# Life as an ML Engineer

David Rasch - Infinia ML

February 13, 2018

# Intro

# Things you already know

1. Interchangable Parts

# Things you already know

1. Interchangable Parts
2. Testing

# Things you already know

1. Interchangable Parts
2. Testing
3. Integration

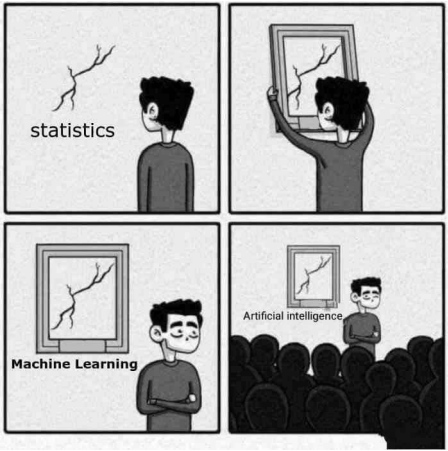
# Objectives

1. Think about ML from an engineering perspective

# Objectives

1. Think about ML from an engineering perspective
2. Learn some of the terminology used to help converse between Data Scientists and Engineers like:

# ai vs statistics



# you're going to need some data

# you need to know what you're trying to do

- look at your data (inputs and outputs)

## you need to know what you're trying to do

- look at your data (inputs and outputs)
- phone a friend

# you need to know what you're trying to do

- look at your data (inputs and outputs)
- phone a friend
  - scikit learn flow chart

# you need to know what you're trying to do

- look at your data (inputs and outputs)
- phone a friend
  - scikit learn flow chart
  - or just use deep learning, it's cool

# you need to know what you're trying to do

- look at your data (inputs and outputs)
- phone a friend
  - scikit learn flow chart
  - or just use deep learning, it's cool
- interpretability

# don't forget to look for prior art

- Look at YOLO, UNet, ResNet51, RetinaNet, BERT, Transformer, and many other hyped algorithms.

# don't forget to look for prior art

- Look at YOLO, UNet, ResNet51, RetinaNet, BERT, Transformer, and many other hyped algorithms.
- Tensorflow has many sets of "pre-trained" weights

# this was a whole section on data prep

- new API

# this was a whole section on data prep

- new API
- new CSV from a customer

# things that matter for ML

- normalizing or "whitening"

## things that matter for ML

- normalizing or "whitening"
- binning

# things that matter for ML

- normalizing or "whitening"
- binning
- missing values

# things that matter for ML

- normalizing or "whitening"
- binning
- missing values
- dimensionality reduction

# things that matter for ML

- normalizing or "whitening"
- binning
- missing values
- dimensionality reduction
- class imbalance

# algorithms

# jargon

- letters

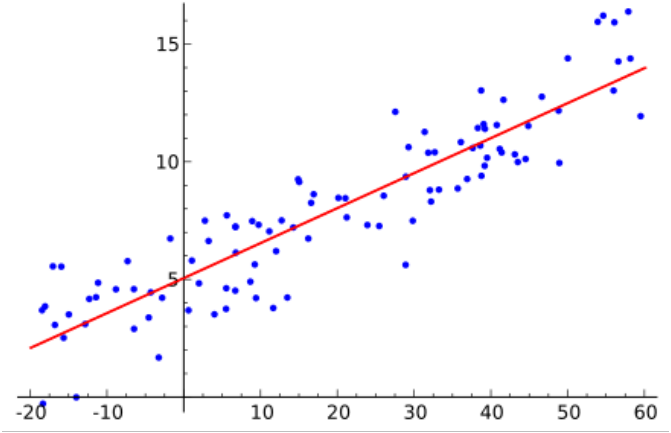
# jargon

- letters
- $Y = mx + b$

# jargon (cont'd)

$$Y = Wx + b$$

# regression



## what if there are multiple variables?

- $y = W_1 x_1 + b$

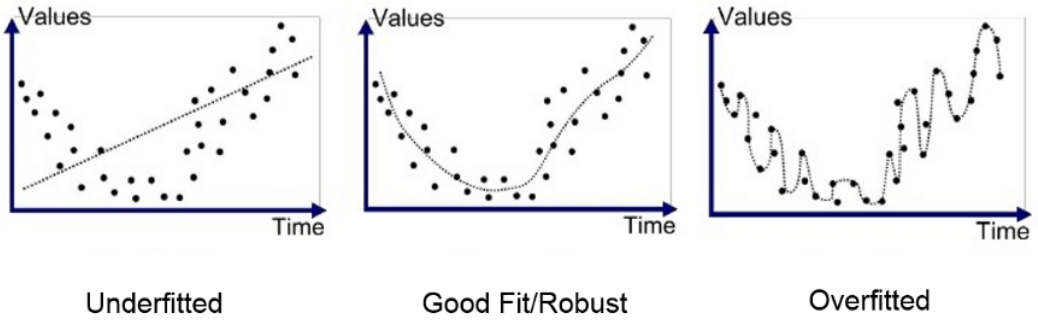
## what if there are multiple variables?

- $y = W_1 x_1 + b$
- $y = W_1 x_1 + W_2 x_2 + \ldots + b$

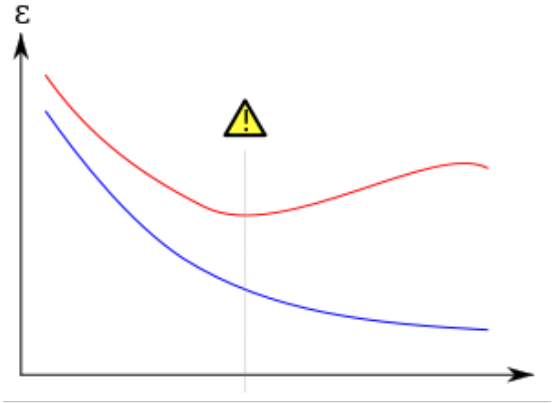
## what if there are multiple variables?

- $y = W_1 x_1 + b$
- $y = W_1 x_1 + W_2 x_2 + \ldots + b$
- $y = Wx + b$

# overfitting



# overfitting



# inference aka "pushing to production"

# scaling (performance, speed)

- easy

# scaling (performance, speed)

- easy
- well defined interfaces

# scaling (performance, speed)

- easy
- well defined interfaces
- shared-nothing

## scaling (performance, speed)

- easy
- well defined interfaces
- shared-nothing
- load balancing

# model health

- what if incoming data is different than training data?

# model health

- what if incoming data is different than training data?
  - e.g., hot dog vs not hot dog, and someone gives it a brautwurst

# model health

- what if incoming data is different than training data?
  - e.g., hot dog vs not hot dog, and someone gives it a brautwurst
  - or a real example, kangaroos on self driving cars

# Operations (checkups)

- get new data! prompt users for wrong responses

# Operations (checkups)

- get new data! prompt users for wrong responses
- online learning: re-train nightly/hourly/steaming w/ new data

# Operations (checkups)

- get new data! prompt users for wrong responses
- online learning: re-train nightly/hourly/steaming w/ new data
- active learning: figure out what labels you need to improve model performance

# tensors and flow graph

# tensors

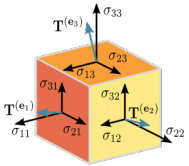
- linear relation between vectors, scalars, or other tensors

# tensors

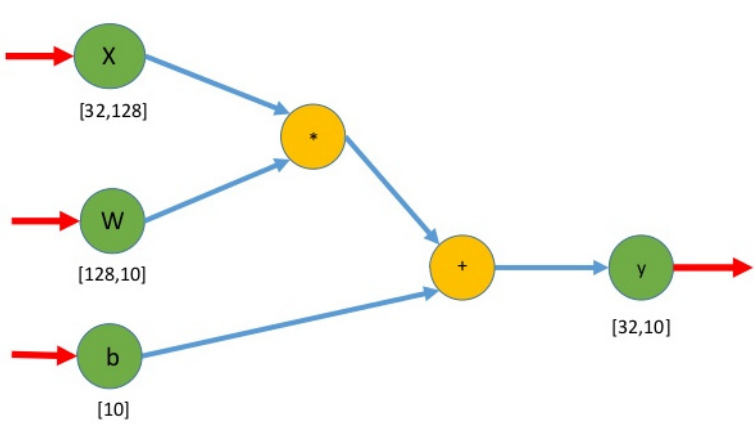
- linear relation between vectors, scalars, or other tensors
- practically: multi-dimensional array

# tensors

- linear relation between vectors, scalars, or other tensors
- practically: multi-dimensional array



# computational flow graph (Directed-acyclic graph)



# questions?

# other resources

# other learning resources

- http://fast.ai
- https://hackernoon.com/choosing-the-right-machine-learning-algorithm-68126944ce1f
- http://ml-cheatsheet.readthedocs.io/en/latest/linear_regression.html

# image credits

- ai vs stats
- regression
- overfitting
- more overfitting
- loss functions
- gradient descent
- tensors
- tensorflow graph