LEDE Algorithms

Richard Dunks
Chase Davis

WEEK 4
CLASS 2

Theses slides are based on the slides by

- Tan, Steinbach and Kumar (textbook authors)
- Eamonn Koegh (UC Riverside)
- Andrew Moore (CMU/Google)

MID-COURSE SURVEY

http://goo.gl/forms/jOs1Vj0ij3

DO IT NOW

4-2_DoNow

Goals for today

- Expand on our discussion of entropy and information gain
- Discuss the meaning of a confusion matrix
- Expand on our discussion of feature engineering
- Discuss logistic regression
- Discuss conditional probability and Bayes Theorem
- Demonstrate the use of the Naive Bayes classifier

FIRST SOME HOUSEKEEPING

Entropy

Entropy (disorder, impurity) of a set of examples, S, relative to a binary classification is:

$$Entropy(S) = -p_1 \log_2(p_1) - p_0 \log_2(p_0)$$

where p1 is the fraction of positive examples in S and p0 is the fraction of negatives

Examples for Computing Entropy

$$Entropy(t) = -\sum_j p(j\,|\,t)\log_2 p(j\,|\,t)$$

NOTE: $p(j \mid t)$ is computed as the relative frequency of class *j* at node *t*

C1	**0**
C2	**6**

P(C1) = 0/6 = 0   P(C2) = 6/6 = 1

Entropy = – 0 $\log_2$ 0 – 1 $\log_2$ 1 = – 0 – 0 = 0

C1	**1**
C2	**5**

P(C1) = 1/6   P(C2) = 5/6

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (5/6) = 0.65

C1	**2**
C2	**4**

P(C1) = 2/6   P(C2) = 4/6

Entropy = – (2/6) $\log_2$ (2/6) – (4/6) $\log_2$ (4/6) = 0.92

C1	**3**
C2	**3**

P(C1) = 3/6=1/2   P(C2) = 3/6 = 1/2

Entropy = – (1/2) $\log_2$ (1/2) – (1/2) $\log_2$ (1/2)

= -(1/2)(-1) – (1/2)(-1) = ½ + ½ = 1

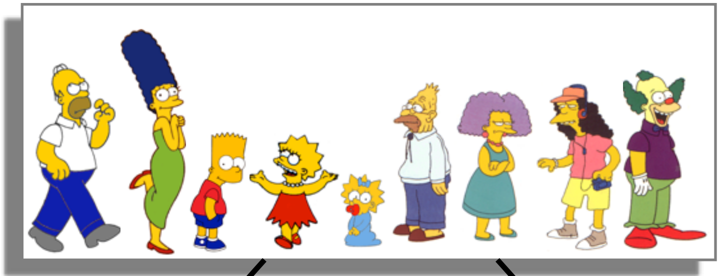
Calculating Information Gain

- Measures reduction in entropy achieved because of the split
- Choose the split that achieves most reduction (maximizes GAIN)

$$Gain(A) = E(Current\ set) - \sum E(all\ child\ sets)$$

Person	Hair Length	Weight	Age	Class
 Homer	0”	250	36	M
 Marge	10”	150	34	F
 Bart	2”	90	10	M
 Lisa	6”	78	8	F
 Maggie	4”	20	1	F
 Abe	1”	170	70	M
 Selma	8”	160	41	F
 Otto	10”	180	38	M
 Krusty	6”	200	45	M

 Comic	8”	290	38	?



$$Entropy(S) = -\frac{p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$$

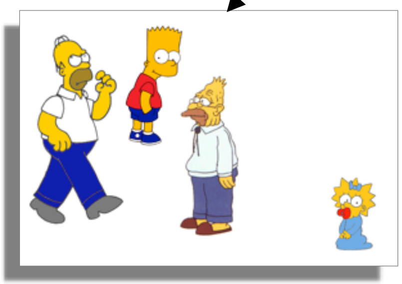
$Entropy(4F,5M) = -(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$

$= 0.9911$

yes

no

Hair Length <= 5?



Let us try splitting on *Hair length*

$Entropy(1F,3M) = -(1/4)\log_2(1/4) - (3/4)\log_2(3/4)$

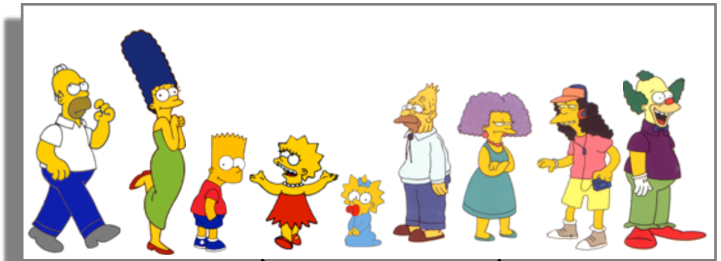
$= 0.8113$

$Entropy(3F,2M) = -(3/5)\log_2(3/5) - (2/5)\log_2(2/5)$

$= 0.9710$

$$Gain(A) = E(Current\ set) - \sum E(all\ child\ sets)$$

Gain(Hair Length <= 5) = **0.9911** – (4/9 * **0.8113** + 5/9 * **0.9710**) = **0.0911**



$$Entropy(S) = -\frac{p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$$

Entropy(4**F**,5**M**) = $-(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$

= **0.9911**

yes

no

Weight <= 160?

Let us try splitting on *Weight*

Entropy(4**F**,1**M**) = $-(4/5)\log_2(4/5) - (1/5)\log_2(1/5)$

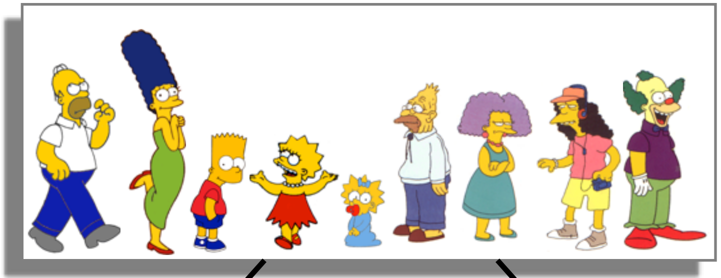
= **0.7219**

Entropy(0**F**,4**M**) = $-(0/4)\log_2(0/4) - (4/4)\log_2(4/4)$

= ***0***

$$Gain(A) = E(Current\ set) - \sum E(all\ child\ sets)$$

Gain(Weight <= 160) = **0.9911** – (5/9 * **0.7219** + 4/9 * **0**) = **0.5900**



$$Entropy(S) = -\frac{p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$$

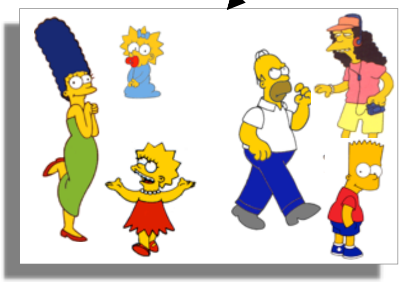
Entropy(4**F**,5**M**) = $-(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$

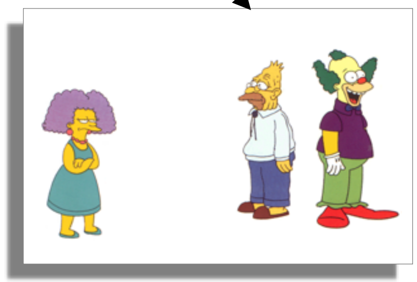
= **0.9911**

yes

no

age <= 40?





Let us try splitting on *Age*

Entropy(3**F**,3**M**) = $-(3/6)\log_2(3/6) - (3/6)\log_2(3/6)$

= ***1***

Entropy(1**F**,2**M**) = $-(1/3)\log_2(1/3) - (2/3)\log_2(2/3)$

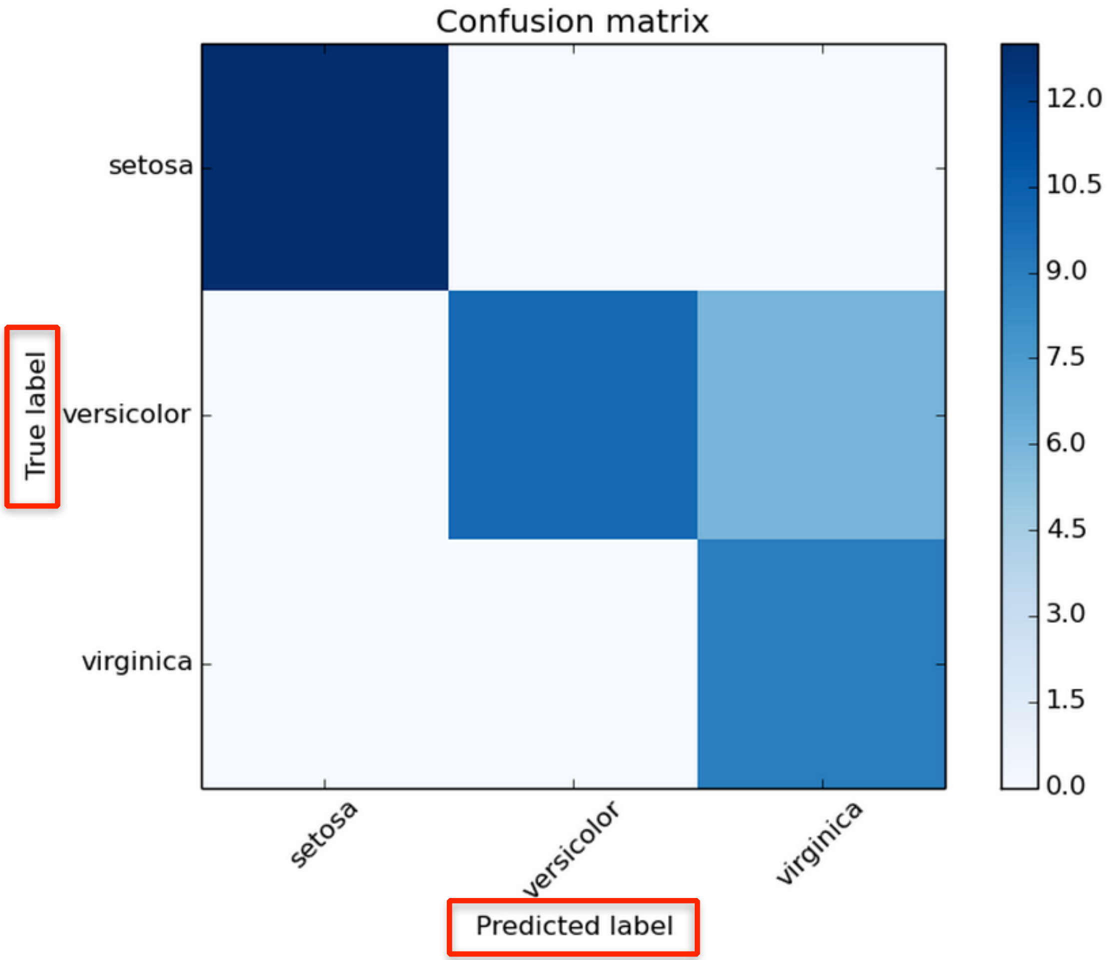
= ***0.9183***

$$Gain(A) = E(Current\ set) - \sum E(all\ child\ sets)$$

Gain(Age <= 40) = **0.9911** – (6/9 * **1** + 3/9 * **0.9183**) = **0.0183**

YOU WILL NEVER HAVE TO CALCULATE INFORMATION GAIN

But you should understand the concepts



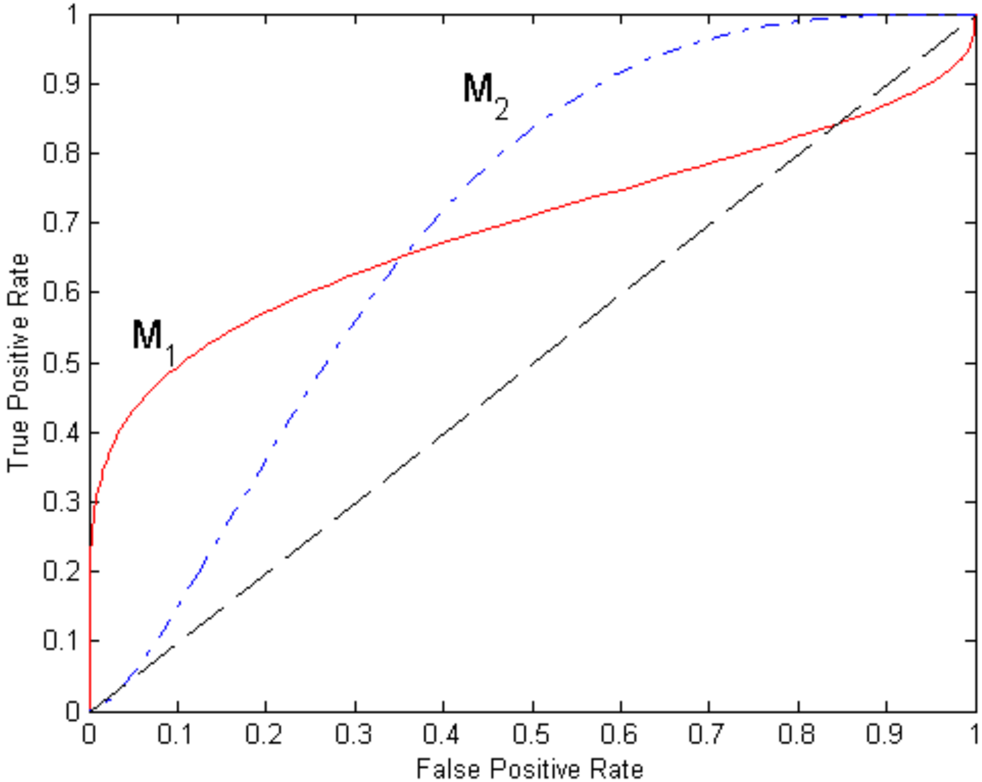
Script output:

```
Confusion matrix, without normalization
[[13  0  0]
 [ 0 10  6]
 [ 0  0  9]]
```

http://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html

Using ROC for Model Comparison

- No model consistently outperform the other
  - M1 is better for small FPR
  - M2 is better for large FPR
- Area Under the ROC curve
- Ideal:
  - Area = 1
- Random guess:
  - Area = 0.5



Feature Engineering

The process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data

http://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/

Feature Engineering

- The algorithms we’re using can’t read the information as we do
- Need to translate the information into a format that can be incorporated into the algorithms
- Typically this means taking text values and **encoding** them into numbers

COMMON METHODS OF CREATING FEATURES

Transform Existing Values

- Use integers to encode categories

"Will someone click on this ad?"	0 or 1 (no or yes)
"What number is this (image recognition)?"	0, 1, 2, etc.
"What is this news article about?"	"Sports"
"Is this spam?"	0 or 1
"Is this pill good for headaches?"	0 or 1

Transform Existing Values

- Discretize continuous values

```
df = pd.read_csv('data/ontime_reports_may_2015_ny.csv')
```

```
#filter DEP_DELAY NaNs
df = df[pd.notnull(df['DEP_DELAY'])]
```

```
#code whether delay or not delayed
df['IS_DELAYED'] = df['DEP_DELAY'].apply(lambda x: 1 if x>0 else 0 )
```

Transform Existing Values

- Create dummy variables

```
df = pd.DataFrame({'key':['b','b','a','c','a','b'],'data1':range(6)})
```

```
df
```

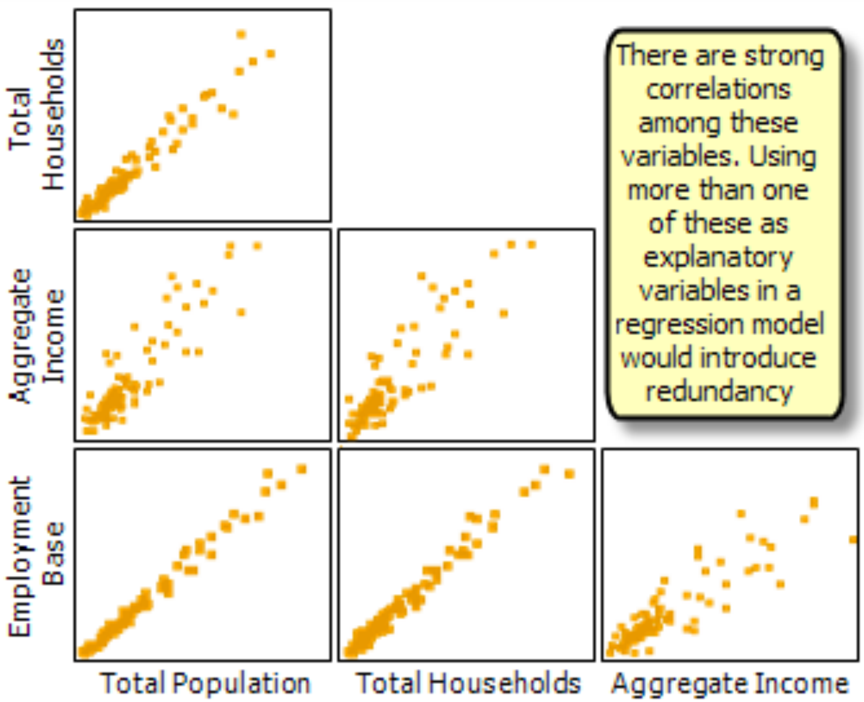
	data1	key
0	0	b
1	1	b
2	2	a
3	3	c
4	4	a
5	5	b

```
pd.get_dummies(df['key'],prefix='key')
```

	key_a	key_b	key_c
0	0	1	0
1	0	1	0
2	1	0	0
3	0	0	1
4	1	0	0
5	0	1	0

Multicollinearity

- Exists when predicator values are correlated
- Violation of independence assumption



http://resources.esri.com/help/9.3/arcgisdesktop/com/gp_toolref/spatial_statistics_toolbox/multicollinearity.htm

Dummy Variable Trap

- Creating dummy variables for all values of an attribute creates perfect multicollinearity
- Leave one value out as the baseline
- The coefficients for the other dummy variables will indicate the effect of the dummy variable left out

Transform Existing Values

- Normalize number range between 0 and 1

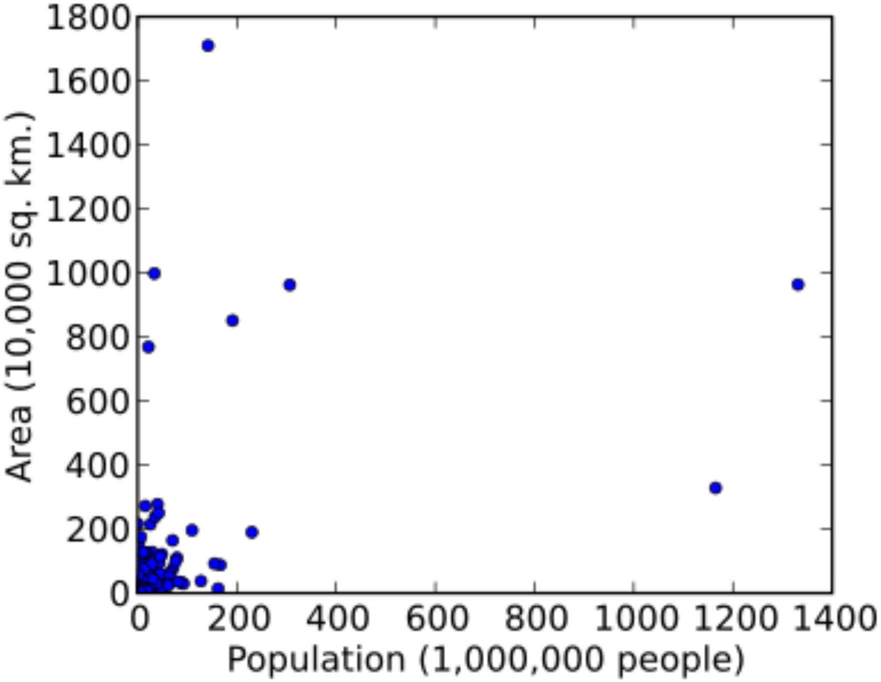
```
#Normalize the data attributes for the Iris dataset
# Example from Jump Start Scikit Learn https://machinelearningmastery.com/jump-start-scikit-learn/
from sklearn.datasets import load_iris
from sklearn import preprocessing #load the iris dataset
iris=load_iris()
X=iris.data
y=iris.target #normalize the data attributes
normalized_X = preprocessing.normalize(X)
```

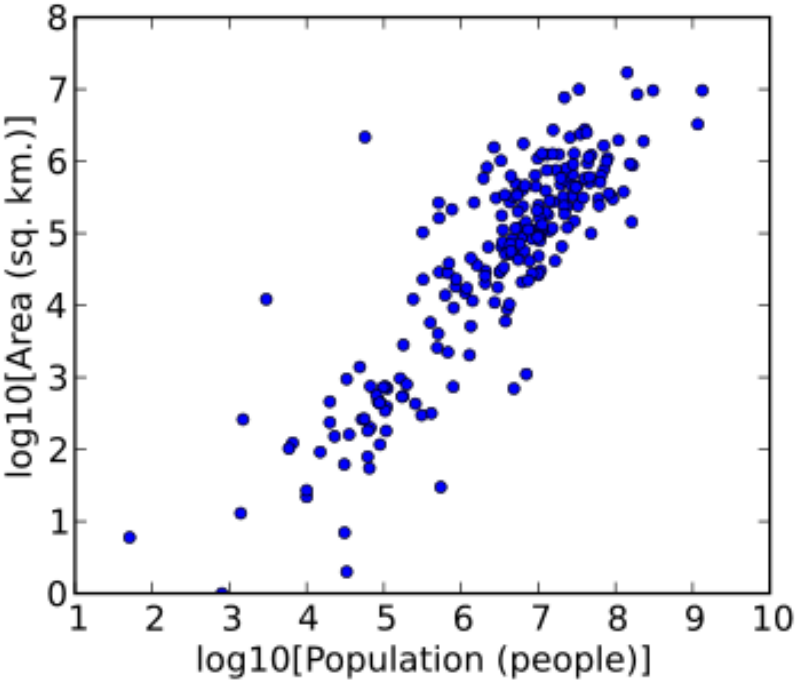
```
zip(X,normalized_X)
```

```
[(array([ 5.1,  3.5,  1.4,  0.2]),
  array([ 0.80377277,  0.55160877,  0.22064351,  0.0315205 ])),
 (array([ 4.9,  3. ,  1.4,  0.2]),
  array([ 0.82813287,  0.50702013,  0.23660939,  0.03380134])),
 (array([ 4.7,  3.2,  1.3,  0.2]),
  array([ 0.80533308,  0.54831188,  0.2227517 ,  0.03426949])),
 (array([ 4.6,  3.1,  1.5,  0.2]),
  array([ 0.80003025,  0.53915082,  0.26087943,  0.03478392])),
```

Transform Existing Values

- Logarithmic transformation



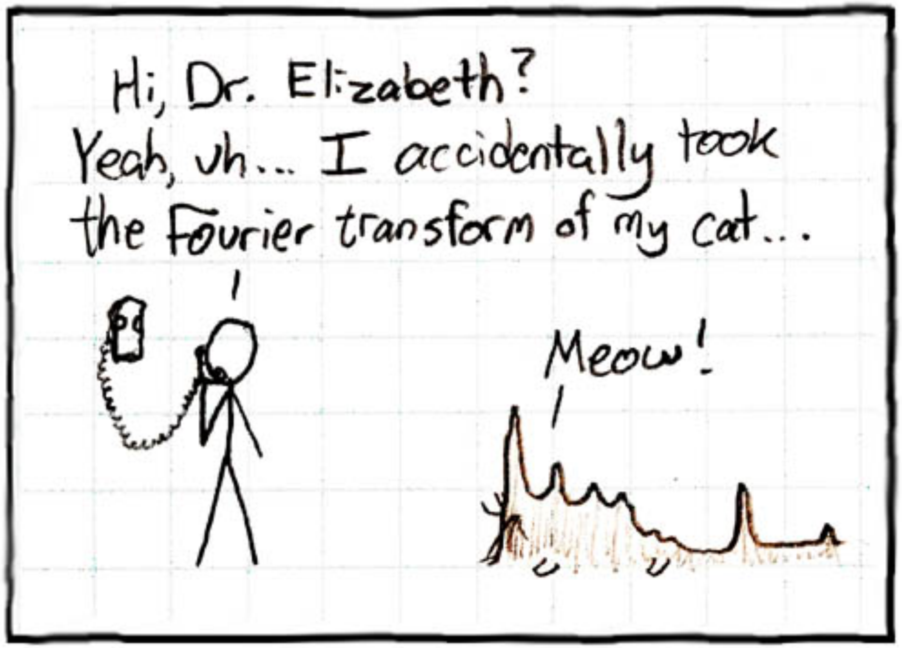


Create new values

- Calculate values based on available data
- Examples
    - Calculating distance from two points
    - Calculating the difference in time

APPROPRIATE NUMERICAL VALUES BASED ON YOUR DATA FEATURES

10 MIN BREAK



https://xkcd.com/26/

Simple Logistic Regression

- Predicts the probability an event occurs (classification)
- Fits data to a logit function logistic curve
- Used to explore associations between one binary outcome and one (continuous, ordinal, or categorical) feature
- Lets you answer questions like, "how does gender affect the probability of having hypertension?”

Multiple Logistic Regression

- Used to explore associations between one binary outcome variable and two or more features (which may be continuous, ordinal or categorical)
- Coefficients describe the nature of the relationship

Logit Versus Inverse-logit

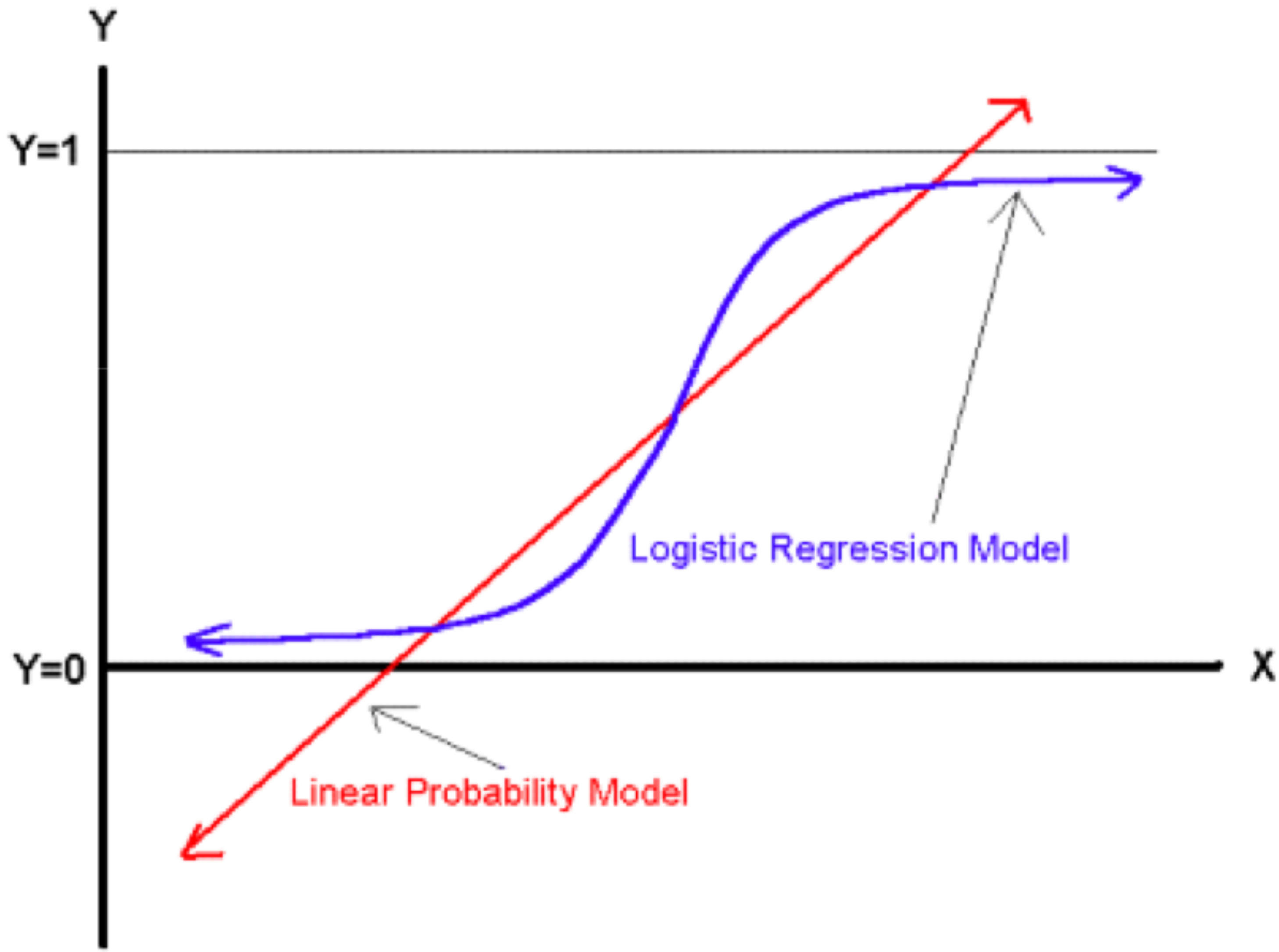
The logit function takes x values in the range $[0,1]$ and transforms them to y values along the entire real line:

$$logit(p) = log\left(\frac{p}{1-p}\right) = log(p) - log(1-p)$$

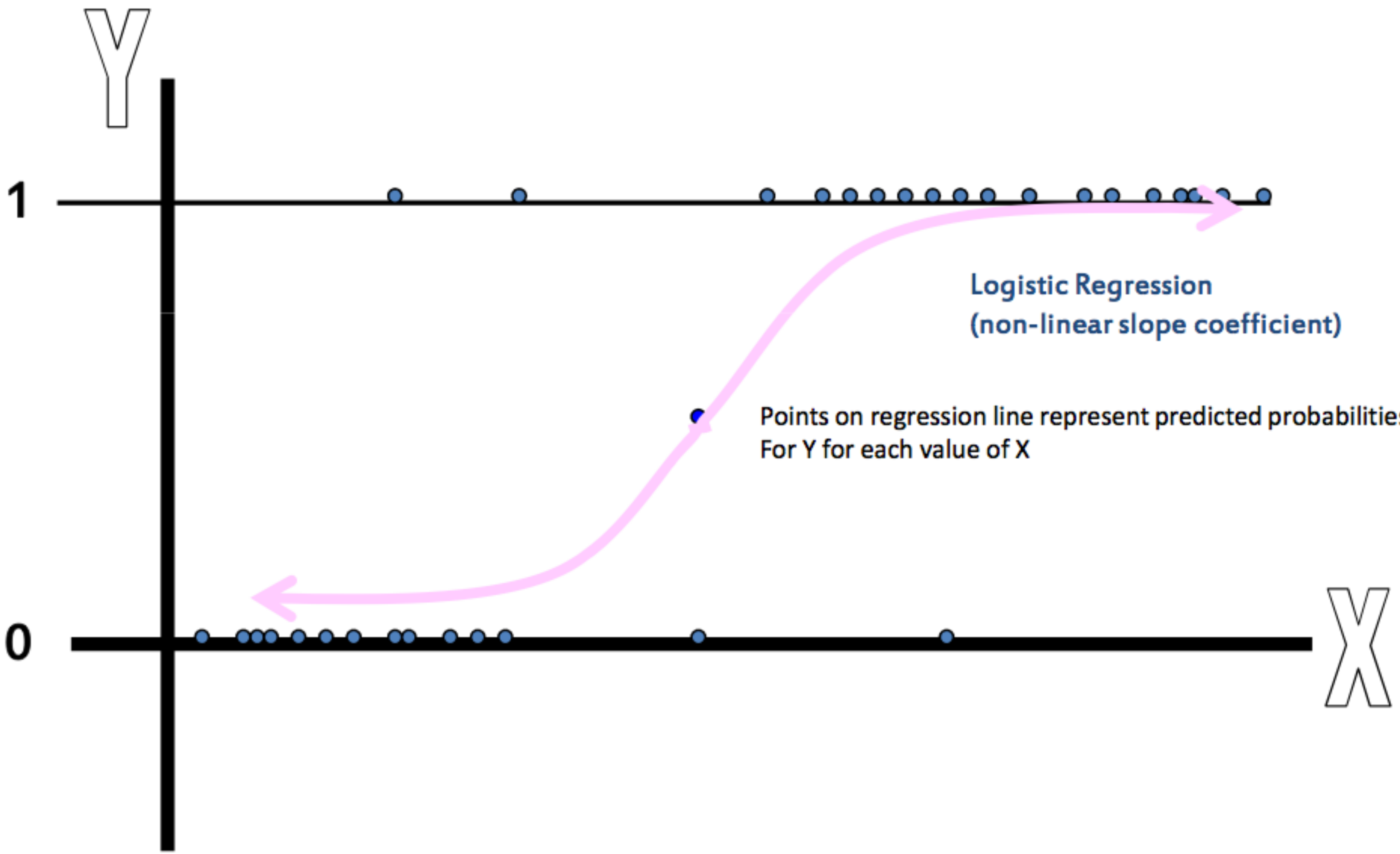
The inverse-logit does the reverse, and takes x values along the real line and tranforms them to y values in the range $[0,1]$.

From *Doing Data Science*, p.

Comparing the Logistic & Linear Regression Models

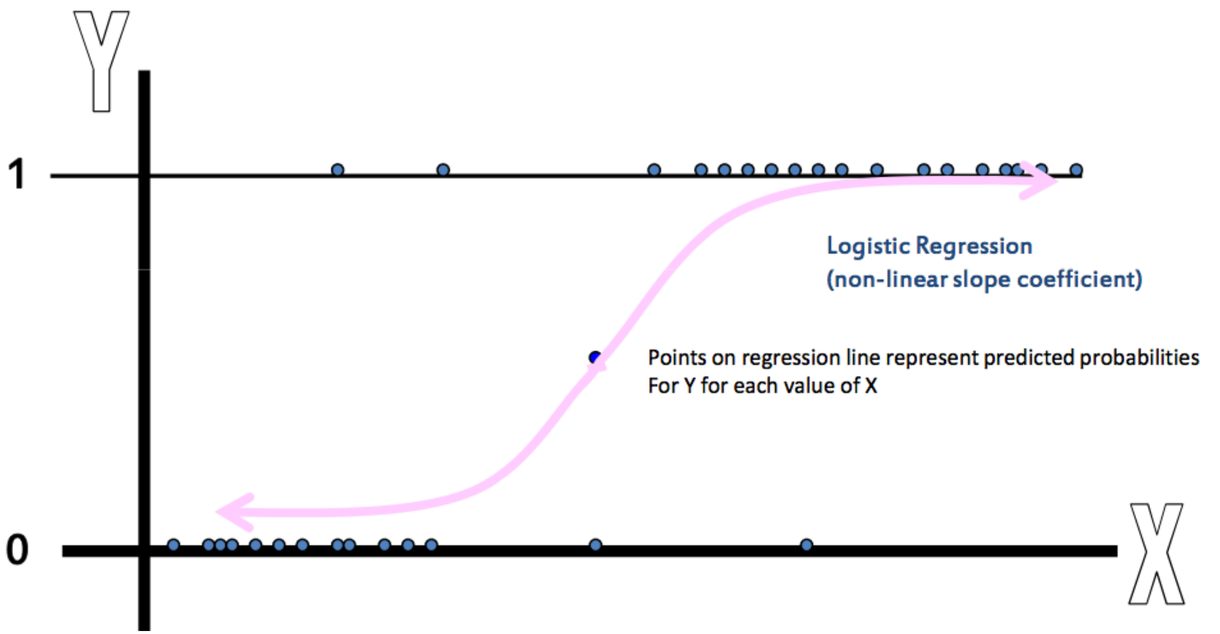


Picture of Logistic Regression



How It Works

- Given a set of features, the algorithm calculates the probability it belongs to class 1
  - P(y=1)/P(y=0)



How It Works

- Given a set of features, the algorithm calculates the probability it belongs to class 1
  - P(y=1)/P(y=0)
- Then take the logarithm of the odds ratio
  - Provides a result bounded by 0 and 1
  - Predicts the class for the instance

LET’S GIVE IT A TRY

Logistic_Regression.ipynb

Interpreting Coefficients

- Positive coefficients indicate a positive effect on the outcome
- Negative coefficients indicate a negative effect on the outcome
- Greater distance from 0, the greater the effect
- Possible to trim features with coefficients near 0 to get better results

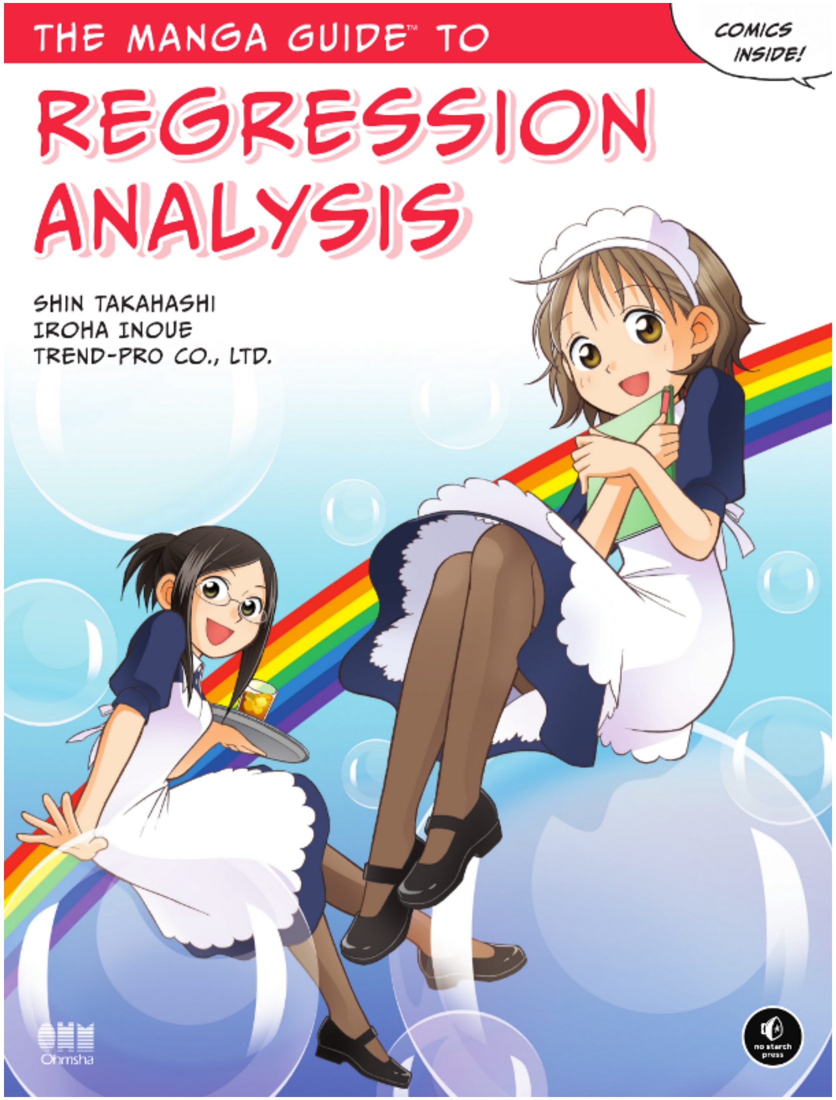
Value of Logistic Regression

- Like decision trees, they’re easy to understand and explain
- Different from more “black box” algorithms we’ll be studying later

Resources

- Logistic regression in Python
  http://blog.yhathq.com/posts/logistic-regression-and-python.html
- Logistic regression with scikit-learn
  http://nbviewer.ipython.org/github/justmarkham/gadsdc1/blob/master/logistic_assignment/kevin_logistic_sklearn.ipynb
- Classification in Python with scikit-learn:
  http://nbviewer.ipython.org/urls/s3.amazonaws.com/datarobotblog/notebooks/classification_in_python.ipynb

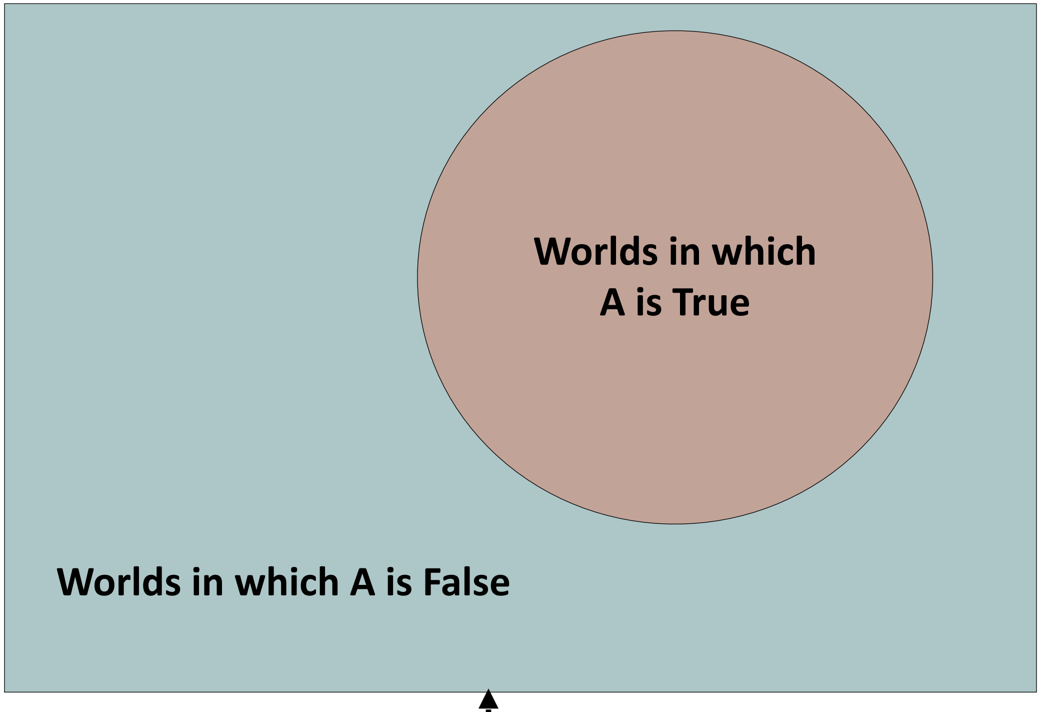
10 MIN BREAK



https://www.nostarch.com/mg_regressionanalysis.htm

CONDITIONAL PROBABILITY AND BAYES THEOREM

Event Space in Probability

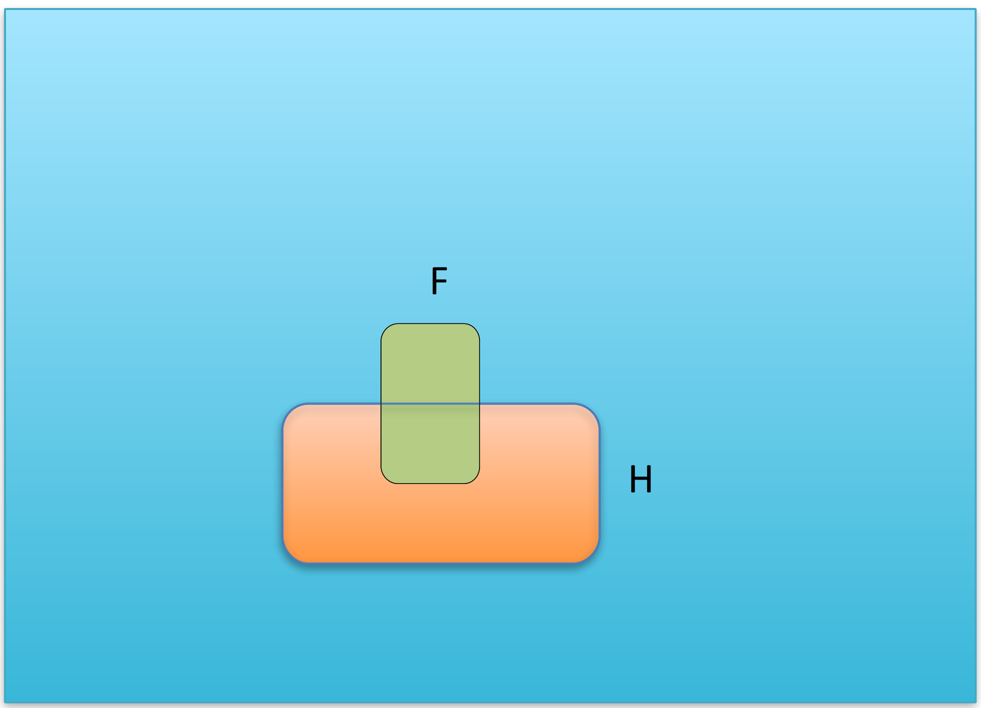


P(A) = Area of reddish oval

Event space of all possible worlds (with an area of 1)

Conditional Probability

- P(A|B) = Fraction of worlds in which B is true that also have A true



H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

"Headaches are rare and flu is rarer, but if you're coming down with 'flu there's a 50-50 chance you'll have a headache."

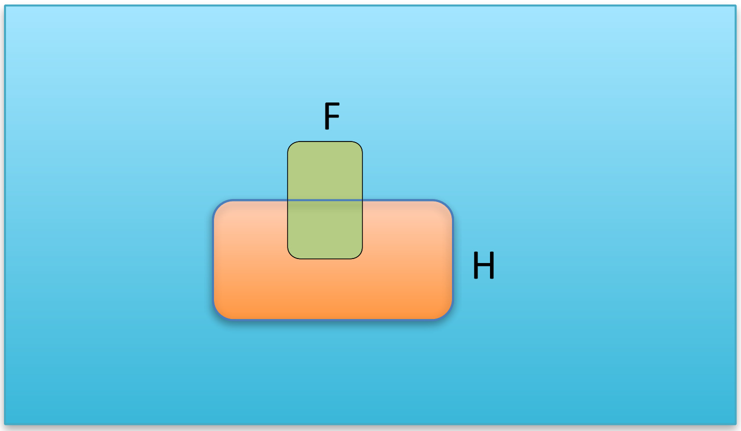
Definition of Conditional Probability

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

Corollary: The Chain Rule

$$P(A \text{ and } B) = P(A|B)\, P(B)$$

Probabilistic Inference



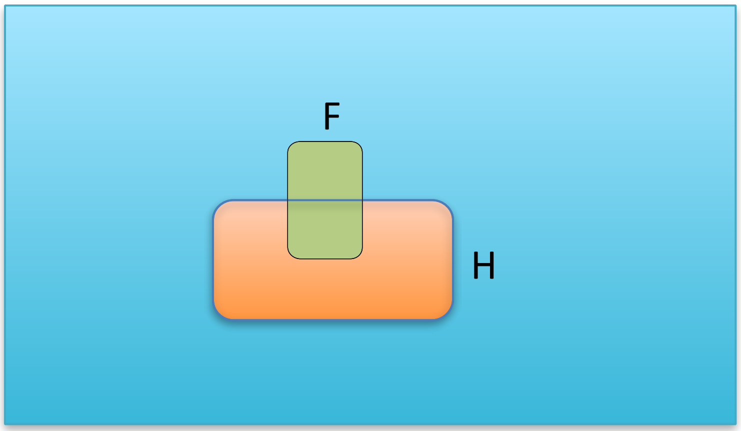
H = “Have a headache”
F = “Coming down with Flu”

$P(H) = 1/10$
$P(F) = 1/40$
$P(H|F) = 1/2$

One day you wake up with a headache. You think: “Drat! 50% of flus are associated with headaches so I must have a 50-50 chance of coming down with flu”

Is this reasoning valid?

Probabilistic Inference



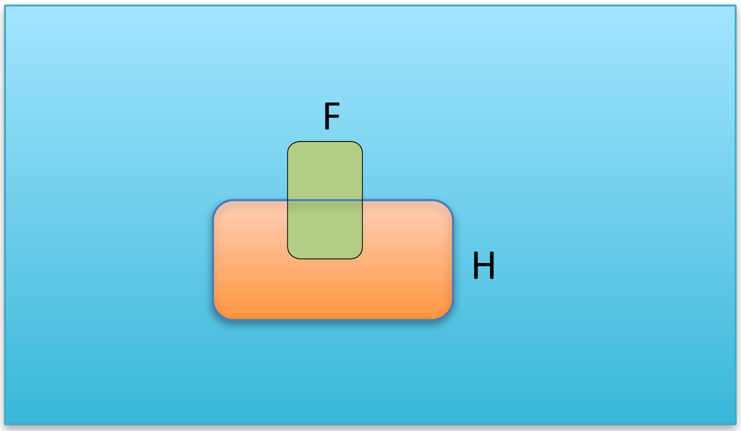
H = “Have a headache”
F = “Coming down with Flu”

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(F and H) = …

P(F|H) = …

Probabilistic Inference



H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

$$P(F \text{ and } H) = P(H \mid F) \times P(F) = \frac{1}{2} \times \frac{1}{40} = \frac{1}{80}$$

$$P(F \mid H) = \frac{P(F \text{ and } H)}{P(H)} = \frac{1/80}{1/10} = \frac{1}{8}$$

What we just did…

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

This is Bayes Rule

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London,* **53:370-418**

Some more terminology

- The Prior Probability is the probability assuming no specific information.
  - Thus we would refer to P(A) as the prior probability of even A occurring
  - We would not say that P(A|C) is the prior probability of A occurring
- The Posterior probability is the probability given that we know something
  - We would say that P(A|C) is the posterior probability of A (given that C occurs)

Example of Bayes Theorem

- Given:
  - A doctor knows that meningitis causes stiff neck 50% of the time
  - Prior probability of any patient having meningitis is 1/50,000
  - Prior probability of any patient having stiff neck is 1/20

- If a patient has stiff neck, what's the probability he/she has meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

How is this relevant to data mining?

- The features (attribute values) observed are evidence of one hypothesis (class) or another, say spam or not spam.
- Can we devise a learning method based on this idea?

$$P(\overset{\text{Class}}{\overset{\downarrow}{C}} = c \mid \overbrace{A_1 = a_1 \,\& \ldots \&\, A_k = a_k}^{\text{Evidence (observed features)}}) = \frac{P(A_1 = a_1 \,\& \ldots \&\, A_k = a_k \mid C = c)P(C = c)}{P(A_1 = a_1 \,\& \ldots \&\, A_k = a_k)}$$

Naïve Bayes

- The "naïve" assumption: The value of each attribute is independent of the values of all other attributes, given the class
- Given the Naïve assumption

$$P(A_1 = a_1 \mid A_2 = a_2 \,\&\ldots\&\, A_k = a_k \,\&\, C = c) = P(A_1 = a_1 \mid C = c)$$

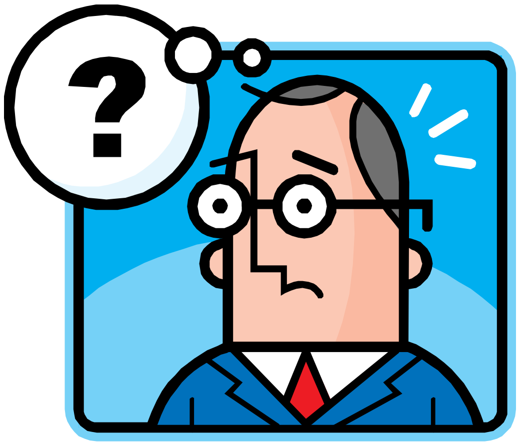
- Thus,

$$P(A_1 = a_1 \,\&\ldots\&\, A_k = a_k \mid C = c) =$$
$$P(A_1 = a_1 \mid C = c) * P(A_2 = a_2 \mid C = c) * \ldots * P(A_k = a_k \mid C = c)$$

- Each factor in the above product is estimated from training data by :

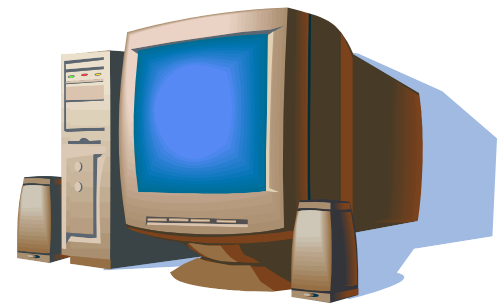
$$P(A_i = a_i \mid C = c) = \frac{count(A_i = a_i \wedge C = c)}{count(C = c)}$$

Example



•I am 35-year old
•I earn $40,000
•My credit rating is fair

Will he buy a computer?



- E : 35 years old customer with an income of $40,000 and fair credit rating.
- H : Hypothesis that the customer will buy a computer.

Data Table

Rec	Age	Income	Student	Credit_rating	Buys_computer
1	<=30	High	No	Fair	No
2	<=30	High	No	Excellent	No
3	31..40	High	No	Fair	Yes
4	>40	Medium	No	Fair	Yes
5	>40	Low	Yes	Fair	Yes
6	>40	Low	Yes	Excellent	No
7	31..40	Low	Yes	Excellent	Yes
8	<=30	Medium	No	Fair	No
9	<=30	Low	Yes	Fair	Yes
10	>40	Medium	Yes	Fair	Yes
11	<=30	Medium	Yes	Excellent	Yes
12	31..40	Medium	No	Excellent	Yes
13	31..40	High	Yes	Fair	Yes
14	>40	Medium	No	Excellent	No

Bayes Theorem

$$P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}$$

- P(H|E) : Probability that the customer will buy a computer given that we know his age, credit rating and income. (Posterior Probability of H)
- P(H) : Probability that the customer will buy a computer regardless of age, credit rating, income (Prior Probability of H)
- P(E|H) : Probability that the customer is 35 yrs old, have fair credit rating and earns $40,000, given that he has bought our computer (Posterior Probability of E)
- P(E) : Probability that a person from our set of customers is 35 yrs old, have fair credit rating and earns $40,000. (Prior Probability of X)

Example. Description

- The data samples are described by attributes *age*, *income*, *student*, and *credit*. The class label attribute, *buy*, tells whether the person buys a computer, has two distinct values, yes (Class C1) and no (Class C2).

- The sample we wish to classify is
  E = (age <= 30, income = medium, student = yes, credit = fair)

- We need to maximize P(E|Ci)P(Ci), for i = 1, 2.
- P(Ci), the a priori probability of each class, can be estimated based on the training samples:

$$P(buy = yes) = \frac{9}{15}$$

$$P(buy = no) = \frac{5}{15}$$

Example. Description

- **E = (age <= 30, income = medium, student = yes, credit = fair)**
- To compute P(E|Ci), for i = 1, 2, we compute the following conditional probabilities:

$$P(age <= 30 \mid buy = yes) = \frac{2}{9}$$

$$P(age <= 30 \mid buy = no) = \frac{3}{5}$$

$$P(income = medium \mid buy = yes) = \frac{4}{9}$$

$$P(income = medium \mid buy = no) = \frac{2}{5}$$

$$P(student = yes \mid buy = yes) = \frac{6}{9}$$

$$P(student = yes \mid buy = no) = \frac{1}{5}$$

$$P(credit = fair \mid buy = yes) = \frac{6}{9}$$

$$P(credit = fair \mid buy = no) = \frac{2}{5}$$

Example. Description

- **E = (age <= 30, income = medium, student = yes, credit = fair)**
- Using probabilities from the two previous slides:

$$P(E \mid buy = yes) = P(age <= 30 \mid buy = yes) *$$
$$P(income = medium \mid buy = yes) *$$
$$P(student = yes \mid buy = yes) *$$
$$P(credit = fair \mid buy = yes) *$$
$$P(buy = yes) / D =$$
$$(\frac{2}{9} * \frac{4}{9} * \frac{6}{9} * \frac{6}{9} * \frac{9}{15}) / D = 0.0263 / D$$

$$P(C_i \mid E) = \frac{P(E \mid C_i) P(C_i)}{P(E)}$$

$$P(E \mid buy = no) = P(age <= 30 \mid buy = no) *$$
$$P(income = medium \mid buy = no) *$$
$$P(student = yes \mid buy = no) *$$
$$P(credit = fair \mid buy = no) *$$
$$P(buy = no) / D =$$
$$(\frac{3}{5} * \frac{2}{5} * \frac{1}{5} * \frac{2}{5} * \frac{5}{15}) / D = 0.0064 / D$$

Bayes Classifiers

That was a visual intuition for a simple case of the Bayes classifier, also called:

- **Naïve** Bayes: we assume independence of attributes

We are about to see some of the mathematical formalisms, and more examples, but keep in mind the basic idea.

Find out the probability of the previously unseen instance belonging to each class, then simply pick the most probable class.

When is it used?

- Recommending web pages
- Spam filtering
- Sentiment analysis for marketing
- Personalized news articles
- Email messages
  - Routing
  - Prioritizing
  - Folderizing
  - spam filtering
- Nate Silver’s election predictions

WRAP-UP

Readings

- *Doing Data Science* “Logistic Regression”, “Naive Bayes”
- *Building Machine Learning Systems with Python* “Logistic Regression”