Introduction to Machine Learning

Machine Learning: Jordan Boyd-Graber
University of Maryland
FEATURE ENGINEERING

Slides adapted from Eli Upfal

What does it mean to learn something?

- What are the things that we're learning?
- What does it mean to be learnable?
- Provides a framework for reasoning about what we can *theoretically* learn

What does it mean to learn something?

- What are the things that we're learning?
- What does it mean to be learnable?
- Provides a framework for reasoning about what we can *theoretically* learn
    - Sometime theoretically learnable things are very difficult
    - Sometimes things that should be hard actually work

Example

- Californian just moved to Colorado
- When is it “nice” outside?
- Has a perfect thermometer, but natives call 50F (10C) “nice”

Example

- Californian just moved to Colorado
- When is it “nice” outside?
- Has a perfect thermometer, but natives call 50F (10C) “nice”
- Each temperature is an observation x
- Coloradan concept of “nice” $c(x)$
- Californian wants to learn hypothesis $h(x)$ close to $c(x)$

Example

- Californian just moved to Colorado
- When is it “nice” outside?
- Has a perfect thermometer, but natives call 50F (10C) “nice”
- Each temperature is an observation x
- Coloradan concept of “nice” $c(x)$
- Californian wants to learn hypothesis $h(x)$ close to $c(x)$

Generalization error

$$R(h) = \Pr_{x\sim D}[h(x) \neq c(x)] = \mathbb{E}_{x\sim D}[\mathbb{1}[h(x) \neq c(x)]] \quad (1)$$

[Notation $\mathbb{1}[x] = 1$ iff x is true, 0 otherwise]

Example

- Californian just moved to Colorado
- When is it “nice” outside?
- Has a perfect thermometer, but natives call 50F (10C) “nice”
- Each temperature is an observation x
- Coloradan concept of “nice” $c(x)$
- Californian wants to learn hypothesis $h(x)$ close to $c(x)$

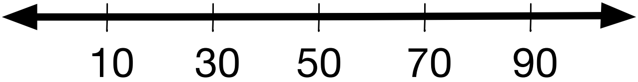
Generalization error

$$R(h) = \Pr_{x \sim D}\left[h(x) \neq c(x)\right] = \mathbb{E}_{x \sim D}\left[\mathbb{1}\left[h(x) \neq c(x)\right]\right] \quad (1)$$

[Notation $\mathbb{1}\left[x\right] = 1$ iff x is true, 0 otherwise]

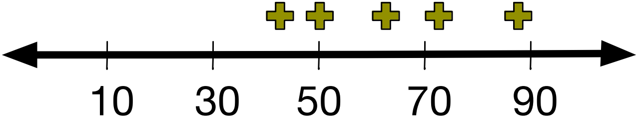
Probably Correct

The Californian gets n random examples.



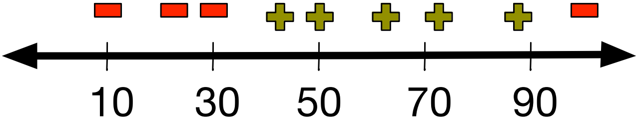
Probably Correct

The Californian gets n random examples.



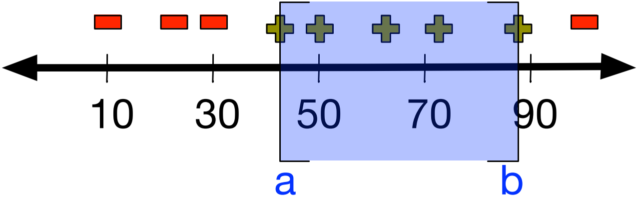
Probably Correct

The Californian gets *n* random examples.

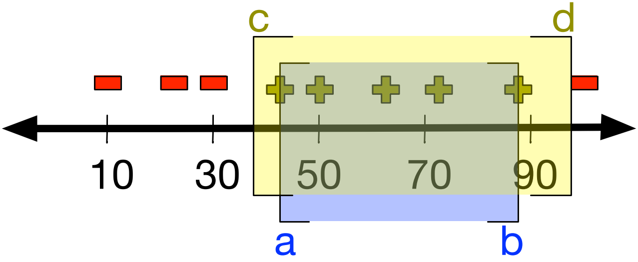


Probably Correct

The best rule that conforms with the examples is $[a, b]$.



Probably Correct



Let $[c, d]$ be the correct (unknown) rule. Let Δ be the gap between. The probability of being wrong is the probability that n samples missed Δ_{ca} and Δ_{bd}.

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}[R(h_S) \leq \epsilon] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

The sample we learn from

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

The data distribution the sample comes from

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

The hypothesis we learn

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

Generalization error

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S\sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

Our bound on the generalization error (e.g., we want it to be better than 0.1)

PAC-learning definition

Definition

PAC-learnable A concept C is PAC-learnable if $\exists$ algorithm $\mathscr{A}$ and a polynomial function f such that for any ϵ and δ, $\forall D(X)$ and $c \in C$

$$\Pr_{S \sim D^m}\left[R(h_S) \leq \epsilon\right] \geq 1 - \delta \tag{2}$$

for any sample size $m \geq f\left(\frac{1}{\epsilon}, \frac{1}{\delta}, n, |c|\right)$

The probability of learning a hypothesis with error greater than ϵ (e.g., 0.05)

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr\left[x_1 \not\in \Delta_{ca} \wedge \cdots \wedge x_m \not\in \Delta_{ca}\right] = \prod_i^m \Pr\left[x_i \not\in \Delta_{ca}\right] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr\left[x_1 \not\in \Delta_{ca} \wedge \cdots \wedge x_m \not\in \Delta_{ca}\right] = \left(1 - \epsilon\right)^m \leq e^{-\epsilon m} \tag{4}$$

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = \prod_i^m \Pr[x_i \notin \Delta_{ca}] \tag{3}$$

Independence!

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

Is a Californian learning temperature PAC learnable?

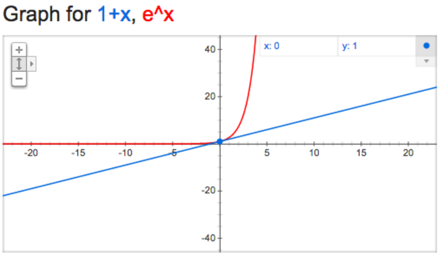
- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = \prod_i^m \Pr[x_i \notin \Delta_{ca}] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

Useful inequality: $1 + x \leq e^x$



Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = \prod_i^m \Pr[x_i \notin \Delta_{ca}] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

- We want the generalization to violate ϵ less than δ, solving for m:

$$\Pr[R(h) \geq \epsilon] \leq \delta \tag{5}$$

$$2e^{-\epsilon m} \leq \delta \tag{6}$$

$$-\epsilon m \leq \ln \frac{\delta}{2} \tag{7}$$

$$\frac{1}{\epsilon} \ln \frac{2}{\delta} \leq m \tag{8}$$

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = \prod_i^m \Pr\left[x_i \notin \Delta_{ca}\right] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

- We want the generalization to violate ϵ less than δ, solving for m:

$$\begin{aligned}
\Pr\left[R(h) \geq \epsilon\right] &\leq \delta && (5)\\
2e^{-\epsilon m} &\leq \delta && (6)\\
-\epsilon m &\leq \ln \frac{\delta}{2} && (7)\\
\frac{1}{\epsilon} \ln \frac{2}{\delta} &\leq m && (8)
\end{aligned}$$

Analysis is symmetrical for Δ_{ca} and Δ_{bd}

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = \prod_i^m \Pr\left[x_i \notin \Delta_{ca}\right] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

- We want the generalization to violate ϵ less than δ, solving for m:

$$\Pr\left[R(h) \geq \epsilon\right] \leq \delta \tag{5}$$

$$2e^{-\epsilon m} \leq \delta \tag{6}$$

$$-\epsilon m \leq \ln \frac{\delta}{2} \tag{7}$$

$$\frac{1}{\epsilon} \ln \frac{2}{\delta} \leq m \tag{8}$$

δ corresponds to the probability of bad hypothesis

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = \prod_i^m \Pr\left[x_i \notin \Delta_{ca}\right] \tag{3}$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = (1-\epsilon)^m \leq e^{-\epsilon m} \tag{4}$$

- We want the generalization to violate ϵ less than δ, solving for m:

$$\Pr\left[R(h) \geq \epsilon\right] \leq \delta \tag{5}$$

$$2e^{-\epsilon m} \leq \delta \tag{6}$$

$$-\epsilon m \leq \ln \frac{\delta}{2} \tag{7}$$

$$\frac{1}{\epsilon} \ln \frac{2}{\delta} \leq m \tag{8}$$

Take log of both sides

Is a Californian learning temperature PAC learnable?

- Bad event happens if no training point in Δ_{ca} or Δ_{bd}.

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = \prod_i^m \Pr\left[x_i \notin \Delta_{ca}\right] \quad (3)$$

- We want the probability of a point landing there (or to be less than ϵ

$$\Pr\left[x_1 \notin \Delta_{ca} \wedge \cdots \wedge x_m \notin \Delta_{ca}\right] = (1-\epsilon)^m \leq e^{-\epsilon m} \quad (4)$$

- We want the generalization to violate ϵ less than δ, solving for m:

$$\Pr\left[R(h) \geq \epsilon\right] \leq \delta \quad (5)$$

$$2e^{-\epsilon m} \leq \delta \quad (6)$$

$$-\epsilon m \leq \ln \frac{\delta}{2} \quad (7)$$

$$\frac{1}{\epsilon} \ln \frac{2}{\delta} \leq m \quad (8)$$

Direction of inequality flips when you divide by $-m$

Consistent Hypotheses, Finite Spaces

- Possible to prove that specific problems are learnable (and we will!)
- Can we do something more general?
- Yes, for **finite** hypothesis spaces $c \in H$
- That are also consistent with training data

Theorem

Learning bounds for finite H, consistent Let H be a finite set of functions mapping from $\mathscr{X}$ to $\mathscr{Y}$. Let $\mathscr{A}$ be an algorithm that for a iid sample S returns a consistent hypothesis (training error $\hat{R}(h) = 0$), then for any $\epsilon, \delta > 0$, the concept is PAC learnable with samples

$$m \geq \frac{1}{\epsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) \tag{9}$$

Proof: Setup

We want to bound the probability that some $h \in H$ is consistent and has error more than ϵ.

$$
\begin{aligned}
& \Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (10)\\
=& \Pr\left[\left(h_1 \in H \wedge \hat{R}(h_1) = 0 \wedge R(h_1) > \epsilon\right) \vee \cdots \vee \left(h_i \in H \wedge \hat{R}(h_i) = 0 \wedge R(h_i) > \epsilon\right)\right] \\
\leq& \sum_h \Pr\left[\hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (11)\\
\leq& \sum_h \Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] && (12)
\end{aligned}
$$

Proof: Setup

We want to bound the probability that some $h \in H$ is consistent and has error more than ϵ.

$$
\begin{aligned}
& \Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (10)\\
= & \Pr\left[\left(h_1 \in H \wedge \hat{R}(h_1) = 0 \wedge R(h_1) > \epsilon\right) \vee \cdots \vee \left(h_i \in H \wedge \hat{R}(h_i) = 0 \wedge R(h_i) > \epsilon\right)\right] \\
\leq & \sum_h \Pr\left[\hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (11)\\
\leq & \sum_h \Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] && (12)
\end{aligned}
$$

Proof: Setup

We want to bound the probability that some $h \in H$ is consistent and has error more than ϵ.

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \tag{10}$$

$$= \Pr\left[\left(h_1 \in H \wedge \hat{R}(h_1) = 0 \wedge R(h_1) > \epsilon\right) \vee \cdots \vee \left(h_i \in H \wedge \hat{R}(h_i) = 0 \wedge R(h_i) > \epsilon\right)\right]$$

$$\leq \sum_h \Pr\left[\hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \tag{11}$$

$$\leq \sum_h \Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] \tag{12}$$

Union bound

Proof: Setup

We want to bound the probability that some $h \in H$ is consistent and has error more than ϵ.

$$
\begin{aligned}
&\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (10)\\
=&\Pr\left[\left(h_1 \in H \wedge \hat{R}(h_1) = 0 \wedge R(h_1) > \epsilon\right) \vee \cdots \vee \left(h_i \in H \wedge \hat{R}(h_i) = 0 \wedge R(h_i) > \epsilon\right)\right]\\
\leq &\sum_h \Pr\left[\hat{R}(h) = 0 \wedge R(h) > \epsilon\right] && (11)\\
\leq &\sum_h \Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] && (12)
\end{aligned}
$$

Definition of conditional probability

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis h.

$$\Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] \leq (1 - \epsilon)^m \tag{13}$$

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis h.

$$\Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] \leq (1-\epsilon)^m \tag{13}$$

but this must be true of all of the hypotheses in H,

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \leq |H|(1-\epsilon)^m \tag{14}$$

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis h.

$$\Pr\left[\hat{R}(h) = 0 \mid R(h) > \epsilon\right] \leq (1-\epsilon)^m \tag{13}$$

but this must be true of all of the hypotheses in H,

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \leq |H|(1-\epsilon)^m \tag{14}$$

$|H|(1-\epsilon)^m \leq |H|e^{-m\epsilon} = \delta$ we set the RHS to be equal to δ

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis h.

$$\Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] \leq (1-\epsilon)^m \tag{13}$$

but this must be true of all of the hypotheses in H,

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \leq |H|(1-\epsilon)^m \tag{14}$$

$$|H|(1-\epsilon)^m \leq |H|e^{-m\epsilon} = \delta$$

$$\ln\delta = \ln|H| - m\epsilon$$

apply log to both sides

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis h.

$$\Pr\left[\hat{R}(h) = 0 \,|\, R(h) > \epsilon\right] \leq (1-\epsilon)^m \tag{13}$$

but this must be true of all of the hypotheses in H,

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \wedge R(h) > \epsilon\right] \leq |H|(1-\epsilon)^m \tag{14}$$

$$|H|(1-\epsilon)^m \leq |H|e^{-m\epsilon} = \delta$$

$$\ln\delta = \ln|H| - m\epsilon$$

$$-\ln\frac{1}{\delta} - \ln|H| = -m\epsilon$$

move $\ln|H|$ to the other side, and rewrite $\ln\delta = -0 - (-\ln\delta) = -1(\ln 1 - \ln\delta) = -\ln(\frac{1}{\delta})$

Proof: Connection back to interval learning

The generalization error is greater than ϵ, so we bound probability of no inconsistent points in training for a single hypothesis *h*.

$$\Pr\left[\hat{R}(h) = 0 \mid R(h) > \epsilon\right] \leq (1-\epsilon)^m \tag{13}$$

but this must be true of all of the hypotheses in *H*,

$$\Pr\left[\exists h \in H : \hat{R}(h) = 0 \land R(h) > \epsilon\right] \leq |H|(1-\epsilon)^m \tag{14}$$

$$\begin{aligned}
|H|(1-\epsilon)^m \leq |H|e^{-m\epsilon} &= \delta \\
\ln \delta &= \ln |H| - m\epsilon \\
-\ln \frac{1}{\delta} - \ln |H| &= -m\epsilon \\
\frac{1}{\epsilon}\left(\ln |H| + \ln \frac{1}{\delta}\right) &= m
\end{aligned}$$

Divide by $-\epsilon$

But what does it all mean?

$$m \geq \frac{1}{\epsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) \tag{15}$$

- **Confidence**
- **Complexity**

But what does it all mean?

$$m \geq \frac{1}{\epsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) \tag{15}$$

- **Confidence**: More certainty means more training data
- **Complexity**

But what does it all mean?

$$m \geq \frac{1}{\epsilon}\left(\ln |H| + \ln \frac{1}{\delta}\right) \tag{15}$$

- **Confidence**: More certainty means more training data
- **Complexity**: More complicated hypotheses need more training data

But what does it all mean?

$$m \geq \frac{1}{\epsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) \quad (15)$$

- **Confidence**: More certainty means more training data
- **Complexity**: More complicated hypotheses need more training data

Scary Question

What's $|H|$ for logistic regression?

What's next . . .

- In class: examples of PAC learnability
- Next time: how to deal with infinite hypothesis spaces

What's next . . .

- In class: examples of PAC learnability
- Next time: how to deal with infinite hypothesis spaces
- Takeaway
  - Even though we can't prove anything about logistic regression, it still works
  - However, using the theory will lead us to a better classification technique: support vector machines