Large Scale Hypothesis Testing

CSE545 - Spring 2022

Stony Brook University

H. Andrew Schwartz

Big Data Analytics, The Class

Goal: Generalizations
A *model* or *summarization* of the data.

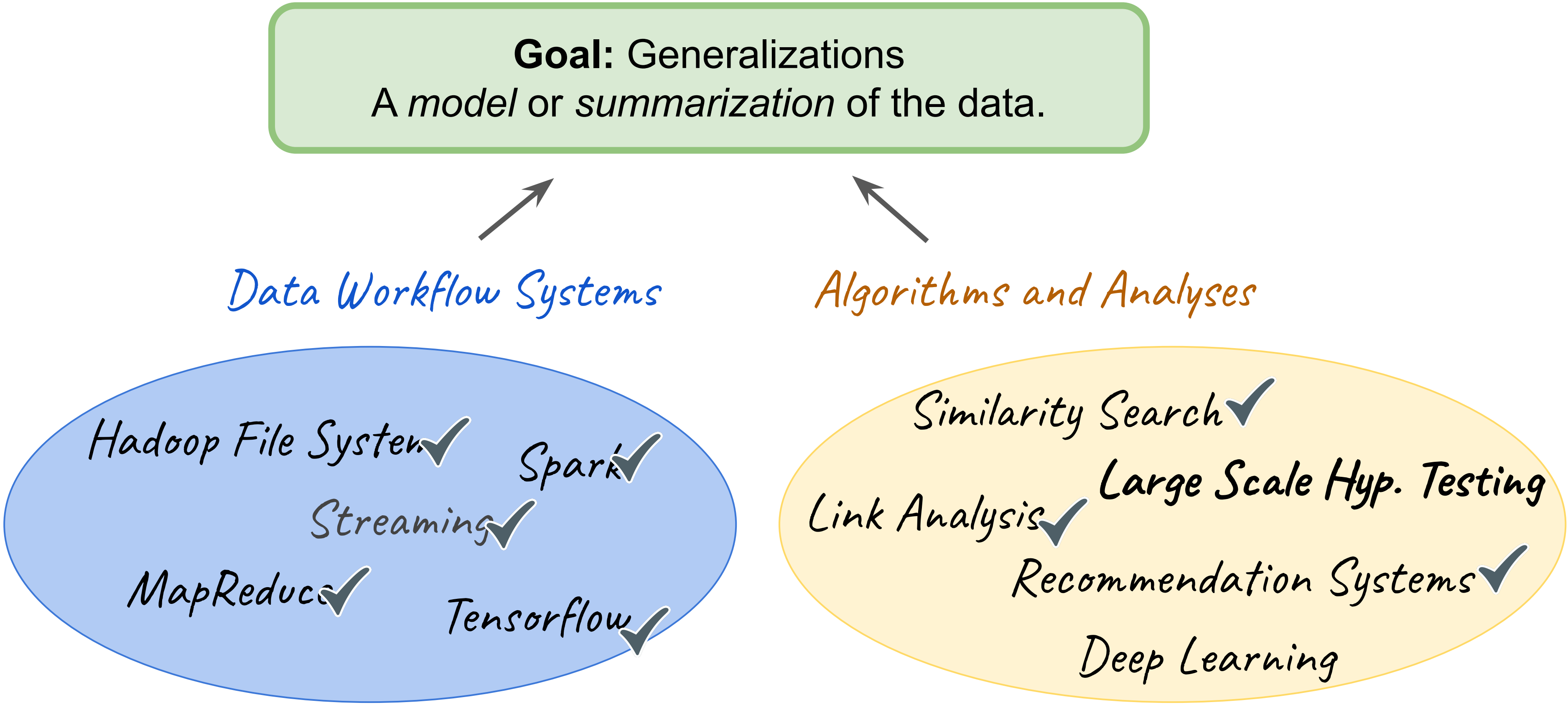
Data Workflow Systems

Algorithms and Analyses

Hadoop File System ✓
Spark ✓
Streaming ✓
MapReduce ✓
Tensorflow ✓

Similarity Search ✓
Large Scale Hyp. Testing
Link Analysis ✓
Recommendation Systems
Deep Learning

Big Data Analytics, The Class



Goal of Data Science

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

The Data Whisperer

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Goal of Data Science

DATA

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Data-driven (evidence-based) **decision**

Goal of Data Science

DATA

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Discrete **Finding(s)**
$\mathcal{F}$ is (likely) True

Data-driven (evidence-based) **decision**

Goal of Data Science

DATA

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Discrete **Finding(s)**
$\mathcal{F}$ is (likely) True

Data-driven (evidence-based) **decision**

Blue cell phones cases are selling the most.

The ResImageGenNet model is most accurate.

Those >70 have a greater mortality rate from the viral infection.

Goal of Data Science

DATA

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Discrete **Finding(s)**
$\mathcal{F}$ is (likely) True

Data-driven (evidence-based) **decision**

Hypotheses!
Potential findings -- to be tested for happenstance.

Blue cell phones cases are selling the most.

The ResImageGenNet model is most accurate.

Those >70 have a greater mortality rate from the viral infection.

The Data Whisperer

Hypotheses!
Potential findings -- to be tested for happenstance.

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

The Data Whisperer

Hypotheses!
Potential findings -- to be tested for happenstance.

Goal: Generalizations
A ***model*** or ***summarization*** of the data.

Hypothesis Testing

Also known as… "Don't be Dilbert's Boss!"

Hypothesis -- something one asserts to be true.

Hypothesis Testing

Also known as… “Don’t be Dilbert’s Boss!”

Hypothesis -- something one asserts to be true.

Formally, two types:

H_0: *null hypothesis* -- some “default” value; “null”: nothing changes

H_1: the alternative -- the opposite of the null => a change or difference

Hypothesis Testing

H_0: null hypothesis -- some "default" value; "null": nothing changes

H_1: the alternative -- the opposite of the null => a change or difference

Goal: Make sure what we observed was unlikely to happen by chance.

Thus, we want to know:

Given null, what is the probability of the observation or worse

Hypothesis Testing

H_0: null hypothesis -- some "default" value; "null": nothing changes

H_1: the alternative -- the opposite of the null => a change or difference

Goal: Make sure what we observed was unlikely to happen by chance.

Thus, we want to know:

Given null, what is the probability of the observation or worse?

-> If low enough, then we "reject the null (H_0) in favor of H_1."

Hypothesis Testing

H_0: null hypothesis -- some "default" value; "null": nothing changes

H_1: the alternative -- the opposite of the null => a change or difference

Goal: Make sure what we observed was unlikely to happen by chance.

Thus, we want to know:

Given null, what is the probability of the observation or worse?

-> If <u>low enough</u>, then we "reject the null (H_0) in favor of H_1."

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

observations (i.e. data)

level of significance

Input: H_0, obs, α

Output: decision

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

observations (i.e. data)

level of significance

```
Input: H0, obs, α
```

probability of what we observed or worse (i.e. more extreme)

```
    p(x>=obs | H0) < α
```

```
Output: decision
```

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

```
Input: H₀, obs, α




if p(x>=obs | H₀) < α:
    decision = "Reject H₀!"
else:
    decision = "Accept H₀."
Output: decision
```

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α
```

Conditional is sometimes evaluated indirectly by first finding the "critical value" of some measurement such that:
if measurement > critical_value then p(obs|H0) < α

```
if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

```
Input: H₀, obs, α




if p(x>=obs | H₀) < α:
    decision = "Reject H₀!"
else:
    decision = "Accept H₀."
Output: decision
```

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"

Input: H_0, obs, α

Need to estimate

What is the distribution of values we would expect if the null was true?
-- the "null distribution"

```
if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

H_0: The blue case is not selling more than average.

Probability Distributions: Review

X: A mapping from **Ω** to ℝ that describes the question we care about in practice.

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *discrete random variable* if it takes only a countable number of values.

Probability Distributions: Review

X: A mapping from **Ω** to ℝ that describes the question we care about in practice.

↓

"sample space", set of all possible outcomes.

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *discrete random variable* if it takes only a countable number of values.

Probability Distributions: Review

X: A mapping from **Ω** to ℝ that describes the question we care about in practice.

"sample space", set of all possible outcomes.

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *discrete random variable* if it takes only a countable number of values.

Error of RedImageGenNet Classifier

Amount of sales of a blue case

Continuous Distributions

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *continuous random variable* if there exists a function *fx* such that:

$$f_X(x) \geq 0, \text{for all } x \in X,$$

$$\int_{-\infty}^{\infty} f_X(x)dx = 1, \quad \text{and}$$

$$\mathrm{P}(a < X < b) = \int_a^b f_X(x)dx$$

Continuous Distributions

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *continuous random variable* if there exists a function *fx* such that:

$$f_X(x) \geq 0, \text{ for all } x \in X,$$

$$\int_{-\infty}^{\infty} f_X(x)dx = 1, \quad \text{and}$$

$$\mathrm{P}(a < X < b) = \int_a^b f_X(x)dx$$

***fx* : "probability density function" (pdf)**

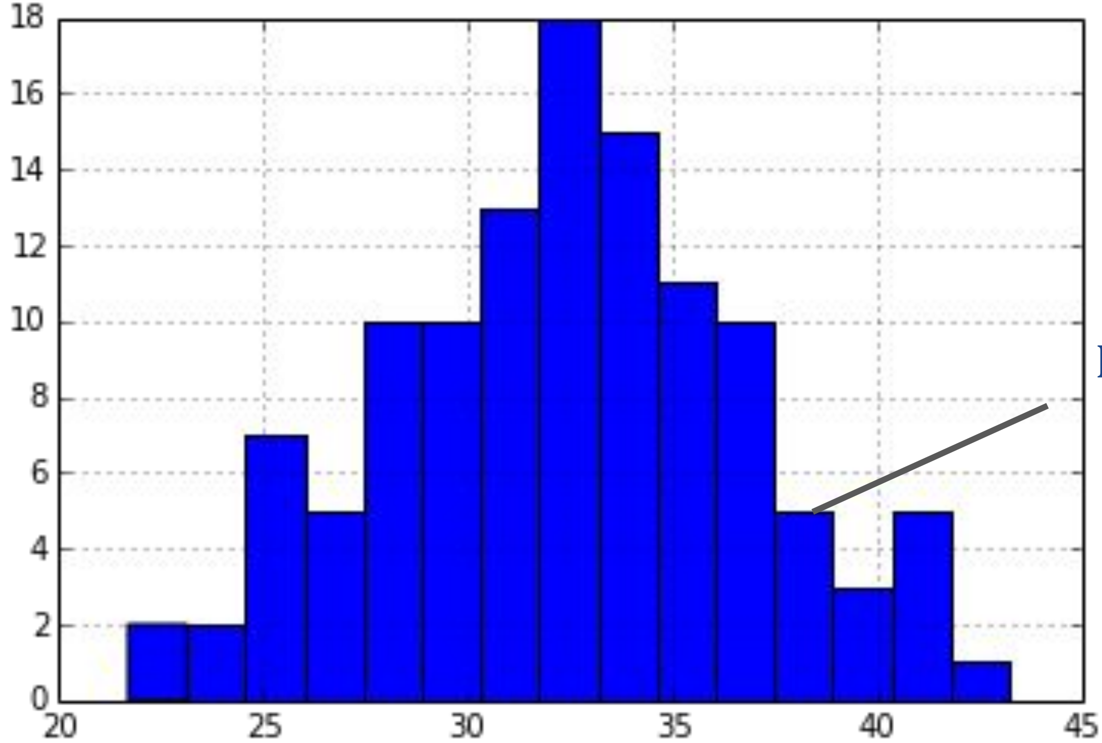
Continuous Distributions

Discretize them!
(group into discrete bins)

How to model?

Continuous Distributions

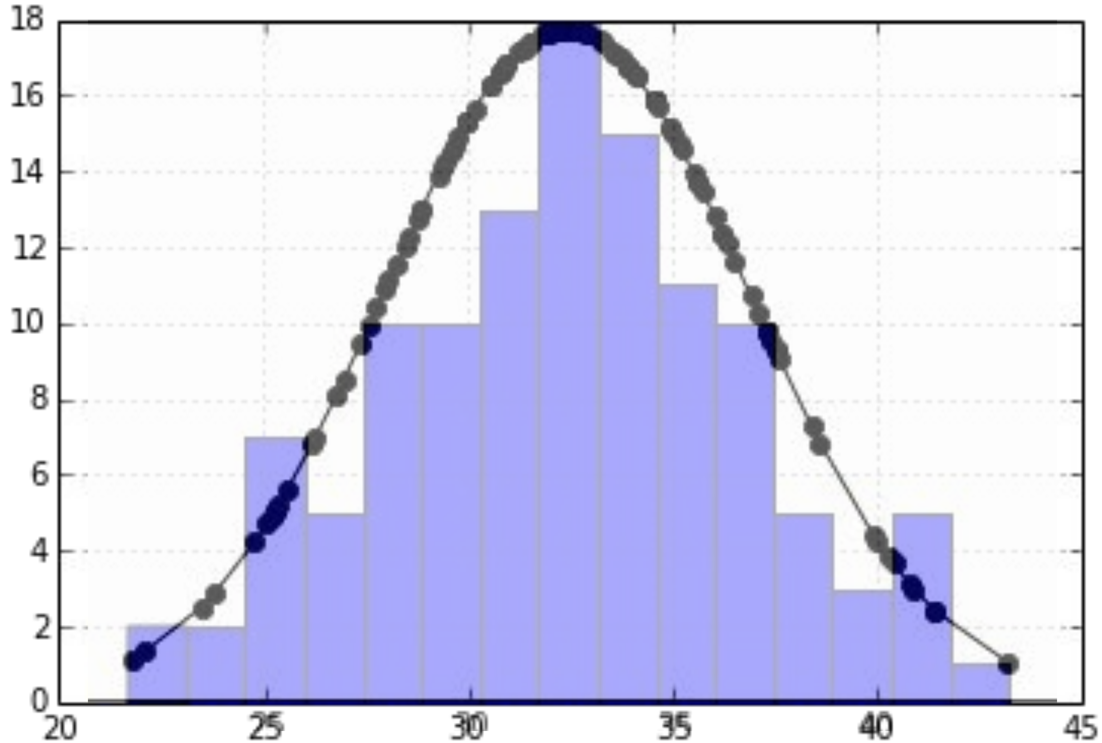
$P(bin=8) = .32$



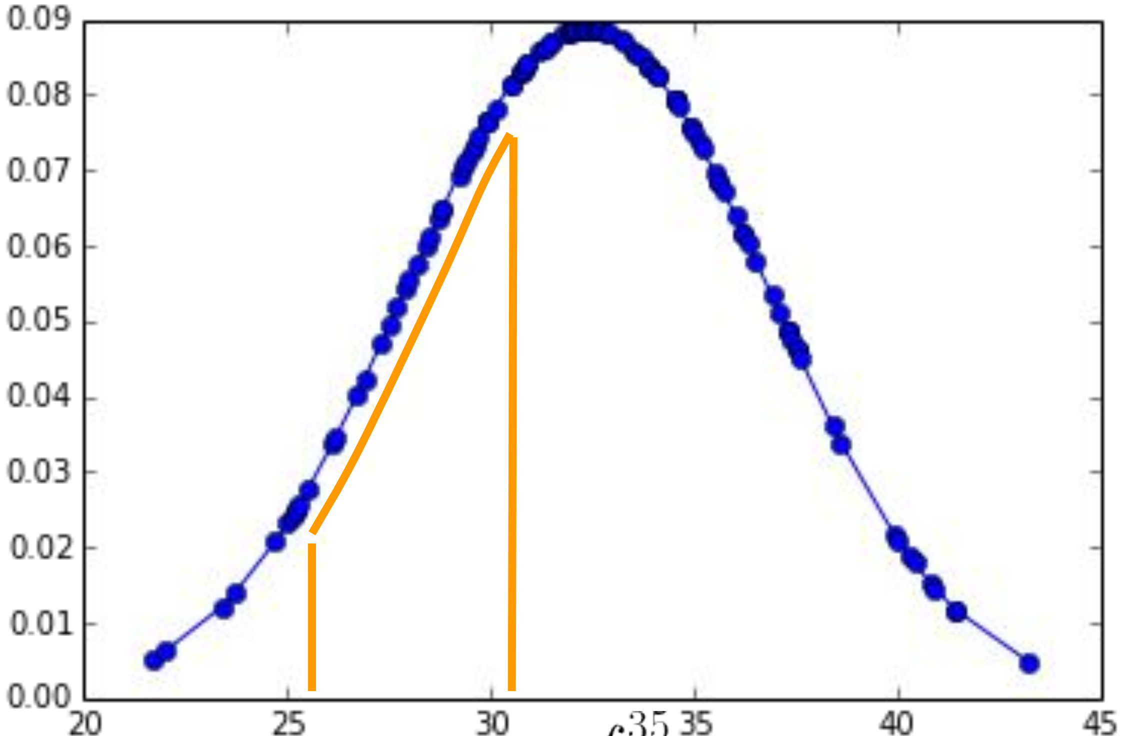
$P(bin=12) = .08$

But aren't we throwing away information?

Continuous Distributions



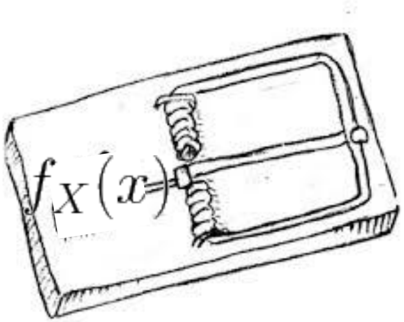
Continuous Distributions



$$P(25 < X < 35) = \int_{25}^{35} fx(x)dx$$

Continuous Distributions

Common Trap

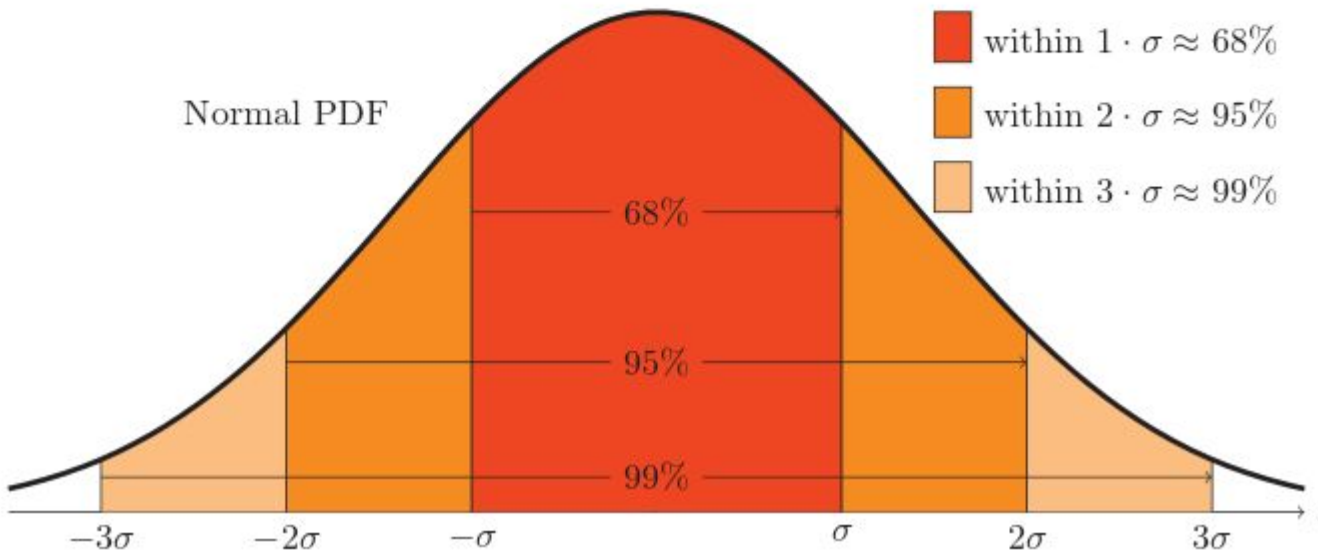


- $f_X(x)$ does not yield a probability
  - $\int_a^b f_X(x)dx$ does
  - x may be anything ($\mathbb{R}$)
    - thus, $f_X(x)$ may be > 1

Continuous Distributions

Common *pdfs*: Normal(0, 1)

$$P(-1 \le Z \le 1) \approx .68, \quad P(-2 \le Z \le 2) \approx .95, \quad P(-3 \le Z \le 3) \approx .99$$



Credit: MIT Open Courseware: Probability and Statistics

Continuous Distributions

Common *pdfs*: Normal(0, 1) ("standard normal")

How to "standardize" any normal distribution:

1. subtract the mean, μ (aka "mean centering")
2. divide by the standard deviation, σ

$z = (x - \mu) / \sigma$, (aka "z score")

Probability Distributions: Review

X: A mapping from **Ω** to ℝ that describes the question we care about in practice.

"sample space", set of all possible outcomes.

X is a *continuous random variable* if it can take on an infinite number of values between any two given values.

X is a *discrete random variable* if it takes only a countable number of values.

Error of RedImageGenNet Classifier

Amount of sales of a blue case

Discrete Random Variables

For a given *discrete* random variable X,
probability mass function (*pmf*),
fx: $\mathbb{R} \rightarrow [0, 1]$, is defined by:

$$f_X(x) = \mathrm{P}(X = x)$$

X is a *discrete random variable* if it takes only a countable number of values.

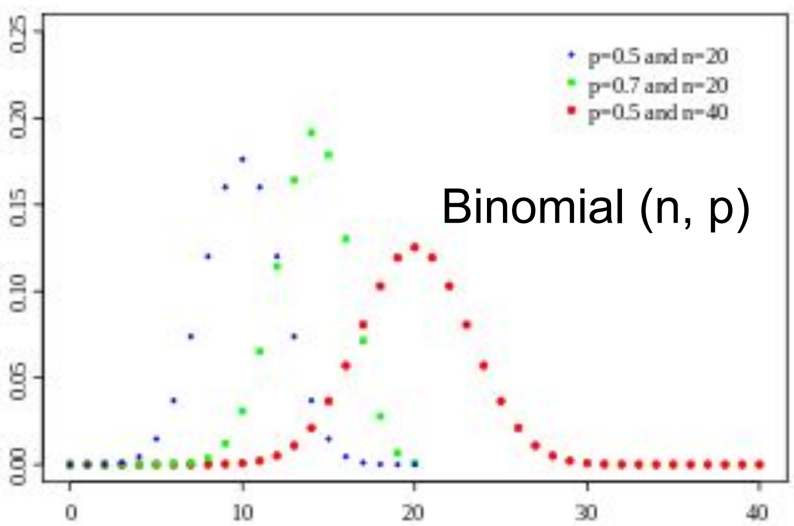
Amount of sales of a blue case

Was a single sale a blue case: {0, 1}

Discrete Random Variables

For a given *discrete* random variable X, *probability mass function* (*pmf*), *fx*: $\mathbb{R} \to [0, 1]$, is defined by:

$$f_X(x) = \mathrm{P}(X = x)$$



X is a *discrete random variable* if it takes only a countable number of values.

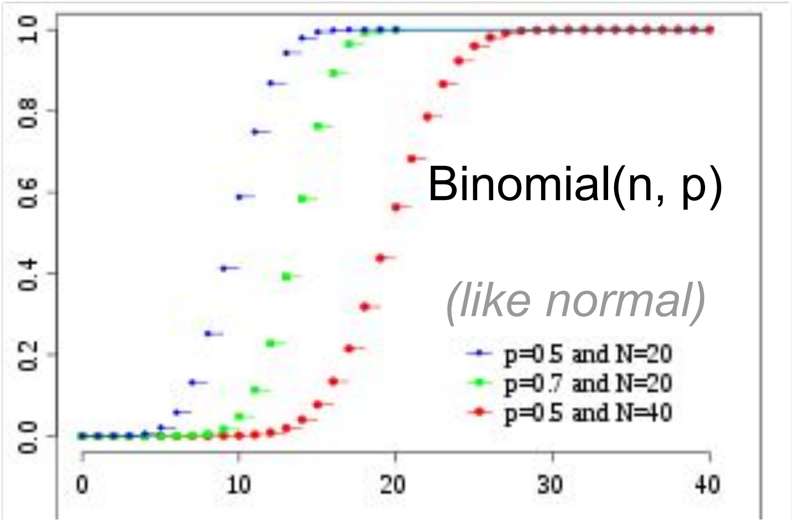
Amount of sales of a blue case

Was a single sale a blue case: {0, 1}

Discrete Random Variables

For a given random variable X, the *cumulative distribution function* (CDF), *Fx:* $\mathbb{R} \to [0, 1]$, is defined by:

$$F_X(x) = \mathrm{P}(X \leq x)$$



X is a *discrete random variable* if it takes only a countable number of values.

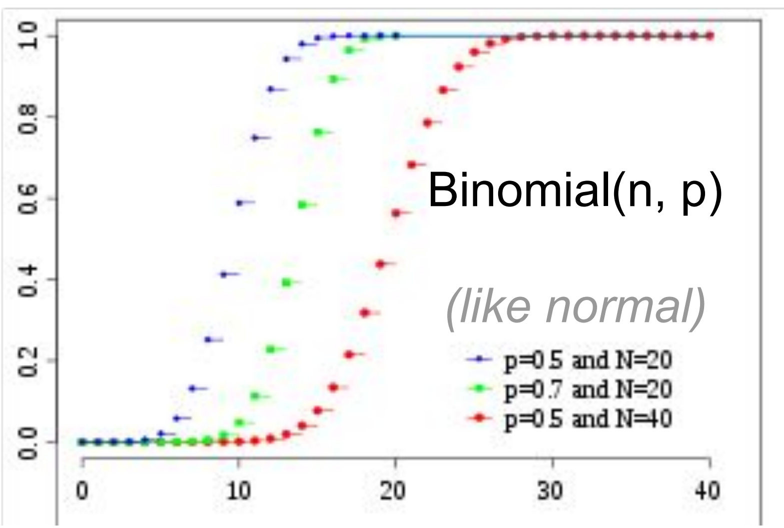
Amount of sales of a blue case

Was a single sale a blue case: {0, 1}

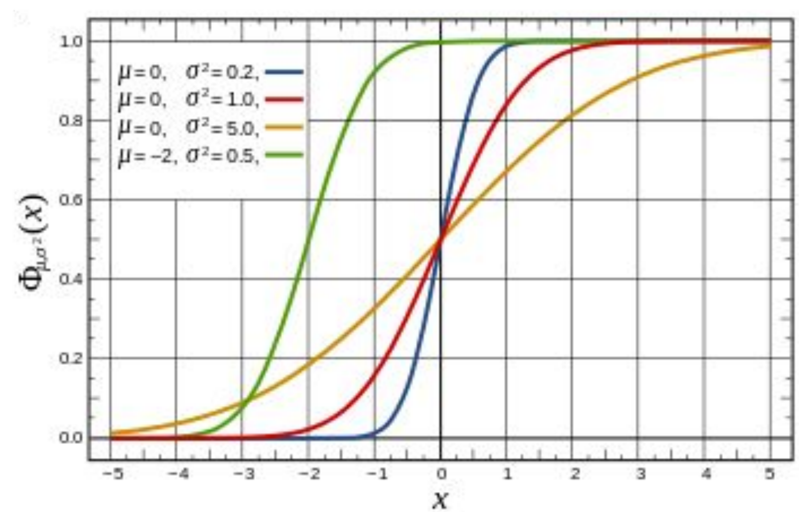
Discrete Random Variables

For a given random variable X, the *cumulative distribution function* (CDF), *Fx:* $\mathbb{R} \to [0, 1]$, is defined by:

$$F_X(x) = \mathrm{P}(X \leq x)$$



Normal



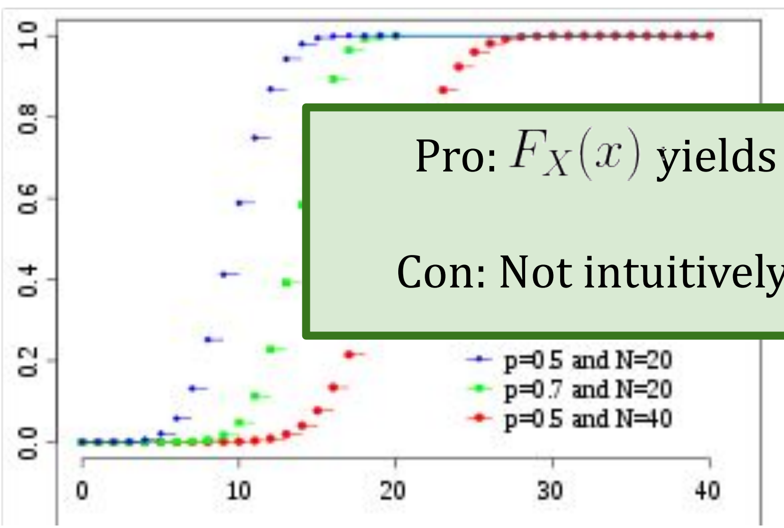
Discrete Random Variables

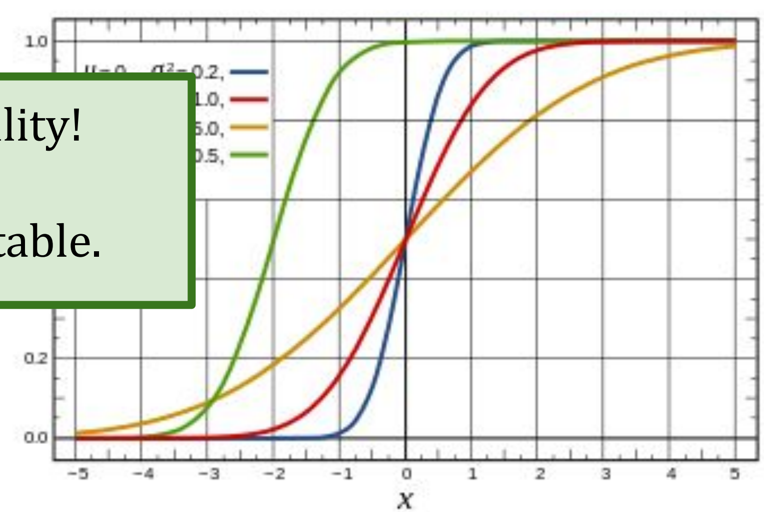
For a given random variable X, the *cumulative distribution function* (CDF), *Fx:* $\mathbb{R} \rightarrow [0, 1]$, is defined by:

$$F_X(x) = \mathrm{P}(X \leq x)$$

Normal

⇩





Pro: $F_X(x)$ yields a probability!

Con: Not intuitively interpretable.

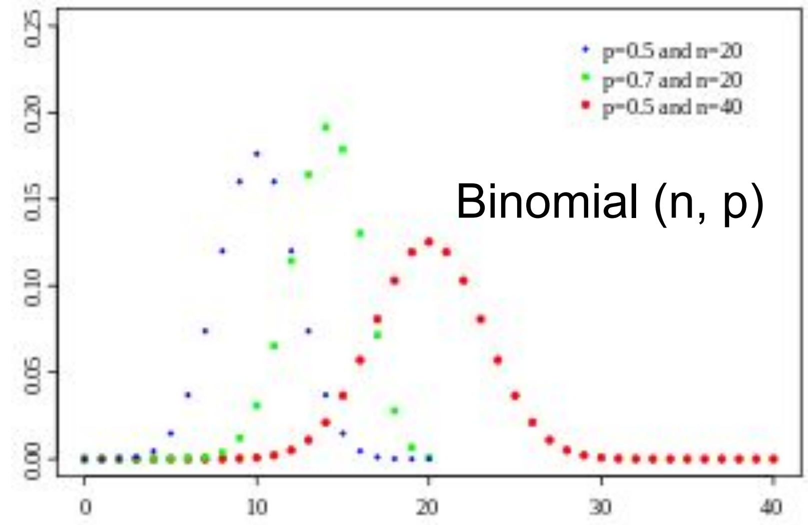
Discrete RVs

For a given random variable X, the *cumulative distribution function* (CDF), *Fx:* $\mathbb{R} \rightarrow [0, 1]$, is defined by:

$$F_X(x) = \mathrm{P}(X \leq x)$$

For a given *discrete* random variable X, *probability mass function* (*pmf*), *fx:* $\mathbb{R} \rightarrow [0, 1]$, is defined by:

$$f_X(x) = \mathrm{P}(X = x)$$



X is a *discrete random variable* if it takes only a countable number of values.

$$\sum_i f_X(x) = 1$$

$$F_X(x) = \mathrm{P}(X \leq x) = \sum_{x_i \leq x} f_X(x)$$

The Hypothesis Test "Algorithm"

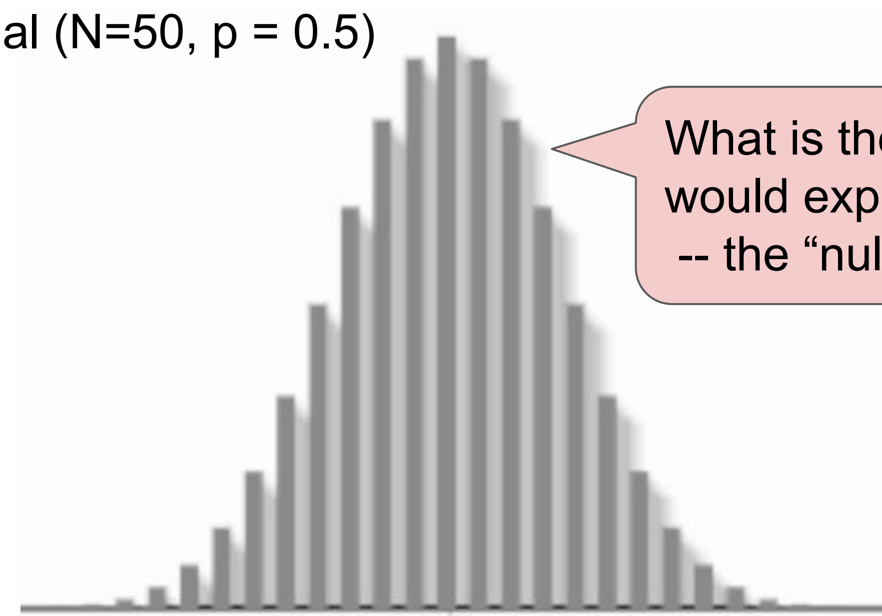
```
Input: H0, observations, α

if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

Need to estimate

What is the distribution of values we would expect if the null was true?
-- the "null distribution"

H_0: The blue case is not selling more than average.

The Hypothesis Test "Algorithm"



What is the distribution of values we would expect if the null was true?
-- the "null distribution"

H_0: The blue case is not selling more than average.

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H₀, obs, α

null_dist = distribution of expected values under H₀



if p(x>=obs | H₀) < α:
    decision = "Reject H₀!"
else:
    decision = "Accept H₀."
Output: decision
```

H_0: The blue case is not selling more than average.

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

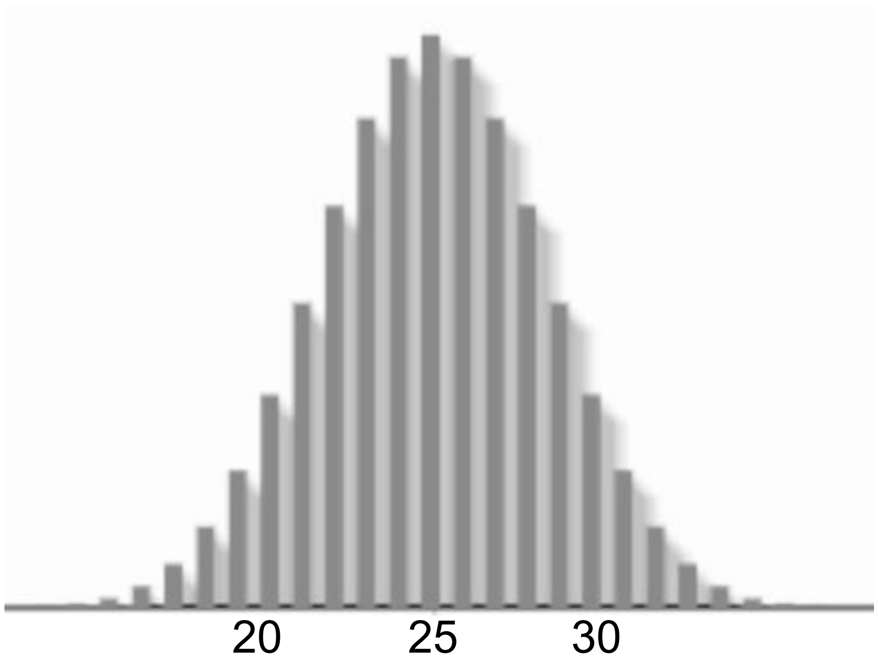
The Hypothesis Test "Algorithm"

```
Input: H_0, obs, α

null_dist = distribution of expected values under H_0




if p(x>=obs | H_0) < α:
    decision = "Reject H_0!"
else:
    decision = "Accept H_0."
Output: decision
```

H_0: The blue case is not selling more than average. **Observed 33 blue sales**

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

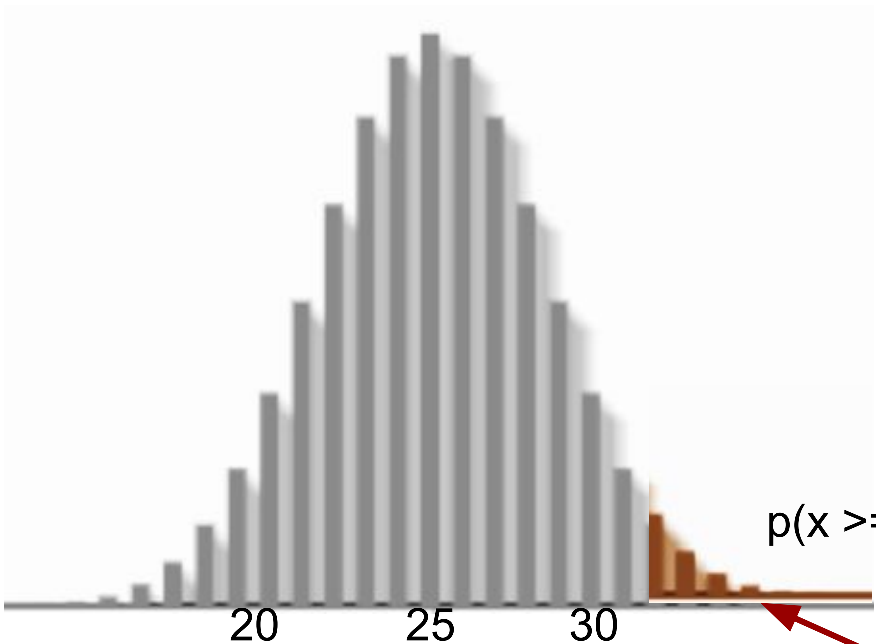
The Hypothesis Test “Algorithm”



H_0: The blue case is not selling more than average. **Observed 33 blue sales**

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"



p(x >= 33; Binomial(50, 0.5)) = 0.016

Observed 33 blue sales

H_0: The blue case is not selling more than average.

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H_0, obs, α

null_dist = distribution of expected values under H_0

p(x>=obs | H_0) =

if p(x>=obs | H_0) < α:
    decision = "Reject H_0!"
else:
    decision = "Accept H_0."
Output: decision
```

H_0: The blue case is not selling more than average. Observed 33 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

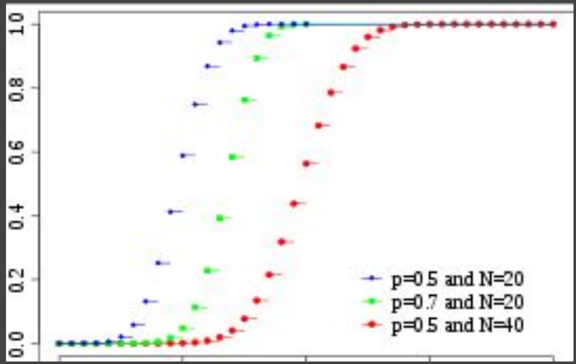
```
Input: H0, obs, α

null_dist = distribution of expected values under H0

p(x>=obs | H0) = sum(pmf(null_dist, o) for o in range(obs,))

if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

H_0: The blue case is not selling more than average. Observed 33 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

null_dist = distribution of expected values under H0

p(x>=obs | H0) = 1 - cdf(null_dist, obs)

if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

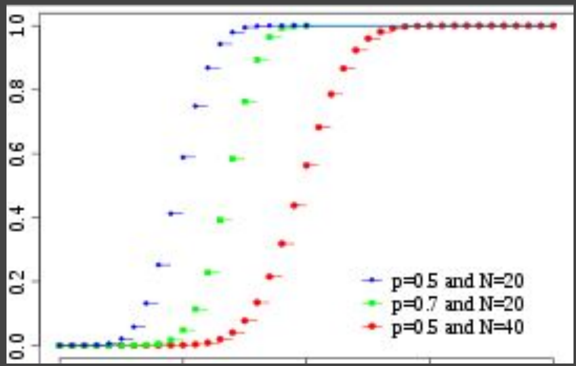


H_0: The blue case is not selling more than average. Observed 33 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

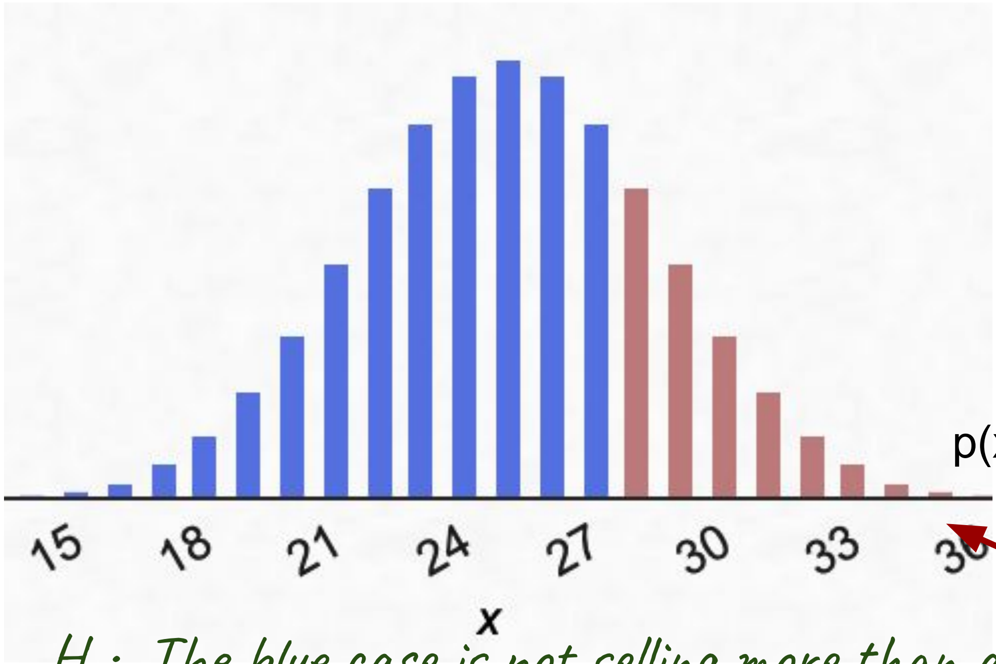
```
Input: H0, obs, α

null_dist = distribution of expected values under H0

p(x>=obs | H0) = 1 - cdf(null_dist, obs) = 0.016

if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```



H_0: The blue case is not selling more than average. **Observed 33 blue sales**

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"



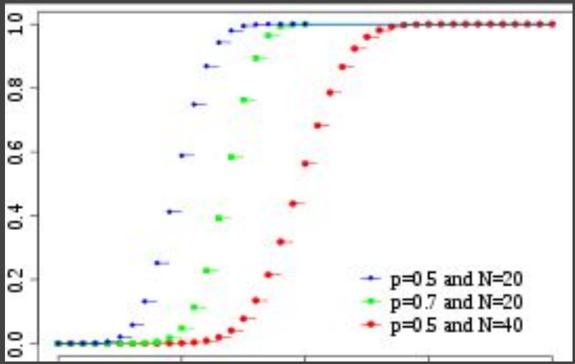
p(x >= 28; Binomial(50, 0.5)) = 0.239

H_0: The blue case is not selling more than average. Observed 28 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

null_dist = distribution of expected values under H0

p(x>=obs | H0) = 1 - cdf(null_dist, obs) = 0.239

if p(x>=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```



H_0: The blue case is not selling more than average. Observed 28 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

null_dist = distribution of expected values under H0

p(x<=obs | H0) = cdf(null_dist, obs)

if p(x<=obs | H0) < α:
    decision = "Reject H0!"
else:
    decision = "Accept H0."
Output: decision
```

H_0: The blue case is not selling **less than** average. Observed 32 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

 null_dist = distribution of expected obs under H0
 p(x<=obs | H0) = cdf(null_dist, obs)
 if p(x<=obs | H0) < α:
    decision = "Reject H0!"
 else:
    decision = "Accept H0."

Output: decision
```

H_0: The blue case is not selling **less than** average. Observed 36 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

 obs_ts = test_stat(obs)
 null_dist = distribution of expected obs under H0
 p(x<=obs | H0) = cdf(null_dist, obs)
 if p(x<=obs | H0) < α:
    decision = "Reject H0!"
 else:
    decision = "Accept H0."

Output: decision
```

H_0: The blue case is not selling **less than** average. Observed 36 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

 obs_ts = test_stat(obs)
 null_dist = distribution of expected test_stat under H0
 p(x<=obs_ts | H0) = cdf(null_dist, obs_ts)
 if p(x<=obs_ts | H0) < α:
    decision = "Reject H0!"
 else:
    decision = "Accept H0."

Output: decision
```

H_0: The blue case is not selling **less than** average. Observed 36 blue sales

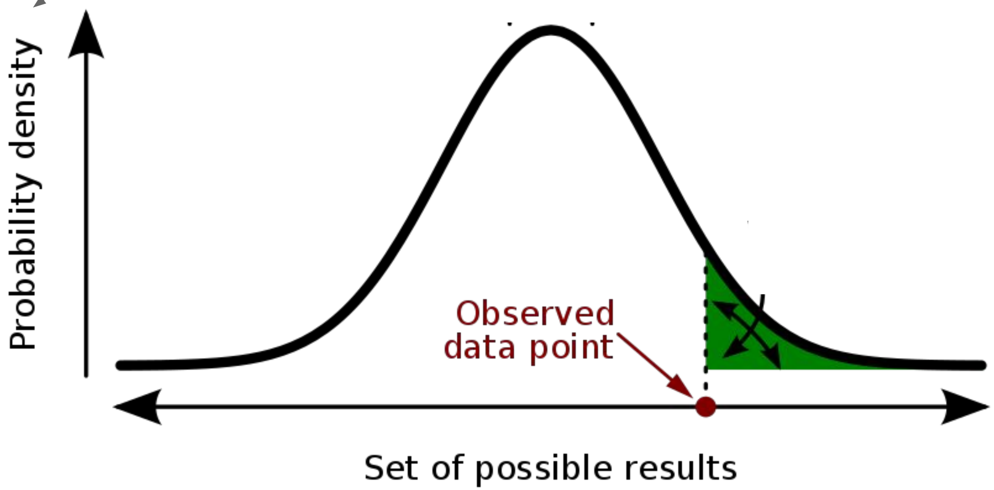
50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

Hypothesis Testing

P(D|H_0): *Given null, what is the probability of the observed data or worse?*

-> If low enough, then we "reject the null (H_0) in favor of H_1."

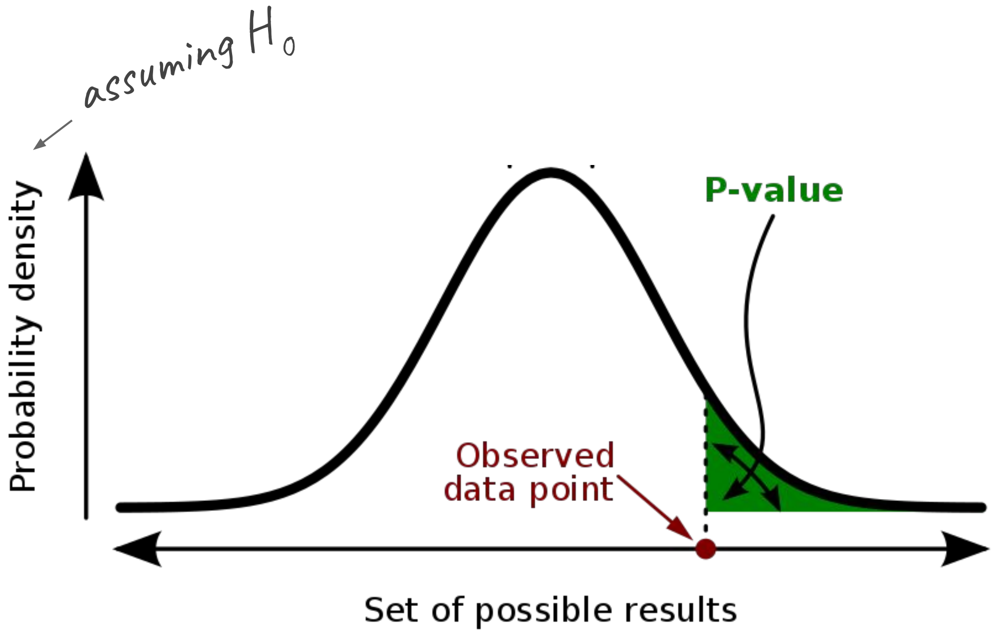
assuming H_0



Hypothesis Testing

P(D|H_0): *Given null, what is the probability of the observed data or worse?*

-> If low enough, then we "reject the null (H_0) in favor of H_1."

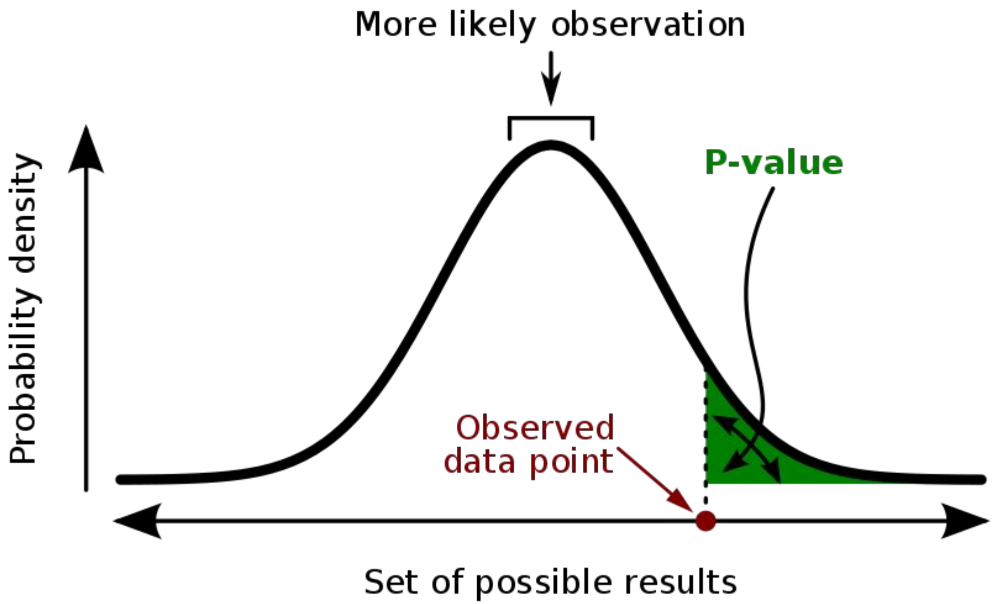


A **p-value** (shaded green area) is the probability of an observed (or more extreme) result assuming that the null hypothesis is true.

Hypothesis Testing

$P(D|H_0)$: *Given null, what is the probability of the observed data or worse?*

-> If low enough, then we "reject the null (H_0) in favor of H_1."

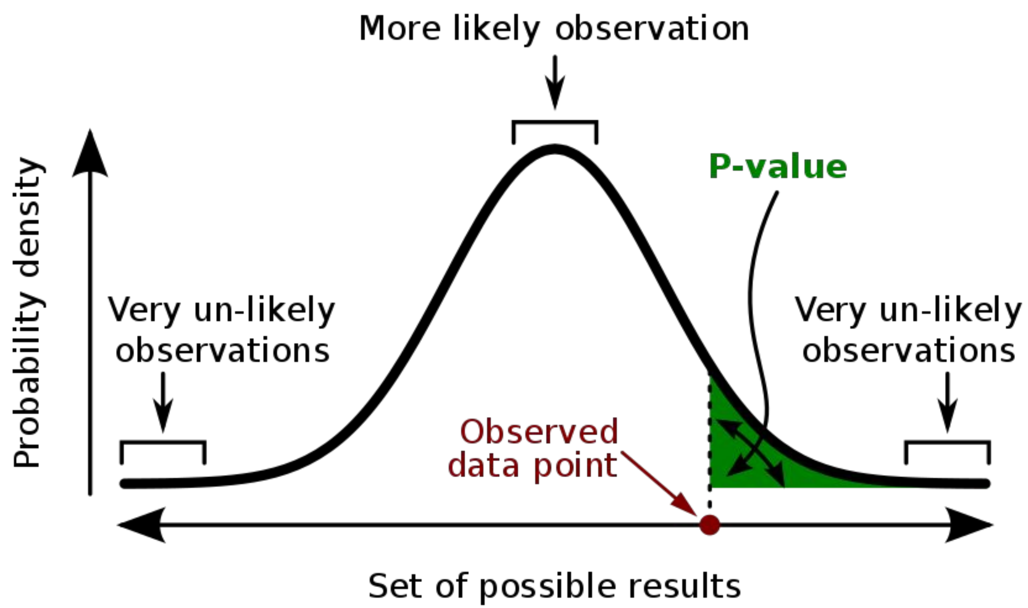


A **p-value** (shaded green area) is the probability of an observed (or more extreme) result assuming that the null hypothesis is true.

Hypothesis Testing

$P(D|H_0)$: *Given null, what is the probability of the observed data or worse?*

-> If low enough, then we "reject the null (H_0) in favor of H_1."



A **p-value** (shaded green area) is the probability of an observed (or more extreme) result assuming that the null hypothesis is true.

(thanks, [Wikipedia]())

The Hypothesis Test "Algorithm"

```
Input: H0, obs, α

 obs_ts = test_stat(obs)
 null_dist = distribution of expected test_stat under H0
 p(x<=obs_ts | H0) = cdf(null_dist, obs_ts)
 if p(x<=obs_ts | H0) < α:
    decision = "Reject H0!"
 else:
    decision = "Accept H0."

Output: decision
```

H_0: The blue case is not selling **less than** average. Observed 36 blue sales

50 sales; 2 colors (blue and red); Thus, average would be 25 blue sales

Hypothesis Testing

Why?

Hypothesis Testing

Why?

A general framework for answering (yes/no) questions!

Hypothesis Testing

Why?

A general framework for answering (yes/no) questions!

- *Are height and baldness related?*
- *Is my deep predictive model better than the state of the art?*

Hypothesis Testing

Why?

A general framework for answering (yes/no) questions!

- *Are height and baldness related?*
- *Is my deep predictive model better than the state of the art?*
- *Is the heat index of a community related to poverty?*
- *Is the heat index of a community related to poverty* ***controlling for education rates?***
- *Does my website receive a higher average number of monthly visitors?*

Hypothesis Testin

Failing to "reject the null" does not mean the null is true.

Why?

A general framework for answering (yes/**maybe**) questions!

- *Are height and baldness related?*
- *Is my deep predictive model better than the state of the art?*
- *Is the heat index of a community related to poverty?*
- *Is the heat index of a community related to poverty **controlling for education rates?***
- *Does my website receive a higher average number of monthly visitors?*

Hypothesis Testin

Failing to "reject the null" does not mean the null is true. However, if the sample is large enough, it may be enough to say that the effect size (correlation, difference value, etc...) is not very meaningful.

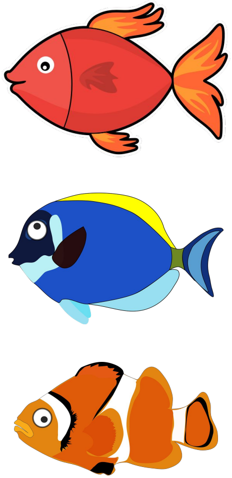
Why?

A general framework for answering (yes/**maybe**) questions!

- *Are height and baldness related?*
- *Is my deep predictive model better than the state of the art?*
- *Is the heat index of a community related to poverty?*
- *Is the heat index of a community related to poverty **controlling for education rates**?*
- *Does my website receive a higher average number of monthly visitors?*

Bonferroni’s Cats

General Question: Which fish do cats like?

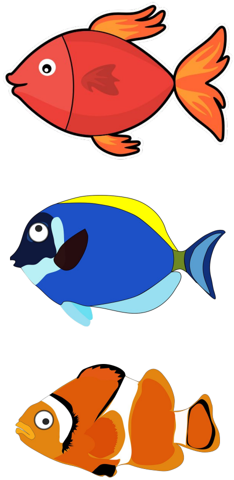


⋮

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats

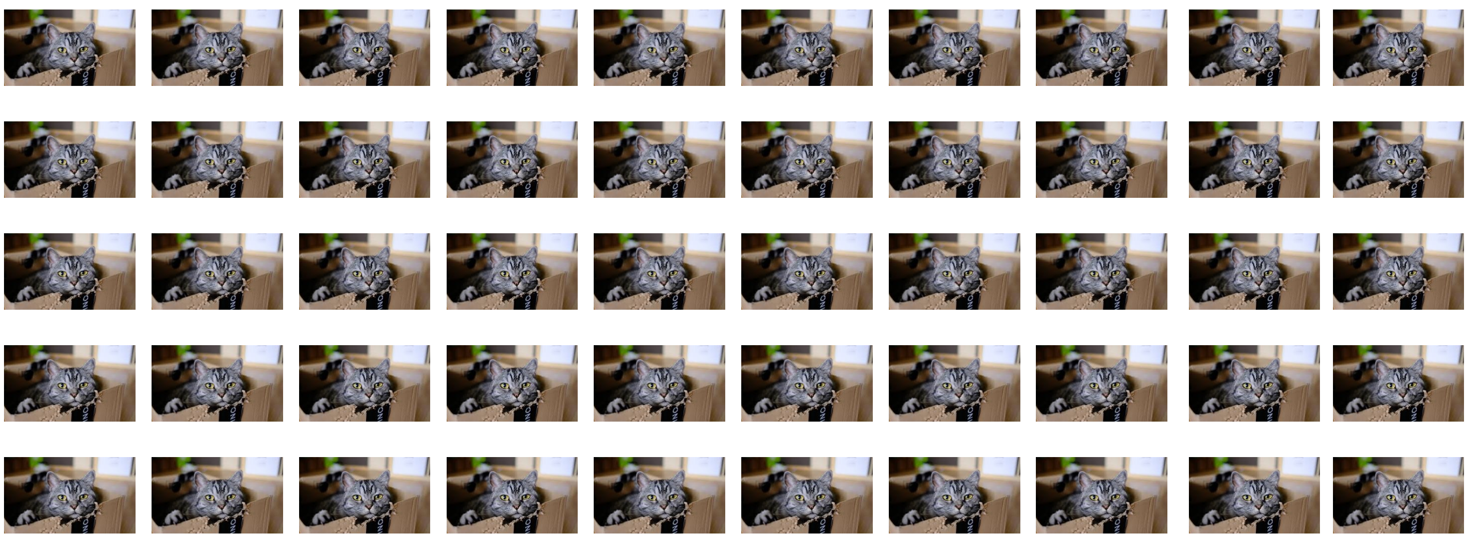


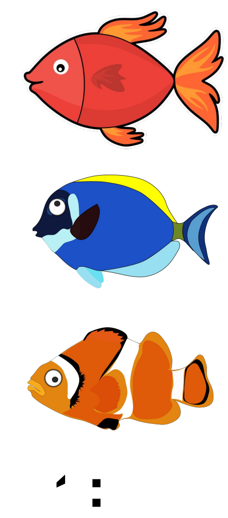
⋮

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats

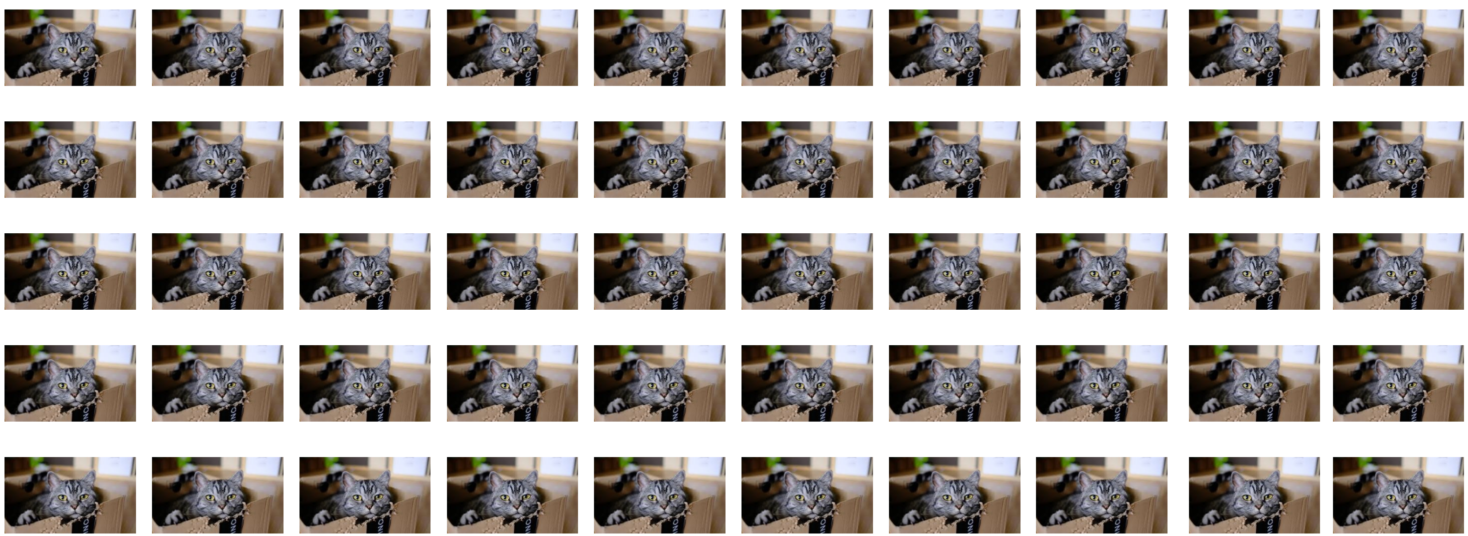


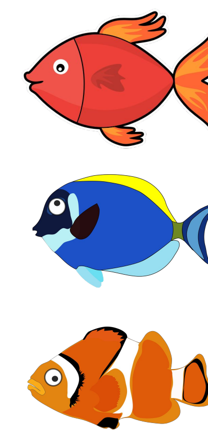


Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats





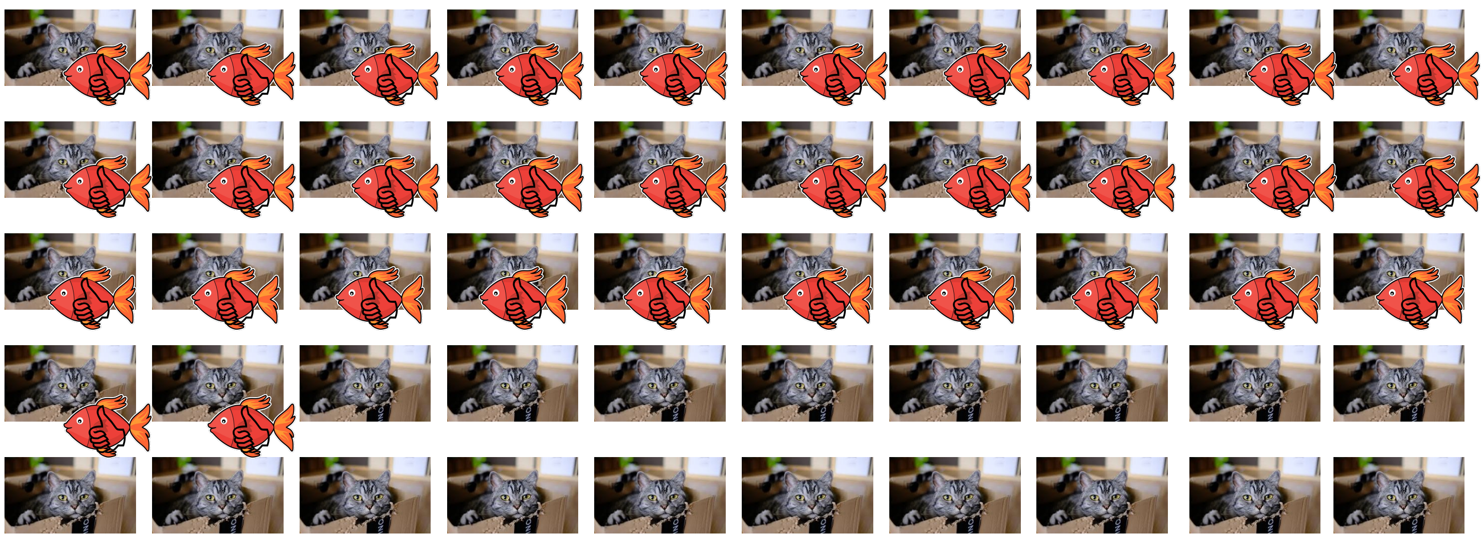
⋮

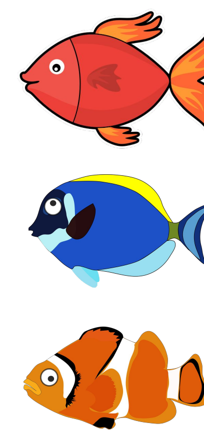
H_1: Most cats like redfish. H_0: Most cats don't like redfish.

Bonferroni’s Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; p = 0.016*



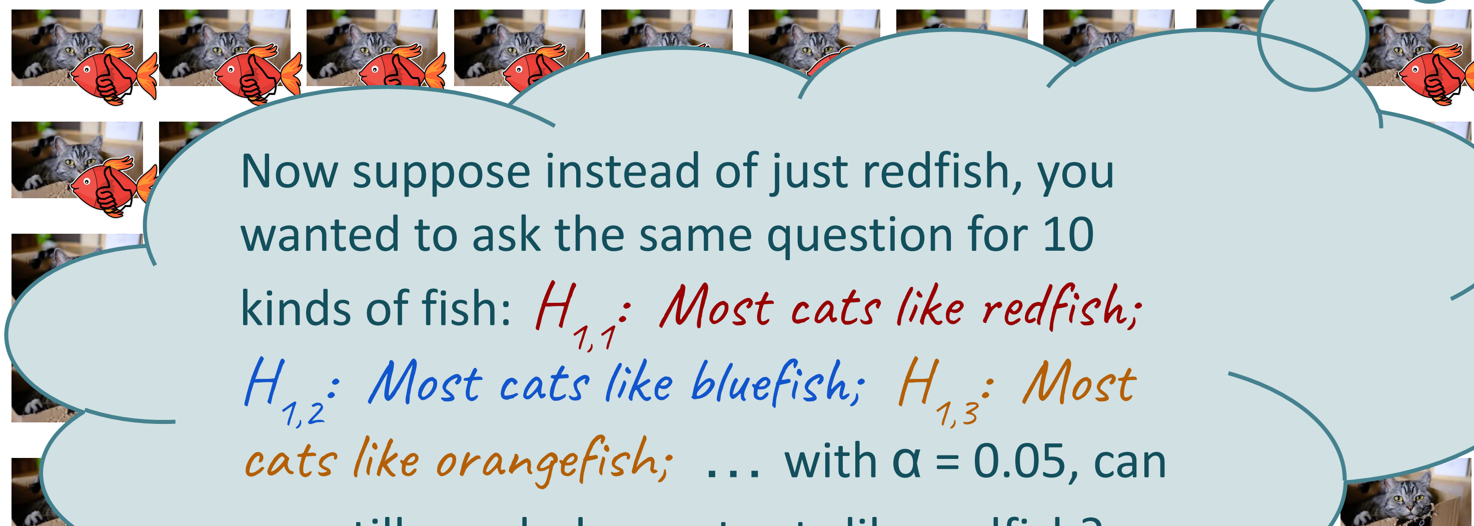


H_1: Most cats like redfish. *H_0: Most cats don’t like redfish.*

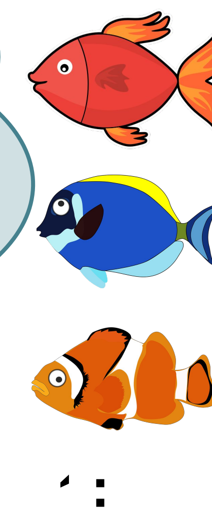
Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; p = 0.016*



Now suppose instead of just redfish, you wanted to ask the same question for 10 kinds of fish: *$H_{1,1}$: Most cats like redfish;* *$H_{1,2}$: Most cats like bluefish;* *$H_{1,3}$: Most cats like orangefish;* … with α = 0.05, can you still conclude most cats like redfish?

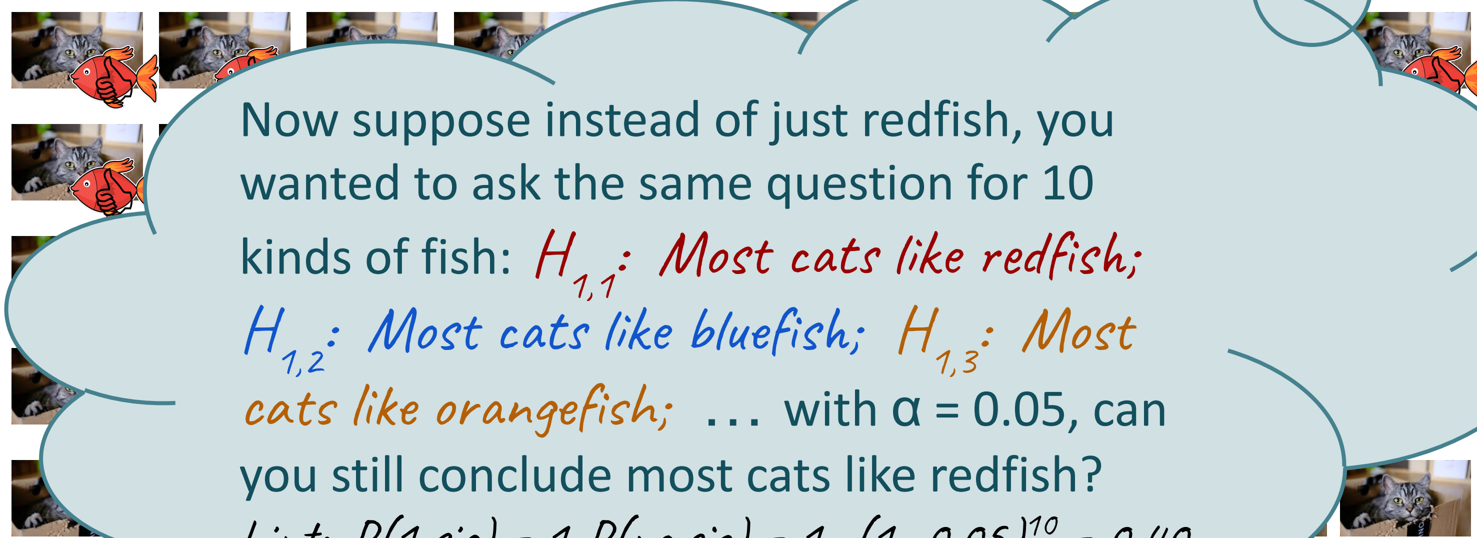


H_1: Most ca… don't like redfish.

Bonferroni's Cats

General Question: Which fish do cats like?

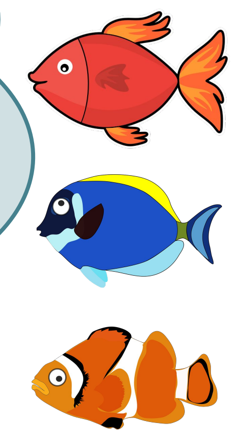
N = 50 cats; *32 like redfish; $p = 0.016$*



Now suppose instead of just redfish, you wanted to ask the same question for 10 kinds of fish: *$H_{1,1}$: Most cats like redfish; $H_{1,2}$: Most cats like bluefish; $H_{1,3}$: Most cats like orangefish;* … with α = 0.05, can you still conclude most cats like redfish?

hint: $P(1\ sig) = 1 - P(no\ sig) = 1 - (1 - 0.05)^{10} = 0.40$

H_1: Most ca… don't like redfish.

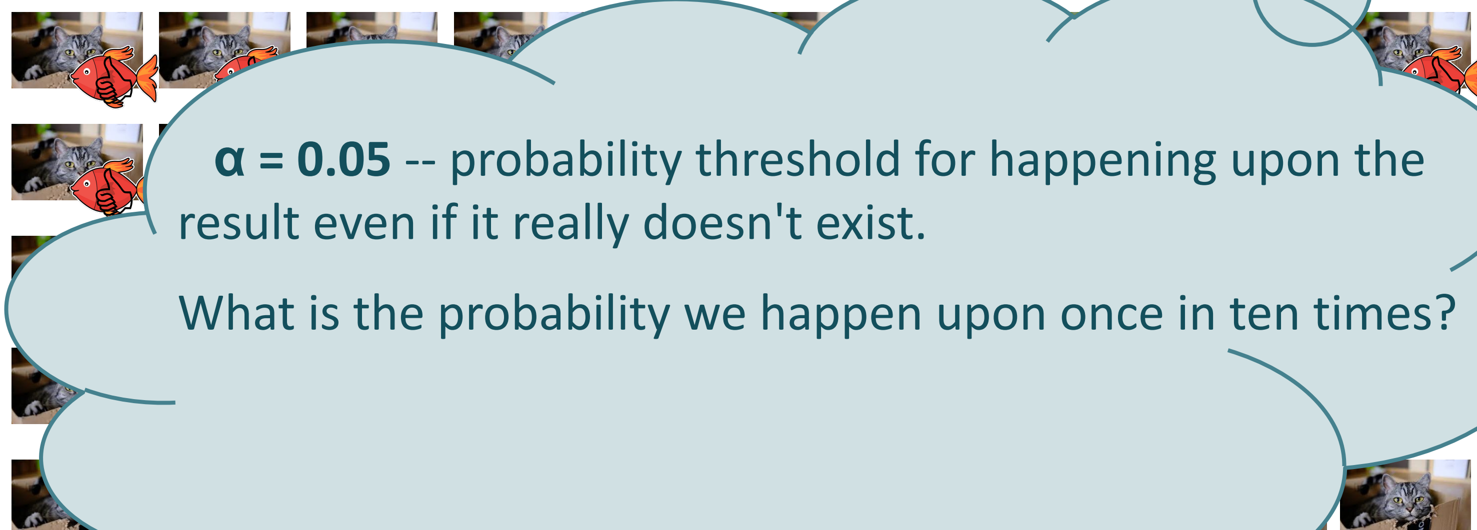


⋮

Bonferroni's Cats

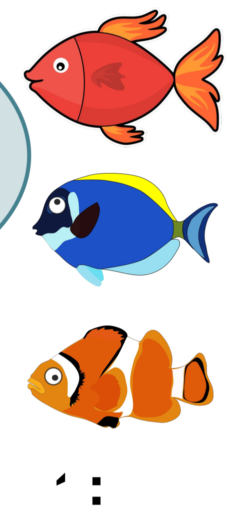
General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; $p = 0.016$*



α = 0.05 -- probability threshold for happening upon the result even if it really doesn't exist.

What is the probability we happen upon once in ten times?

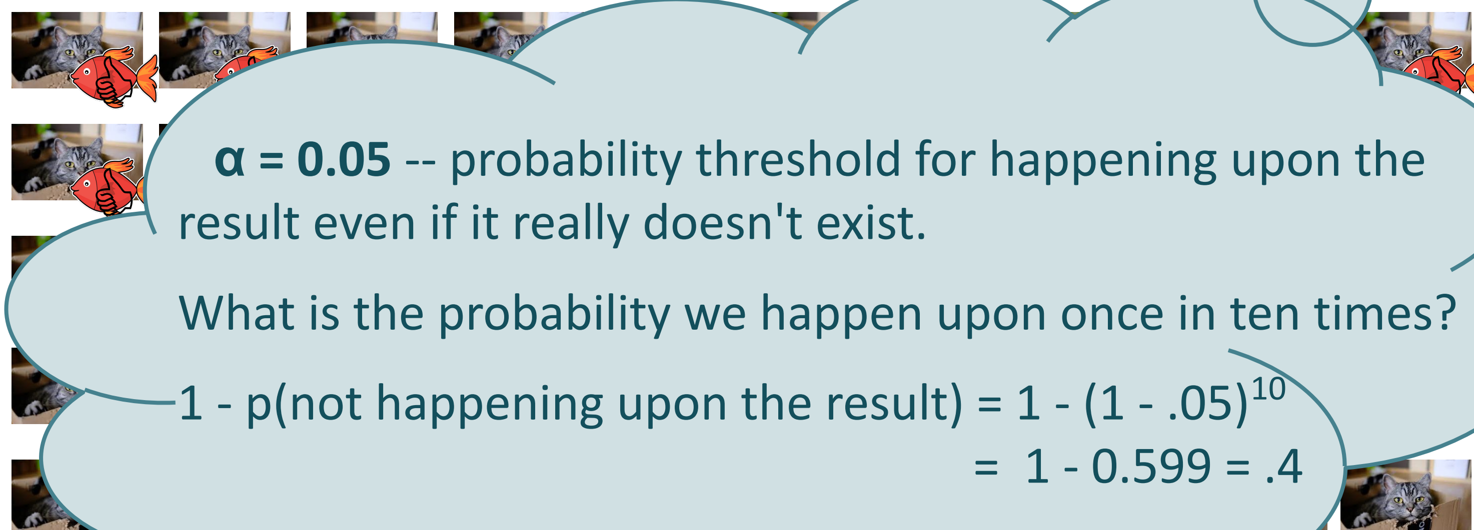


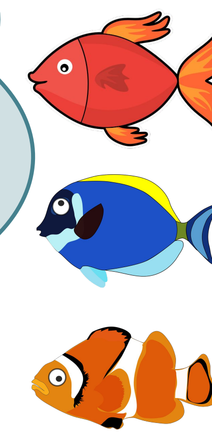
H_1: Most ca... don't like redfish.

Bonferroni’s Cats

General Question: Which fish do cats like?

N = 50 cats; 32 like redfish; $p = 0.016$





α = 0.05 -- probability threshold for happening upon the result even if it really doesn't exist.

What is the probability we happen upon once in ten times?

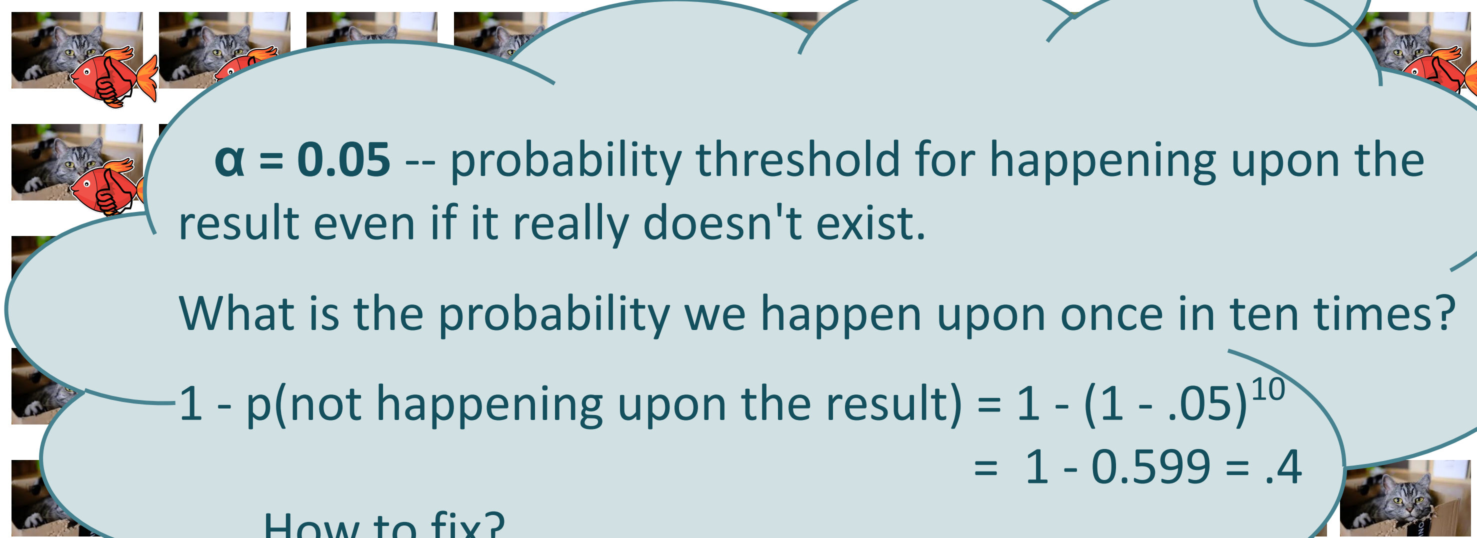
$$1 - p(\text{not happening upon the result}) = 1 - (1 - .05)^{10}$$
$$= 1 - 0.599 = .4$$

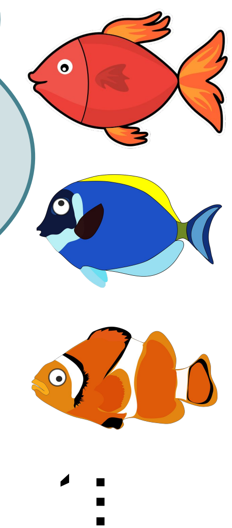
H_1: Most ca... don't like redfish.

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; p = 0.016*





α = 0.05 -- probability threshold for happening upon the result even if it really doesn't exist.

What is the probability we happen upon once in ten times?

$$1 - p(\text{not happening upon the result}) = 1 - (1 - .05)^{10}$$
$$= 1 - 0.599 = .4$$

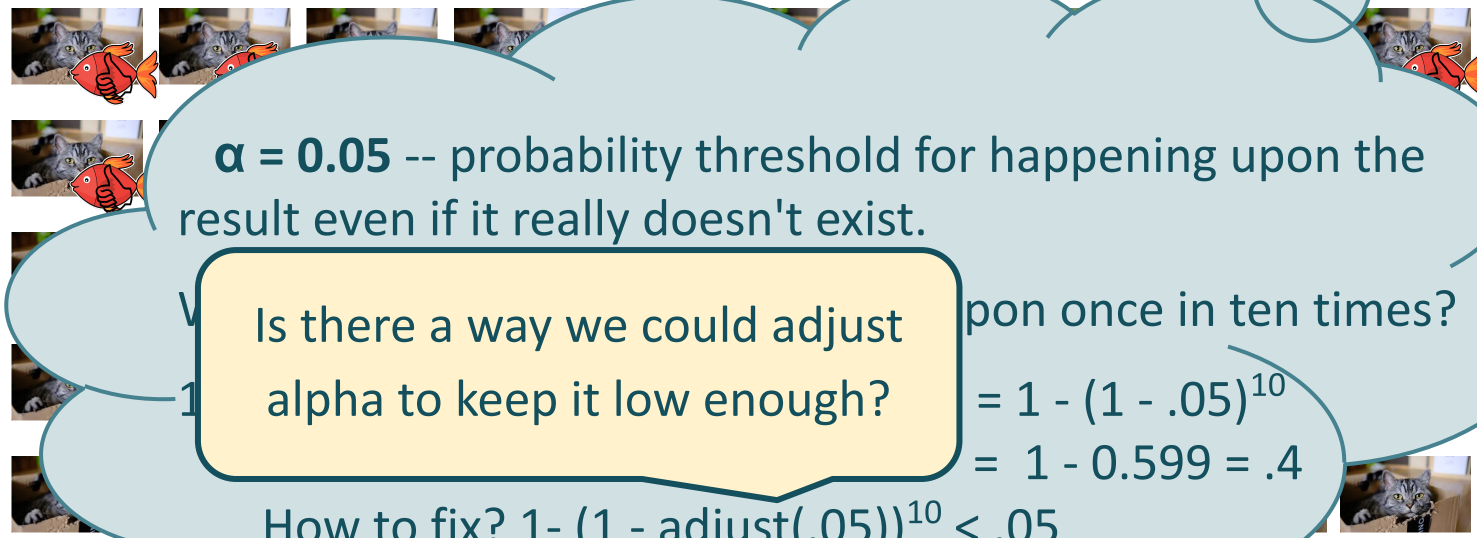
How to fix?

H_1: Most cats ... don't like redfish.

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; p = 0.016*



α = 0.05 -- probability threshold for happening upon the result even if it really doesn't exist.

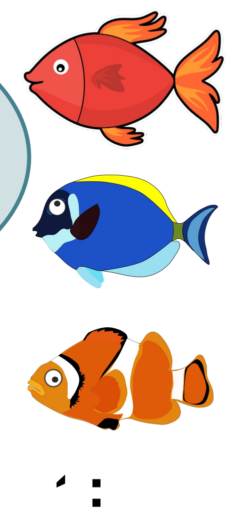
W… pon once in ten times?

1… $= 1 - (1 - .05)^{10}$

$= 1 - 0.599 = .4$

Is there a way we could adjust alpha to keep it low enough?

How to fix? $1 - (1 - \text{adjust}(.05))^{10} < .05$



H_1: Most ca… don't like redfish.

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; p = 0.016*

r happening upon the

n

W

pon once in ten times?

1

$= 1 - (1 - .05)^{10}$

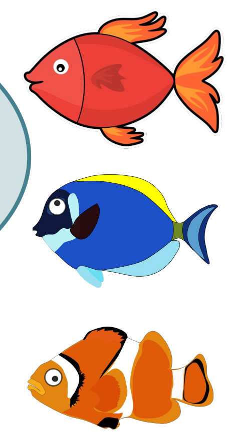
$= 1 - 0.599 = .4$

How to fix? $1 - (1 - \text{adjust}(.05))^{10} < .05$

Is there a way we could adjust alpha to keep it low enough?

The Bonferroni correction:

$$\alpha_{bonf} = \alpha \,/\, |h|$$



H_1: Most cat... don't like redfish.

Bonferroni's Cats

General Question: Which fish do cats like?

N = 50 cats; *32 like redfish; $p = 0.016$*

Is there a way we could adjust alpha to keep it low enough?

The Bonferroni correction:

$$\alpha_{bonf} = \alpha \,/\, |h|$$

r happening upon the

n

W

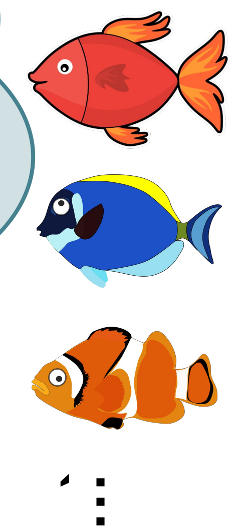
pon once in ten times?

1

$= 1 - (1 - .05)^{10}$

$= 1 - 0.599$

How to fix? $1 - (1 - \mathbf{(.05/10)})^{10} = .0488$



H_1: Most ca

don't like redfish.

Multi-test Correction

Type I, Type II Errors

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

significance level ("p-value") = P(type I error) = **P(Reject H_0 | H_0)**
(probability we are incorrect)

	H_0	H_A
Reject H_0	**P(Reject H_0 | H_0)**	

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

significance level ("p-value") = P(type I error) = **P(Reject H_0 | H_0)**
(probability we are incorrect)

power = 1 - P(type II error) = **P(Reject H_0 | H_1)**
(probability we are correct)

	H_0	H_A
Reject H_0	**P(Reject H_0 | H_0)**	**P(Reject H_0 | H_A)**

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

FWER: Family-wise error rate (Bonferroni Corrects)

The probability of making >=1 type 1 error.

$$FWER = Pr(\text{type1s}>0) = 1 - Pr(\text{type1s}=0) = 1 - (1 - a)^m$$

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

FWER: Family-wise error rate (Bonferroni Corrects)

The probability of making >=1 type 1 error.

$$FWER = Pr(\text{type1s}>0) = 1 - Pr(\text{type1s}=0) = 1 - (1 - a)^m$$

$1 - (1 - \mathbf{(.05/10)})^{10} = .0488$

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

FWER: Family-wise error rate (Bonferroni corrects)

The probability of making >=1 type 1 error.

$$FWER = Pr(\text{type1s}>0) = 1 - Pr(\text{type1s}=0) = 1 - (1 - a)^m$$

FDR: False discovery rate (Benjamini-Hochberg corrects)

type1s / (type1s + correctRejects)

		True state of nature	
		H_0	H_A
Our decision	Reject H_0	Type I error	correct decision
	'Accept' H_0	correct decision	Type II error

(Orloff & Bloom, 2014)

Multi-test Correction

FWER: Family-wise error rate (Bonferroni corrects)

The probability of making >=1 type 1 error.

$$FWER = Pr(\text{type1s>0}) = 1 - Pr(\text{type1s=0}) = 1 - (1 - a)^m$$

FDR: False discovery rate (Benjamini-Hochberg corrects)

type1s / (type1s + correctRejects)

Proportion of false positives among *all* significant results.

(Orloff & Bloom, 2014)

The Hypothesis Test "Algorithm"

Input: H_0, obs, α

```
obs_ts = test_stat(obs)
null_dist = distribution of expected test_stat under H_0
p(x<=obs_ts | H_0) = cdf(null_dist, obs_ts)
if p(x<=obs_ts | H_0) < α:
   decision = "Reject H_0!"
else:
   decision = "Accept H_0."
```

Output: decision

The Multi-test "Algorithm"

```
Input: H0s, obs, α

decisions = []
α_adj = adjust(α)

for H0 in H0s
   obs_ts = test_stat(obs)
   null_dist = distribution of expected test_stat under H0
   p(x<=obs_ts | H0) = cdf(null_dist, obs_ts)
   if p(x<=obs_ts | H0) < α_adj:
     decisions.append("Reject H0!")
   else:
     decisions.append("Accept H0.")

Output: decisions
```

The Multi-test "Algorithm"

```
Input: H_0s, obs, α

decisions = []
α_adj = adjust(α) #e.g. adjust(α) = α/len(H_0s)

for H_0 in H_0s
    obs_ts = test_stat(obs)
    null_dist = distribution of expected test_stat under H_0
    p(x<=obs_ts | H_0) = cdf(null_dist, obs_ts)
    if p(x<=obs_ts | H_0) < α_adj:
      decisions.append("Reject H_0!")
    else:
      decisions.append("Accept H_0.")

Output: decisions
```

Multi-test "Algorithm" **Alternative**

```
Input: H0s, obs, α

decisions = []
for H0 in H0s
   obs_ts = test_stat(obs)
   null_dist = distribution of expected test_stat under H0
   p(x<=obs_ts|H0) = cdf(null_dist, obs_ts)
   p_adj = inverse_adjust(p(x<=obs_ts|H0))#e.g. p*len(H0s)
   if p_adj < α:
     decisions.append("Reject H0!")
   else:
     decisions.append("Accept H0.")

Output: decisions
```

Statistical Considerations for Big Data

1. Average multiple models (ensemble techniques)
2. Correct for multiple tests (Bonferonni's Principle)
3. Smooth data
4. "Plot" data (or figure out a way to look at a lot of it "raw")
5. Interact with data
6. Know your "real" sample size
7. Correlation is not causation
8. Define metrics for success (set a baseline)
9. Share code and data
10. The problem should drive solution

(http://simplystatistics.org/2014/05/22/10-things-statistics-taught-us-about-big-data-analysis/)

Comparing Variables

- **Linear Regression**
- Pearson Product-Moment Correlation
- Multiple Linear Regression
- ~~(Multiple) Logistic Regression~~
- ~~Ridge Regression (L2 Penalized)~~
- ~~Lasso Regression (L1 Penalized)~~

Comparing Variables

Finding a linear function based on *X* to best yield *Y.*

X = "covariate" = "feature" = "predictor" = "regressor" = "independent variable"

Y = "response variable" = "outcome" = "dependent variable"

Regression: $$r(x) = \mathrm{E}(Y|X = x)$$

goal: estimate function *r*

The **expected** value of *Y*, given that the random variable *X* is equal to some specific value, *x*.

Linear Regression

Finding a linear function based on *X* to best yield *Y.*

X = “covariate” = “feature” = “predictor” = “regressor” = “independent variable”

Y = “response variable” = “outcome” = “dependent variable”

Regression: $$r(x) = \mathrm{E}(Y|X = x)$$

goal: estimate the function *r*

Linear Regression (univariate version):

$$r(x) = \beta_0 + \beta_1 x$$

goal: find β_0, β_1 such that

$$r(x) \approx \mathrm{E}(Y|X = x)$$

Linear Regression

Simple Linear Regression $$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\text{where } \mathbf{E}(\epsilon_i | X_i) = 0 \textit{ and } \mathbf{V}(\epsilon_i | X_i) = \sigma^2$$

more precisely

$$r(x) = \beta_0 + \beta_1 x$$

Linear Regression

intercept slope error

Simple Linear Regression

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\text{where } \mathbf{E}(\epsilon_i | X_i) = 0 \text{ } and \text{ } \mathbf{V}(\epsilon_i | X_i) = \sigma^2$$

expected variance

Linear Regression: Estimating Params

Simple Linear Regression $$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\text{where } \mathbf{E}(\epsilon_i | X_i) = 0 \; and \; \mathbf{V}(\epsilon_i | X_i) = \sigma^2$$

How to estimate intercept ($\square_0$) and slope intercept ($\square_1$)?

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n} (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; rss$^{(0)} = \infty$
for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if rss$^{(t\text{-}1)}$ - rss$^{(t)}$ < ε: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha(\sum_{i=1}^{n} \hat{Y}_i - Y_i)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha(\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i))$$

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

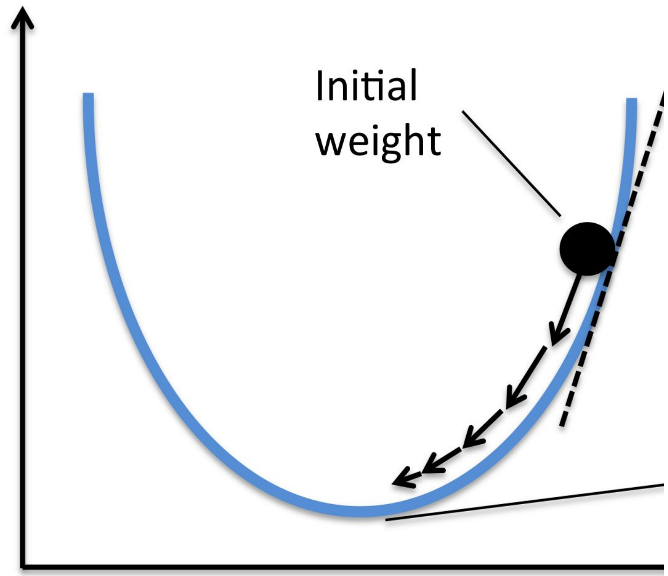
initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; $\text{rss}^{(0)} = \infty$

for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if $\text{rss}^{(t\text{-}1)} - \text{rss}^{(t)} < \varepsilon$: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha\left(\sum_{i=1}^{n} \hat{Y}_i - Y_i\right)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha\left(\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i)\right)$$



(http://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/)

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; rss$^{(0)} = \infty$

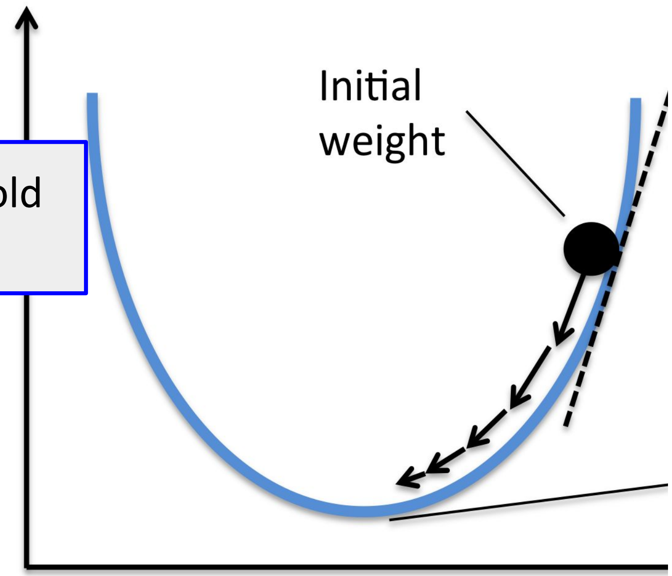
for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if rss$^{(t\text{-}1)}$ - rss$^{(t)} < \varepsilon$: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha(\sum_{i=1}^{n} \hat{Y}_i - Y_i)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha(\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i))$$

convergence threshold (e.g. .00001)



(http://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/)

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; $rss^{(0)} = \infty$

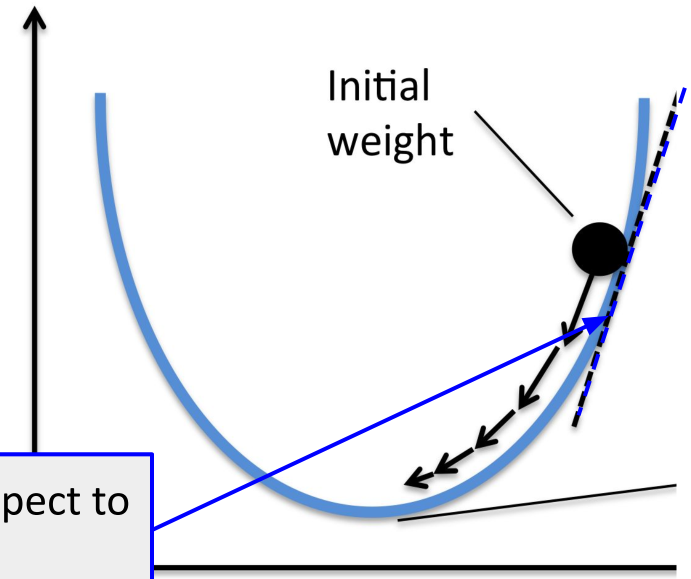
for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if $rss^{(t-1)} - rss^{(t)} < \varepsilon$: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha\left(\boxed{\sum_{i=1}^{n} \hat{Y}_i - Y_i}\right)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha\left(\boxed{\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i)}\right)$$

gradient with respect to given □.



(http://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/)

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(□) = RSS = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; $rss^{(0)} = \infty$

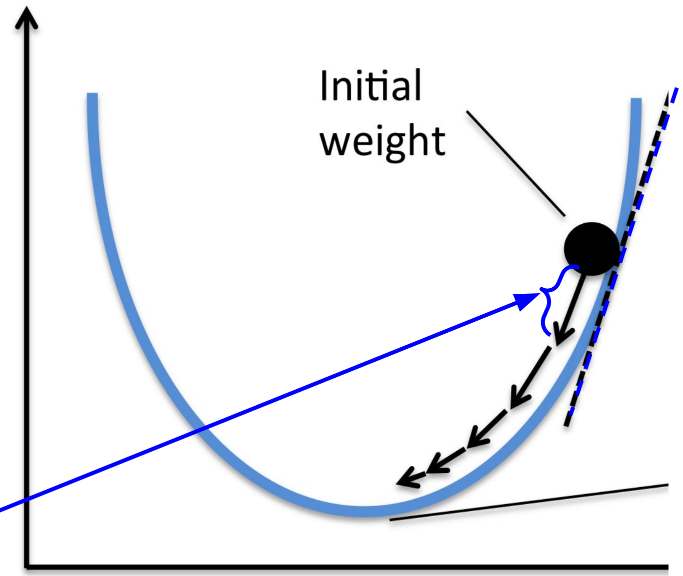
for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if $rss^{(t\text{-}1)} - rss^{(t)} < \varepsilon$: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \boxed{\alpha}\left(\sum_{i=1}^{n} \hat{Y} \dots\right.$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \boxed{\alpha}\left(\sum_{i=1}^{n} X_i(Y \dots\right.$$

learning rate: scales the size of the update.



(http://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/)

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

Linear Regression: Estimating Params

Method 1: Gradient Descent

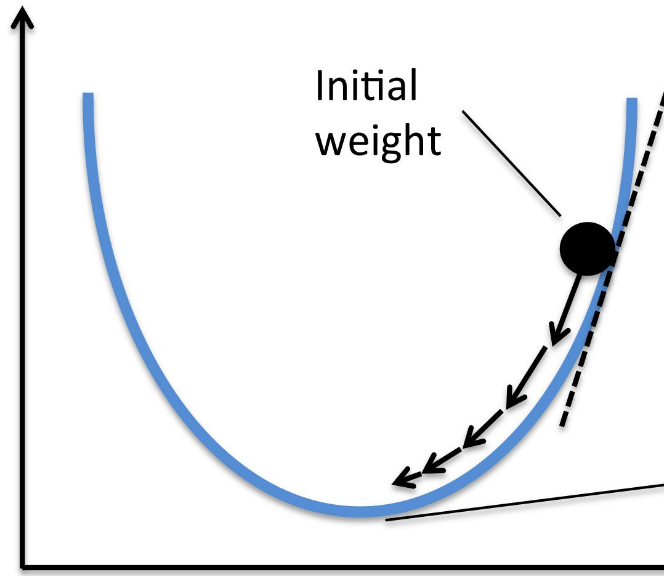
initialize: $\hat{\beta}_0 = \hat{\beta}_1 = 0$; rss$^{(0)} = \infty$

for t in range(1, limit):

1. calculate all $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$
2. if rss$^{(t\text{-}1)}$ - rss$^{(t)}$ < ε: break #converged
3. set:

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha(\sum_{i=1}^{n} \hat{Y}_i - Y_i)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha(\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i))$$



(http://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/)

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$J(\square) = RSS = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Linear Regression

via Gradient Descent

Start with $\hat{\beta}_0 = \hat{\beta}_1 = 0$

Repeat until convergence:

Calculate all $\hat{Y}_i$

$$\hat{\beta}_0 = \hat{\beta}_0 - \alpha(\sum_{i=1}^{n} \hat{Y}_i - Y_i)$$

$$\hat{\beta}_1 = \hat{\beta}_1 - \alpha(\sum_{i=1}^{n} X_i(\hat{Y}_i - Y_i))$$

via Direct Estimates (normal equations)

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

Least Squares Estimate. Find $\hat{\beta}_0$ and $\hat{\beta}_1$ which minimizes the *residual sum of squares:*

$$RSS = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n}(Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

Pearson Product-Moment Correlation

Covariance

$$
\begin{aligned}
Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\
&= \mathbf{E}\left((X - \bar{X})(Y - \bar{Y})\right)
\end{aligned}
$$

Pearson Product-Moment Correlation

Covariance

$$\begin{aligned} Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\ &= \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right) \end{aligned}$$

Correlation

$$r = r_{X,Y} = \frac{Cov(X,Y)}{s_X s_Y}$$

Pearson Product-Moment Correlation

Covariance

$$\begin{aligned} Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\ &= \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right) \end{aligned}$$

Correlation *(standardized covariance)*

$$r = r_{X,Y} = \frac{Cov(X,Y)}{s_X s_Y}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i-\bar{X}}{s_X}\right)\left(\frac{Y_i-\bar{Y}}{s_Y}\right)$$

Pearson Product-Moment Correlation

Covariance

$$\begin{aligned} Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\ &= \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right) \end{aligned}$$

Correlation

$$\begin{aligned} r = r_{X,Y} &= \frac{Cov(X,Y)}{s_X s_Y} \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i-\bar{X}}{s_X}\right)\left(\frac{Y_i-\bar{Y}}{s_Y}\right) \end{aligned}$$

Method 2: Direct Estimates (normal equations)

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sum_{i=1}^{n}(X_i-\bar{X})^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1\bar{X}$$

Pearson Product-Moment Correlation

Covariance

$$
\begin{aligned}
Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\
&= \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right)
\end{aligned}
$$

Correlation

$$
\begin{aligned}
r = r_{X,Y} &= \frac{Cov(X,Y)}{s_X s_Y} \\
&= \frac{1}{n-1}\sum_{i=1}^{n} \boxed{\left(\frac{X_i - X}{s_X}\right)\left(\frac{Y_i - Y}{s_Y}\right)}
\end{aligned}
$$

Method 2: Direct Estimates (normal equations)

$$
\hat{\beta}_1 = \boxed{\frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sum_{i=1}^{n}(X_i-\bar{X})^2}}
$$

$$
\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1\bar{X}
$$

Pearson Product-Moment Correlation

Covariance

$$\begin{aligned} Cov(X,Y) &= \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) \\ &= \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right) \end{aligned}$$

Correlation

$$r = r_{X,Y} = \frac{Cov(X,Y)}{s_X s_Y}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n} \boxed{\left(\frac{X_i - X}{s_X}\right)\left(\frac{Y_i - Y}{s_Y}\right)}$$

Method 2: Direct Estimates (normal equations)

$$\hat{\beta}_1 = \boxed{\frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

If one standardizes *X* and *Y* (i.e. subtract the mean and divide by the standard deviation) before running linear regression, then:

??

Pearson Product-Moment Correlation

Covariance

$$Cov(X,Y) = \mathbf{E}(XY) - \mathbf{E}(X)\mathbf{E}(Y) = \mathbf{E}\left((X-\bar{X})(Y-\bar{Y})\right)$$

Correlation

$$r = r_{X,Y} = \frac{Cov(X,Y)}{s_X s_Y} = \frac{1}{n-1}\sum_{i=1}^{n} \boxed{\left(\frac{X_i - X}{s_X}\right)\left(\frac{Y_i - Y}{s_Y}\right)}$$

Method 2: Direct Estimates (normal equations)

$$\hat{\beta}_1 = \boxed{\frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sum_{i=1}^{n}(X_i-\bar{X})^2}}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1\bar{X}$$

If one standardizes *X* and *Y* (i.e. subtract the mean and divide by the standard deviation) before running linear regression, then:

$\hat{\beta}_0 = 0$ and $\hat{\beta}_1 = r$ --- *i.e.* $\hat{\beta}_1$ *is the Pearson correlation!*

Comparing Variables

- **Linear Regression**
- **Pearson Product-Moment Correlation**
- Multiple Linear Regression
- ~~(Multiple) Logistic Regression~~
- ~~Ridge Regression (L2 Penalized)~~
- ~~Lasso Regression (L1 Penalized)~~

Comparing Variables

- Linear Regression
- Pearson Product-Moment Correlation
- **Multiple Linear Regression**
- ~~(Multiple) Logistic Regression~~
- ~~Ridge Regression (L2 Penalized)~~
- ~~Lasso Regression (L1 Penalized)~~

Multiple Linear Regression

Simple Linear Regression $$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\text{where } \mathbf{E}(\epsilon_i | X_i) = 0 \textit{ and } \mathbf{V}(\epsilon_i | X_i) = \sigma^2$$

expected variance

Estimated intercept and slope

$$\hat{r}(x) = \hat{\beta}_0 + \hat{\beta}_1 x \qquad \hat{Y}_i = \hat{r}(X_i)$$

Residual: $$\hat{\epsilon}_i = Y_i - \hat{Y}_i$$

Multiple Linear Regression

Suppose we have multiple X that we'd like to fit to Y at once:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

If we include and $X_{oi} = 1$ for all i (i.e. adding the intercept to X), then we can say:

$$Y_i = \sum_{j=0}^{m} \beta_j X_{ij} + \epsilon_i$$

Multiple Linear Regression

Suppose we have multiple X that we'd like to fit to Y at once:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

If we include and $X_{oi} = 1$ for all i, then we can say:

$$Y_i = \sum_{j=0}^{m} \beta_j X_{ij} + \epsilon_i$$

Or in vector notation across all i:

$$Y = X\beta + \epsilon$$

where β and ϵ are vectors and X is a matrix.

Multiple Linear Regression

Suppose we have multiple X that we'd like to fit to Y at once:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

If we include and $X_{oi} = 1$ for all i, then we can say:

$$Y_i = \sum_{j=0}^{m} \beta_j X_{ij} + \epsilon_i$$

Or in vector notation across all i:

$$Y = X\beta + \epsilon$$

where β and ϵ are vectors and X is a matrix.

Estimating β :

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

Multiple Linear Regression

Suppose we have multiple independent variables that we'd like to fit to our dependent variable: $Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$

If we include and X_{oi} = 1 for all i. Then we can say:

$$Y_i = \sum_{j=0}^{m} \beta_j X_{ij} + \epsilon_i$$

To test for significance of individual coefficient, *j*:

$$t = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)} = \frac{\hat{\beta}_j}{\sqrt{\frac{s^2}{\sum_{i=1}^{n}(X_{ij} - \bar{X}_j)^2}}}$$

Or in vector notation across all i: $Y = X\beta + \epsilon$

Where β and ϵ are vectors and X is a matrix.

Estimating β :

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

Significance Testing

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

$$s^2 = \frac{RSS}{df}$$

To test for significance of individual coefficient, *j*:

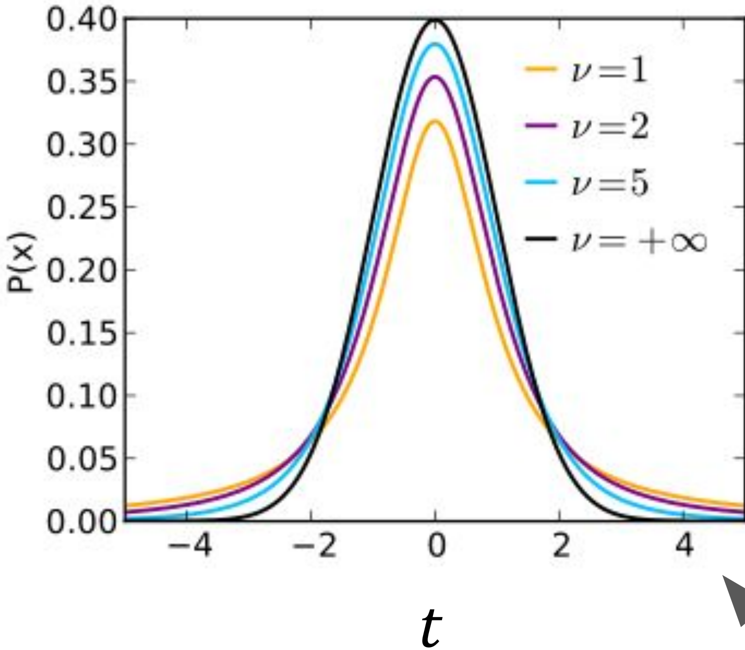
$$t = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)} = \frac{\hat{\beta}_j}{\sqrt{\frac{s^2}{\sum_{i=1}^{n}(X_{ij} - \bar{X}_j)^2}}}$$

T-Test for significance of hypothesis:

1) Calculate t
2) Calculate degrees of freedom:

$$df = N - (m+1)$$

3) Check probability in a *t* distribution:



$$\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

To test for significance of individual coefficient, j:

$$t = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)} = \frac{\hat{\beta}_j}{\sqrt{\frac{s^2}{\sum_{i=1}^{n}(X_{ij} - \bar{X}_j)^2}}}$$

T-Test for significance of hypothesis:

1) Calculate t
2) Calculate degrees of freedom:

$$df = N - (m+1)$$

3) Check probability in a t distribution: ($df = v$)

Summary

Hypothesis Testing:

A framework for deciding which differences/relationships matter.

- Random Variables
- Distributions
- Hypothesis Testing Framework

Comparing Variables:

Metrics to quantify the difference or relationship between variables.

- Simple Linear Regression, Correlation, Multiple Linear Regression,
- Comparing Variables and Hypothesis Testing
- ~~Regularized Linear Regression (for supervised ML)~~
- Multiple Hypothesis Testing

Large-Scale Hypothesis Testing

- Findings and Uncertainty

- Hypothesis Testing

- Bonferroni’s Cats
- Multi-test Corrections
  - Family-wise Error Rate
  - False-Discovery Rate
- Correlation Metrics
  - Effect Size (coefficient)
  - Significance (whether p-value is below significance level)

Supplement: Not on exam

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want “classification”)

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$

Note: this is a probability here.

In simple linear regression we wanted an expectation:

$$r(x) = \mathrm{E}(Y | X = x)$$

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \underbrace{\mathbf{P}(Y_i = 1 | X = x)} = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$

Note: this is a probability here.
In simple linear regression we wanted an expectation:

$$r(x) = \mathrm{E}(Y | X = x)$$

(i.e. if $p > 0.5$ we can confidently predict $Y_i = 1$)

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$

$$logit(p_i) = log\left(\frac{p_i}{1 - p_i}\right) = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}$$

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$

$$logit(p_i) = log\left(\frac{p_i}{\boxed{1 - p_i}}\right) = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}$$

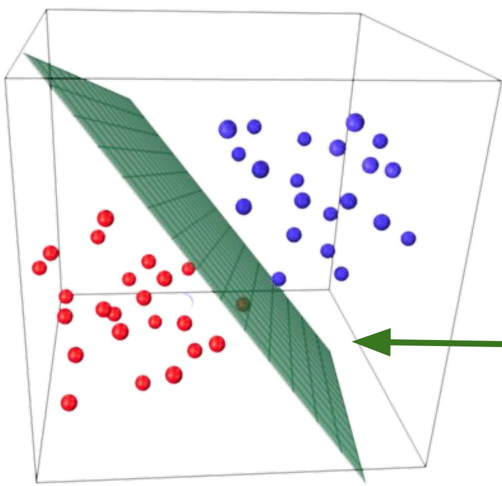
$P(Y_i = 0 \mid X = x)$

Thus, 0 is class 0 and 1 is class 1.

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}}}$$



$$logit(p_i) = log\left(\frac{p_i}{1 - p_i}\right) = \beta_0 + \sum_{j=1}^{m} \boxed{\beta_j x_{ij}}$$

We're still learning a linear *separating hyperplane*, but fitting it to a *logit* outcome.

(https://www.linkedin.com/pulse/predicting-outcomes-probabilities-logistic-regression-konstantinidis/)

Logistic Regression

What if $Y_i \in \{0, 1\}$? (i.e. we want "classification")

$$p_i \equiv p_i(\beta) \equiv \mathbf{P}(Y_i = 1 | X = x) = \frac{e^{\beta_0 + \sum_{j=1}^m \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^m \beta_j x_{ij}}}$$

$$logit(p_i) = log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}$$

To estimate β, one can use *reweighted least squares:*

set $\hat{\beta}_0 = ... = \hat{\beta}_m = 0$ (remember to include an intercept)

1. Calculate p_i and let W be a diagonal matrix where element$(i,i) = p_i(1-p_i)$.
2. Set $z_i = logit(p_i) + \frac{Y_i - p_i}{p_i(1-p_i)} = X\hat{\beta} + \frac{Y_i - p_i}{p_i(1-p_i)}$
3. Set $\hat{\beta} = (X^T W X)^{-1} X^T W z$ //weighted lin. reg. of Z on Y.
4. Repeat from 1 until $\hat{\beta}$ converges.

(Wasserman, 2005; Li, 2010)

Uses of linear and logistic regression

1. Testing the relationship between variables given other variables. β is an “effect size” -- a score for the magnitude of the relationship; can be tested for significance.

2. Building a predictive model that generalizes to new data. $\hat{Y}$ is an estimate value of Y given X.

Uses of linear and logistic regression

1. Testing the relationship between variables given other variables. β is an "effect size" -- a score for the magnitude of the relationship; can be tested for significance.

2. Building a predictive model that generalizes to new data. $\hat{Y}$ is an estimate value of Y given X. However, unless $|X| <<< observatations$ then the model might "overfit".

   -> Regularized linear regression (a ML technique)

Statistical Considerations in Big Data

1. Correct for multiple tests (Bonferonni's Principle)
2. Average multiple models (ensemble techniques)
3. Smooth data
4. "Plot" data (or figure out a way to look at a lot of it "raw")
5. Interact with data

(http://simplystatistics.org/2014/05/22/10-things-statistics-taught-us-about-big-data-analysis/)

Statistical Considerations in Big Data

1. Correct for multiple tests (Bonferonni's Principle)
2. Average multiple models (ensemble techniques)
3. Smooth data
4. "Plot" data (or figure out a way to look at a lot of it "raw")
5. Interact with data
6. Know your "real" sample size
7. Correlation is not causation
8. Define metrics for success (set a baseline)
9. Share code and data
10. The problem should drive solution

(http://simplystatistics.org/2014/05/22/10-things-statistics-taught-us-about-big-data-analysis/)