 1/k% can correct for this
- Can adjust intercepts if applicable:

$$\hat{\beta}_0 - \ln\left[\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]$$

where τ is the base rate of the population and ŷ is the sample base rate.*

3. Be aware of the effect of base rate on evaluation metrics

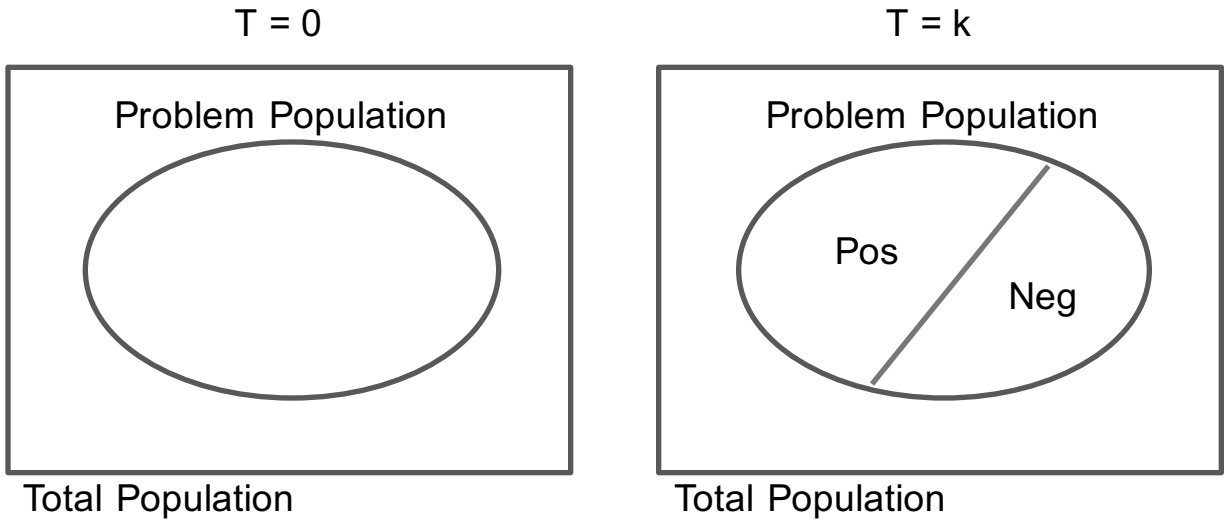
- AUC is invariant to base rate
- Accuracy, precision and lift depend on the base rate

Source: Logistic Regression in Rare Events Data
http://dash.harvard.edu/bitstream/handle/1/4125045/relogit%20rare%20events.pdf?sequence=2

Copyright: Brian d'Alessandro, all rights reserved

COUPLING LABELING W/ SAMPLING

Example labeling processes:
In many problems we can define a population of interest and then collect the labels through some intervention process or observation period. I.e., ad serving, credit risk, fraud,

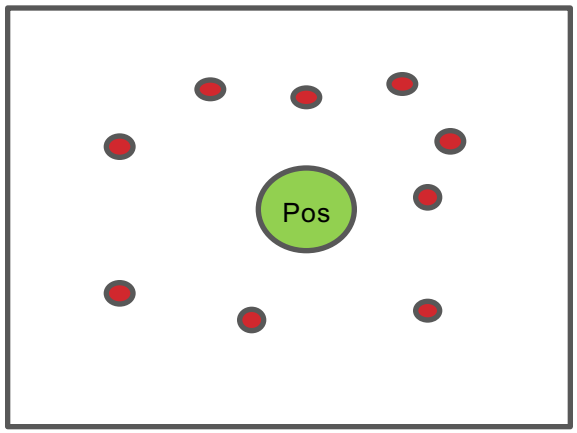


Copyright: Brian d'Alessandro, all rights reserved

COUPLING LABELING W/ SAMPLING

Example labeling processes:
In some problems we sample two populations, labeling each either pos/neg (called case-control study design in medical studies). Retrospective as opposed to prospective.

T = 0



Total Population

Problem: Look-alike modeling – create an audience that is similar in activity and composition to a target audience. (FB, Display Advertising)

Pos: All people that have purchased a particular product or have been to a particular site.

Neg: Randomly selected people who are not in the Pos population.

Problem: Tumor detection in fMRIs – use fMRI images to detect the presence of a brain tumor

Pos: All positively diagnosed patients with said tumors

Neg: fMRIs for Randomly selected people who are not in the Pos population.

See: http://archive.nyu.edu/bitstream/2451/31708/2/Provost%201_2013.pdf

Copyright: Brian d'Alessandro, all rights reserved