"The mind is a neural computer, fitted by natural selection with combinatorial algorithms for causal and probabilistic reasoning about plants, animals, objects, and people."

...

"In a universe with any regularities at all, decisions informed about the past are better than decisions made at random. That has always been true, and we would expect organisms, especially informavores such as humans, to have evolved acute intuitions about probability. The founders of probability, like the founders of logic, assumed they were just formalizing common sense."

Steven Pinker, How the Mind Works, 1997, pp. 524, 343.

Learning

Learning Overview

Supervised Learning

Learning

Learning is the ability to improve one's behavior based on experience.

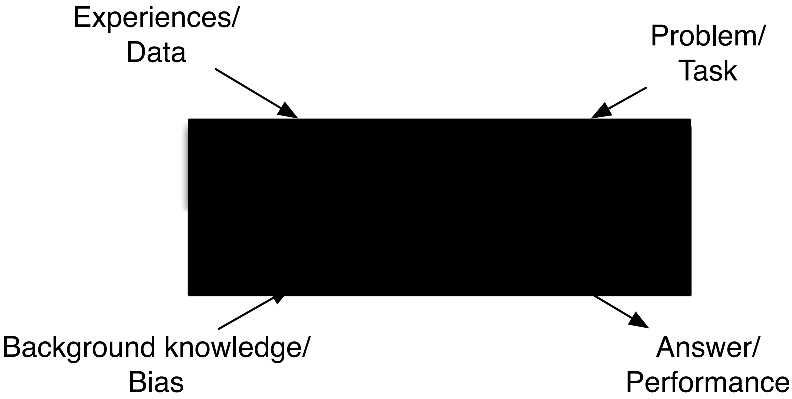
- The range of behaviors is expanded: the agent can do more.
- The accuracy on tasks is improved: the agent can do things better.
- The speed is improved: the agent can do things faster.

Components of a learning problem

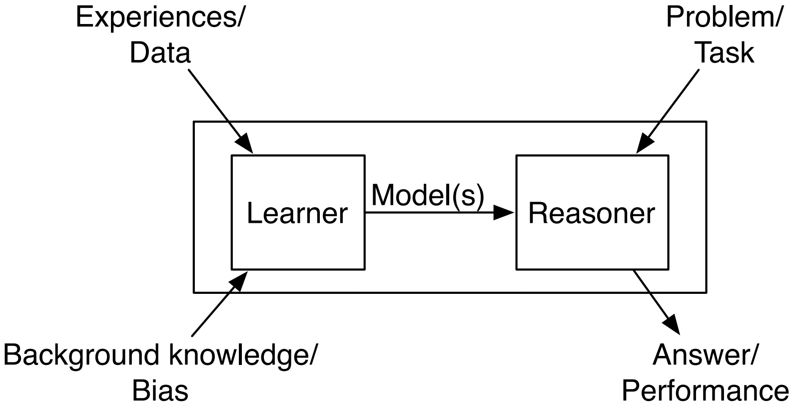
The following components are part of any learning problem:

- task The behavior or task that's being improved.
  For example: classification, acting in an environment
- data The experiences that are being used to improve performance in the task.
- measure of improvement How can the improvement be measured?
  For example: increasing accuracy in prediction, new skills that were not present initially, improved speed.

Black-box Learner



Learning architecture



Common Learning Tasks

- Supervised classification Given a set of pre-classified training examples, classify a new instance.
- Unsupervised learning Find natural classes for examples.
- Reinforcement learning Determine what to do based on rewards and punishments.
- Analytic learning Reason faster using experience.
- Inductive logic programming Build richer models in terms of logic programs.
- Statistical relational learning learning relational representations that also deal with uncertainty.

Example Classification Data

Training Examples:

	Action	Author	Thread	Length	Where
e1	skips	known	new	long	home
e2	reads	unknown	new	short	work
e3	skips	unknown	old	long	work
e4	skips	known	old	long	home
e5	reads	known	new	short	home
e6	skips	known	old	long	work

New Examples:

e7	???	known	new	short	work
e8	???	unknown	new	short	work

We want to classify new examples on feature *Action* based on the examples' *Author*, *Thread*, *Length*, and *Where*.

Feedback

Learning tasks can be characterized by the feedback given to the learner.

- Supervised learning What has to be learned is specified for each example.
- Unsupervised learning No classifications are given; the learner has to discover categories and regularities in the data.
- Reinforcement learning Feedback occurs after a sequence of actions.

Measuring Success

- The measure of success is not how well the agent performs on the training examples, but how well the agent performs for new examples.

Measuring Success

- The measure of success is not how well the agent performs on the training examples, but how well the agent performs for new examples.
- Consider two agents:
  - P claims the negative examples seen are the only negative examples. Every other instance is positive.
  - N claims the positive examples seen are the only positive examples. Every other instance is negative.

Measuring Success

- The measure of success is not how well the agent performs on the training examples, but how well the agent performs for new examples.
- Consider two agents:
  - P claims the negative examples seen are the only negative examples. Every other instance is positive.
  - N claims the positive examples seen are the only positive examples. Every other instance is negative.
- Both agents correctly classify every training example, but disagree on every other example.

Bias

- The tendency to prefer one hypothesis over another is called a bias.
- Saying a hypothesis is better than N's or P's hypothesis isn't something that's obtained from the data.

Bias

- The tendency to prefer one hypothesis over another is called a bias.
- Saying a hypothesis is better than N's or P's hypothesis isn't something that's obtained from the data.
- To have any inductive process make predictions on unseen data, an agent needs a bias.

Bias

- The tendency to prefer one hypothesis over another is called a bias.
- Saying a hypothesis is better than N's or P's hypothesis isn't something that's obtained from the data.
- To have any inductive process make predictions on unseen data, an agent needs a bias.
- What constitutes a good bias is an empirical question about which biases work best in practice.

Learning as search

- Given a representation, data, and a bias, the problem of learning can be reduced to one of search.

Learning as search

- Given a representation, data, and a bias, the problem of learning can be reduced to one of search.
- Learning is search through the space of possible representations looking for the representation or representations that best fits the data, given the bias.

Learning as search

- Given a representation, data, and a bias, the problem of learning can be reduced to one of search.
- Learning is search through the space of possible representations looking for the representation or representations that best fits the data, given the bias.
- These search spaces are typically prohibitively large for systematic search. E.g., use gradient descent.

Learning as search

- Given a representation, data, and a bias, the problem of learning can be reduced to one of search.
- Learning is search through the space of possible representations looking for the representation or representations that best fits the data, given the bias.
- These search spaces are typically prohibitively large for systematic search. E.g., use gradient descent.
- A learning algorithm is made of a search space, an evaluation function, and a search method.

Data

- Data isn't perfect:
  - the features given are inadequate to predict the classification
  - there are examples with missing features
  - some of the features are assigned the wrong value
  - there isn't enough data to determine the correct hypothesis

Data

- Data isn't perfect:
  - the features given are inadequate to predict the classification
  - there are examples with missing features
  - some of the features are assigned the wrong value
  - there isn't enough data to determine the correct hypothesis
- overfitting occurs when distinctions appear in the training data, but not in the unseen examples.

Errors in learning

Errors in learning are caused by:

- Limited representation (representation bias)

Errors in learning

Errors in learning are caused by:

- Limited representation (representation bias)
- Limited search (search bias)

Errors in learning

Errors in learning are caused by:

- Limited representation (representation bias)
- Limited search (search bias)
- Limited data (variance)

Errors in learning

Errors in learning are caused by:

- Limited representation (representation bias)
- Limited search (search bias)
- Limited data (variance)
- Limited features (noise)

Choosing a representation for models

- The richer the representation, the more useful it is for subsequent problem solving.
- The richer the representation, the more difficult it is to learn.

"bias-variance tradeoff"

Characterizations of Learning

- Find the best representation given the data.
- Delineate the class of consistent representations given the data.
- Find a probability distribution of the representations given the data.

Learning

Learning Overview

Supervised Learning

Supervised Learning

Given:

- a set of inputs features $X_1, \ldots, X_n$
- a set of target features $Y_1, \ldots, Y_k$
- a set of training examples where the values for the input features and the target features are given for each example
- a new example, where only the values for the input features are given

predict the values for the target features for the new example.

Supervised Learning

Given:

- a set of inputs features $X_1, \ldots, X_n$
- a set of target features $Y_1, \ldots, Y_k$
- a set of training examples where the values for the input features and the target features are given for each example
- a new example, where only the values for the input features are given

predict the values for the target features for the new example.

- classification when the Y_i are discrete
- regression when the Y_i are continuous

Example Data Representations

A travel agent wants to predict the preferred length of a trip, which can be from 1 to 6 days. (No input features).

Example Data Representations

A travel agent wants to predict the preferred length of a trip, which can be from 1 to 6 days. (No input features).

Two representations of the same data:

— Y is the length of trip chosen.

— Each Y_i is an indicator variable that has value 1 if the chosen length is i, and is 0 otherwise.

Example	Y
e_1	1
e_2	6
e_3	6
e_4	2
e_5	1

Example Data Representations

A travel agent wants to predict the preferred length of a trip, which can be from 1 to 6 days. (No input features).

Two representations of the same data:

— Y is the length of trip chosen.

— Each Y_i is an indicator variable that has value 1 if the chosen length is i, and is 0 otherwise.

Example	Y
e_1	1
e_2	6
e_3	6
e_4	2
e_5	1

Example	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6
e_1	1	0	0	0	0	0
e_2	0	0	0	0	0	1
e_3	0	0	0	0	0	1
e_4	0	1	0	0	0	0
e_5	1	0	0	0	0	0

What is a prediction?

Evaluating Predictions

Suppose we want to make a prediction of a value for a target feature on example e:

- o_e is the observed value of target feature on example e.
- p_e is the predicted value of target feature on example e.
- The error of the prediction is a measure of how close p_e is to o_e.
- There are many possible errors that could be measured.

Sometimes p_e can be a real number even though o_e can only have a few values.

Measures of error

E is the set of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- absolute error $L_1(E) = \sum_{e \in E} |o_e - p_e|$

Measures of error

E is the set of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- absolute error $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- sum of squares error $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$

Measures of error

E is the set of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- absolute error $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- sum of squares error $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- worst-case error: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$

Measures of error

E is the set of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- absolute error $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- sum of squares error $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- worst-case error: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$
- number wrong: $L_0(E) = \#\{e : o_e \neq p_e\}$

Measures of error

E is the set of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- absolute error $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- sum of squares error $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- worst-case error: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$
- number wrong: $L_0(E) = \#\{e : o_e \neq p_e\}$
- A cost-based error takes into account costs of errors.

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- likelihood of the data

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- likelihood of the data

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

- log likelihood

$$\sum_{e \in E} (o_e \log p_e + (1 - o_e) \log(1 - p_e))$$

log loss is the negative of log likelihood.

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- likelihood of the data

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

- log likelihood

$$\sum_{e \in E} (o_e \log p_e + (1 - o_e) \log(1 - p_e))$$

log loss is the negative of log likelihood.
in terms of bits:

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- likelihood of the data

$$\prod_{e \in E} p_e^{o_e}(1 - p_e)^{(1-o_e)}$$

- log likelihood

$$\sum_{e \in E} (o_e \log p_e + (1 - o_e) \log(1 - p_e))$$

  log loss is the negative of log likelihood.
  in terms of bits: negative of number of bits to encode the data given a code based on p_e.

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish 2 items

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items
- n items can be distinguished using

Information theory overview

- A bit is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items
- n items can be distinguished using $\log_2 n$ bits
- Can we do better?

Information and Probability

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111

The string *aacabbda* has code

Information and Probability

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111

The string *aacabbda* has code 00110010101110.
The code 0111110010100 represents string

Information and Probability

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111

The string *aacabbda* has code 00110010101110.
The code 0111110010100 represents string *adcabba*

Information and Probability

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111

The string *aacabbda* has code 00110010101110.
The code 0111110010100 represents string *adcabba*
This code uses 1 to 3 bits. On average, it uses

Information and Probability

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111

The string *aacabbda* has code 00110010101110.
The code 0111110010100 represents string *adcabba*
This code uses 1 to 3 bits. On average, it uses

$$\begin{aligned} & P(a) \times 1 + P(b) \times 2 + P(c) \times 3 + P(d) \times 3 \\ = \; & \frac{1}{2} + \frac{2}{4} + \frac{3}{8} + \frac{3}{8} = 1\frac{3}{4} \text{ bits.} \end{aligned}$$

Information Content

- To identify x, we need $-\log_2 P(x)$ bits.
- Give a distribution over a set, to a identify a member, the expected number of bits

$$\sum_x -P(x) \times \log_2 P(x).$$

is the information content or entropy of the distribution.

Information Content

- To identify x, we need $-\log_2 P(x)$ bits.
- Give a distribution over a set, to a identify a member, the expected number of bits

$$\sum_x -P(x) \times \log_2 P(x).$$

is the information content or entropy of the distribution.

- The expected number of bits it takes to describe a distribution given evidence e:

$$I(e) = \sum_x -P(x|e) \times \log_2 P(x|e).$$

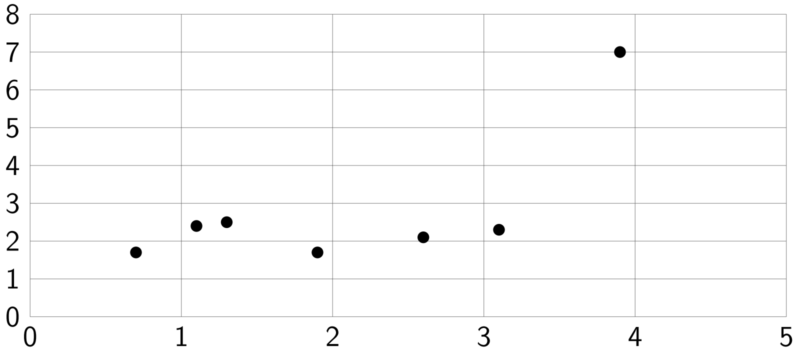
Information Gain

Given a test that can distinguish the cases where α is true from the cases where α is false, the information gain from this test is:

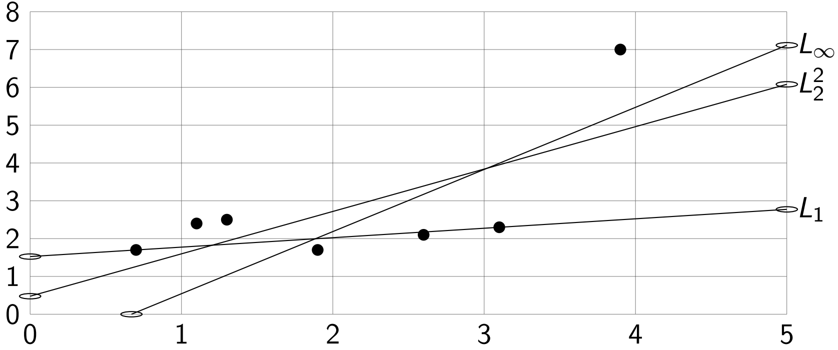
$$I(true) - (P(\alpha) \times I(\alpha) + P(\neg\alpha) \times I(\neg\alpha)).$$

- $I(true)$ is the expected number of bits needed before the test
- $P(\alpha) \times I(\alpha) + P(\neg\alpha) \times I(\neg\alpha)$ is the expected number of bits after the test.

Linear Predictions



Linear Predictions



Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical probability.

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical probability.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical probability.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is the empirical probability.

Point Estimates

To make a single prediction for feature Y, with examples E.

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y.
- The prediction that minimizes the absolute error on E is the median value of Y.
- The prediction that minimizes the number wrong on E is the mode of Y.
- The prediction that minimizes the worst-case error on E is $(maximum + minimum)/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical probability.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is the empirical probability.

But that doesn't mean that these predictions minimize the error for future predictions....

Training and Test Sets

To evaluate how well a learner will work on future predictions, we divide the examples into:

- training examples that are used to train the learner
- test examples that are used to evaluate the learner

...these must be kept separate.

Simplest case of learning

To predict the value of a Boolean variable X from data.

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.
- Generate n samples from the distribution with $P(X) = p$. Observe n_0 samples with $X = false$, and n_1 samples with $X = true$, so $n = n_0 + n_1$.

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.
- Generate n samples from the distribution with $P(X) = p$. Observe n_0 samples with $X = \mathit{false}$, and n_1 samples with $X = \mathit{true}$, so $n = n_0 + n_1$.
- Which predictor is best on test cases (other cases sampled from $P(X) = p$)?

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.
- Generate n samples from the distribution with $P(X) = p$. Observe n_0 samples with $X = false$, and n_1 samples with $X = true$, so $n = n_0 + n_1$.
- Which predictor is best on test cases (other cases sampled from $P(X) = p$)? When error is
  - absolute error?

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.
- Generate n samples from the distribution with $P(X) = p$. Observe n_0 samples with $X = false$, and n_1 samples with $X = true$, so $n = n_0 + n_1$.
- Which predictor is best on test cases (other cases sampled from $P(X) = p$)? When error is
  - absolute error?
  - sum-of-squares error?

Simplest case of learning

To predict the value of a Boolean variable X from data.
Consider the following scenario:

- Pick random number $p \in [0, 1]$. This will be the ground truth of $P(X)$.
- Generate n samples from the distribution with $P(X) = p$. Observe n_0 samples with $X = \mathit{false}$, and n_1 samples with $X = \mathit{true}$, so $n = n_0 + n_1$.
- Which predictor is best on test cases (other cases sampled from $P(X) = p$)? When error is
  - absolute error?
  - sum-of-squares error?
  - log loss?