Art of Feature Engineering For Data Science

Nabeel Sarwar

Machine Learning Engineer at Comcast

Spark Summit Europe - October 26th, 2017

What is Comcast NBCUniversal?

- Much like Ireland's Sky
- Provide Broadband Services in the United States
  - Internet
  - Video
  - Media
  - Mobile
- ~25 million customers on combination of broadband services
- Use Case: machine learning to optimize customer experience
  - While complying with all applicable privacy policies and laws

COMCAST

COMCAST NBCUNIVERSAL

Uniquely positioned at the intersection of media and technology

COMCAST

NBCUniversal

Comcast Cable

xfinity

XFINITY TV
XFINITY Internet
XFINITY Voice
XFINITY Home

WATCHABLE

COMCAST BUSINESS

COMCAST SPOTLIGHT

Comcast Spectacor

COMCAST SPECTACOR

SPECTRA BY COMCAST SPECTACOR

xfinity Live! PHILADELPHIA

Other

COMCAST VENTURES

Cable Networks

USA
E!
MSNBC
CNBC
chiller
oxygen
UNIVERSAL HD
NBCSN
The Weather Channel weather.com *
sprout
GOLF
Esquire NETWORK
Bravo
cloo
NBC UNIVERSO
Syfy
Comcast SPORTSNET

Digital & Other *

Broadcast

NBC

NBC Owned Television Stations

TELEMUNDO

NBC Sports

NBC NEWS

Film

UNIVERSAL A COMCAST COMPANY

FOCUS FEATURES

UNIVERSAL UNIVERSAL STUDIOS HOME ENTERTAINMENT

Parks

Universal Studios Hollywood
Universal Orlando Resort
Universal Studios Japan
Universal Studios Singapore
CityWalk
CityWalk

*

NBC ENTERTAINMENT DIGITAL
NBC NEWS DIGITAL
BuzzFeed
FANDANGO
GOLFNOW
MSNBC.COM
CNBC WORLD
VOX MEDIA
hulu
SEESO
CNBC DIGITAL

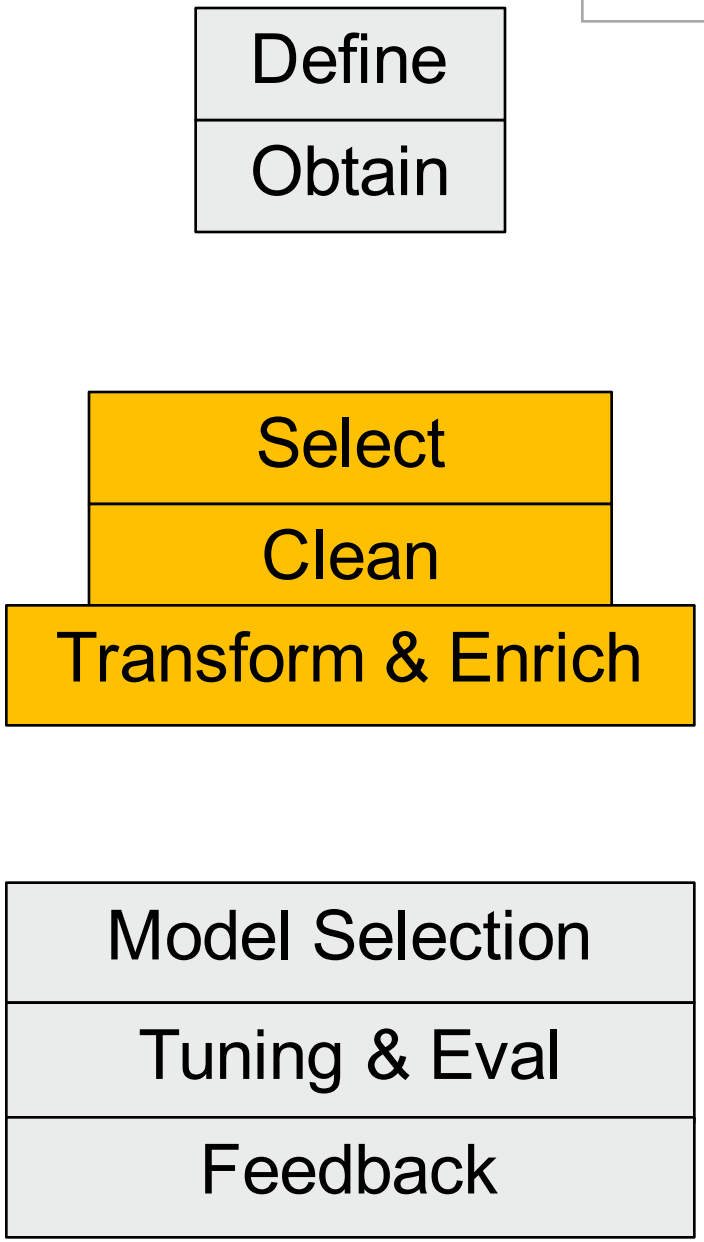
*Minority interest and/or non-controlling interest.

Slide is not comprehensive of all Comcast NBCUniversal assets
Updated: December 22, 2015

COMCAST

Where in the process?

Imagine Arrows Everywhere



COMCAST

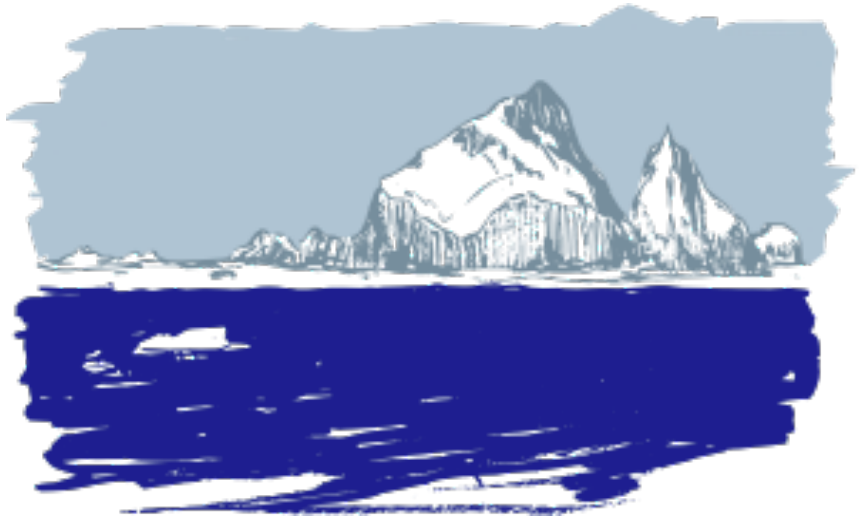
Feature Engineering 101

- Cheat sheet to some of the terminology and best practices
- Why is feature engineering necessary?
- How and how much feature engineering?
- Feature Engineering Process variations
- Select
- Clean
- Normalize and Transform
- Philosophies
  - Confirm a hypothesis
  - Explore trends
  - Middle ground approach

Features

- ML algorithms need pieces of information
- Each individual piece in a sample: **feature**
- Encoded in all sort of different ways
- Differentiate by **use case**
  - Insurance: Model of car
  - Imaging: Color of pixels
  - General Relativity (black hole detection): coordinates in space
  - Online Food Ordering: what you ate last
- Differentiate by **model**
  - Sequencing: Last event states
  - Imaging: Channels of the pixels
  - Reinforcement Learning: Actions from state to state
- Differentiate by **available data**:
  - How do you partition?
  - Feature selection

Example: Titanic Survival Prediction



- Who survives on the titanic?
- Select the features
- Prioritize who survives
  - Age
  - Role on ship
  - Speaking tongue of passenger
- Pick Decision Tree
- Bucket Age
- Keep role as is
- Encode speaking tongue into binary variable whether they speak English or not

How to actually do it

Categorical Variables

- Discrete (but not necessarily disjoint) levels
- Can often seem numeric
- Encoding methods
  - Leave as is (but encode into some number - **StringIndexer**)
  - One Hot Encoding
  - Binary Encoding
- Text: Tokenizing & Word2Vec
- Decision Trees and variants can often do worse with one hot encoding

Numerical Variables

- Can be continuous or discrete
- Can become categorical through binning and bucketting
- To scale or not variables
  - Some variables might be weighted higher than others (KNN?)
  - Per mini-batch in neural network training
  - Normalize or max-min scaling
- Always determine relevant stats:
  - Mean
  - Variance
  - Kurtosis
  - Mode
  - Median
- Specifically for words: term frequency-inverse document frequency (**tf-idf**)

Cleaning Data

- Correcting for input mistakes or known mistakes in the data (normalization)
  - If predicting text: Knowing the last few sequence was **Iceland** instead of **ice land**
- Transforming missing data and outliers
  - Missing Data -> Imputation
    - Mean
    - Mode
    - Matrix Factorization (A = LDU)
- Converting data into known formats for rest of pipeline
  - Can be very simple: Mapping categories to different numbers
  - Or very difficult: Cleaning irregularly sampled data

Selecting Features

- Domain Knowledge
- Statistically correlate features
  - Weights
  - Gini
  - Correlation
  - Information Gain
  - Information Criteria (AIC, BIC, Bayes Factors)
- Global Gains vs Local Gains
- Throw all the data and then down sample
- Throw in little data and then increase until acceptable tolerance
- Noisy or dirty data
- Some models prefer no intra feature other interactions
  - Ordinary Least Squares and some exponential families

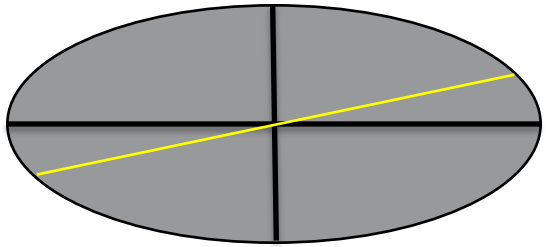
Transformations and Enrichment

- Build Features from Features
  - Add
  - Subtract (the mean)
  - Multiply
  - **Ratios**
  - Polynomials (kernel trick in SVMs)
  - **Rational Differences**
  - Logarithms, Exponentials, Sigmoids
  - Fourier or Laplace Transforms
- Encode categorical variables
- Scaling numerical variables
- Look up tables to bring in additional features
- Transforming dates
  - Year, month, day, hour, minute, second
  - Holidays and Special Events

```
val encoder = new OneHotEncoder()
                .setInputCol(”categories")
                .setOutputCol(”oneHot")
val encodedDF = encoder.transform(categoryDF)
```

COMCAST

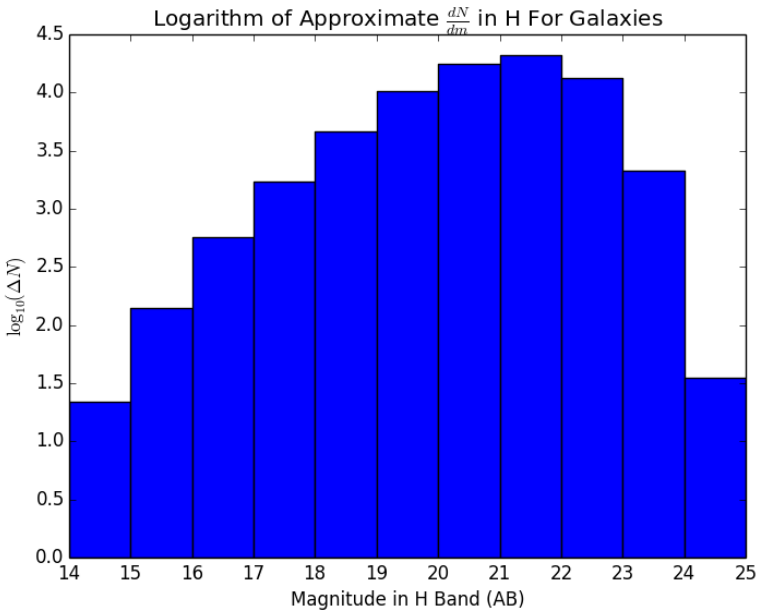
Dimension Reduction

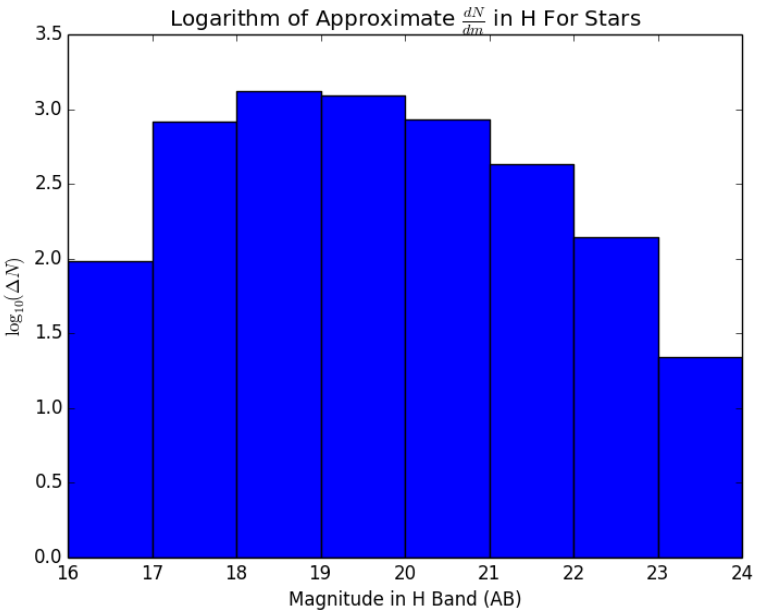


- Curse of Dimensionality
  - Unhealthy prunes
  - Distances look the same
  - Slow training
- Optimization in production
- Reduce number of dimensions by learning the most important “directions” or combination of features
  - Finding the right basis (eigen) vectors
- Principal Component Analysis
  - Singular Value Decomposition on Covariance Matrix
- Factor Analysis
  - Expectation-Maximization to find basis vectors
- Non-negative Matrix Factorization
  - Poisson-gamma factorization
- Just remove features or transform better!

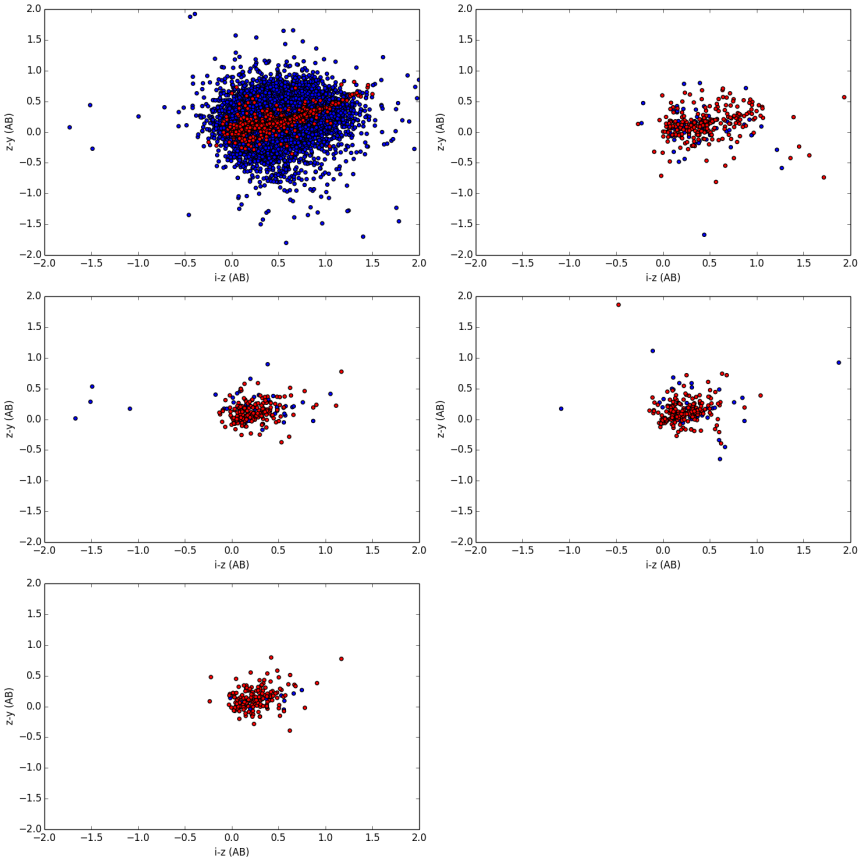
A Harder Use Case: Star-Galaxy Separation

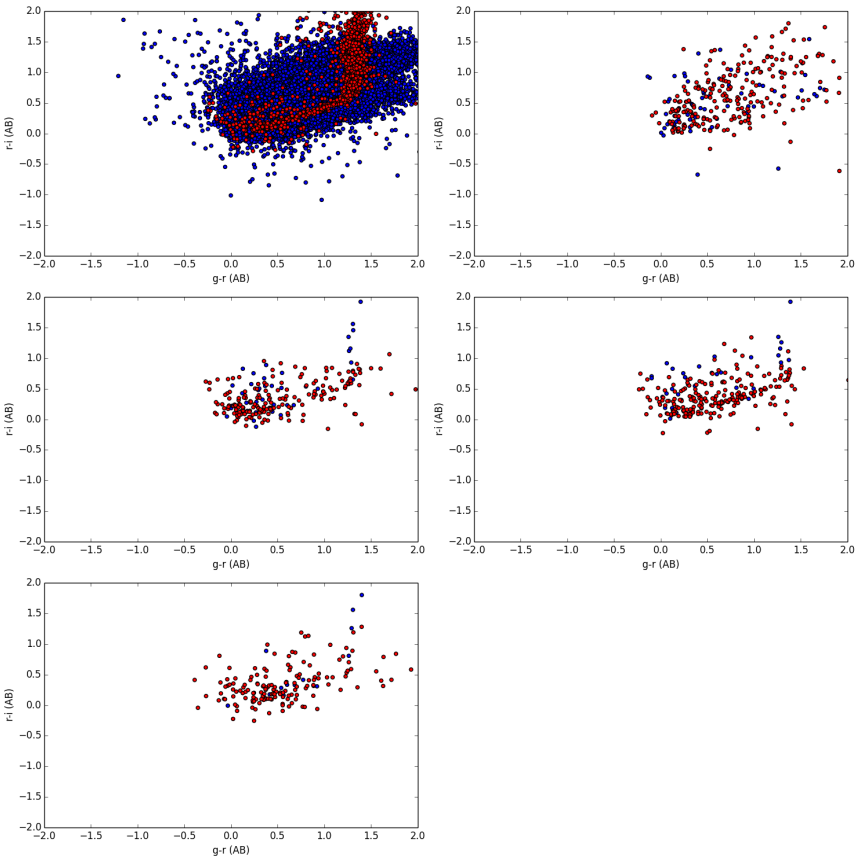
- Stars and galaxies look like little points in space, hard to differentiate on images
- Can read temperatures
- Know that Stars are one-dimensional based on temperature
- Galaxies are composed of various stars
- Bin the temperatures
- Logarithm of temperatures
- Form Features from the combination of differences
- Logarithms on same magnitudes
  - No need to scale


Logarithm of Approximate $\frac{dN}{dm}$ in H For Galaxies
$\log_{10}(\Delta N)$
Magnitude in H Band (AB)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
14
15
16
17
18
19
20
21
22
23
24
25


Logarithm of Approximate $\frac{dN}{dm}$ in H For Stars
$\log_{10}(\Delta N)$
Magnitude in H Band (AB)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
16
17
18
19
20
21
22
23
24

COMCAST





COMCAST

Philosophy

- Explore:
  - Not sure what exactly is there
  - Try out different combinations until you see trends
  - Select all and throw out as needed
- Confirmation:
  - Have some hypothesis on how data is supposed to behave
  - Statistical tests
  - Add Feature if it passes your confirmation test
- Hybrid:
  - Data sets often have a combination of features that fit both philosophies

New Horizons

- Deep Learning: Feature Engineering -> Architecture building
  - Still need good data and good features
- Meta Features
  - Models building features for other models
- Automation!
  - Combinatorial complexity

COMCAST


COMCAST