Applied Machine Learning!!!

W207 Section 9
Rasika Bhalerao
rasikabh@berkeley.edu

Aug 23: Welcome!
Nov 8 and 22: No classes

Schedule

Supervised learning methods

	Sync	Topic
2	Aug 30	Linear Regression / Gradient Descent
3	Sep 6	Feature Engineering Bonus: Naive Bayes
4	Sep 13	Logistic Regression
5	Sep 20	Multiclass classification / Eval Metrics Bonus: Reinforcement learning
6	Sep 27	Neural Networks
7	Oct 4	KNN, Decision Trees, Ensembles

Unsupervised learning methods

	Sync	Topic
8	Oct 11	KMeans and PCA
9	Oct 18	Text Embeddings Bonus: Language models
10	Oct 25	CNNs Bonus: GANs
11	Nov 1	Real data, Baselines, LDA
12	Nov 15	Fairness / Ethics
13	Nov 29	Fancy Neural Networks
14	Dec 6	Final Presentations

Assignment Schedule

Due Date	Assignment
Aug 28	HW1
Sep 4	HW2
Sep 11	HW3
Sep 18	HW4
Sep 25	HW5
Oct 2	HW6
Oct 16	Group project baseline
Oct 23	HW8
Nov 6	HW9
Nov 20	HW10
Dec 4	Final project notebook + presentation

Behavior expectations

- Healthy disagreement is expected
- Be mindful of one another’s schedules
- Be a good listener
- Have fun in a professional manner
- Share related real-world experience
- Ask questions when something is confusing
- Keep it 100 but be respectful
- Be open-minded to new ideas in the real world and when coding
- On time for group meetings

How are final projects going?

Guidelines: https://docs.google.com/document/d/1R7mIHOtYXKU8vEQzw10uofb_iK3sgimw8iZLWSTzdgg/edit?usp=sharing

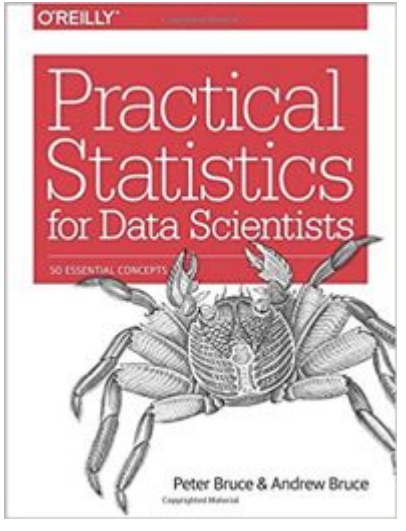
What is a baseline (in ML)?

View options (at the top) → Annotate

Async Practice Quiz Questions (vote!)

The Titanic task involves predicting survival based on individual features like age.	True	False
Imputing missing values as an average is the best way to deal with them.	True	False
If some feature is not highly correlated with the target variable, it will not be useful.	True	False
A baseline should be a simple system that is reasonably difficult to beat, thus helping us calibrate evaluation scores from our models.	True	False
In the functional API, each layer is a function that takes a tensor as input.	True	False
By binning the age feature and using a one-hot representation, our model no longer assumes a linear relationship between age and survival.	True	False
When we design a model architecture with multiple outputs, we must specify the loss for each output.	True	False

Practical, Real Data



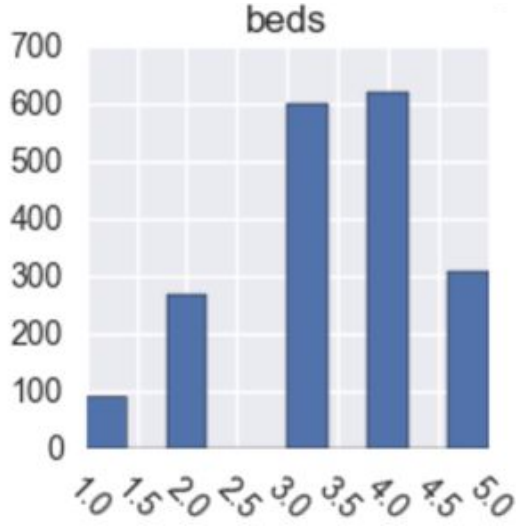
Example Data Set

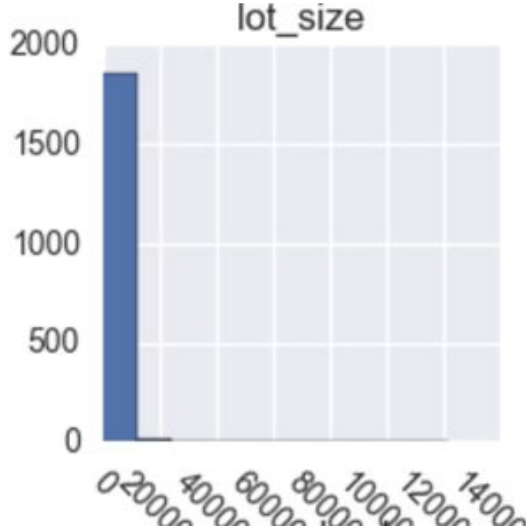
	tx_price	beds	baths	sqft	year_built	lot_size	property_type	exterior_walls	roof	basement
0	295850	1	1	584	2013	0	Apartment / Condo / Townhouse	Wood Siding	NaN	NaN
1	216500	1	1	612	1965	0	Apartment / Condo / Townhouse	Brick	Composition Shingle	1.0
2	279900	1	1	615	1963	0	Apartment / Condo / Townhouse	Wood Siding	NaN	NaN
3	379900	1	1	618	2000	33541	Apartment / Condo / Townhouse	Wood Siding	NaN	NaN
4	340000	1	1	634	1992	0	Apartment / Condo / Townhouse	Brick	NaN	NaN

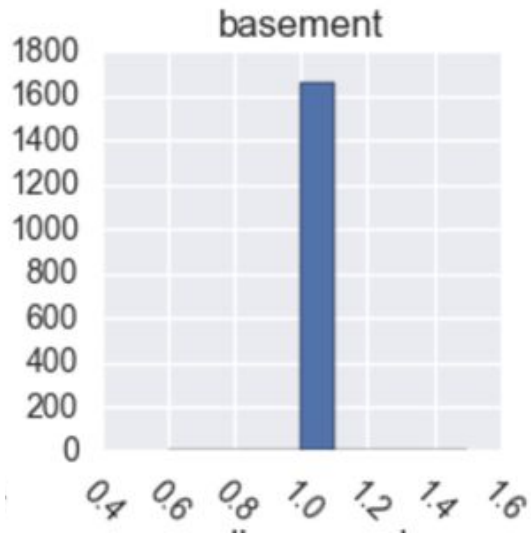
Quickly Examining Data

- Often, a grid of histograms is enough to understand the distributions.
- Here are a few things to look out for:
  - Distributions that are unexpected
  - Potential outliers that don’t make sense
  - Features that should be binary (i.e. wannabe “indicator variables”)
  - Boundaries that don’t make sense
  - Potential measurement errors

Quickly Examining Data

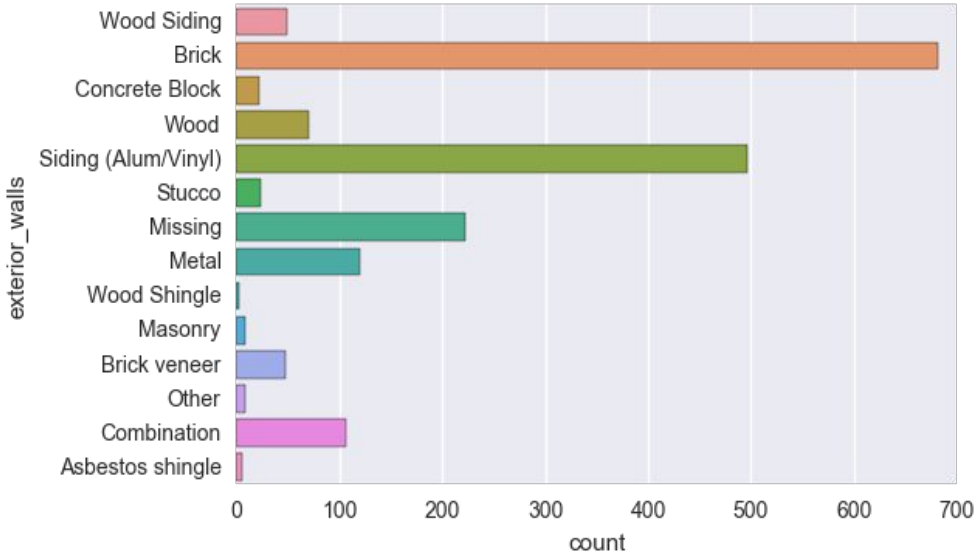






Plot Categorical Distributions

- Categorical features cannot be visualized through histograms
  - Instead, we can use **bar plots**
- A **class** is a unique value for a categorical feature
- **Sparse classes** have very small numbers of observations

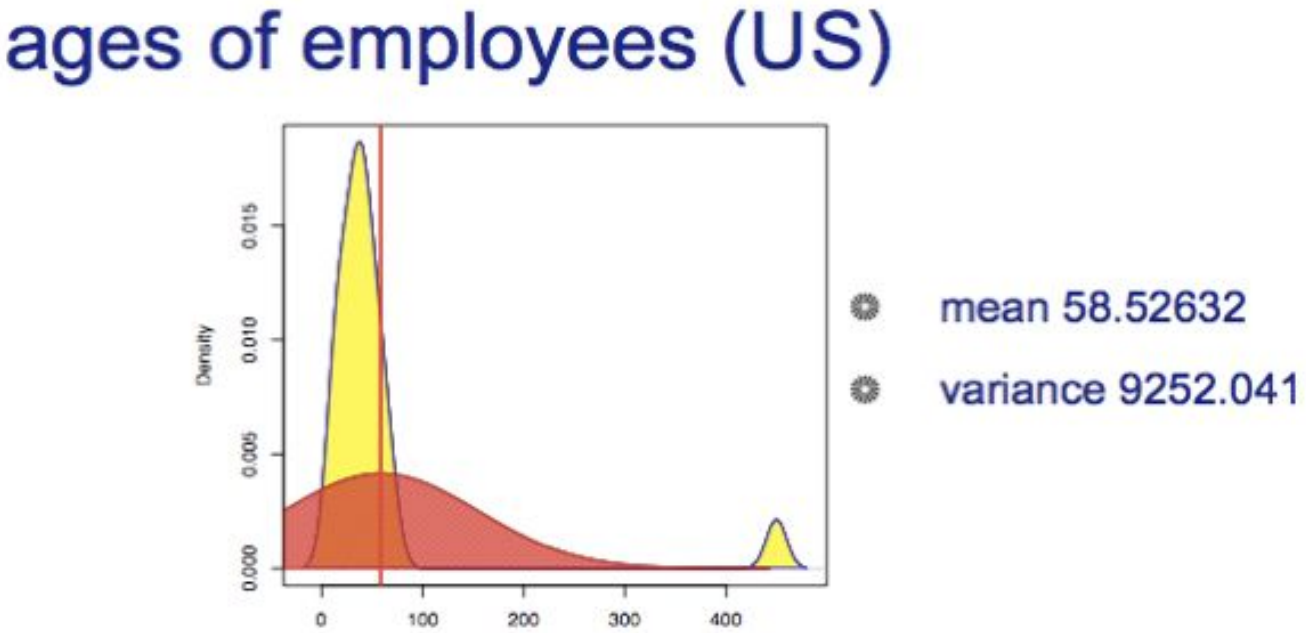


Data Quality Problems

- (Source) data is dirty on its own
- Transformations corrupt the data (complexity of software pipelines)
- Data sets are clean but integration (i.e. combining them) screws them up
- “Rare” errors can become frequent after transformation or integration
- Data sets are clean but suffer “bit rot”
  - Old data loses its value / accuracy over time
- Any combination of the above

Numeric Outliers

12	13	14	21	22	26	33	35	36	37	39	42	45	47	54	57	61	68	450



Adapted from Joe Hellerstein's 2012 CS 194 Guest Lecture

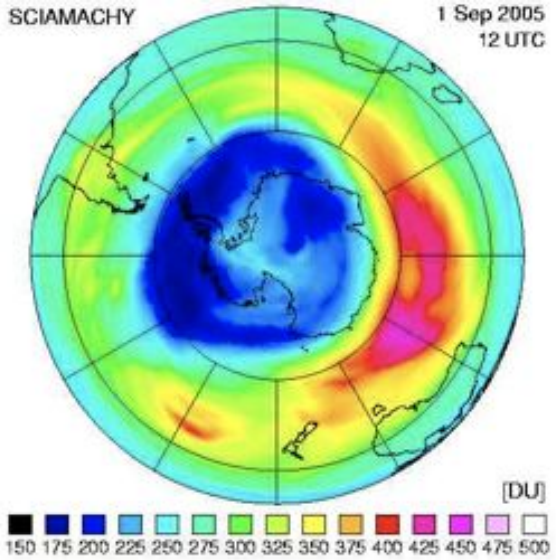
Filter Unwanted Outliers

- If you have a **legitimate** reason to remove an outlier, it will help your model's performance
- Outliers are innocent until proven guilty
  - We should never remove an outlier just because it's a "big number"
- Need a good reason for removing an outlier, such as suspicious measurements that are unlikely to be real data

Data Cleaning Makes Everything Okay?

The appearance of a hole in the earth's ozone layer over Antarctica, first detected in 1976, was so unexpected that scientists didn't pay attention to what their instruments were telling them; they thought their instruments were malfunctioning.

National Center for Atmospheric Research



In fact, the data were rejected as unreasonable by data quality control algorithms (algorithm: below 180 Dobson Units → QC flag → ignore)

Dirty Data

- Everyone cleans their data but nobody really talks about it
- Better data beats fancier algorithms
- Dirty data problems from Stanford Data Integration course:
  - Parsing text into fields (separator issues)
  - Naming conventions (NYC vs. New York)
  - Missing required field
  - Different representations (2 vs. two)
  - Fields too long (get truncated)
  - Primary key violation
  - Redundant records (exact match or other)
  - Formatting issues - especially dates
  - Licensing issues / privacy / keep you from using the data as you would like

Conventional Definition of Data Quality

- Accuracy
  - The data was recorded correctly
- Completeness
  - All relevant data was recorded
- Uniqueness
  - Entities are recorded once
- Timeliness
  - The data is kept up to date
- Consistency
  - The data agrees with itself

Adapted from Ted Johnson’s SIGMOD 2003 Tutorial

Problems with that definition

- Unmeasurable
  - Accuracy and completeness are extremely difficult, perhaps impossible to measure
- Context independent
  - No accounting for what is important. E.g. if you are computing aggregates, you can tolerate a lot of inaccuracy
- Incomplete
  - What about interpretability, accessibility, metadata, analysis, etc.
- Vague
  - The conventional definitions provide no guidance towards practical improvements of the data

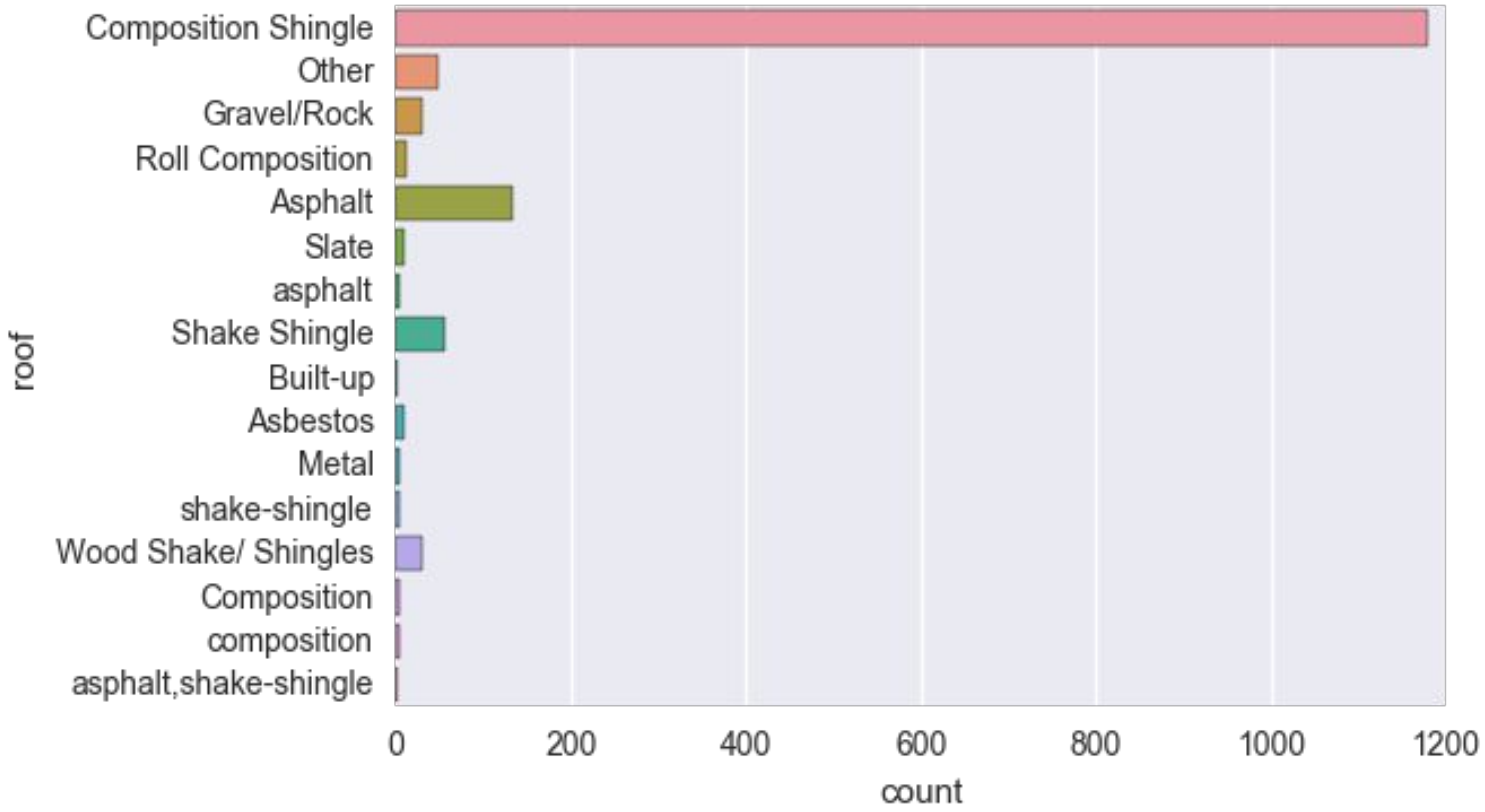
Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Gathering - Solutions

- Preemptive
  - Process architecture (build in integrity checks)
  - Process management (reward accurate data entry, data sharing, data stewards)
- Retrospective
  - Cleaning focus (duplicate removal, merge / purge, name and address matching, field value standardization)
  - Diagnostic focus (automated detection of glitches)

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Everything’s fine



Handle Missing Data

- Do not simply ignore missing values in your dataset
- A few not great solutions:
  - Dropping observations that have missing values
    - The fact that the value was missing may be informative in itself
    - Need to make predictions on new data even if features are missing
  - Imputing the missing values based on other observations
    - Reinforcing patterns already provided by other features

Handle Missing Data

- The best way to handle missing data for **categorical** features is to label them as “missing”
  - You’re essentially adding a new class for the feature
  - This tells the algorithm that the value was missing
  - This also gets around the technical requirement for no missing values
- For missing **numeric** data, we should flag and fill the values
  - Flag the observation with an indicator variable of missingness
  - Then, fill the original missing value with 0 just to meet the technical requirement of no missing values
  - This allows the algorithm to estimate the optimal constant for missingness, instead of just filling it in with the mean

Summary of Data Cleaning

1. Start with computing basic statistics of data
2. Look for errors in the data
3. Clean data to correct errors

Notebook!

To access later:
https://github.com/MIDS-W207/rasikabh/blob/main/live_sessions/Week_11.ipynb