Machine Learning with

Daniel Hinojosa

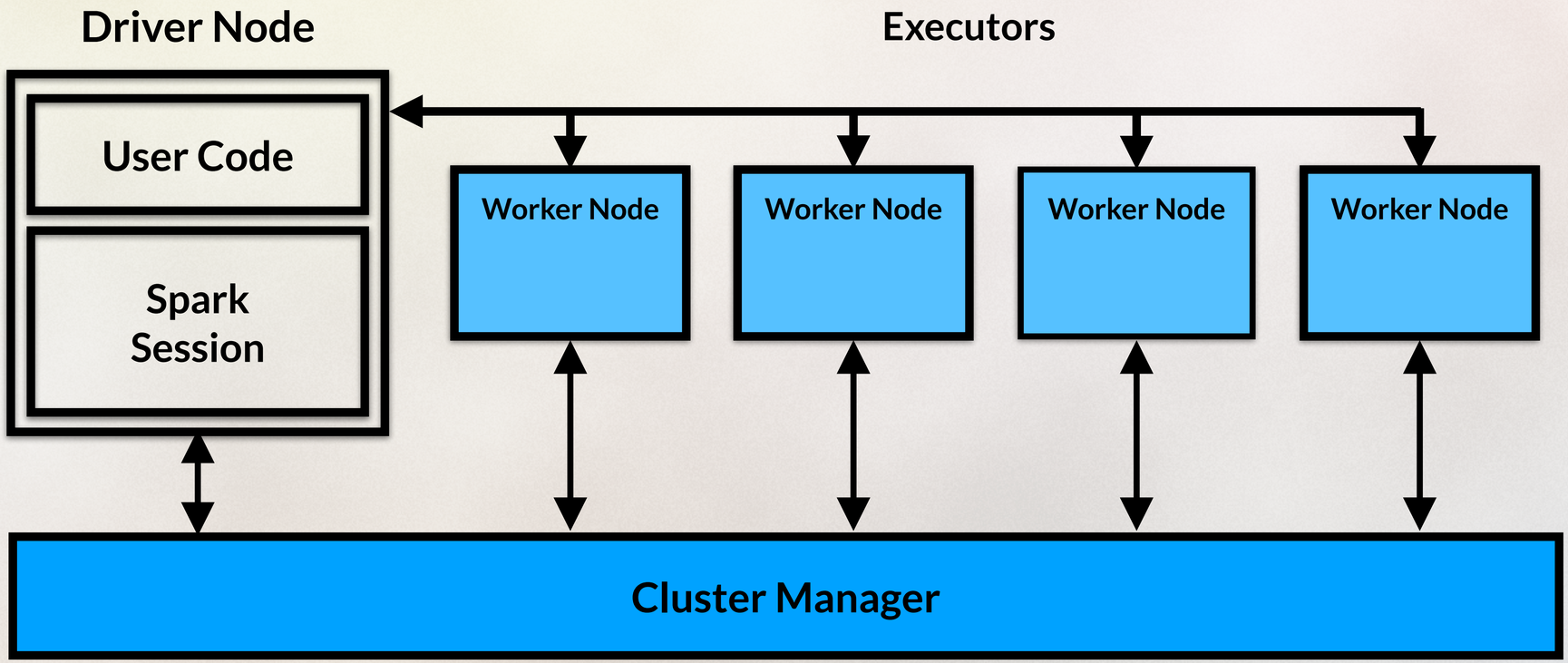
Repository:

https://github.com/dhinojosa/machine-learning-spark

Requirements:

- Sbt - scala.sbt.org
- JDK 8 or higher

About Spark


Driver Node
Executors
User Code
Spark Session
Worker Node
Worker Node
Worker Node
Worker Node
Cluster Manager

MLLib

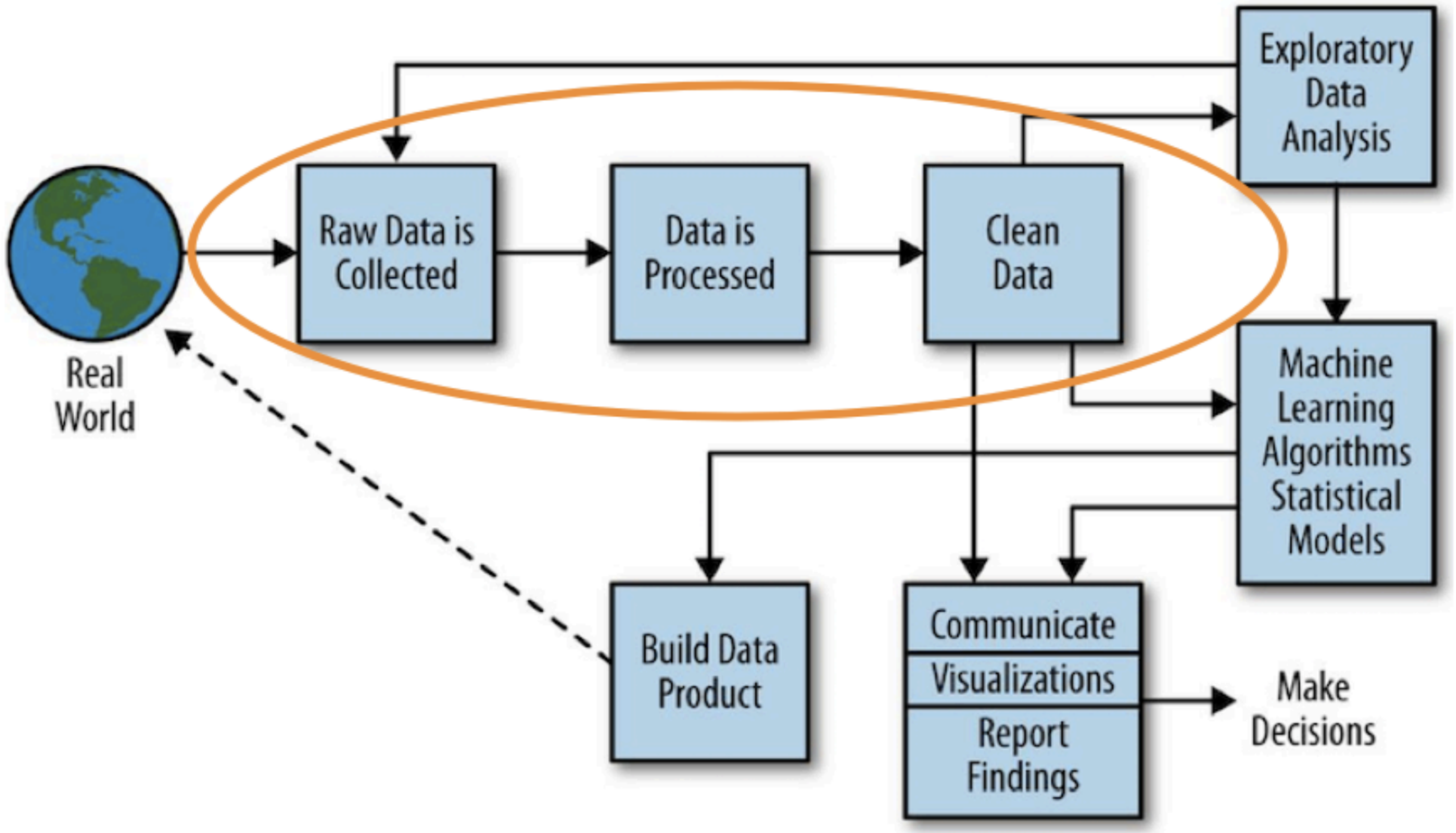
Features:

- Ease of Use
- Machine Learning
- Runs Under Spark
- DataFrame Centric

Performance:

- 100x Faster than MapReduce
- Large Number of Models
- Large Number of Tools

Machine Learning Process


Real
World
Raw Data is
Collected
Data is
Processed
Clean
Data
Exploratory
Data
Analysis
Machine
Learning
Algorithms
Statistical
Models
Build Data
Product
Communicate
Visualizations
Report
Findings
Make
Decisions

Machine Learning Terms

columns = features = dimensions

rows = observations

id	age	sex	trestbps	target
0	20	1	145	1
1	45	1	130	1
2	43	0	130	0
3	33	1	150	0

data

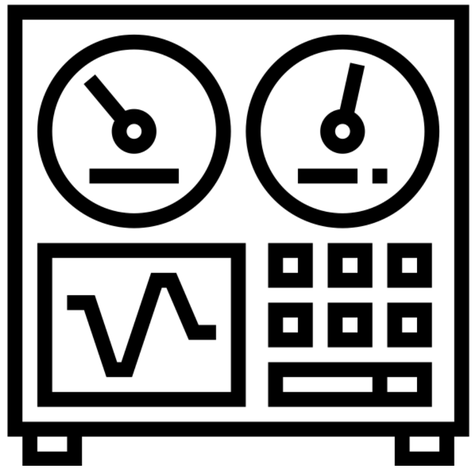
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

training phase



id	age	sex	trestbps	risk
0	20	1	145	1
1	45	1	130	1
2	43	0	130	0
3	33	1	150	0

training

testing

id	age	sex	trestbps	target
0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1
10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

id	age	sex	trestbps	target
0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1
10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

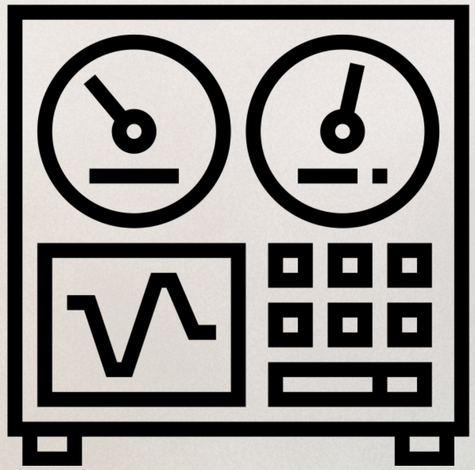
random distribution between training and testing

training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1



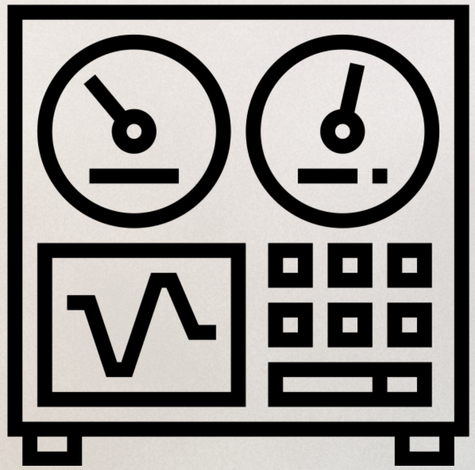
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

model →



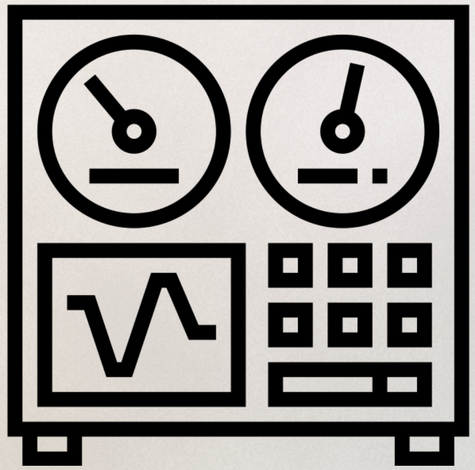
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

training phase

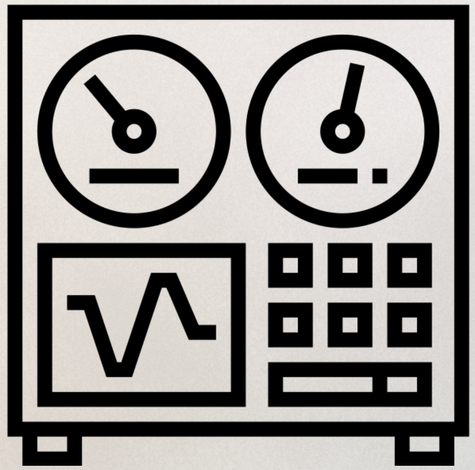


training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1



training

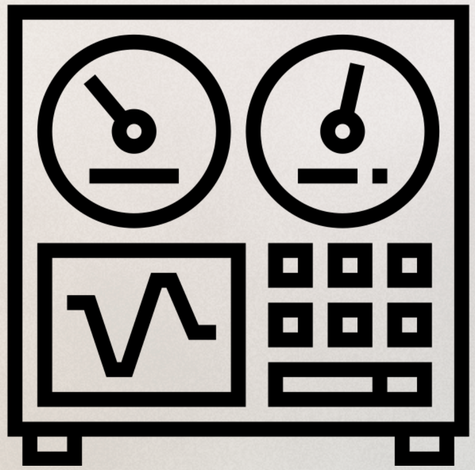
0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239
11	48	0	275
12	49	1	266
13	64	1	211
14	58	0	283

actual

1
0
1
0
1



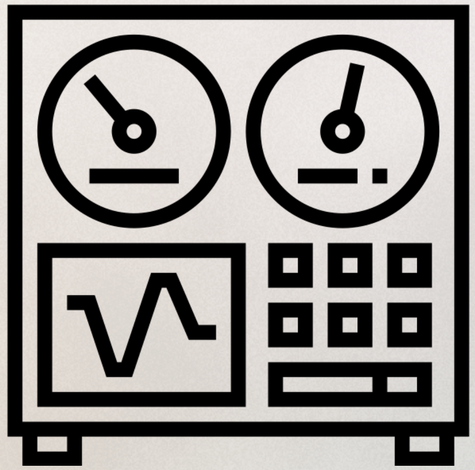
reserve the actual later for verification

training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239
11	48	0	275
12	49	1	266
13	64	1	211
14	58	0	283



testing phase

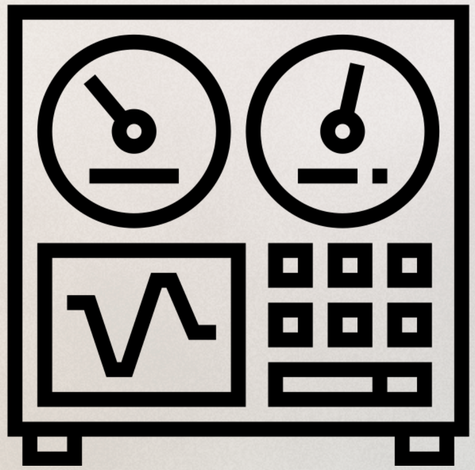
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239
11	48	0	275
12	49	1	266
13	64	1	211
14	58	0	283

!!!!!!!



generated result

1
0
1
1
1

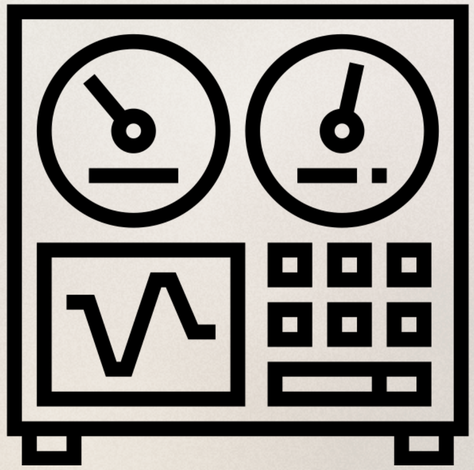
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239
11	48	0	275
12	49	1	266
13	64	1	211
14	58	0	283

!!!!!!!



generated result

1
0
1
1
1

actual

1
0
1
0
1

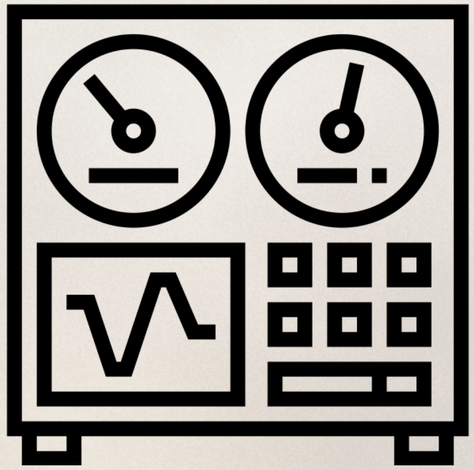
training

0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1

testing

10	54	1	239
11	48	0	275
12	49	1	266
13	64	1	211
14	58	0	283

!!!!!!!



generated result

1
0
1
1
1

actual

1
0
1
0
1

How did we do?

generated result	actual
1	1
0	0
1	1
1	0
1	1

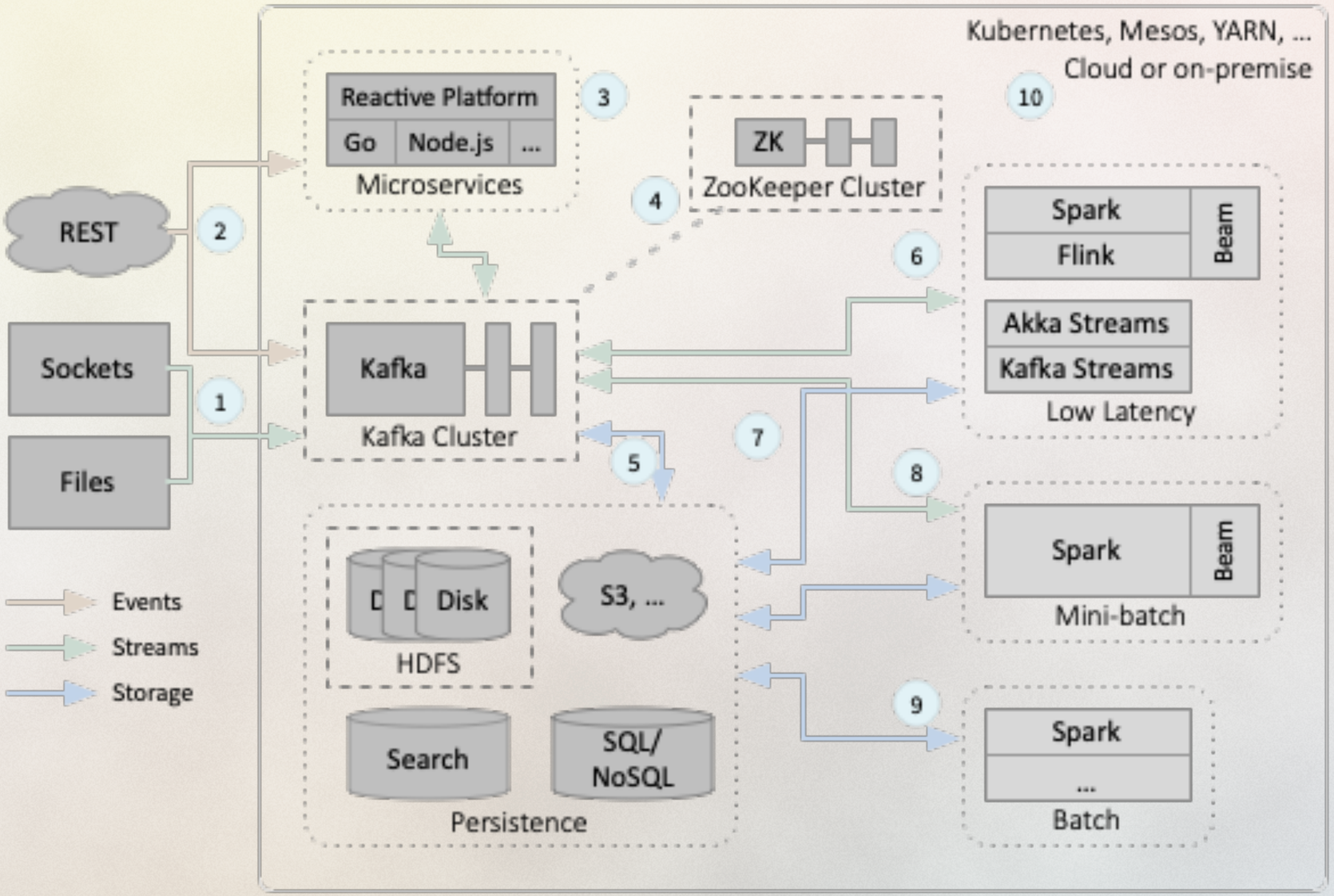
How did we do?

Finding Data

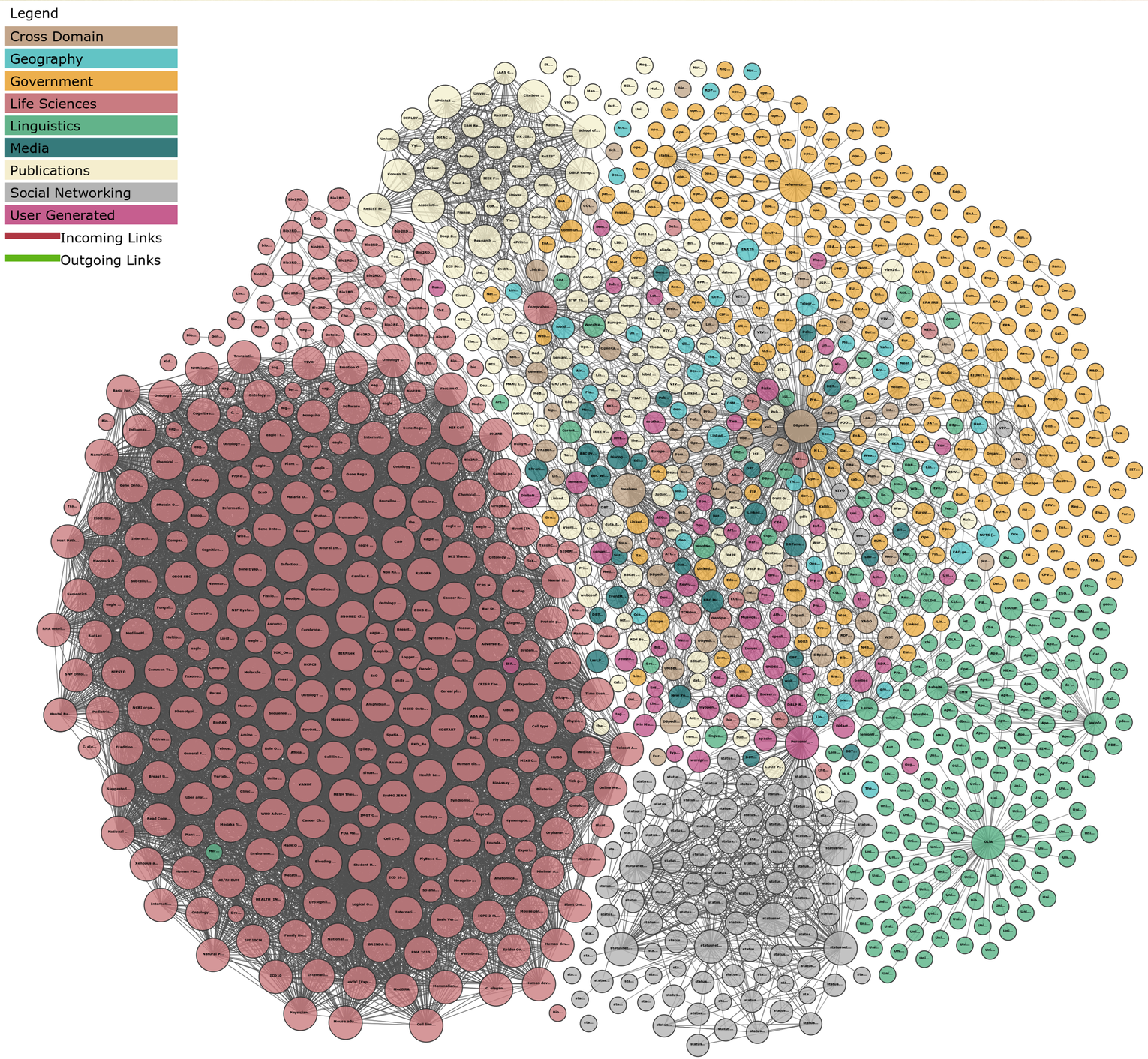
Finding Data Issues

- 60% of time getting to insight
- 37% of time searching for data
- 36% of time preparing data
- 27% of time actual analysis

https://tinyurl.com/ybwdq34u



Credit: Dean Wampler



Linked Data

- Content is interlinked
- RDF
- Queryable using SPARQL

http://lod-cloud.net

kaggle

Wide range of Datasets

Cleaning Data

age	id	sex	trestbps	target
63	0	1	233	1
37	1	1	250	0
41	2	0	204	1
56	3	1	\n	1
57	4	0	354	0
57	5	1	192	0
56	6	0	\n	0
44	7	1	263	1
52	8	1	199	1
57	9	1	168	1
\n	\n	\n	\n	\n
48	11	0	275	0
49	12	1	266	1
64	13	\n	\n	\n
58	14	0	283	1

Get rid of rows where majority is unavailable

age	id	sex	trestbps	target
63	0	1	233	1
37	1	1	250	0
41	2	0	204	1
56	3	1	\n	1
57	4	0	354	0
57	5	1	192	0
56	6	0	\n	0
44	7	1	263	1
52	8	1	199	1
57	9	1	168	1
48	11	0	275	0
49	12	1	266	1
58	14	0	283	1

What if I am missing some values but minimal?

age	id	sex	trestbps	target
63	0	1	233	1
37	1	1	250	0
41	2	0	204	1
56	3	1	\n	1
57	4	0	354	0
57	5	1	192	0
56	6	0	\n	0
44	7	1	263	1
52	8	1	199	1
57	9	1	168	1
48	11	0	275	0
49	12	1	266	1
58	14	0	283	1

How about finding the resting blood pressure of all 56 year olds?

age	id	sex	trestbps	target
63	0	1	233	1
37	1	1	250	0
41	2	0	204	1
56	3	1	200	1
57	4	0	354	0
57	5	1	192	0
56	6	0	200	0
44	7	1	263	1
52	8	1	199	1
57	9	1	168	1
48	11	0	275	0
49	12	1	266	1
58	14	0	283	1

How about finding the resting blood pressure of all 56 year olds?

Feature Engineering

Feature Engineering:

- Applying domain knowledge to features
- Converting Strings to numerical values
- Extracting Date Information (e.g. weekends)

age	sex	trestbps	target
63	male	233	critical
37	male	250	non-critical
41	female	204	critical
56	male	236	critical
57	female	354	non-critical
57	male	192	non-critical
56	female	294	non-critical
44	male	263	critical
52	male	199	critical
57	male	168	critical
54	male	239	critical
48	female	275	non-critical
49	male	266	critical
64	male	211	non-critical
58	female	283	critical

machines don't know what a male/female is or what critical means

Feature Selection

id	age	sex	trestbps	target
0	63	1	233	1
1	37	1	250	0
2	41	0	204	1
3	56	1	236	1
4	57	0	354	0
5	57	1	192	0
6	56	0	294	0
7	44	1	263	1
8	52	1	199	1
9	57	1	168	1
10	54	1	239	1
11	48	0	275	0
12	49	1	266	1
13	64	1	211	0
14	58	0	283	1

Do we need an id?

Feature Selection:

- Determining which features (columns) are needed for learning
- Critical thinking required as to what is important

Notebooks

Zeppelin Notebook Job Search anonymous

HeartNotebook

Head default

FINISHED

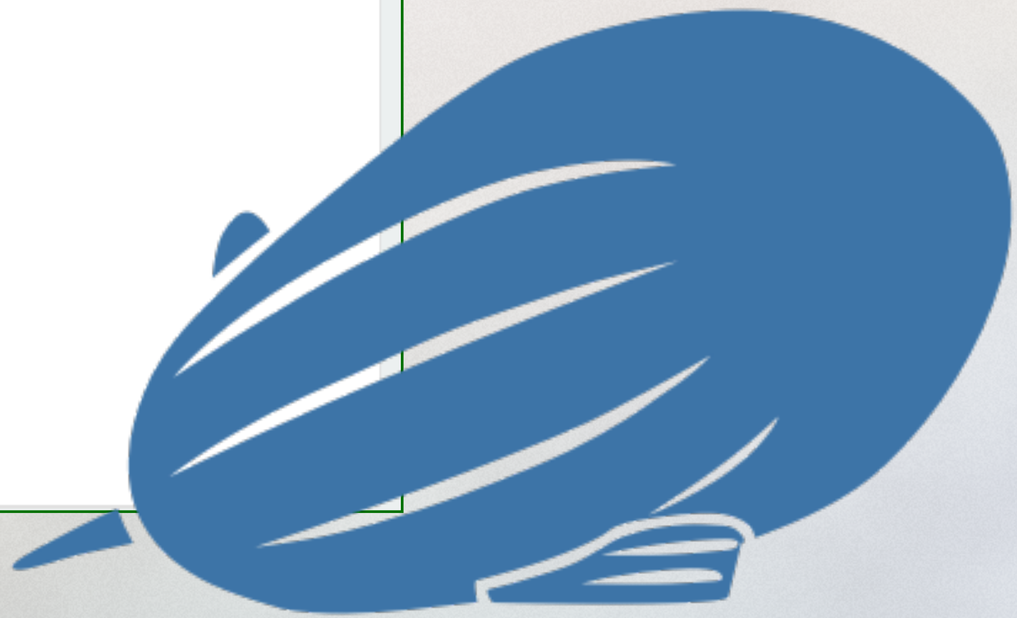
```
%spark
val df = spark.read.format("csv").option("header", "true").option("inferSchema", "true").load("/data/heart.csv")
```

```
df: org.apache.spark.sql.DataFrame = [age: int, sex: int ... 12 more fields]
```

FINISHED

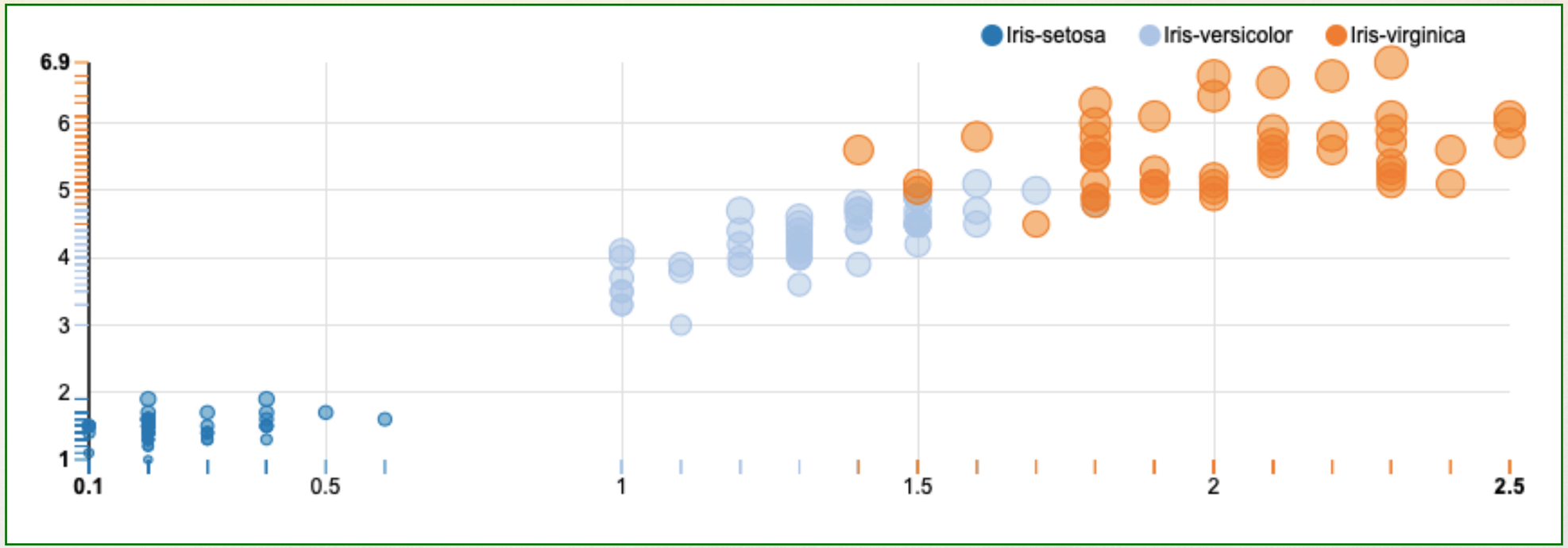
```
%spark
df.show()
```

```
+---+---+---+--------+----+---+-------+-------+-----+-------+-----+---+----+------+
|age|sex| cp|trestbps|chol|fbs|restecg|thalach|exang|oldpeak|slope| ca|thal|target|
+---+---+---+--------+----+---+-------+-------+-----+-------+-----+---+----+------+
| 63|  1|  3|     145| 233|  1|      0|    150|    0|    2.3|    0|  0|   1|     1|
| 37|  1|  2|     130| 250|  0|      1|    187|    0|    3.5|    0|  0|   2|     1|
| 41|  0|  1|     130| 204|  0|      0|    172|    0|    1.4|    2|  0|   2|     1|
| 56|  1|  1|     120| 236|  0|      1|    178|    0|    0.8|    2|  0|   2|     1|
| 57|  0|  0|     120| 354|  0|      1|    163|    1|    0.6|    2|  0|   2|     1|
| 57|  1|  0|     140| 192|  0|      1|    148|    0|    0.4|    1|  0|   1|     1|
| 56|  0|  1|     140| 294|  0|      0|    153|    0|    1.3|    1|  0|   2|     1|
| 44|  1|  1|     120| 263|  0|      1|    173|    0|    0.0|    2|  0|   3|     1|
| 52|  1|  2|     172| 199|  1|      1|    162|    0|    0.5|    2|  0|   3|     1|
| 57|  1|  2|     150| 168|  0|      1|    174|    0|    1.6|    2|  0|   2|     1|
| 54|  1|  0|     140| 239|  0|      1|    160|    0|    1.2|    2|  0|   2|     1|
| 48|  0|  2|     130| 275|  0|      1|    139|    0|    0.2|    2|  0|   2|     1|
| 49|  1|  1|     130| 266|  0|      1|    171|    0|    0.6|    2|  0|   2|     1|
| 64|  1|  3|     110| 211|  0|      0|    144|    1|    1.8|    1|  0|   2|     1|
| 58|  0|  3|     150| 283|  1|      0|    162|    0|    1.0|    2|  0|   2|     1|
```



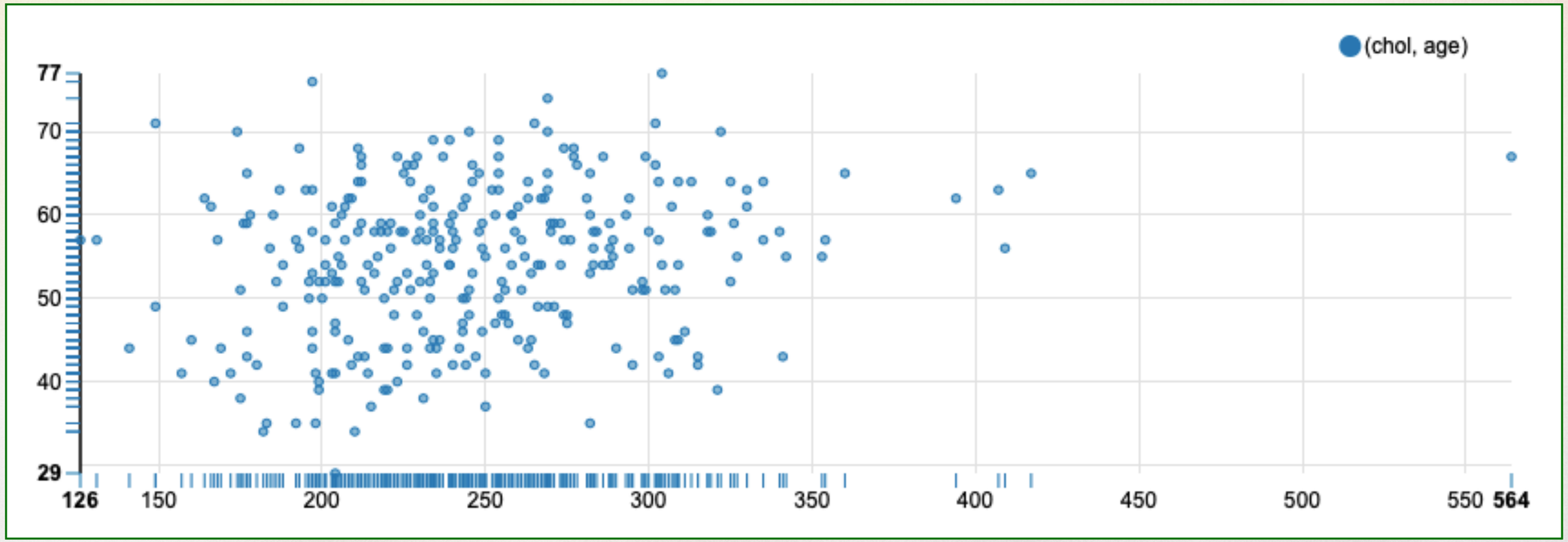
https://zeppelin.apache.org/

Visualizing Data



Visualization:

- See what the data looks like it represents
- This is the iris dataset



Visualization:

- Another Representation
- Cholesterol vs. Age

Types of Learning

Types of Learning:

- **Supervised** - We are providing an answer to guide the model along.
  - Where $f(x) = y$, y is the supervised variable
- **Unsupervised** - We aren't providing the answer

Model Selection

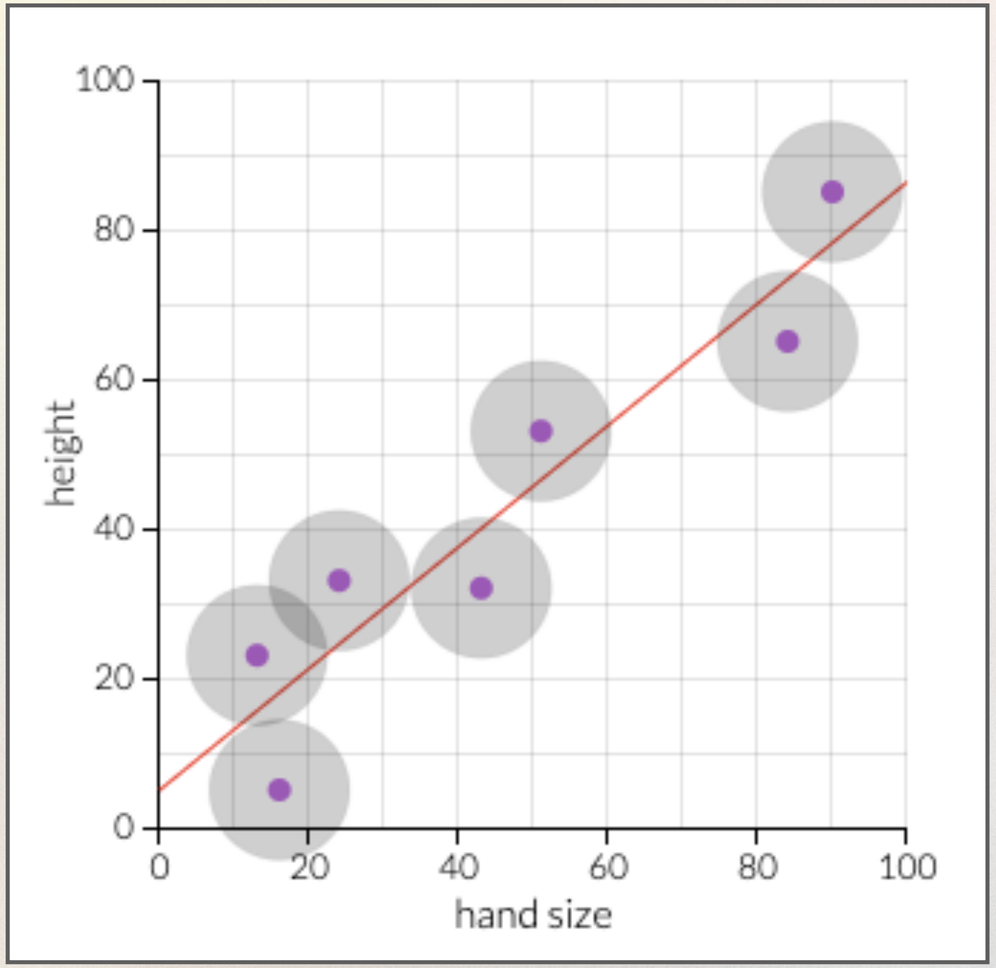
Model Selection:

- What is your goal?
  - **Regression** - What value?
    - If I make $90k what is my credit rating?
  - **Classification** - What category?
    - Sunny, Raining, Overcast
    - Malignant, Benign
    - Static Typed, Dynamic Typed (Ha)

Linear Regression

Linear Regression:

- Works if x has a covariant relationship with y
- Supervised: We know y
- Fits a straight line through points
- Minimizes the error from the line



http://setosa.io/ev/ordinary-least-squares-regression/

Naive Bayes

Naive Bayes:

- Determines probability based on context
- Must know the corpus of values
- Requires very little training data
- Document/Text Classification
- Independence of Features

$P(A|B)$ = Probability of A Given B

$P(A \cap B)$ = Probability of A And B

$P(B)$ = Probability of B

The formula

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

setosa.io/conditional

Algebra Time:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

$$P(A \cap B) = P(B|A) \cdot P(A)$$

$$P(B|A) = \frac{P(B|A) \cdot P(A)}{P(A)}$$

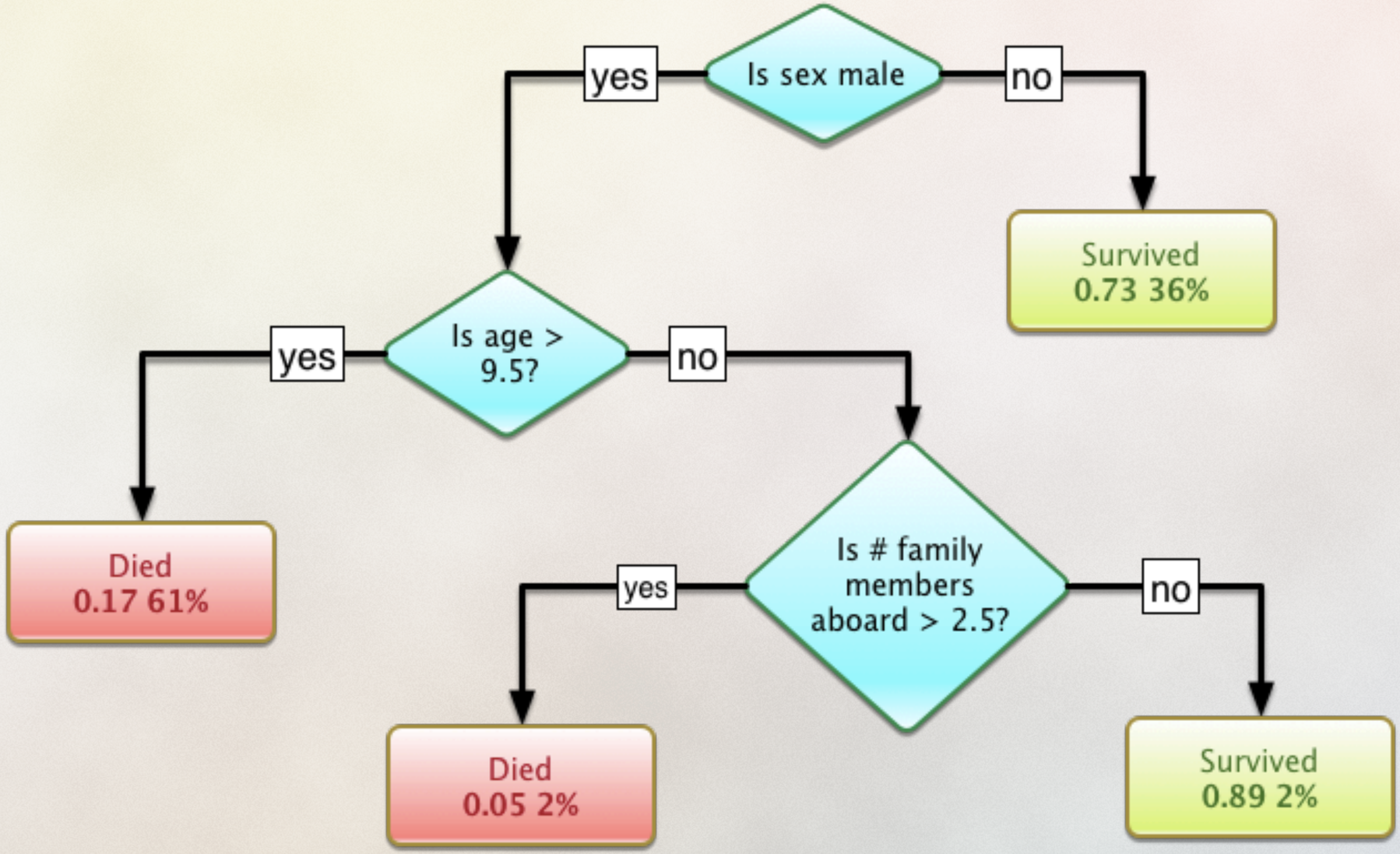
Naive Bayes with Multiple Features:

https://en.wikipedia.org/wiki/Naive_Bayes_classifier

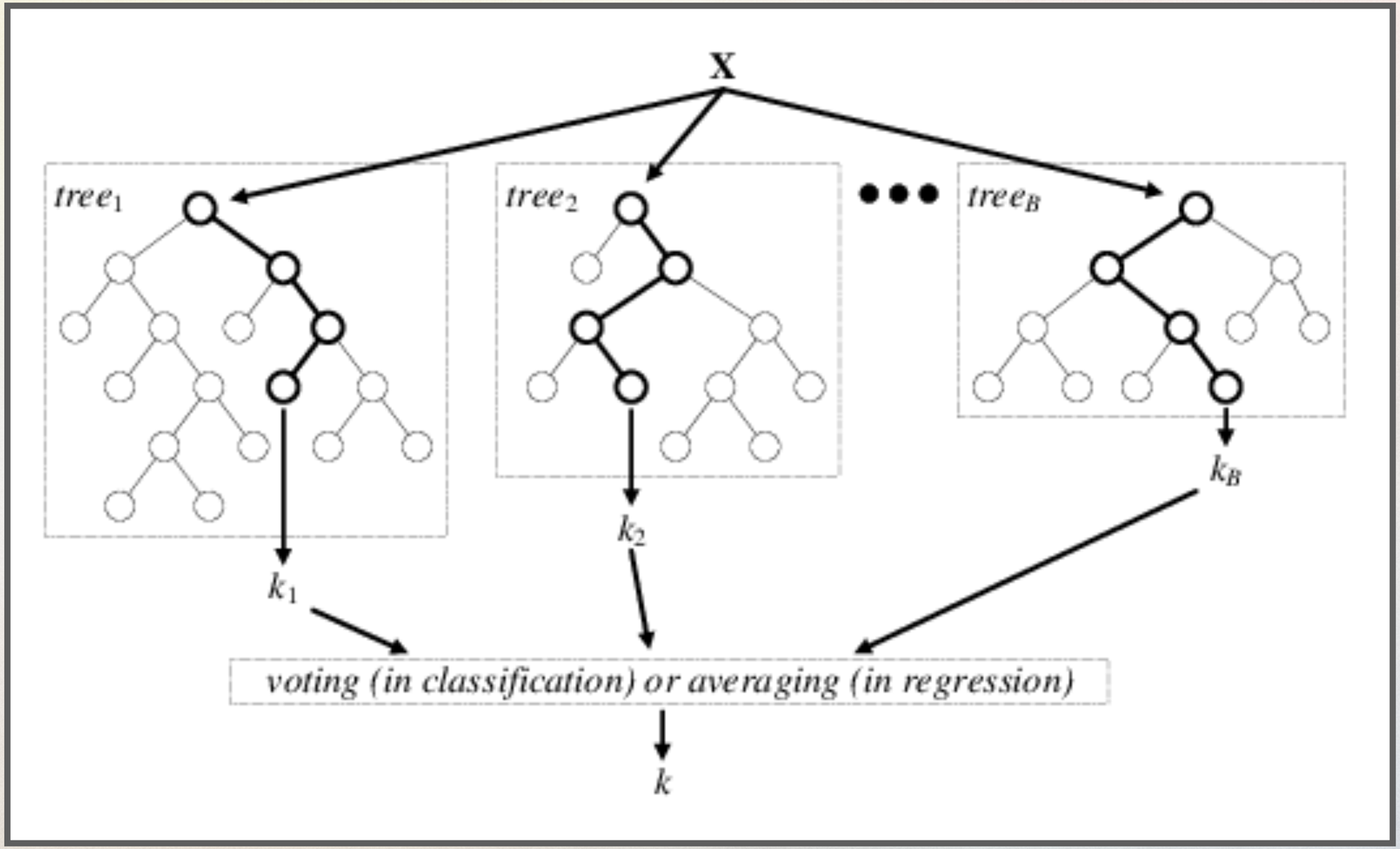
Decision Tree

Decision Trees:

- If-then statements made by machine
- Models relationships between features and outputs
- Easy to Explain
- Feature chosen on best splits
- Best split chosen by implementation (e.g. gini-impurity)


Is sex male
yes
no
Survived
0.73 36%
Is age > 9.5?
yes
no
Died
0.17 61%
Is # family members aboard > 2.5?
yes
no
Died
0.05 2%
Survived
0.89 2%

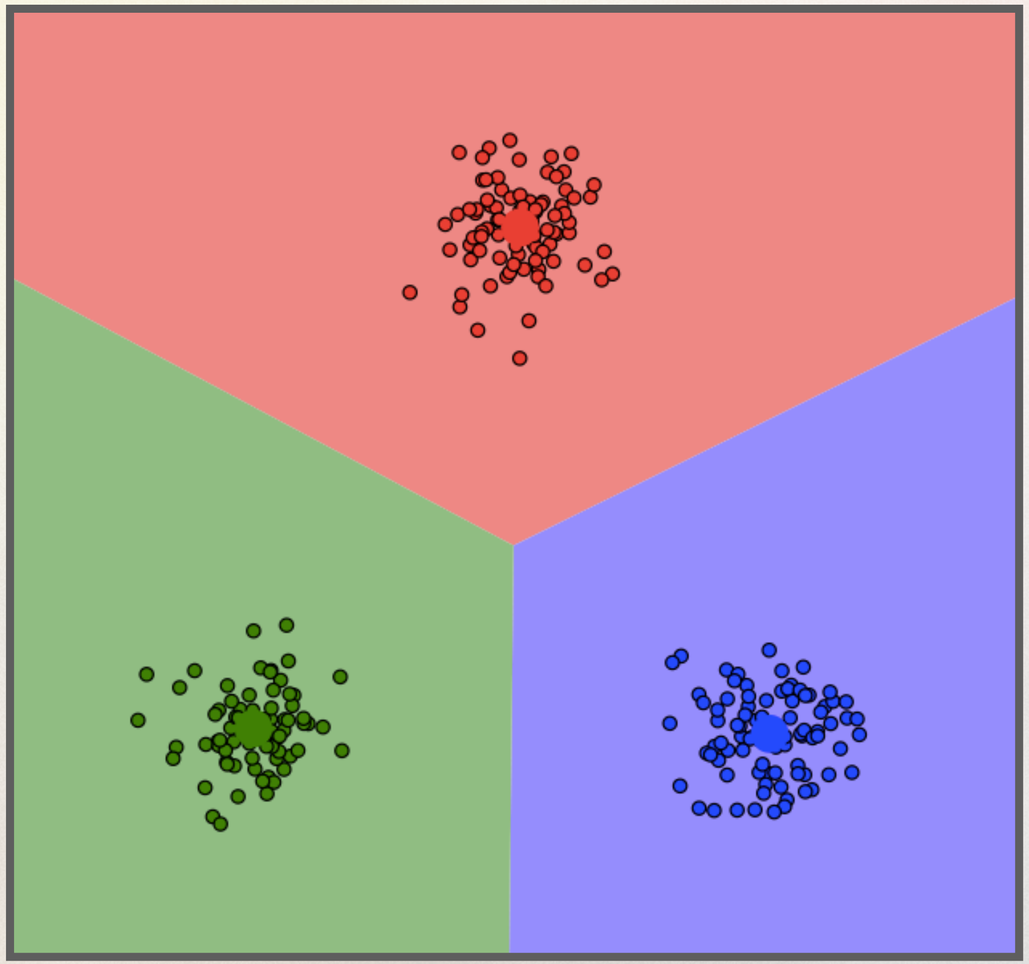
Random Forrest


X
tree1
tree2
treeB
k1
k2
kB
voting (in classification) or averaging (in regression)
k

KMeans

KMeans:

- Unsupervised Learning
- KMeans process:
  - Determine the number of centroids
  - Centroids are calculated by mean
  - Then calculates a new mean
  - Repeat until convergence



https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

Thank You

Daniel Hinojosa
Programmer, Consultant, Trainer
Contact:
Email: dhinojosa@evolutionnext.com
Twitter @dhinojosa