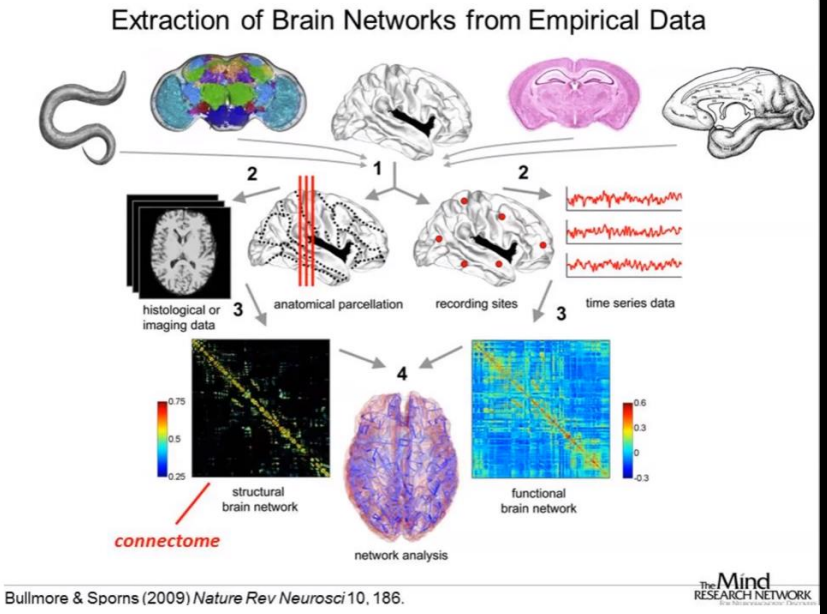
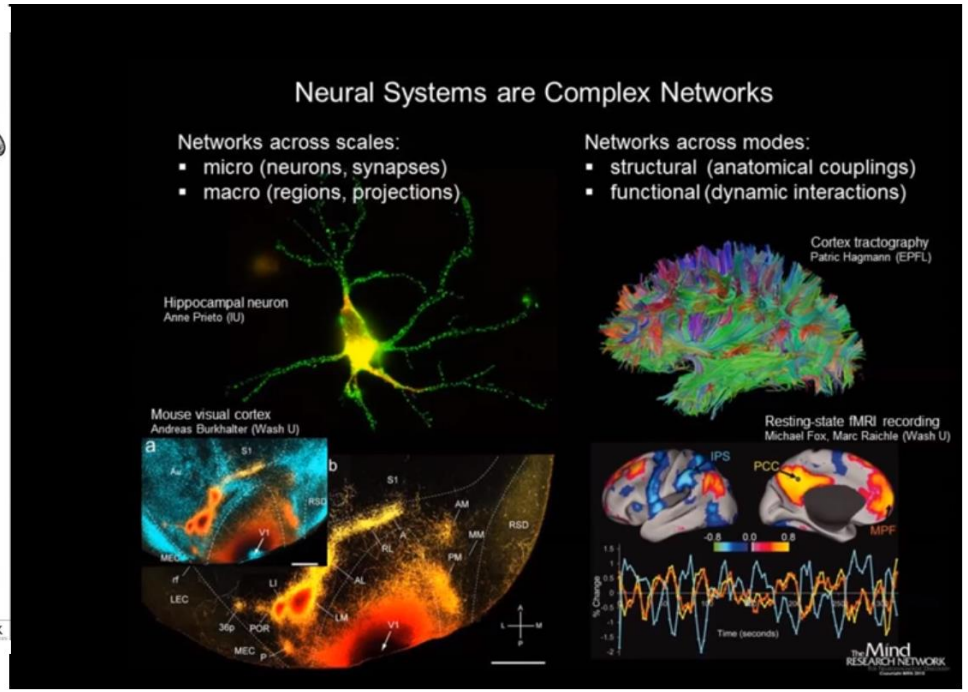




Lecture on Modeling Tools for Null Hypothesis Testing & Correlation

CS – 590.21 Analysis and Modeling of Brain Networks

Department of Computer Science
University of Crete

Greek Diaspora Fellowship Program
ΙΣΝ/SNF
ΙΔΡΥΜΑ ΣΤΑΥΡΟΣ ΝΙΑΡΧΟΣ
STAVROS NIARCHOS FOUNDATION

Acknowledgement – Resources used in the slides

On statistical hypothesis test

- Yiannis Tsamardinos (University of Crete)
- William Morgan (Stanford University)

On clustering

- Jiawei Han, University of Illinois at Urbana-Champaign

Agenda

Basic data analysis & modeling tools that can be employed in the projects

- Null Hypothesis Test
- Temporal correlation metrics
  - Pearson correlation
  - STTC
- Kolmogorov-Smirnov Test
- Clustering

  k-means
- Regression
  - Linear regression
  - Lasso & Ridge regression

Proving Your Hypothesis

Mathematics

1. We already know a set of axioms & theorems, say K
2. We want to show the theorem (hypothesis) H
3. We show: **K, ¬H ⇒ False** (**contradiction**)
4. Thus, if we trust that K holds indeed, **¬H cannot hold**, and

   H must hold

Real World

1. We **already "know" K**
2. We want to show a hypothesis H, e.g., "H: medicine A reduces the mortality of disease B"
3. We gather data from the real world.

   We show that **K, ¬H makes it**

   very unlikely to observe our data
4. We conclude that

   ¬H is very unlikely

   We **reject ¬H**, and **accept H**

Symbol "¬" indicates the negation of a statement

Notation for the following slides

- **Random variables** are denoted with a *capital* letter, e.g., ***X***
- **Observed quantities** of random variables are denoted with their corresponding *small* letter ***x***

Example:

- G is the expression level of a specific gene in a patient
- *g* is the measured expression level of the game in a ***specific*** **patient**

The Null Hypothesis

- The hypothesis we hope to accept is called the *Alternative Hypothesis*

  Sometimes denoted *as* H_1

- The hypothesis we hope to reject, the **negation of the Alternative Hypothesis**, is called the *Null Hypothesis*

  Usually denoted by H_o

Think of the H_o as the "status quo"

Standard Single Hypothesis Testing

1. Form the Null & Alternative Hypothesis
2. Obtain related data
3. Find a **suitable test statistic *T***
4. Find the **distribution of *T given the null***
5. Depending on the **distribution of *T*** & the **observed** $t_o = T(x)$
   decide to reject or not H_0

Test Statistics

- Test statistic is a **function** of our **data X**: ***T(X)*** *(X:* random variable *)*
  e.g., if X contains a single quantity (variable) *T(X)* the **mean** value of *X*

- T is a random variable (since it depends on *X* , our data which is random variable)

- Denote with t_o **= T*(x)*** the **observed value of *T*** in **our data**

- *Instead* of calculating P (**obtaining data similar to *X*** | H_0)
  Calculate ***P (T similar to t_o / H_0)***

- If *P (T* similar to t_o / H_0) is **very low, reject H_0**

Statistical significance tests

- Let's just think about **a two-tailed test**: "difference" or "no difference"
- **Null hypothesis**: there is **no difference between A vs. B**
- Assume that o_A & o_B are "sampled" independently from a "**population**"
- **Test statistic**: a function of the **sample data** **on which the decision is to be based**

  $\mathbf{t}\ (o_1, o_2) = |e(o_1) - e(o_2)|$

  e: evaluation metric
- **Find the distribution of t under the null hypothesis**

  Assume that the null hypothesis is true

- Where does the $t(o_A, o_B)$ lie in this distribution?

  If **it's somewhere unlikely**, that's evidence that the **null hypothesis is false**

"Welcome to Lake Wobegon, where all the women are strong, all the men are good-looking, and all the children are above average."

- Garrison Keillor, *A Prairie Home Companion*

The Lake Wobegon Example: “Where all the children are above average!”

- Let X represent Weschler Adult Intelligence scores (WAIS)
- Typically, X ~ **N(100, 15)** (**$\mu_0 = 100$, $\sigma = 15$**)
- Obtain data: **9 children** from Lake Wobegon population

  Their scores: {116, 128, 125, 119, 89, 99, 105, 116, 118}

  Average of the observations $\bar{x} = 112.8$

Does **sample mean** provide strong evidence that **population mean $\mu > 100$**?

One-Sample *z* Test

1. Hypothesis statements
   H_0: $\mu = \mu_0$
   H_a: $\mu \neq \mu_0$ (two-sided) or
   H_a: $\mu < \mu_0$ (left-sided) or
   H_a: $\mu > \mu_0$ (right-sided)

3. Test statistic

$$z_{stat} = \frac{\bar{x} - \mu_0}{SE_{\bar{x}}} \text{ where } SE_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$

4. P-value: convert z_{stat} to P value

A. Significance statement (usually not necessary)

Example: Two-Sided Hypothesis Test "Lake Wobegon"

1. Formulation of the Hypotheses:

H_0: $\mu = 100$

H_a: $\mu > 100$ (one-sided)

H_a: $\mu \neq 100$ (two-sided)

2. Obtain data …

Obtain data: **9 children** from Lake Wobegon population

Their scores: {116, 128, 125, 119, 89, 99, 105, 116, 118}

Average of the observations = 112.8

Example: Two-Sided Hypothesis Test "Lake Wobegon"

3. Test statistic

$$SE_{\bar{x}} = \frac{\sigma}{\sqrt{n}} = \frac{15}{\sqrt{9}} = 5$$

$$z_{\text{stat}} = \frac{\bar{x} - \mu_0}{SE_{\bar{x}}} = \frac{112.8 - 100}{5} = 2.56$$

Classical CLT [edit]

Let $\{X_1, \ldots, X_n\}$ be a random sample of size n — that is, a sequence of independent and identically distributed random variables drawn from distributions of expected values given by μ and finite variances given by σ^2. Suppose we are interested in the sample average

$$S_n := \frac{X_1 + \cdots + X_n}{n}$$

of these random variables. By the law of large numbers, the sample averages converge in probability and almost surely to the expected value μ as $n \to \infty$. The classical central limit theorem describes the size and the distributional form of the stochastic fluctuations around the deterministic number μ during this convergence. More precisely, it states that as n gets larger, the distribution of the difference between the sample average S_n and its limit μ, when multiplied by the factor $\sqrt{n}$ (that is $\sqrt{n}(S_n - \mu)$), approximates the normal distribution with mean 0 and variance σ^2. For large enough n, the distribution of S_n is close to the normal distribution with mean μ and variance $\frac{\sigma^2}{n}$. The usefulness of the theorem is that the distribution of $\sqrt{n}(S_n - \mu)$ approaches normality regardless of the shape of the distribution of the individual X_i. Formally, the theorem can be stated as follows:

Central Limit Theory

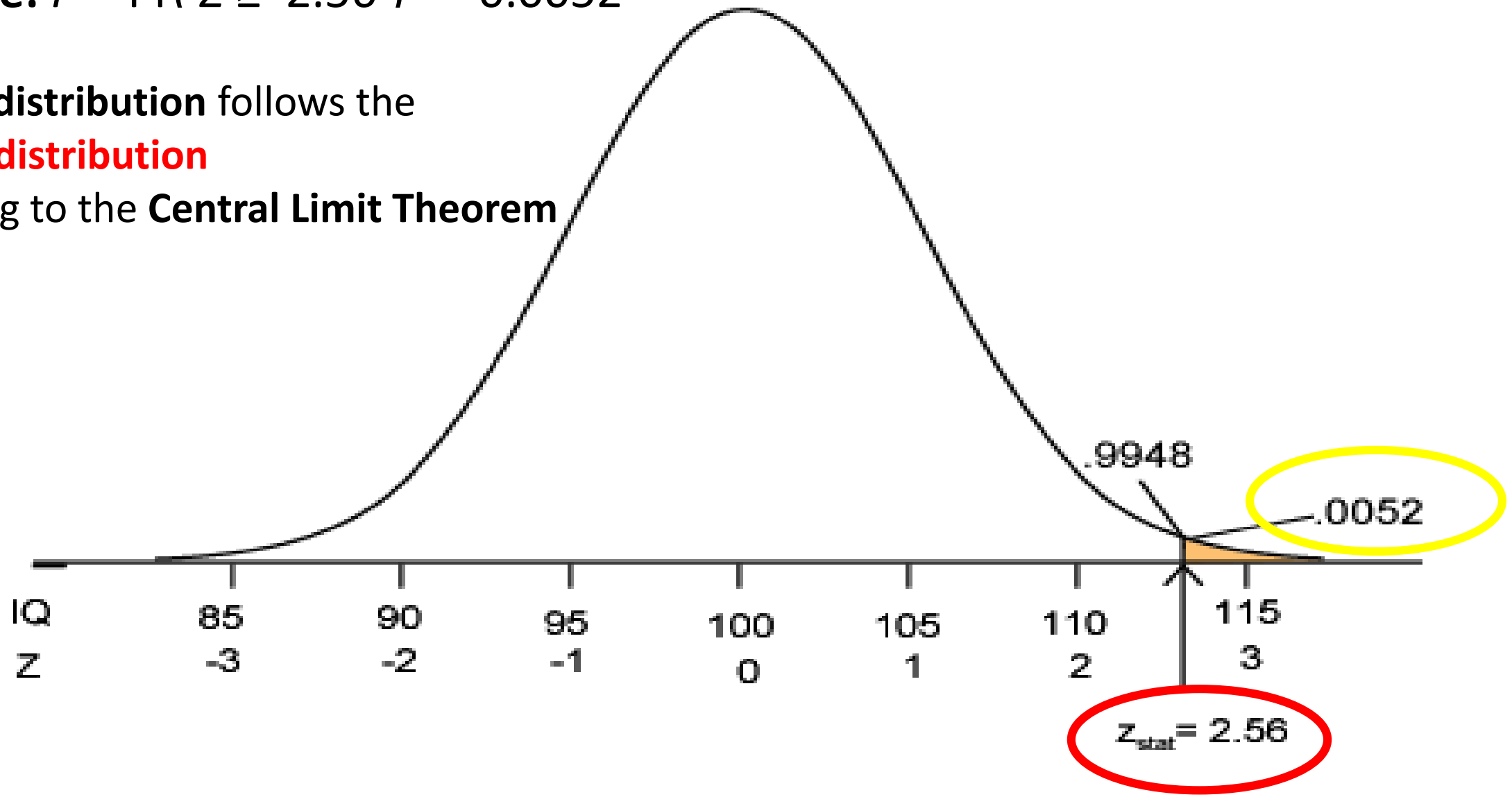
Establishes that, in most situations, when **independent random variables are added**, their **properly normalized sum tends toward a normal distribution** **even if the original variables themselves are *not normally* distributed.**

1. A **sample is obtained containing a large number of observations**, each observation being randomly generated in a way that does not depend on the values of the other observations.
2. If step 1 is performed many times, **the computed values of the average will be distributed according to a normal distribution**.

Example: Flip a coin many times. The probability of getting a given number of heads in a series of **K** flips will approach the normal distr. with mean =K/2

***P*-value:** $P = \Pr(Z \geq 2.56) = 0.0052$

Sample distribution follows the
Normal distribution
according to the **Central Limit Theorem**



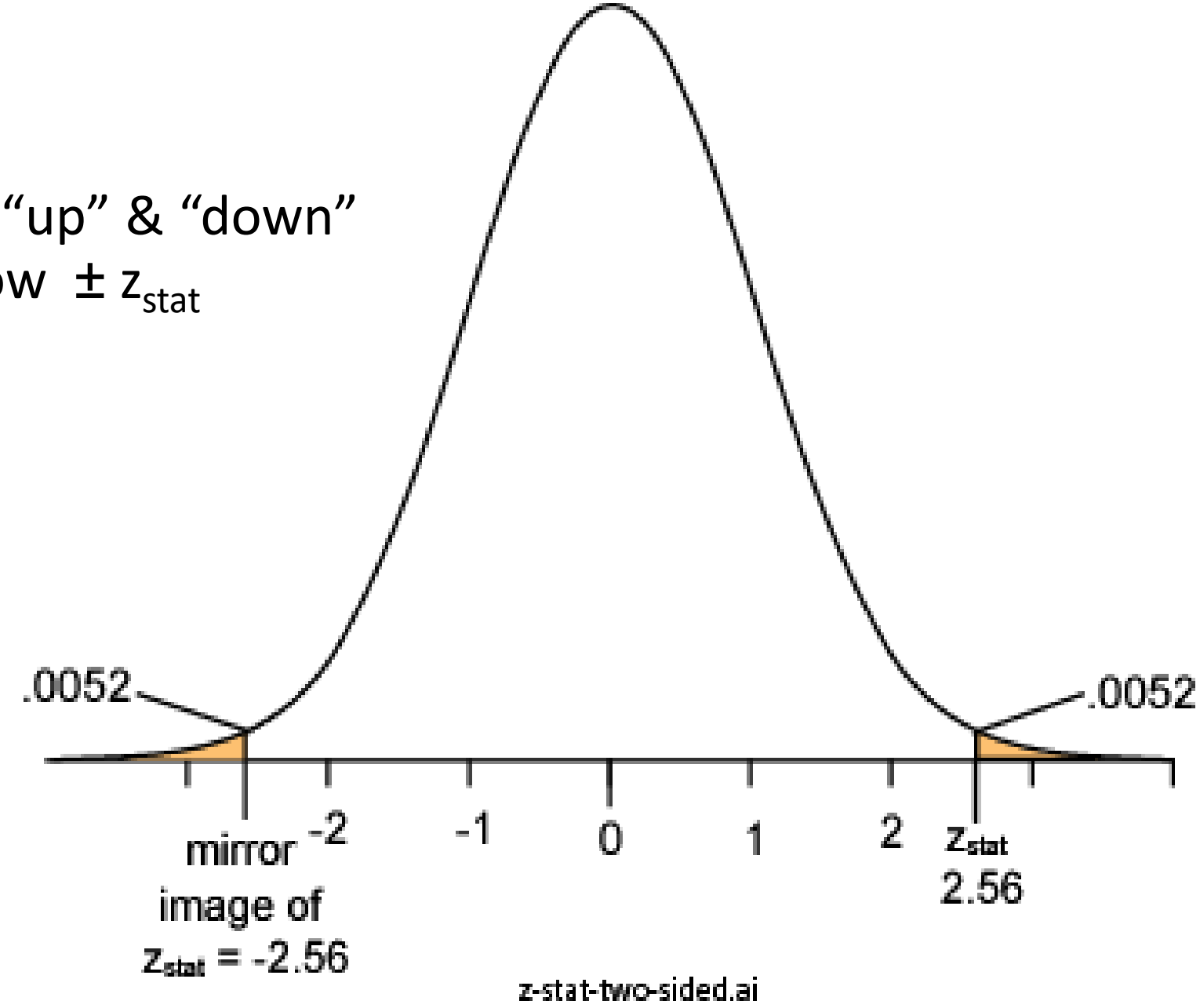
***P* =.0052** ⇒ it is unlikely the sample came from this null distribution ⇒ strong evidence **against H_0**

Example - Two-Sided *P*-value: Lake Wobegon

- H_a: $\mu \neq 100$

Considers random deviations "up" & "down" from $\mu_0 \Rightarrow$ tails above & below $\pm z_{stat}$

Thus, **two-sided P**
= 2 × 0.0052
= 0.0104



Conditions for z Test

1. **Population approximately Normal** or **large sample** (central limit theorem)
2. The **population variance is known!**

If the **population variance is unknown** (and therefore has to be estimated from the sample itself) & the **sample size is not large** ($n < 30$), the **Student's *t*-test** may be more appropriate.

Another Example

- Background knowledge: Breast Cancer is related to mutations in genes BRCA1 & BRCA2
- Hypothesis: Gene G is expressed differently in breast cancer patients with mutation in BRCA1 than BRCA2
- Data: Obtained 7 patients with BRCA1 mutation & 8 with BRCA2 mutation

Patient number	Expression Level of Gene G x_i^1	Have mutation in BRCA1 or not x_i^2
1 (x_1)	98.2244	1
2	69.6810	1
3	118.4339	1
4	115.2322	1
5	150.7729	1
6	117.7385	1
7	80.6921	1
8	142.8455	2
9	156.8692	2
10	151.9287	2
11	147.3357	2
12	131.2094	2
13	150.3127	2
14	147.0670	2
15 (x_{15})	122.3306	2

Hedenfalk et al. N Engl J Med. 2001 Feb 22;344(8):539-48.

1. Form the Null Hypothesis

- Gene G is expressed differently in breast cancer patients with mutation in BRCA1 than BRCA2

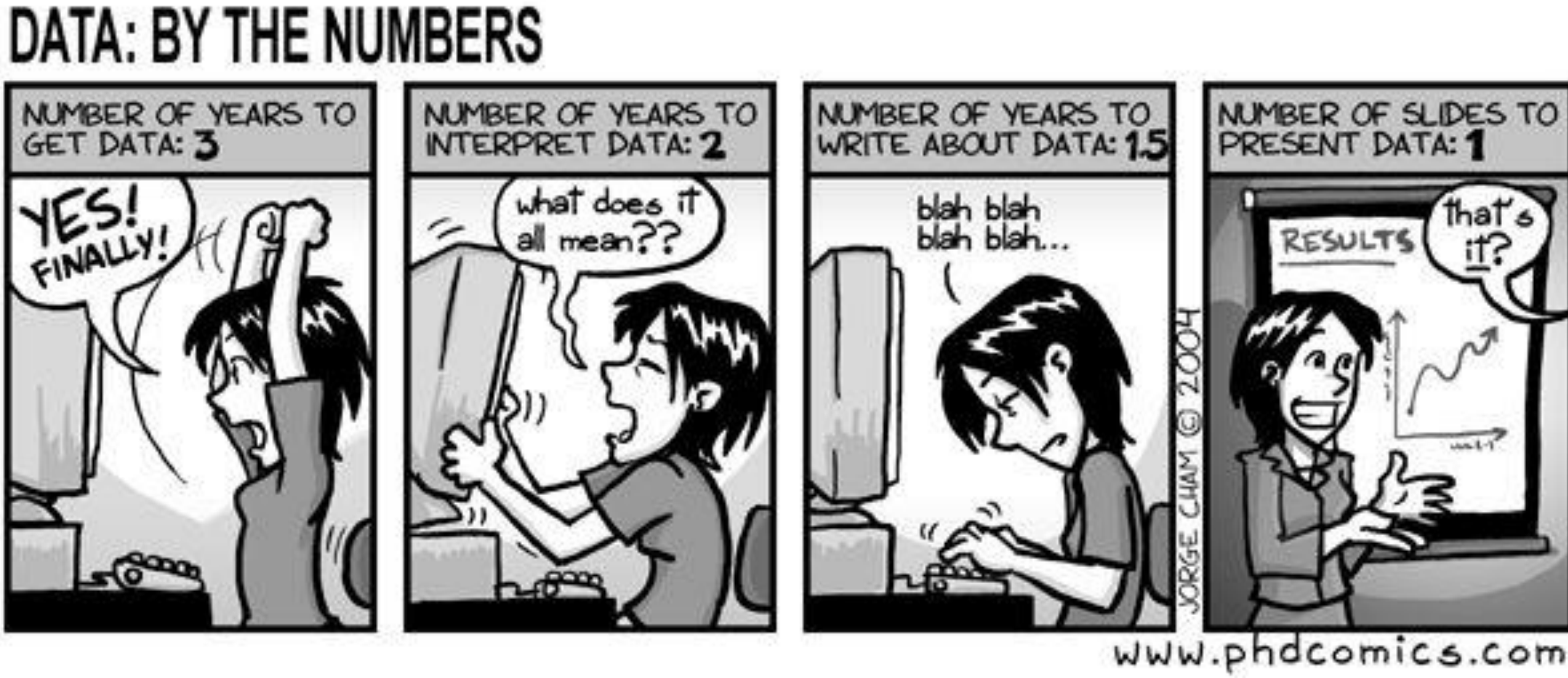
Mathematically

- μ_1 : be the mean expression level of gene G in patients with BRCA1 mutation
- μ_2 : be the mean expression level of gene G in patients with BRCA2 mutation

$H_0 : \mu_1 = \mu_2$

$H_1 : \mu_1 \neq \mu_2$

2. Obtain data….



3. Find a suitable test statistic T (Example)

$$T(x) = \frac{m_1 - m_2}{s\sqrt{\frac{1}{k} + \frac{1}{l}}}$$

Unpaired Two Sample t-test

- The larger the difference of the two means, the larger the statistic
- The larger our sample, the larger the statistic
- The smaller the sample variance, the larger the statistic

So ***T*** **will be quite large (in absolute value), when we can confidently say** H_0 **does not hold**

3. Find a suitable test statistic T (cont'd)

$$T(x) = \frac{m_1 - m_2}{s\sqrt{\frac{1}{k} + \frac{1}{l}}}$$

Unpaired Two Sample t-test

$$k = \#\{x_i : x_i^2 = 1\}, \quad l = \#\{x_i : x_i^2 = 2\}$$

$$m_1 = \frac{1}{k} \sum_{\{x_i : x_i^2 = 1\}} x_i^1, \quad m_2 = \frac{1}{l} \sum_{\{x_i : x_i^2 = 2\}} x_i^1, \quad m = \frac{1}{k+l} \sum_{\{x_i\}} x_i^1$$

$$s = \sqrt{\frac{1}{k+l-1} \sum (x_i^1 - m)^2}$$

3. Find the distribution of *T* (cont'd)

For the test of this specific example, we will make the following assumptions:

a) The **data in both groups are distributed normally** around a mean value μ_1 , μ_2 respectively
b) Their **variance is the same in both groups**
c) Each patient was **sampled independently**

and most importantly that **THE NULL HYPOTHESIS HOLDS**

This is an assumption for ALL tests!

Then

T(X) has a probability density function of:

$$p(t \mid H_0) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-\left(\frac{\nu+1}{2}\right)}$$

where the **degrees of freedom of the test** ***v*** is
15 – 2 = 13 (number of patients – 2)

The *t*-statistic was introduced in 1908 by William Sealy Gosset, a chemist working for the Guinness brewery in Dublin. "Student" was his pen name.

Sampling distribution [edit]

One sample T-distribution

Let $x_1, \ldots, x_n$ be the numbers observed in a sample from a continuously distributed population with expected value μ

$$\bar{x} = \frac{x_1 + \cdots + x_n}{n},$$

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

The resulting *t-value* is

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}}.$$

The *t*-distribution with $n - 1$ degrees of freedom is the sampling distribution of the *t*-value when the samples consist
distributed population. Thus for inference purposes *t* is a useful "pivotal quantity" in the case when the mean and var

Student t-distribution (basics)

Probability density function [edit]

Student's ***t*-distribution** has the probability density function given by

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\,\Gamma(\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}},$$

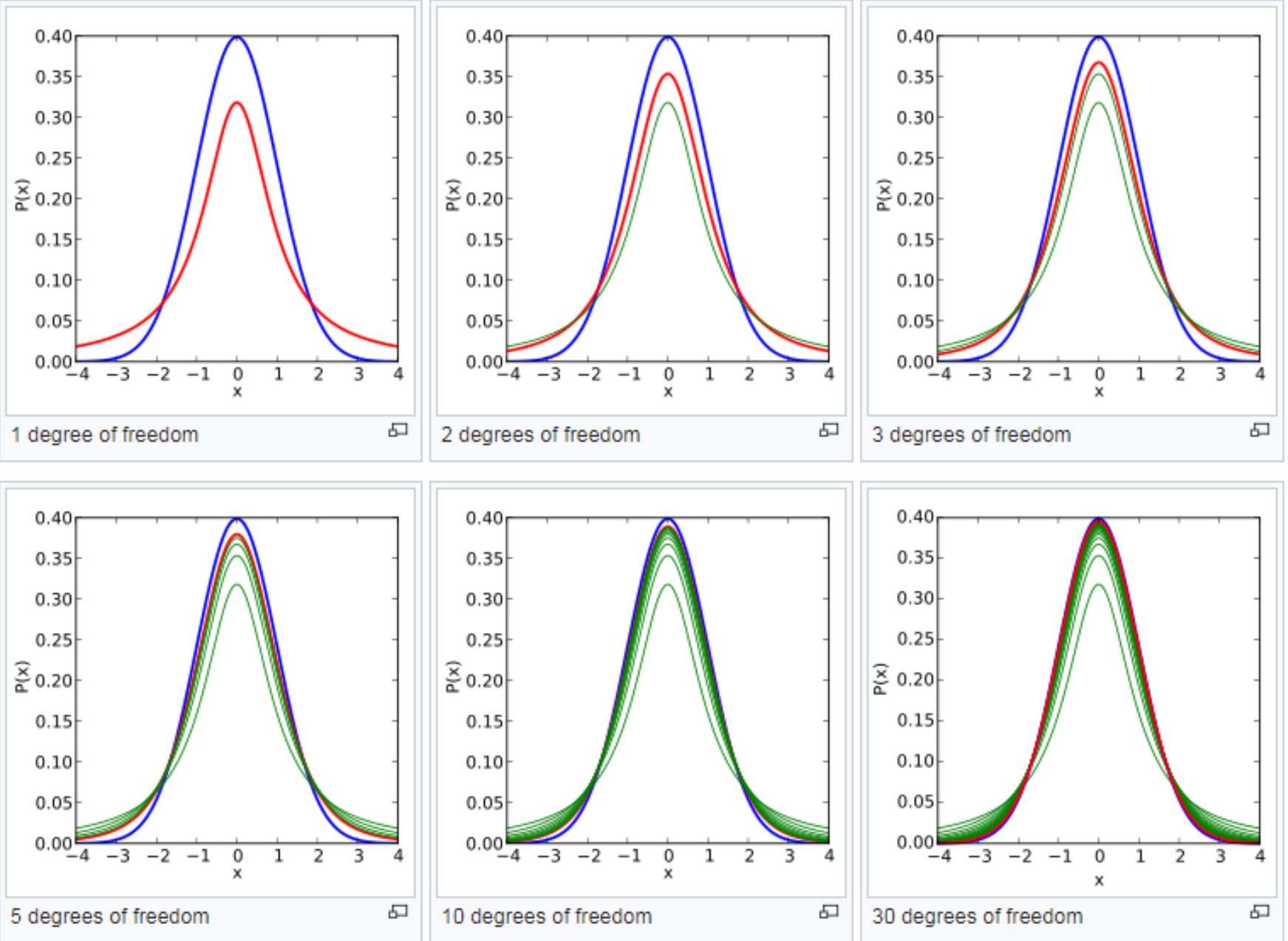
where ν is the number of *degrees of freedom* and Γ is the gamma function. This may also be written as

$$f(t) = \frac{1}{\sqrt{\nu}\,\mathrm{B}(\frac{1}{2},\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}},$$

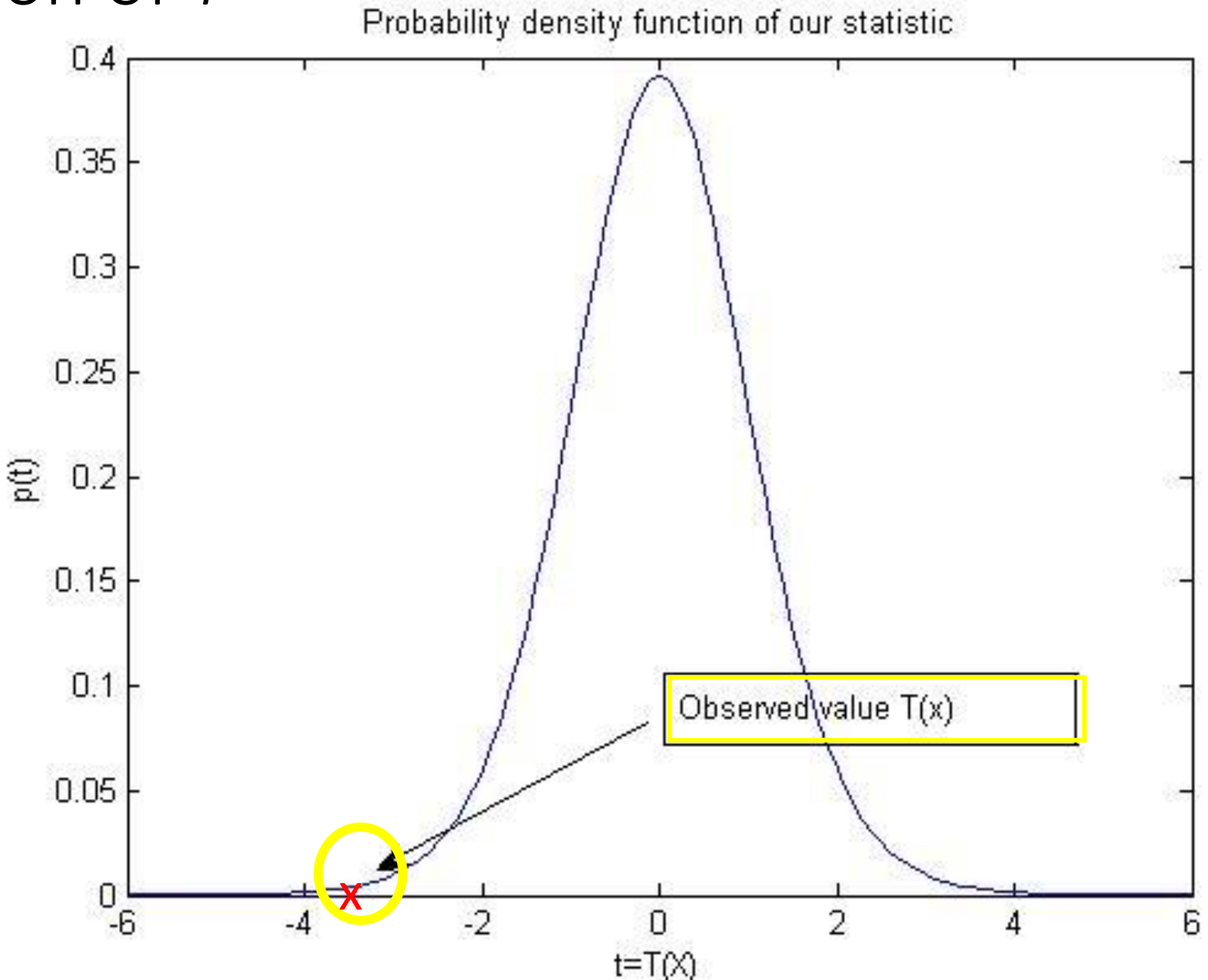
t-distribution (basics)

Density of the *t*-distribution (red) for 1, 2, 3, 5, 10, and 30 degrees of freedom compared to the standard normal distribution (blue).

Previous plots shown in green.



3. Find the distribution of *T*
Example



4. Decide on a Rejection Region

- Decide on a **rejection region Γ** in the range of our statistic
- **If $t_o \in \Gamma$, then reject H_0**
- If $t_o \notin \Gamma$, then ***do not* reject H_0**
  accept H_1 ?

Since the pdf of T ***when the null hypothesis holds is known,***

$P(T \in \Gamma \mid H_0)$ can be calculated

4. Decide on a Rejection Region

- If $P(T \in \Gamma \mid H_0)$ **is too low**, we know we are **safely rejecting H_0**

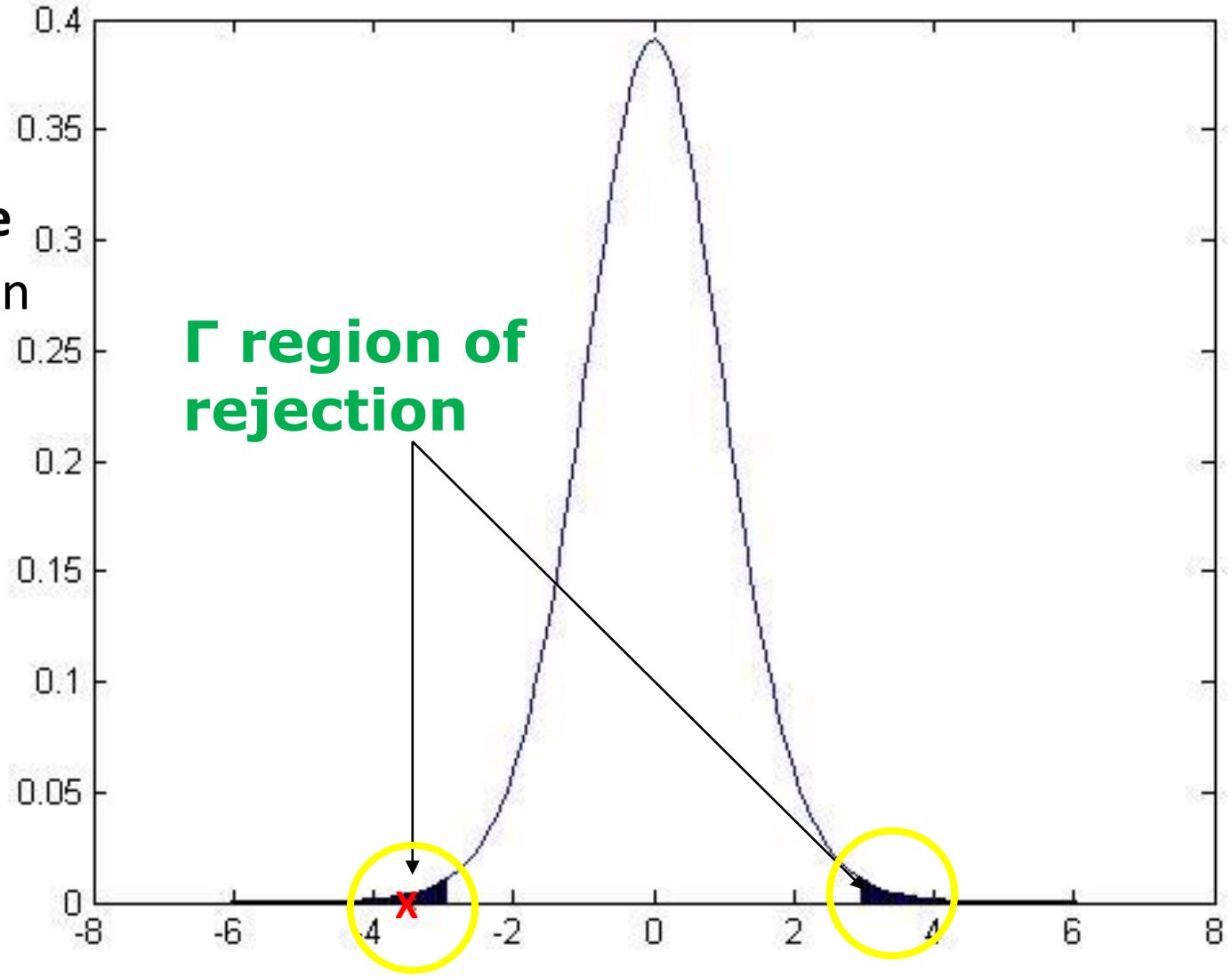
- What should be our rejection region in our example?

4. Decide on a Rejection Region

Where **extreme values of t_o** are:

- **unlikely to come from when H_0 is true**
- could come with high probability, when H_0 is false

$P(T \in \Gamma \mid H_0)$ is the area of the shaded region (can be calculated)



Rejection Procedure

- Pre-select a probability threshold α
- Find a **rejection region** $\Gamma = \{ t: |t|>c \}$, such that $\mathbf{P(T \in \Gamma \mid H_0) = \alpha}$
- Decide
  - **Reject** H_0, if $t_o \in \Gamma$ (recall: t_o **is the observed** T in our data)
  - ***Accept*** H_0, otherwise

What values do we usually use for α in science?

0.05 is typical

Smaller ones are also used: 0.01 , 0.001

When $t_o \in \Gamma$ we say the finding is statistically significant at significance level α

Issues to be Considered

- When there exist two or more tests that are appropriate in a given situation,

  how can the tests be compared to decide which should be used?

- If a **test is derived under specific assumptions** about the **distribution of the population being sampled**,

  how well will the test procedure work **when the assumptions are violated?**

Parametric versus non-Parametric Tests

- **Parametric test**

  Makes the assumption that the **data are sampled** from a **particular class of distributions**

  It then becomes **easier to derive the distribution of the test statistic**

- **Non-Parametric test**

  No assumption about a particular class of distributions

Permutation Testing

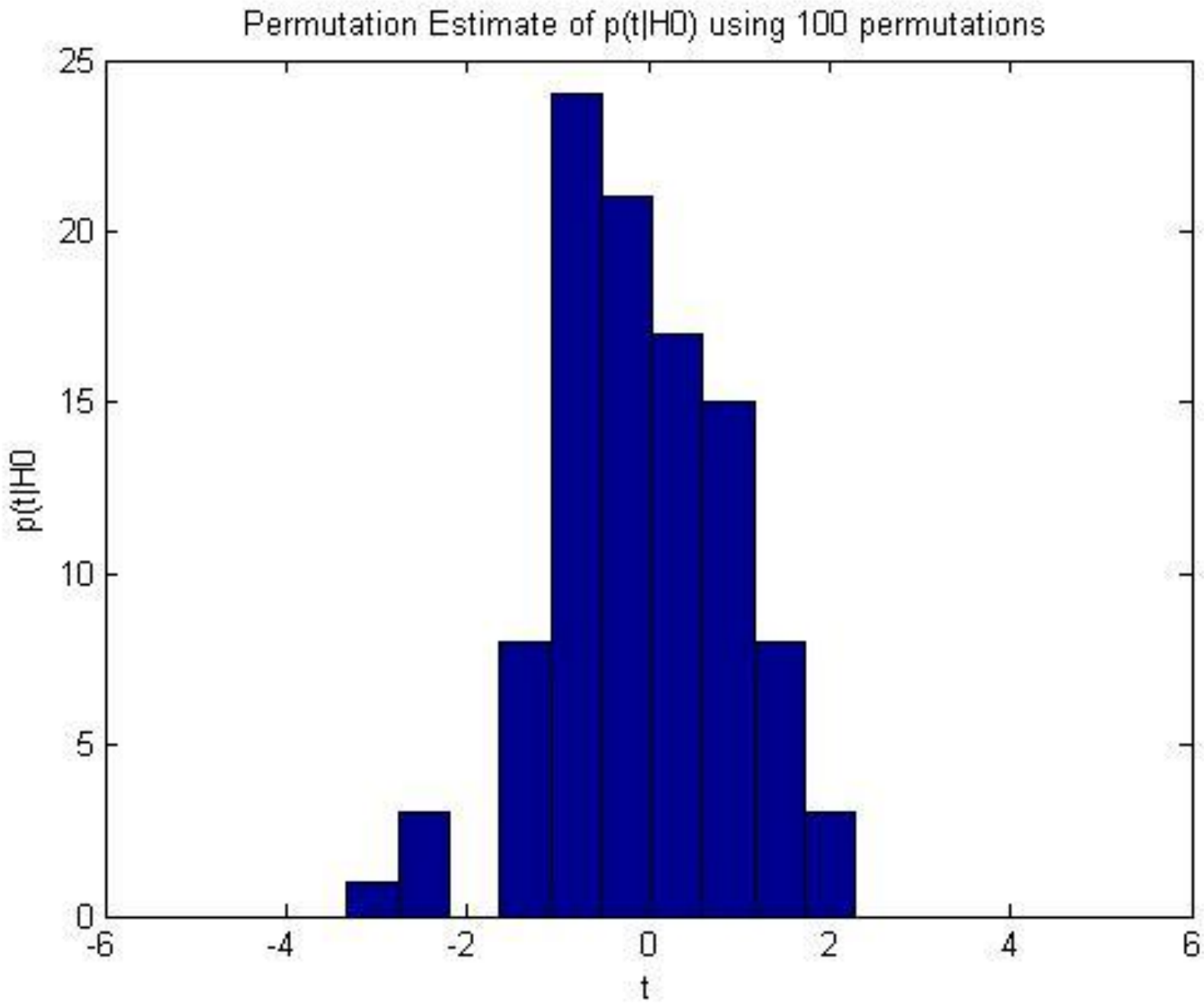
- Often in biological data, we do ***not know much*** **about the data distribution**
- How do we obtain the distribution of our test statistic?
- Great idea in statistics: permutation testing
- Recently practical because it requires computing power (or a lot of patience)

Permutation Testing

1. In our first example, we want to calculate $p\ (\ t\ |\ H_0\ )$
2. **If H_0**, then it does ***not*** matter which group each value x_i^1 comes from
3. Then, if we **permute the group labels**, we would get a value for our test statistic **given the null hypothesis holds**
4. If we get a lot of such values, we can estimate (approximate) $p(\ t\ |\ H_0\ )$

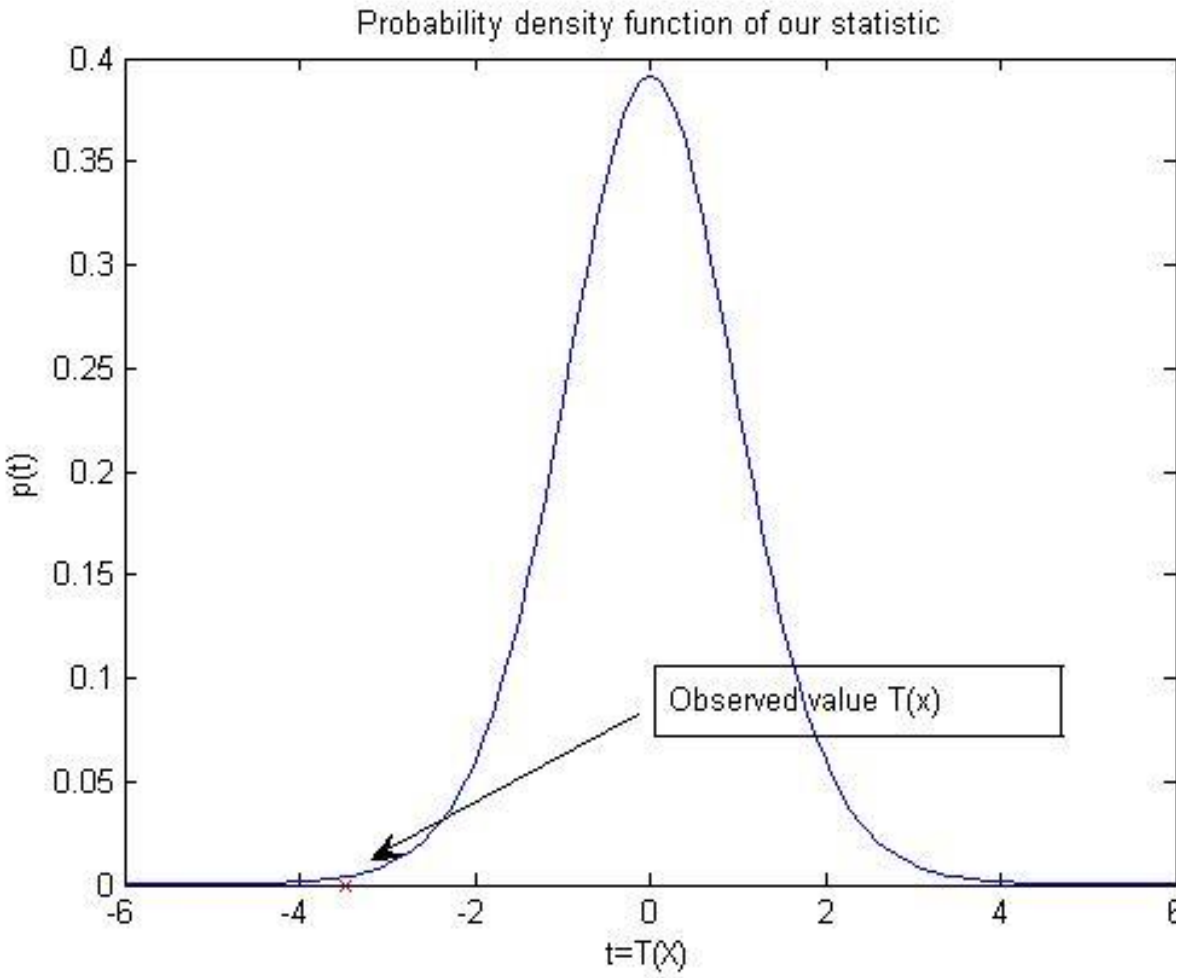
Permutation Testing Revisited

- Decide what can be permuted, if the null hypothesis is true
- For all (as many as possible) permutations of the data, calculate the **test statistic on the permuted data: t_B**
- Estimated p-value = $\#\{\ |t_B| \geq |t_o|\ \} / \#B$

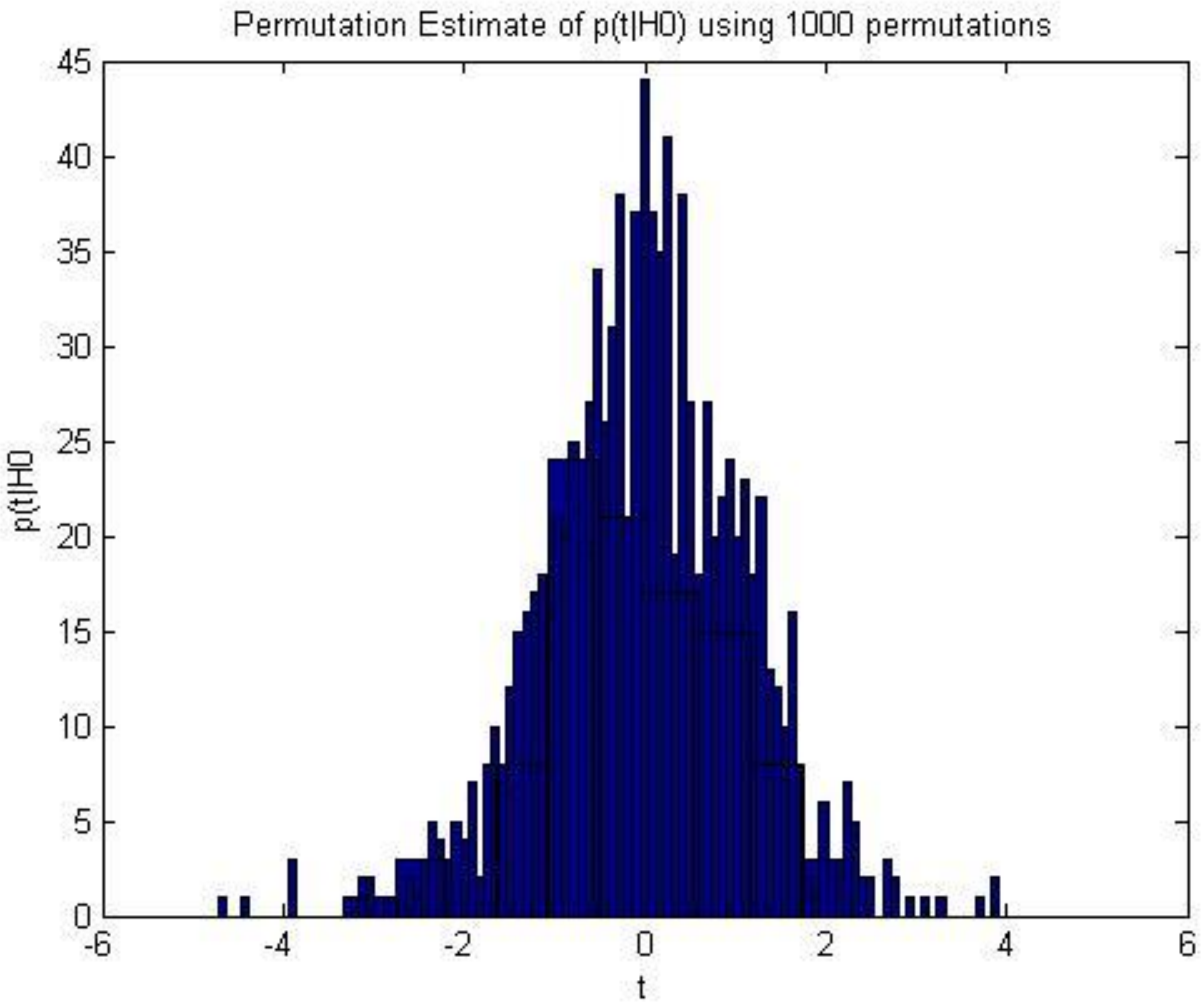


Estimated distribution from our data: **100 permutations**

Does It Really Work?

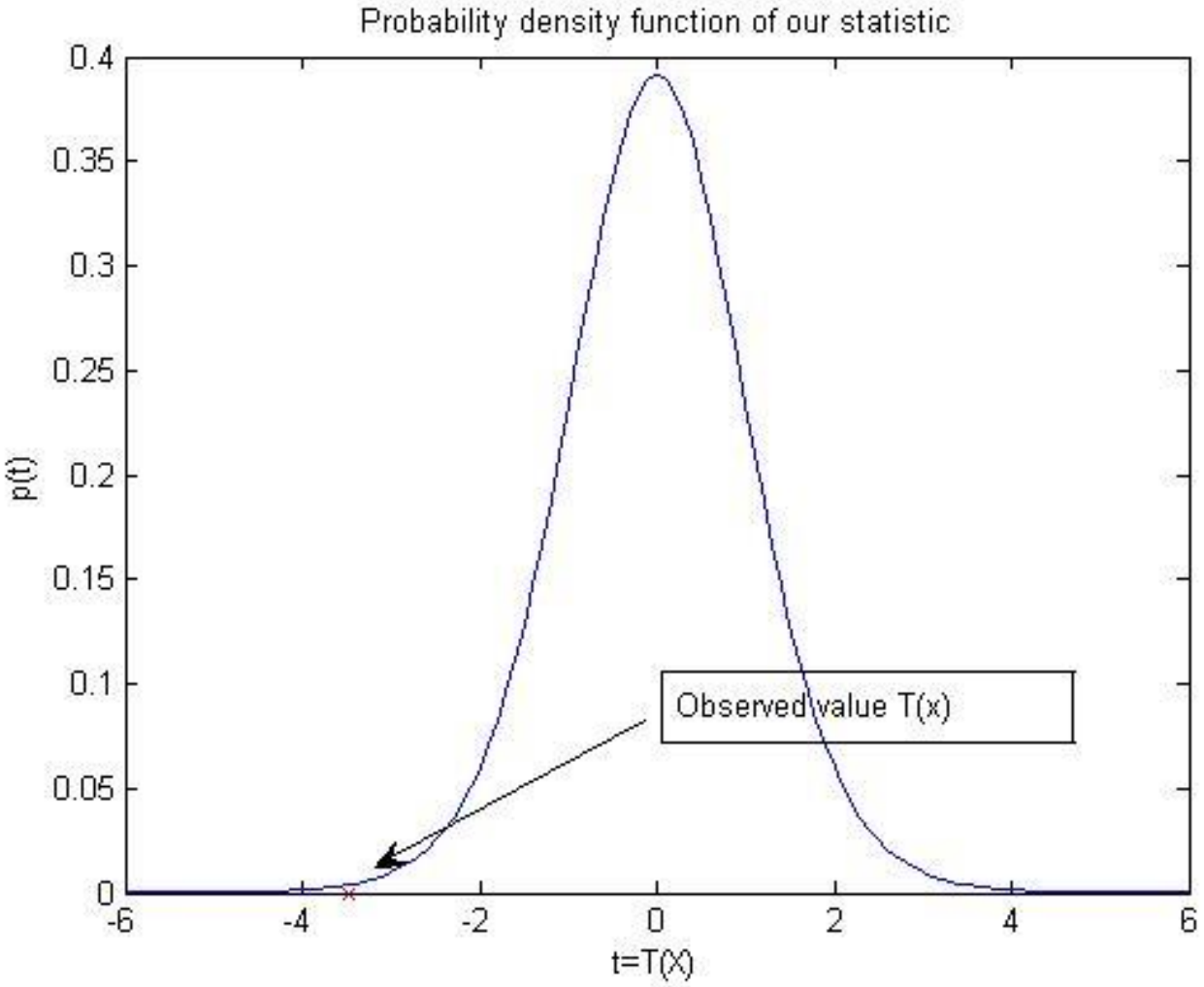


True distribution **calculated theoretically**

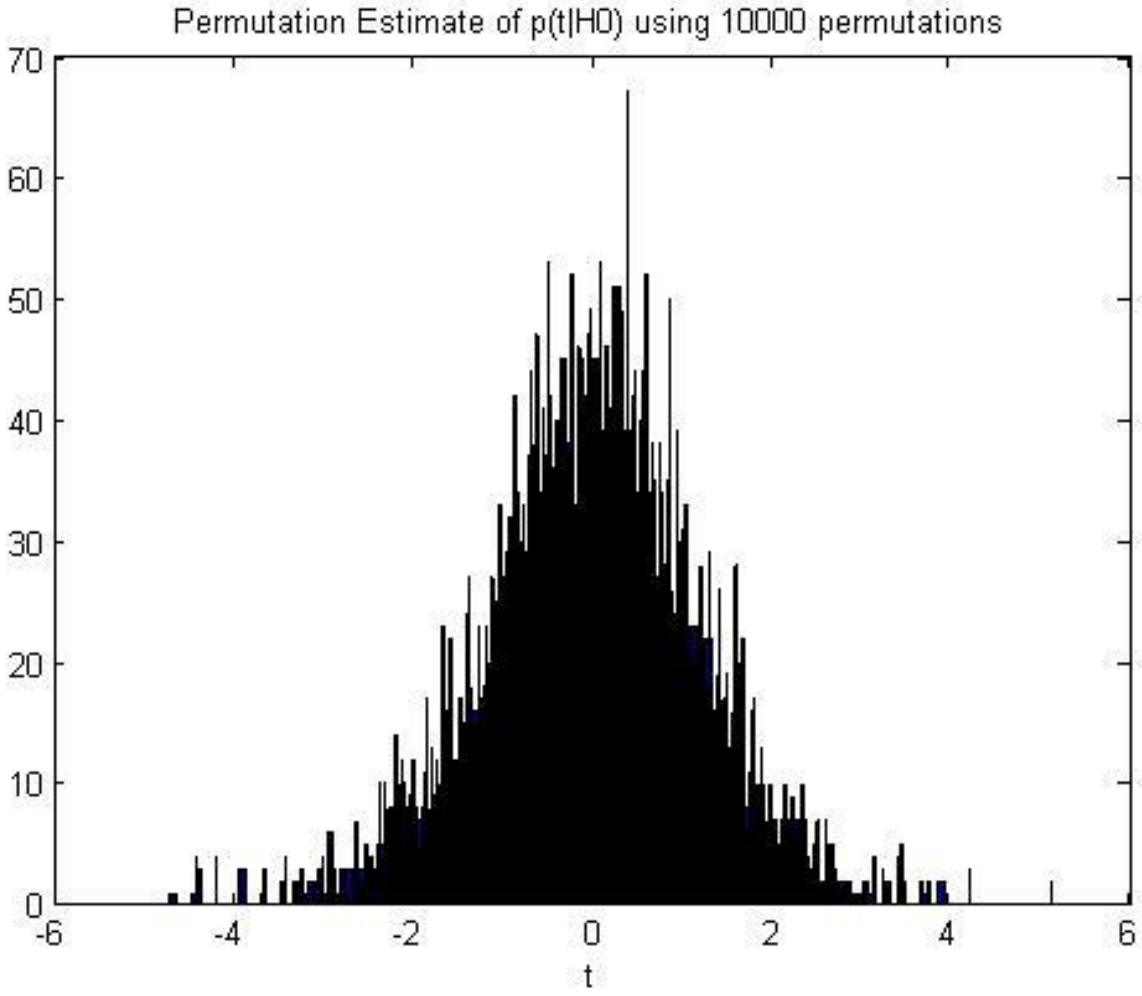


Estimated distribution from our data: **1,000 permutations**

Does It Really Work?

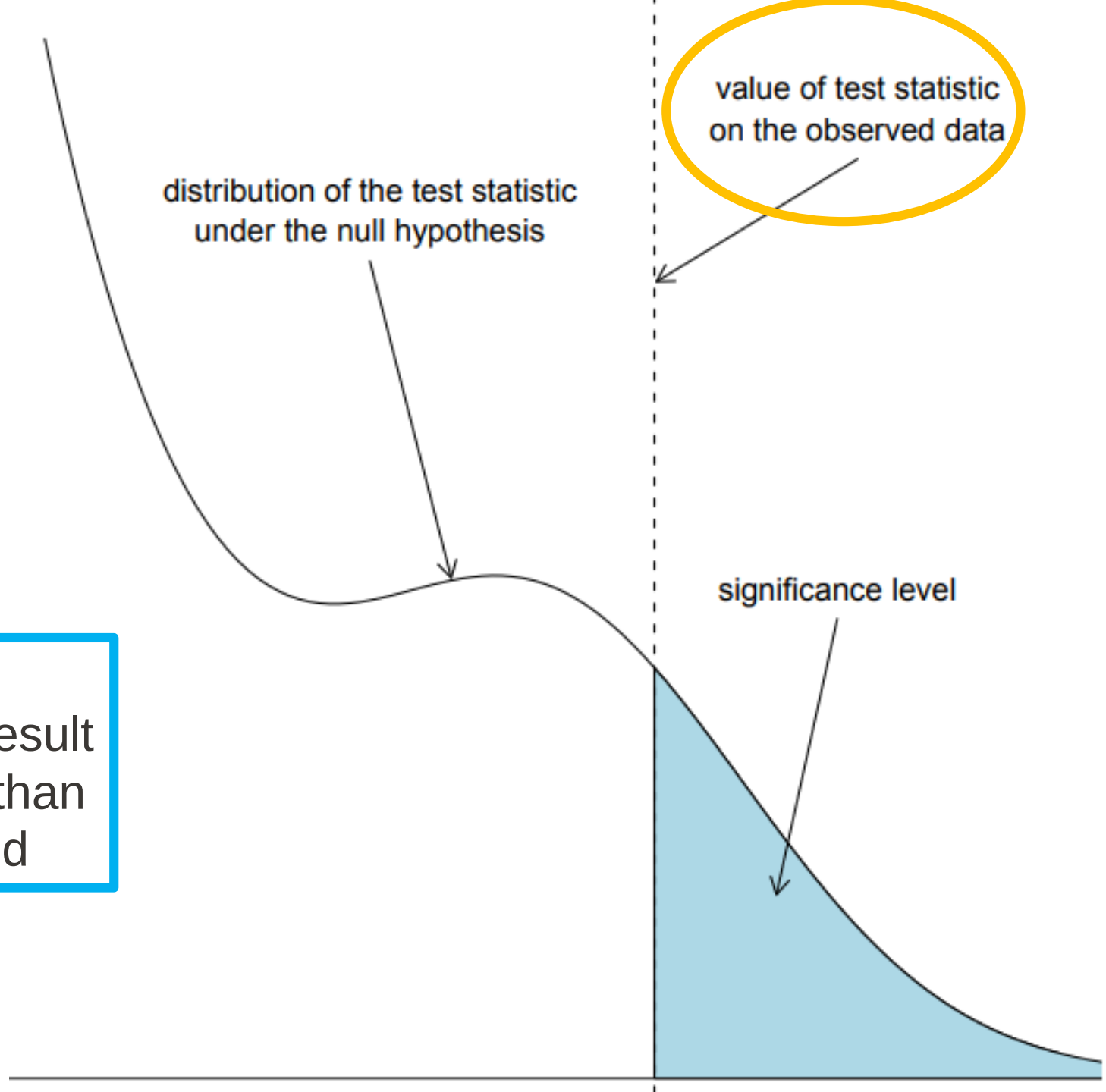


True distribution calculated theoretically



Estimated distribution from our data: **10,000 permutations**

p-value is defined as the probability of obtaining a result equal to or more extreme than what was actually observed



The Significance Level

- The area to the right of $t(o_A, o_B)$ is the "significance level"—the probability that some $t^* \geq t(o_A, o_B)$ would be generated *if the null hypothesis were true*.
  - Also called the p-value.

Small values suggest the null hypothesis is false, given the observation of $t(o_A, o_B)$.

- Corollary: all else being equal, a large difference between $e(o_A)$ and $e(o_B)$ yields a smaller significance level (as one would hope!).
- Values below 0.05 are typically considered "good enough."

So all we have to do is calculate the distribution of t.

Calculating the Distribution

The classical approach:

- Keep adding assumptions until we arrive at a known distribution which we can calculate analytically.
- E.g.: Student's t-test.
  - Assume that $e(o_A)$ and $e(o_B)$ are sample means from a bivariate Normal distribution with zero covariance. Then we know t is distributed according to Student's t-distribution if the null hypothesis is true.
- Back in the stone age, computing with rocks and twigs, making those assumptions made the problem tractable.
- But the problem with this approach is that you may falsely reject the null hypothesis if one of the additional assumptions is violated. (Type I error.)

What you SHOULD do

- Simulate the distribution using a *randomization* test.
- It's just as good as analytical approaches, even when the analytical assumptions are met! (Hoeffding 1952)
- And it's better when they're not. (Noreen 1989)
- Best of all: dirt simple.

Intuition:

- Erase the labels "output of A" or "output of B" from all of the observations.
- Now consider the population of every possible labeling. (Order relevant.)
- If the systems are really different, the observed labeling should be unlikely under this distribution.

Statistical Errors

- **Type 1 Errors**

-Rejecting H_0 when it is actually true

-Concluding a difference when one does ***not actually exist***

- **Type 2 Errors**

-Accepting H_0 when it is actually false (e.g. previous slide)

-Concluding no difference when **one does exist**

Errors can occur due to **biased/inadequate sampling, poor experimental design** or the use of **inappropriate/non-parametric** tests.

Regarding the Choice of a Test

When we cannot reject H_0, it does *not* mean H_1 holds!

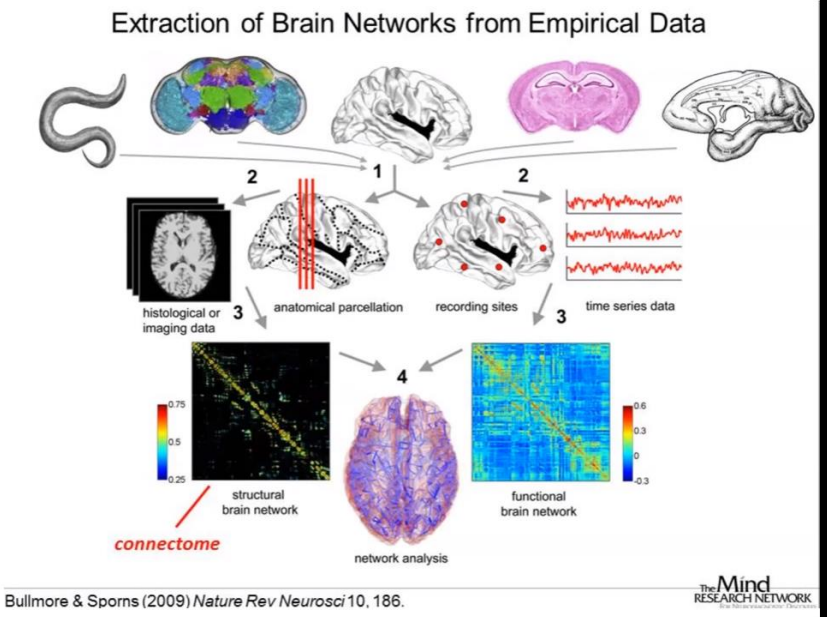
- It could be that we do **not have enough power**, i.e.,

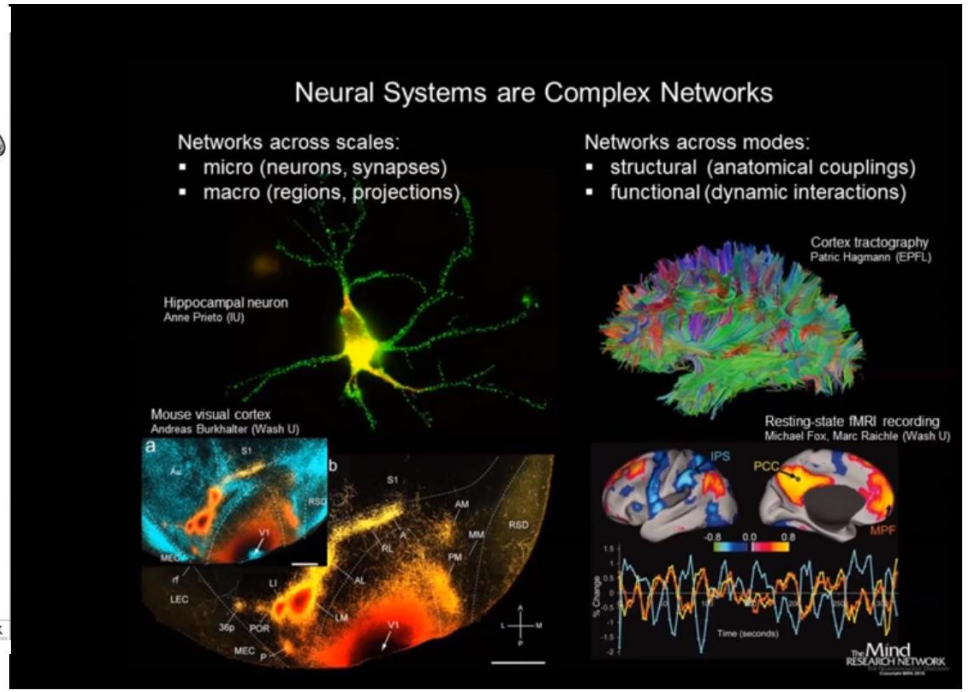
  H_1 is not that "**different enough**" from H_0 to distinguish it with the **given sample size**

  of all possible tests for a hypothesis choose the one with the maximum power

Power analysis methods need to be employed.


EVERYBODY WHO WENT TO THE MOON HAS EATEN CHICKEN!
GOOD GRIEF. CHICKEN MAKES YOU GO TO THE MOON!





Lecture on Modeling Tools for Assessing Temporal Correlation

CS – 590.21 Analysis and Modeling of Brain Networks

Department of Computer Science
University of Crete

Greek Diaspora Fellowship Program
ΙΣΝ/SNF
ΙΔΡΥΜΑ ΣΤΑΥΡΟΣ ΝΙΑΡΧΟΣ
STAVROS NIARCHOS FOUNDATION

Challenges in Quantifying Correlation

1. Correlated neurons fire at **similar times but not precisely synchronously**, so correlation must be defined with **reference to a timescale** within which spikes are considered correlated
2. Spiking is sparse with respect to the recording's sampling frequency & spike duration

   e.g., spiking rate 1 Hz, sampling rate typically 20 kHz (Demas et al., 2003)

This means that conventional approaches to correlation (such as Pearson's correlation coefficient) are unsuitable

- **as periods of quiescence should not count as correlated**
- correlations should **compare spike trains over short timescales, not just instantaneously.**

Pearson Correlation of two variables X & Y ($\rho_{X,Y}$)

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

where:

- cov is the covariance
- σ_X is the standard deviation of X
- σ_Y is the standard deviation of Y

ne formula for ρ can be expressed in terms of mean and expectation. Since

$$\operatorname{cov}(X,Y) = \mathbf{E}[(X-\mu_X)(Y-\mu_Y)],$$[5]

e formula for ρ can also be written as

$$\rho_{X,Y} = \frac{\mathbf{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

where:

- cov and σ_X are defined as above
- μ_X is the mean of X
- $\mathbf{E}$ is the expectation.

Sample Pearson correlation coefficient

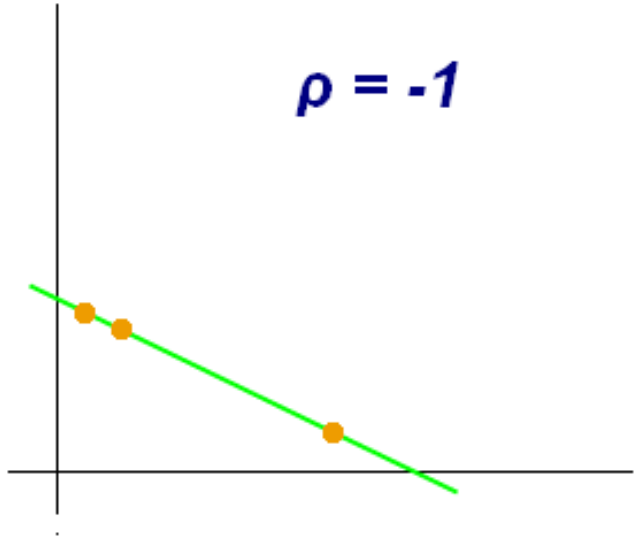
$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

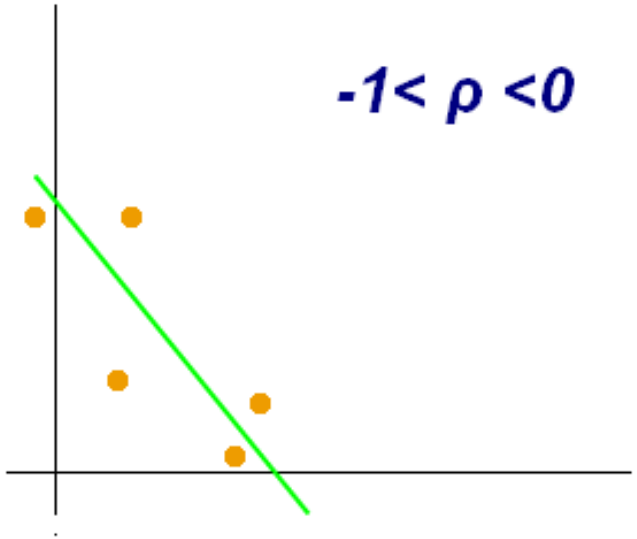
Datasets $\{x_1,\ldots,x_n\}$ & $\{y_1,\ldots,y_n\}$ containing *n* values

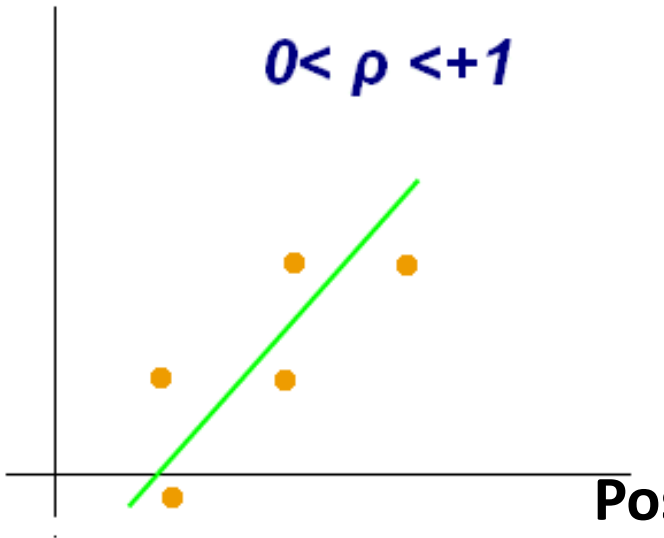
where:

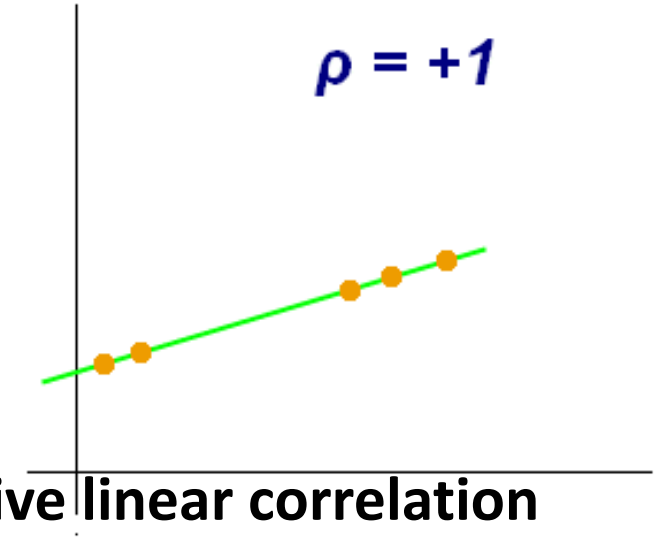
- n is the sample size
- x_i, y_i are the single samples indexed with i
- $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ (the sample mean); and analogously for $\bar{y}$

Pearson correlation: widely-used measure of the **linear correlation** between variables

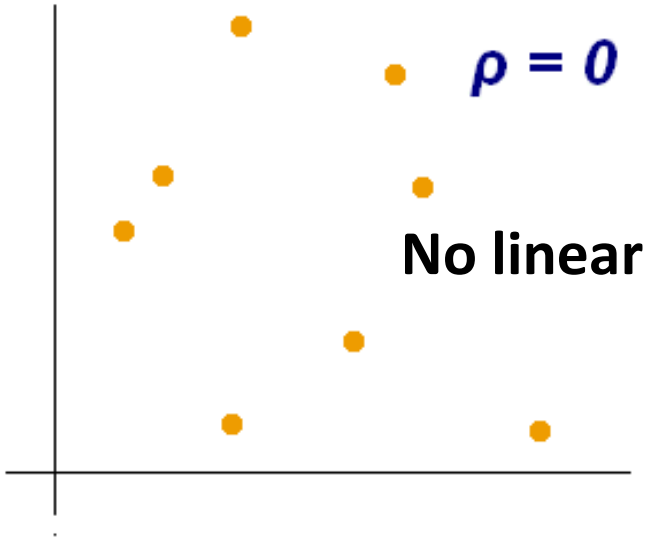






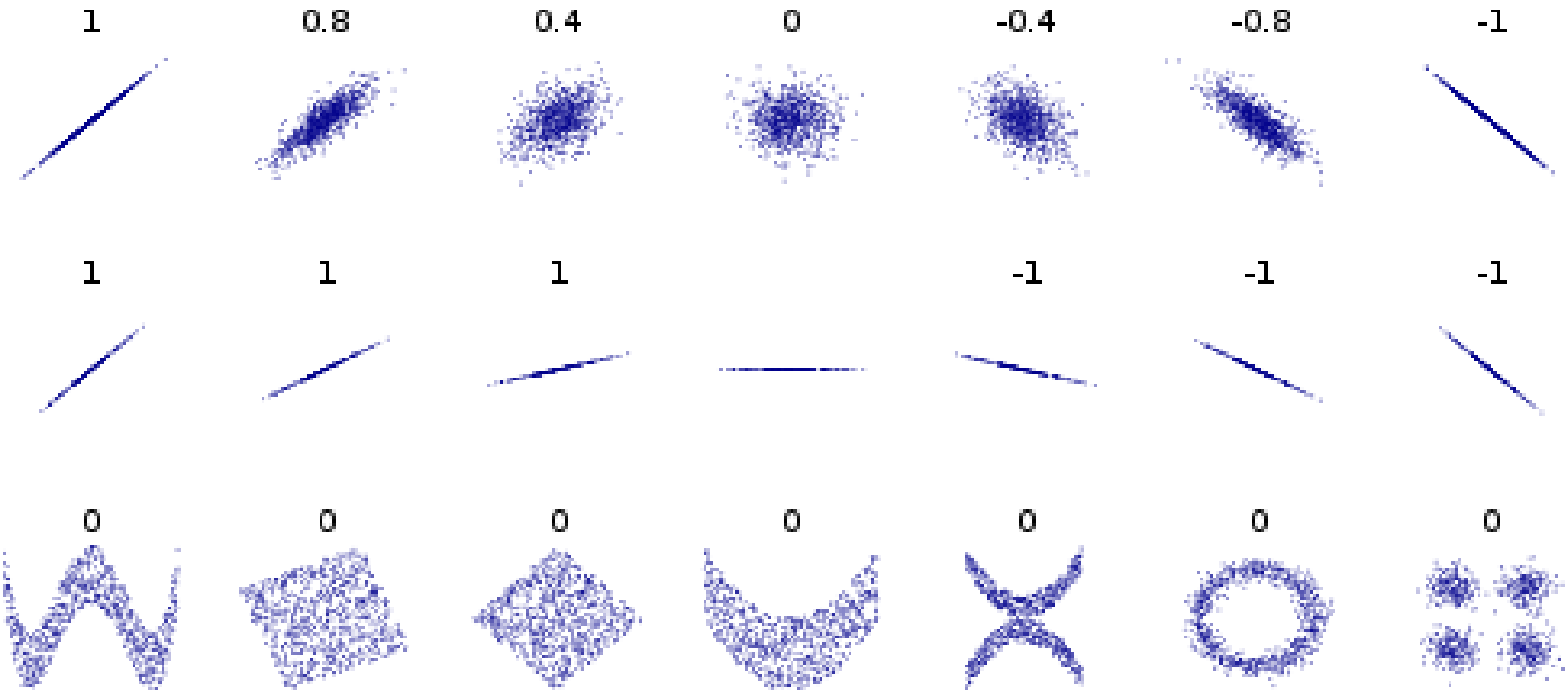


Positive linear correlation



No linear correlation

Examples of Pearson Correlation



Quantification of Correlation between Neural Spike Trains

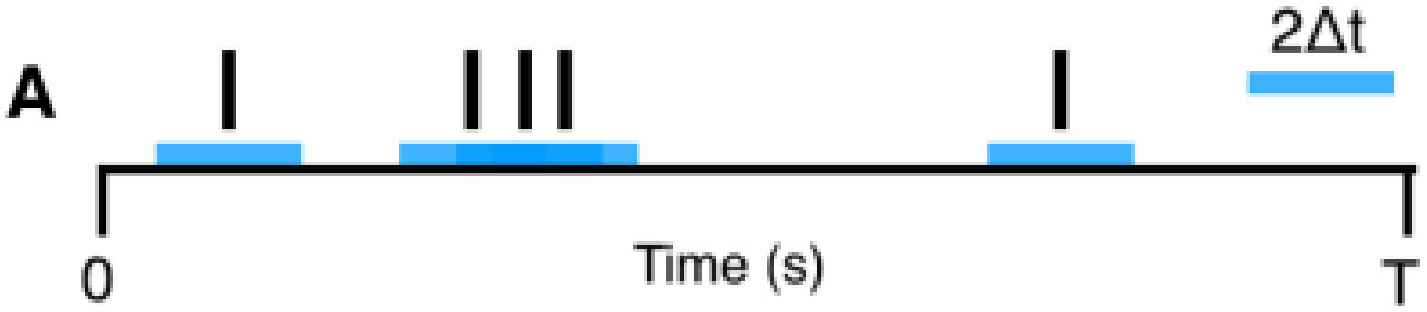
- Key part of the analysis of experimental data
- Neural coordination is thought to play a key role in
  - information propagation & processing
  - self-organization of the neural system during development

Designing the Appropriate Temporal Correlation Metric

- Symmetry
- Treatment of **idle periods**
- **Robustness to variations in firing rate**

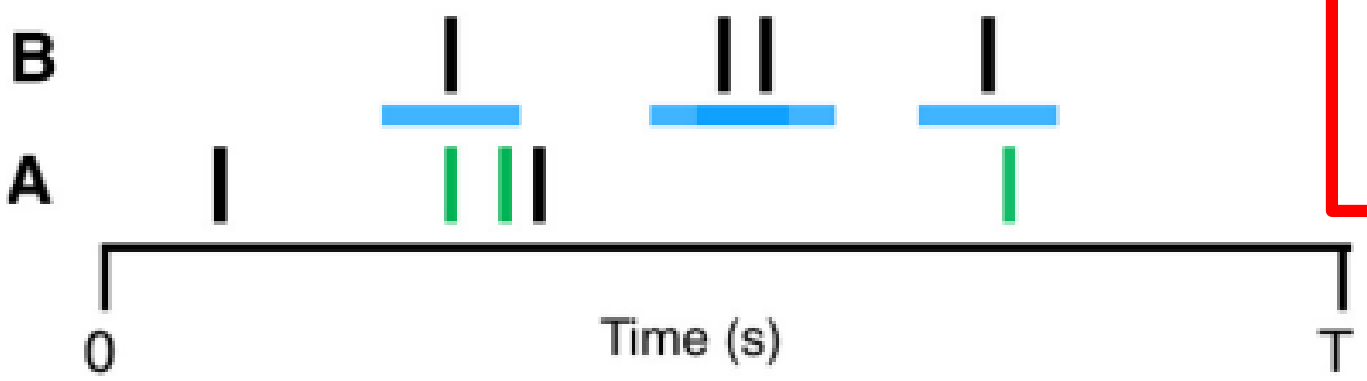
  e.g., doubling the firing rate of two spike trains with a **specific firing structure**, does their correlation remain the same?
- Robust to the **recording duration**
- **Bounded**
- Distinction of the correlation vs. no correlation vs. anti-correlation
- **Minimal assumptions on the underlying structure/distribution of the events**

T_A: the proportion of total recording time which lies within ±Δt of any spike from A. **T_B** calculated similarly.



T_A is given by the fraction of the total recording time (black) which is covered (tiled) by blue bars. Here T_A is 1/3.

P_A: the proportion of spikes from A which lie within ±Δt of any spike from B. **P_B** calculated similarly.



P_A is the number of green spikes in A (3) divided by the total number of spikes in A (5). Here P_A is 3/5.

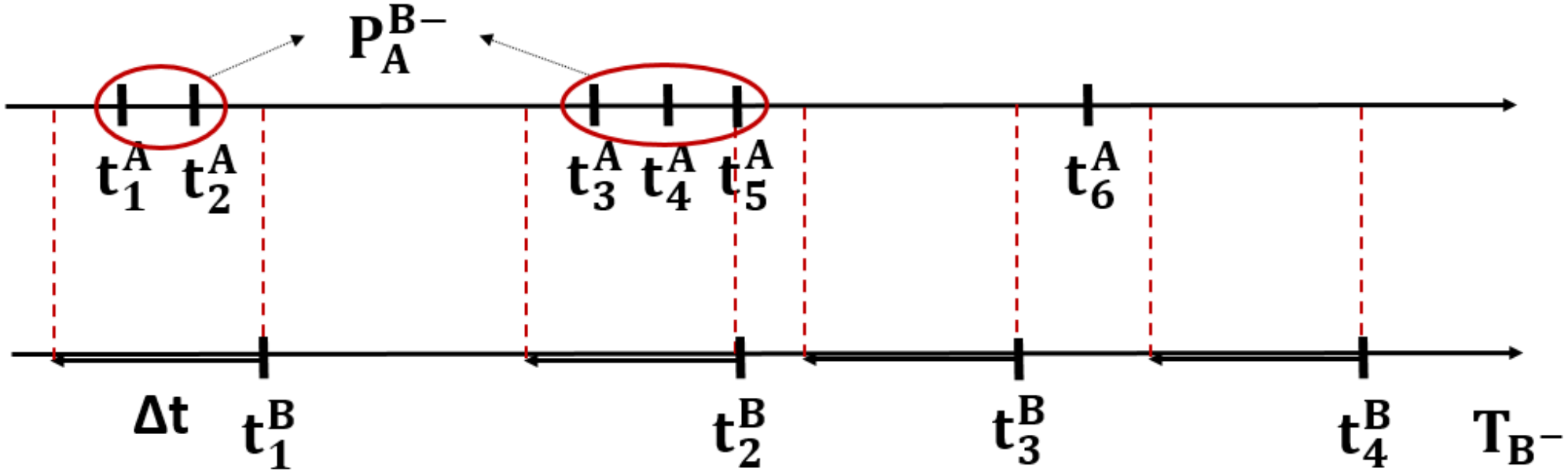
$$STTC = \frac{1}{2}\left(\frac{P_A - T_B}{1 - P_A T_B} + \frac{P_B - T_A}{1 - P_B T_A}\right)$$

Directional STTC
Temporal Correlation Metric

Extended STTC metric to take into consideration the temporal **order** of the correlation of the spike trains of two neurons

Directional $STTC_{AB}$ represents a measure of the chance that firing events of A will **precede** firing events of B

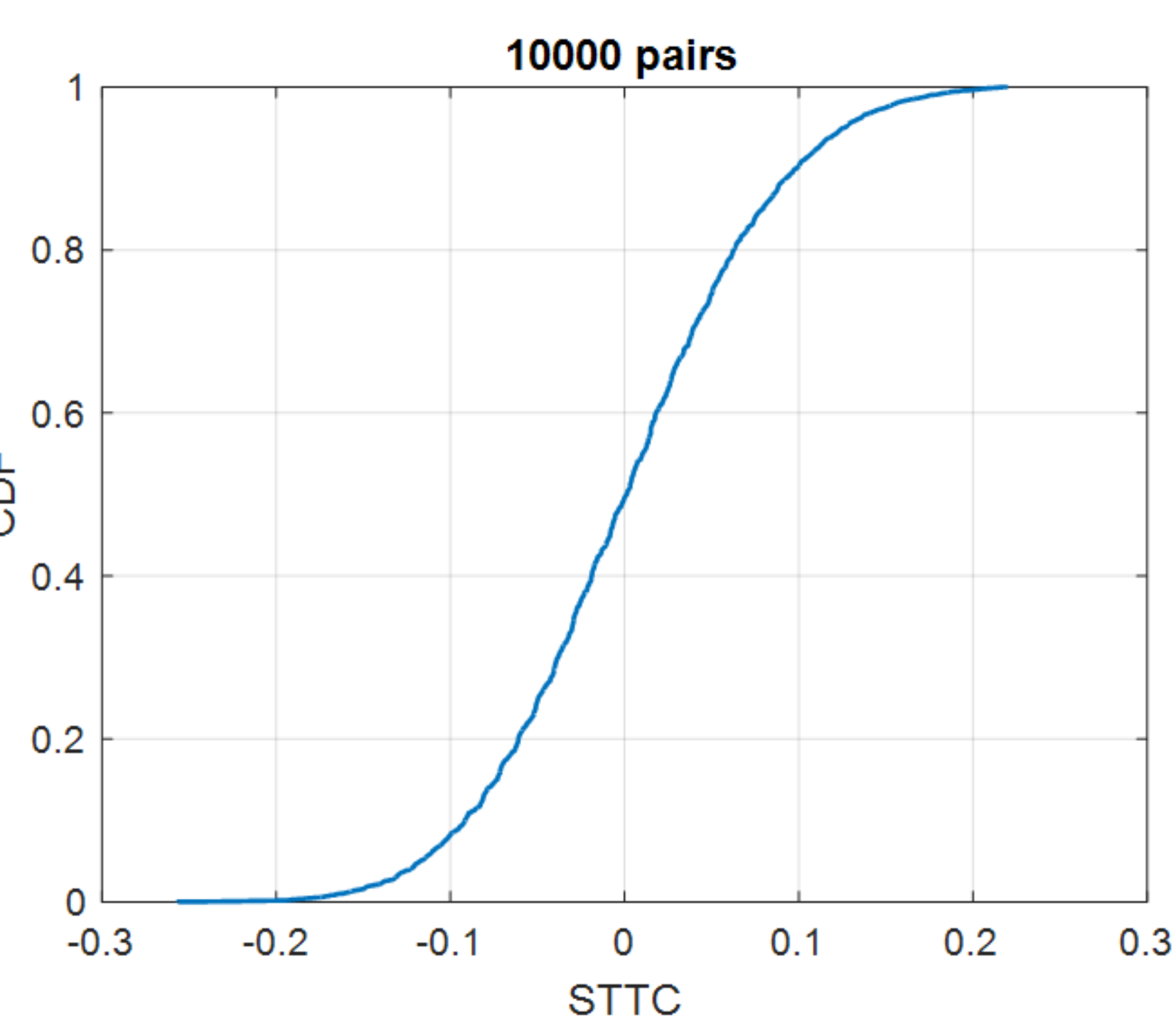
$$STTC_{AB} = \frac{1}{2}\left(\frac{P_A^{B-} - T_{B^-}}{1 - P_A^{B-} T_{B^-}} + \frac{P_B^{A+} - T_{A^+}}{1 - P_B^{A+} T_{A^+}}\right)$$



P_A^{B-}: fraction of firing events of A that occur within an interval Δt prior to firing events of B

T_{B^-}: fraction of total recording time covered by the intervals Δt **prior to each spike of B**

Δt: specific lag (input in directional STTC)



Directional STTC
Synchronous (lag = 0)

Spike trains of 100 time unit
with uniform distr [10, 30] spikes
10,000 pairs

Advantages of Directional STTC

$$STTC_{AB} = \frac{1}{2}\left(\frac{P_A^{B-} - T_{B-}}{1 - P_A^{B-} T_{B-}} + \frac{P_B^{A+} - T_{A+}}{1 - P_B^{A+} T_{A+}}\right)$$

Relative spike-time shifts (lag parameter)

Order between neurons with respect to their firing events

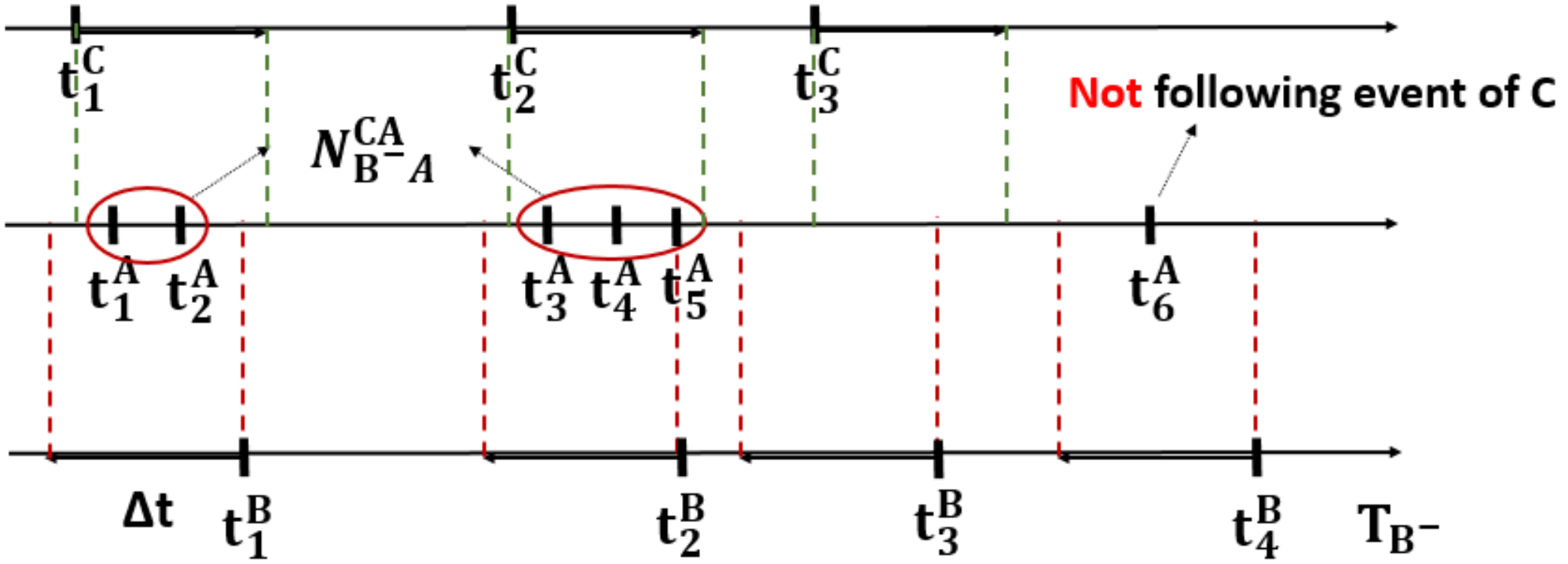
Local fluctuations of neural activity or noise

- accounting the amount of correlation expected by chance

The presence of periods without firing events

- only the firing events contribute

Conditional STTC (A->B |C) represents a measure of the chance that firing events of A will precede firing events of B, given the presence of firing of C



CA

Conditional STTC (A->B |C) $STTC^{C}_{AB}$

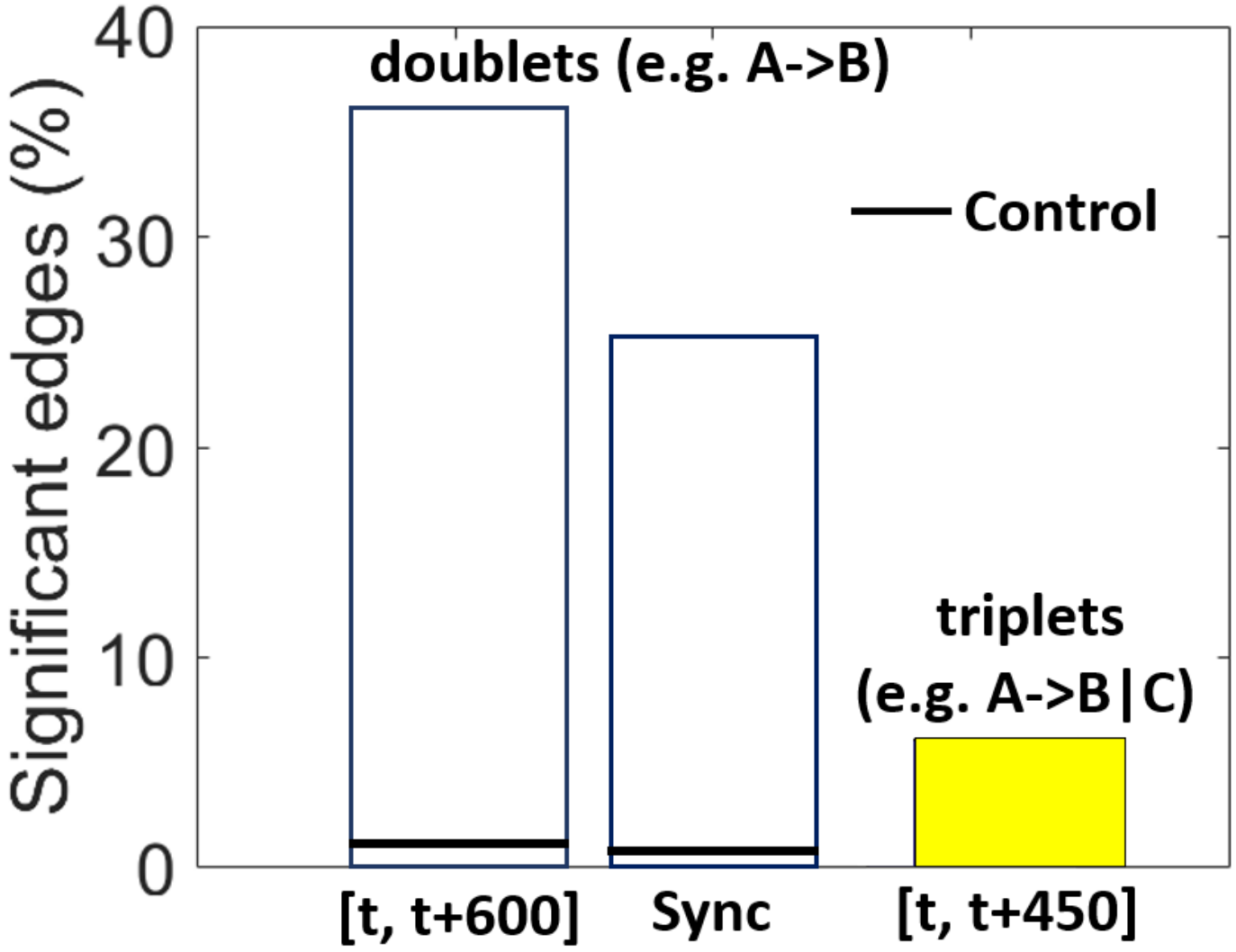
$$STTC^{C}_{AB} = \frac{1}{2}\left(\frac{\frac{N^{CA}_{B^-A}}{N_A} - T_{B^-}}{1 - \frac{N^{CA}_{B^-A}}{N_A} T_{B^-}} + \frac{\frac{N^{CA}_{A^+B}}{N_B} - T_{A^+}}{1 - \frac{N^{CA}_{A^+B}}{N_B} T_{A^+}}\right)$$

N_A is the number of firing event in A & N_B is the number of firing event in B.

T_{A^+} is the fraction of the total recording time which is covered by the tiles $+\Delta t$ after each spike of A, that fall within the tiles Δt after each spike of C.

T_{B^-} is the fraction of the total recording time which is covered by the tiles Δt before each spike of B.

Significant Motifs



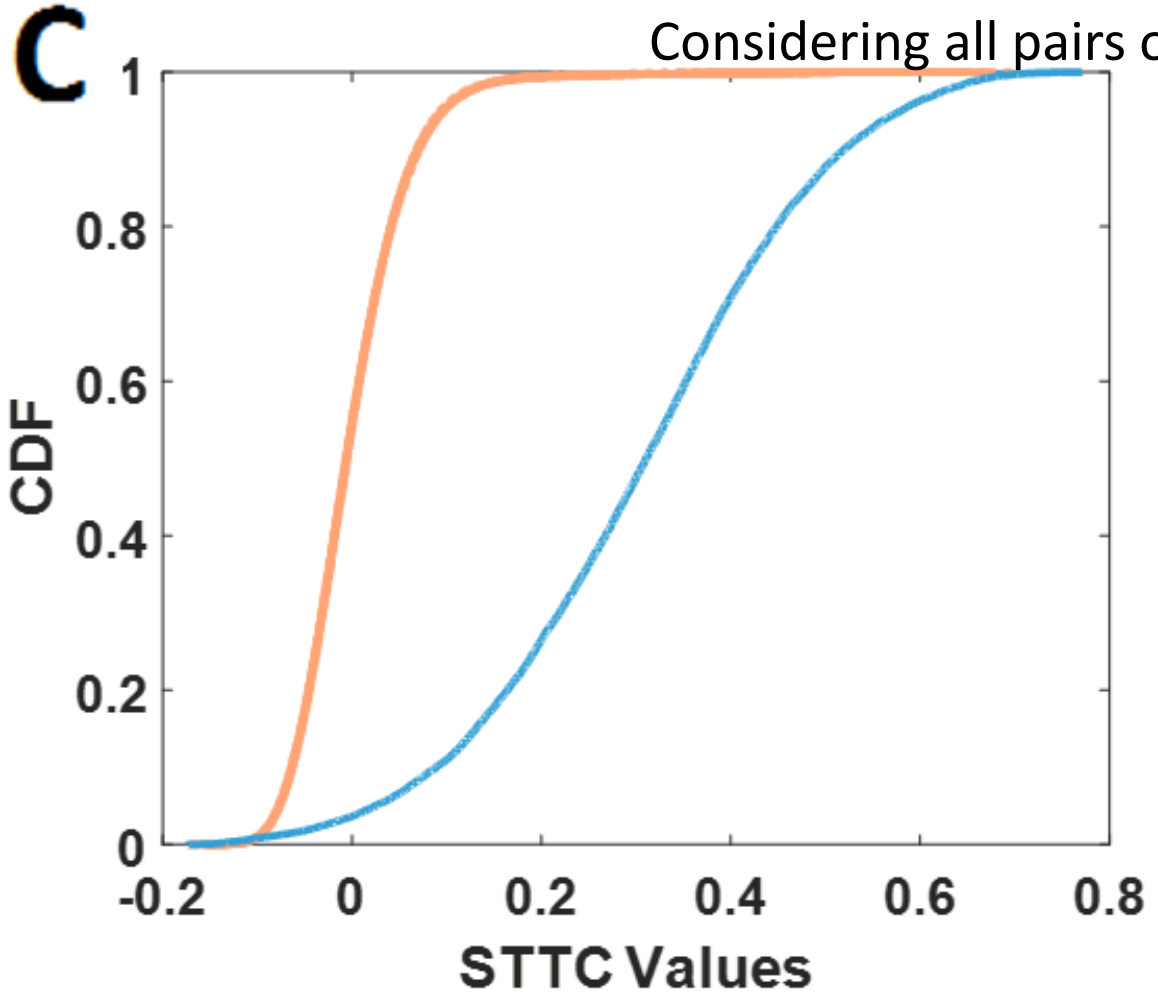
Significant edge: real STTC value > 3 std. dev. of null distribution

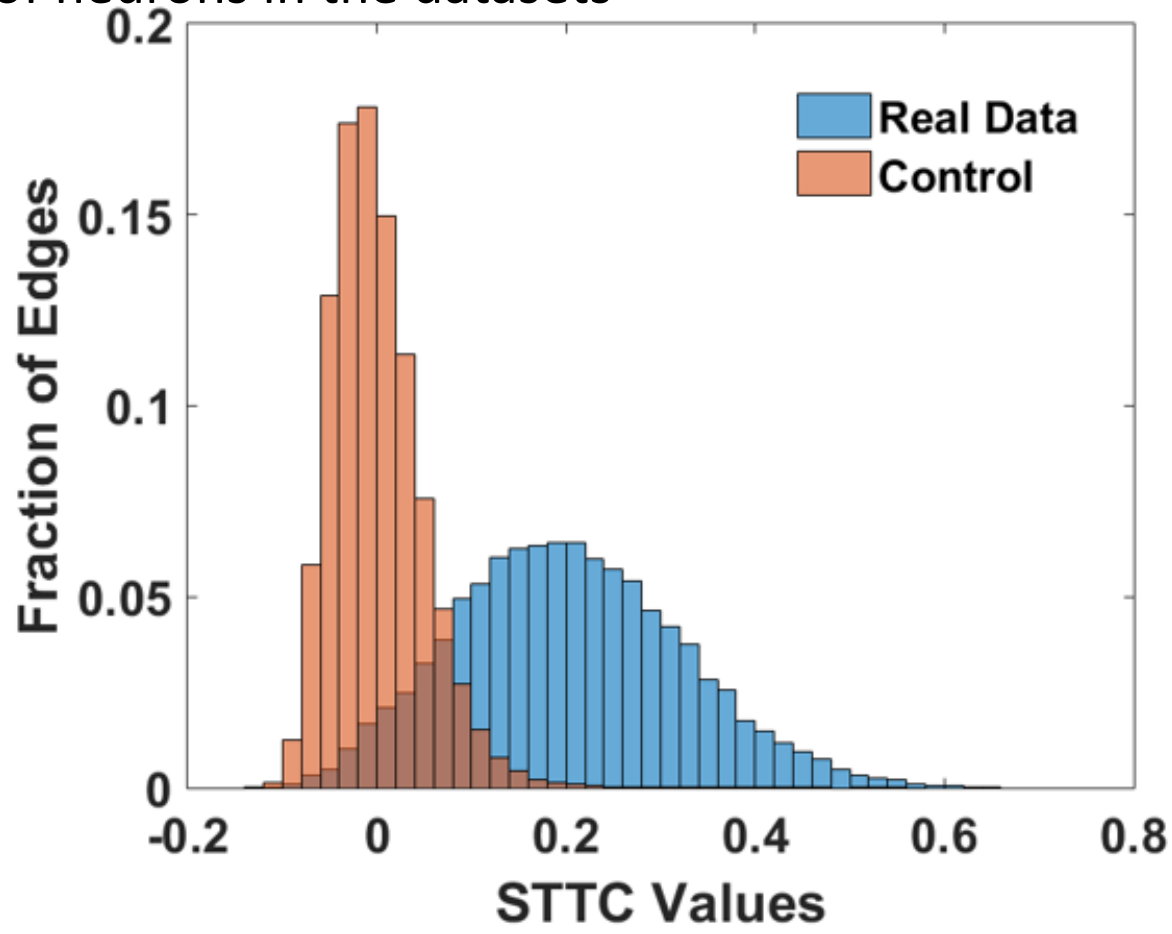
Null distribution: STTC values for the circular shifted neurons (by random delays)

Control (synthetic data)
Each neuron trace is circular shifted by random delay
For each pair of 'shifted' neurons, estimate the directional STTC & null distr.
Identify the significant edges

"A→B" indicates that firing events of **A proceed firing events of B** by a specific lag

Considering all pairs of neurons in the datasets





Control group

Each neuron trace is circular shifted by random delay

For each pair of neurons, estimate the directional STTC & null distribution

Identify the significant edges

The **real neuron traces** appear **higher** values of directional STTC & percentage of significant edges

Null distribution test for directional STTC

For a given pair of neurons i and j (i,j)

Estimate the (observed) STTC(i,j) $STTC_{i,j}^{obs}$

1. Circular shift the spike train of the neuron j (generated spike train j^1)
2. Estimate the directional STTC(i^1, j)

   We will call them synthetic STTC values (called also null or control)

Repeat the above steps a **large** number of times (k=1, … , T)

T depends on the time horizon of the spike trains – the larger is the T the better

3. Estimate the mean $\overline{STTC_{i,j}^{null}}$ & standard deviation $\overline{\sigma_{i,j}^{null}}$ of the obtained **synthetic** STTC values in step2
4. Based on the mean & std dev of the synthetic values, employ a statistical significant threshold (α) & criterion

$$\frac{STTC_{i,j}^{obs} - \overline{STTC_{i,j}^{null}}}{\sigma_{i,j}^{null}} > a$$

Criterion: If the directional STTC (A, B) satisfies the above inequality, the directional STTC (A,B) is statistically significant.

The criterion can be strengthen with more repetitions (T) & larger threshold α

Strengthen the Criterion of Significant Directional STTC (A,B)

Additional requirements

- The total number of spikes of A within a STTC lag of spikes of B is above 3.
- The total number of spikes of B within a STTC lag of spikes of A is above 3.

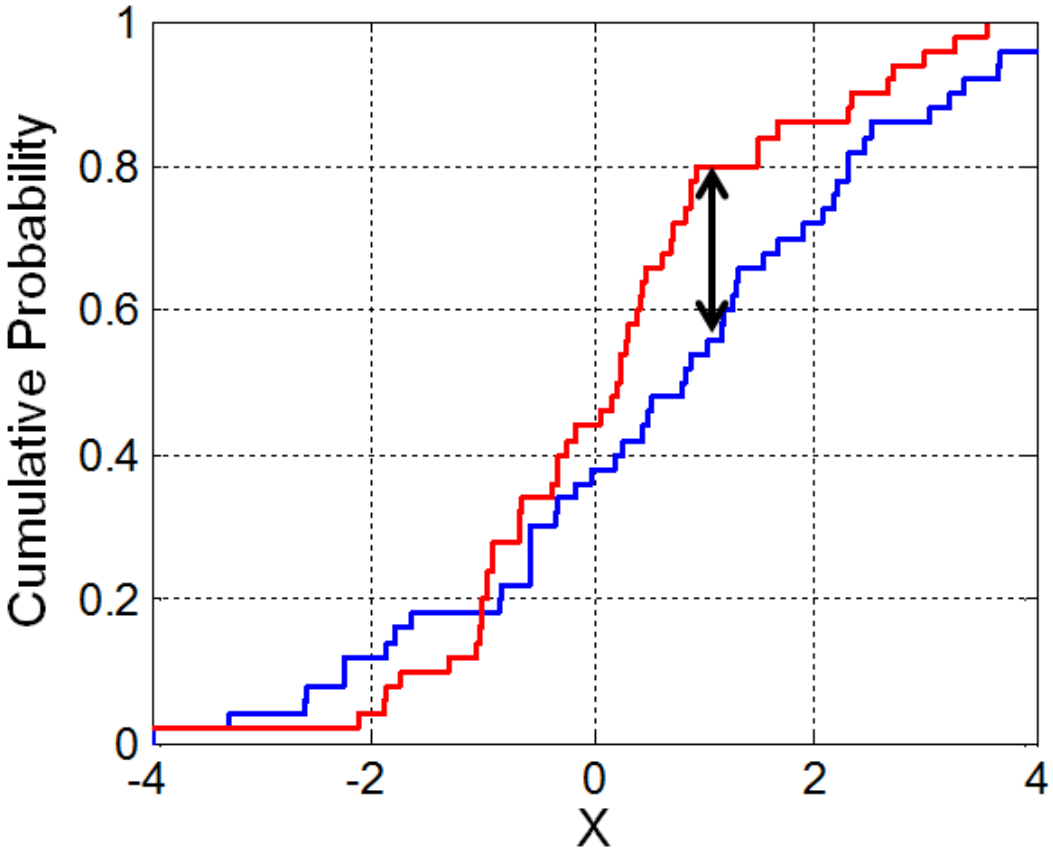
Kolmogorov-Smirnov (K-S) Test

- **Non-parametric** test of the equality of **continuous 1D** probability distributions
- Quantifies a **distance between two distribution** functions
- Can serve as a **goodness of fit test**

- **Null hypothesis**

  H_0: Two samples drawn from **populations with same distribution**

The maximum absolute difference between the two CDFs



Kolmogorov-Smirnov (K-S) Test

$$D_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)|,$$

where $F_{1,n}$ and $F_{2,m}$ are the empirical distribution fun

The null hypothesis is rejected at level α if

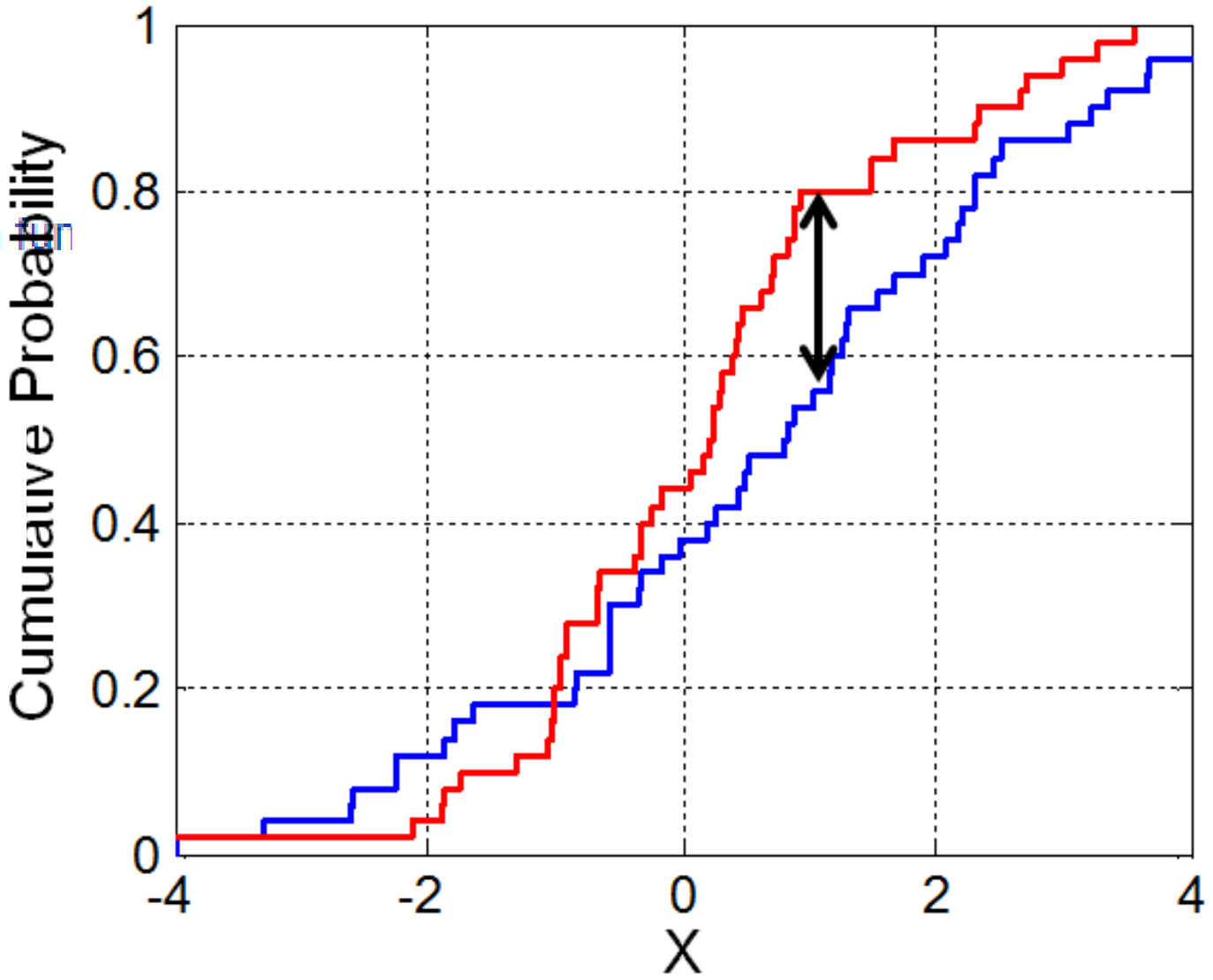
$$D_{n,m} > c(\alpha)\sqrt{\frac{n+m}{nm}}.$$

n & m: size of the sample datasets

α	0.10	0.05	0.025	0.01	0.005	0.001
$c(\alpha)$	1.22	1.36	1.48	1.63	1.73	1.95

and in general by

$$c(\alpha) = \sqrt{-\frac{1}{2}\ln\left(\frac{\alpha}{2}\right)}.$$



Kolmogorov-Smirnov (K-S) Test

Kolmogorov computed the expected distribution of the distance of the two CDFs when the null hypothesis is true.

Example: Kolmogorov-Smirnov Test

	Decision		p-value		Distance	
Lag	**True Null**	**Null Null**	**True Null**	**Null Null**	True Null	Null Null
1	1	0	0	0.5427	0.79	0.0076
2	1	0	0	0.2126	0.78	0.0100
3	1	0	0	0.98485	0.75	0.0043
4	1	0	0	0.9937	0.72	0.0040
5	1	0	0	0.9769	0.68	0.00453

Distance of two distributions in sup norm

For **all neuron pairs (**A, B), populate the following distributions with

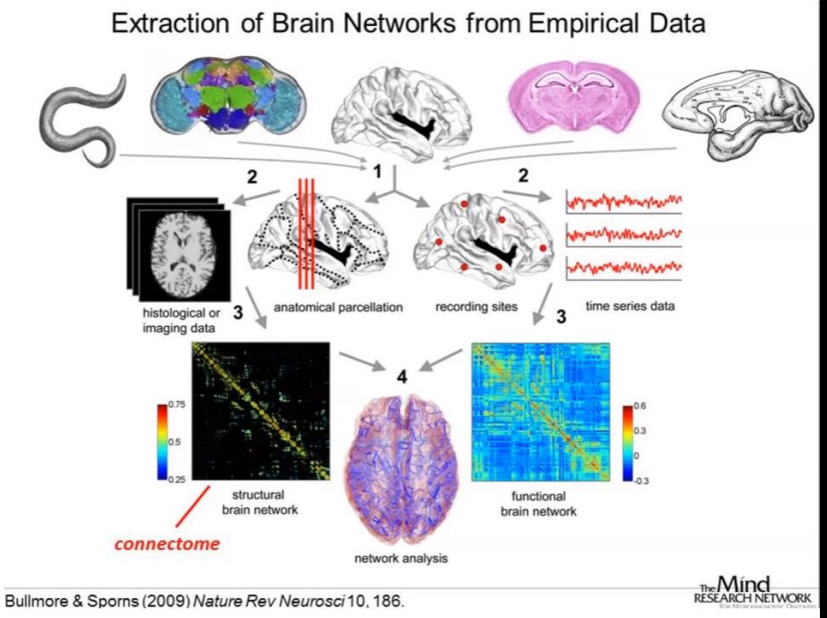
Population 1: real STTC of the pair (A,B)

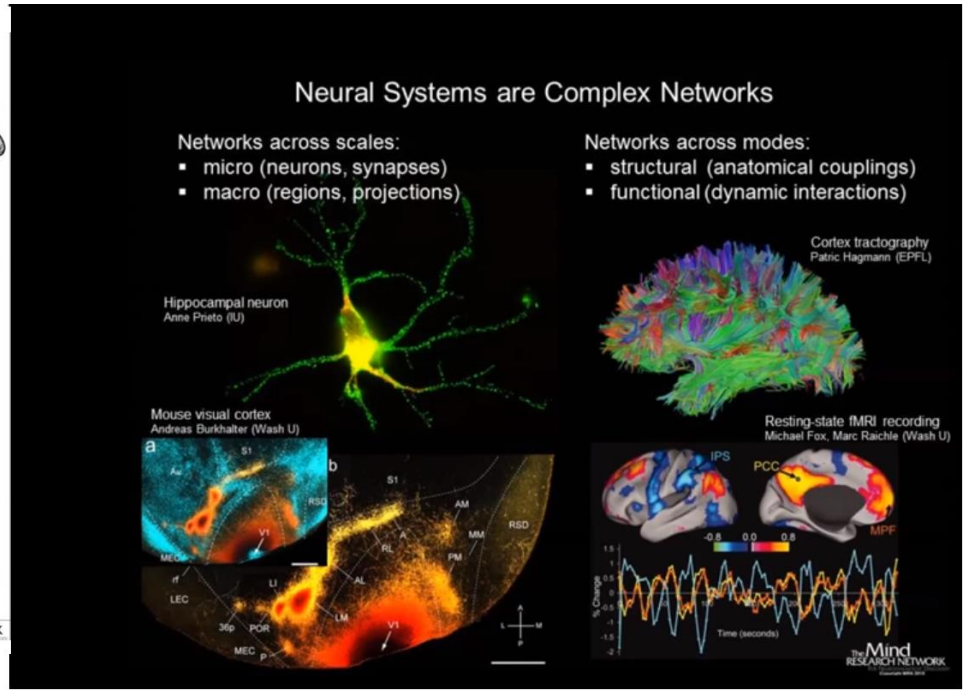
Population 2: random circular shift in one of the two spike trains of (A,B)

Population 3: random circular shift in one of the two spike trains of (A,B)

True Null: Population 1 vs. Population 2

Null Null: Population 2 vs. Polulation 3





Lecture on Modeling Tools for Regression & Clustering

CS – 590.21 Analysis and Modeling of Brain Networks

Department of Computer Science
University of Crete

Greek Diaspora Fellowship Program
ΙΣΝ/SNF
ΙΔΡΥΜΑ ΣΤΑΥΡΟΣ ΝΙΑΡΧΟΣ
STAVROS NIARCHOS FOUNDATION


DOGBERT CONSULTS
YOU NEED TO DO DATA MINING TO UNCOVER HIDDEN SALES TRENDS.
IF YOU MINE THE DATA HARD ENOUGH, YOU CAN ALSO FIND MESSAGES FROM GOD.
© 1999 United Feature Syndicate, Inc.
1/3/00
...SALES TO LEFT-HANDED SQUIRRELS ARE UP...AND GOD SAYS YOUR TIE DOESN'T GO WITH THAT SHIRT.

Data Clustering – Overview

- Organizing data into sensible groupings is critical for understanding and learning.
- Cluster analysis: methods/algorithms for grouping objects according to measured or perceived **intrinsic characteristics or similarity**.
- Cluster analysis does **not** use **category labels** that tag objects with prior identifiers, i.e., class labels.

The absence of category labels distinguishes data clustering (**unsupervised learning**) from **classification (supervised learning**).

- Clustering aims to find **structure in data and is therefore exploratory in nature**.

- Clustering has a long rich history in various scientific fields.

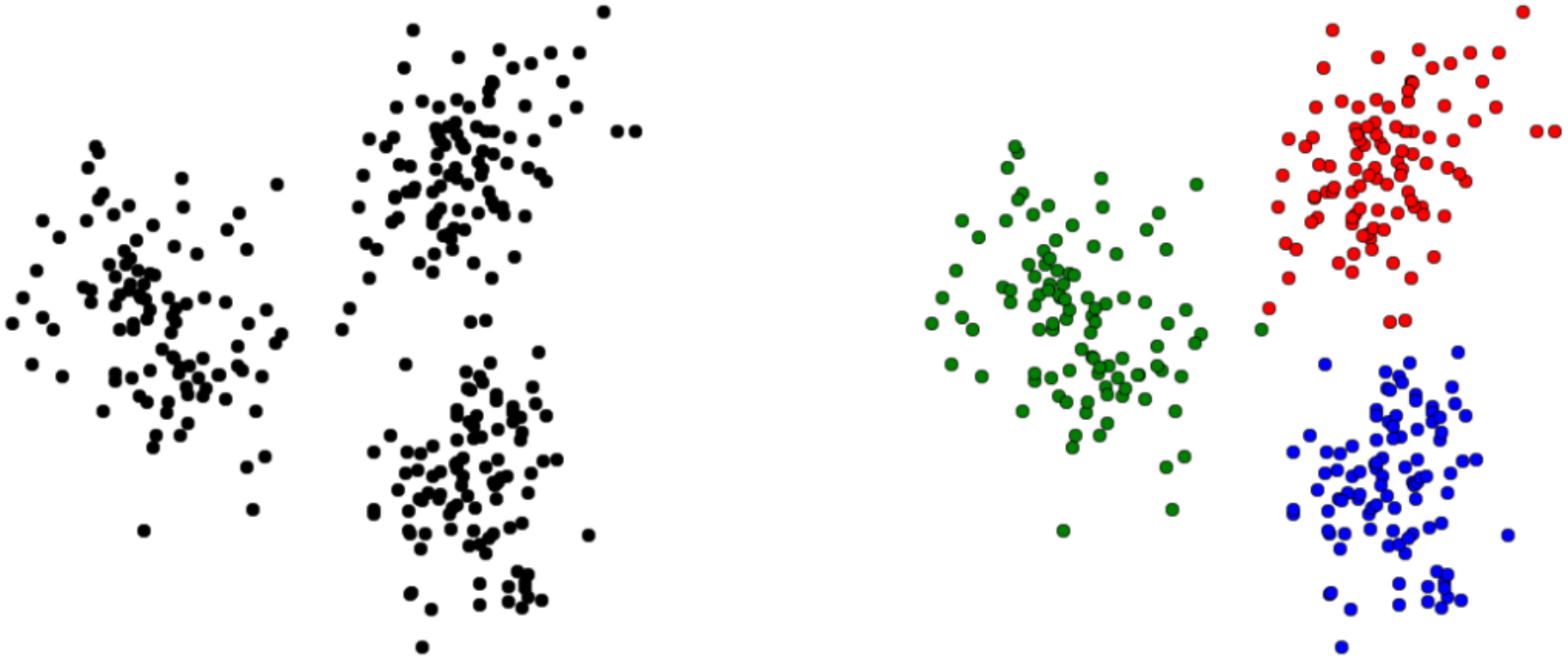
  K-means (1955): One of the most popular simple clustering algorithms

  Still widely-used.

  The design of a general purpose clustering algorithm is a difficult task

Clustering

- given N n-vectors $x_1, \ldots, x_N$
- goal: partition (divide, cluster) into k groups
- want vectors in the same group to be close to one another



Clustering objectives

- $G_j \subset \{1, \ldots, N\}$ is group j, $j = 1, \ldots, k$
- c_i is group that x_i is in: $x_i \in G_{c_i}$
- group *representatives*: n-vectors $z_1, \ldots, z_k$

- clustering objective is

$$J = \frac{1}{N} \sum_{i=1}^{N} \|x_i - z_{c_i}\|^2$$

  mean square distance from vectors to associated representative
- J small means good clustering
- goal: choose clustering c_i and representatives z_j to minimize J

k-means: simplest unsupervised learning algorithm

Iterative greedy algorithm (K):

1. Place K points into the space represented by the objects that are being clustered
   These points represent **initial group centroids** (e.g., start by **randomly** selecting K centroids)
2. Assign each object to the group that has the closest centroid (e.g., Euclidian distance)
3. When all objects have been assigned, recalculate the positions of the K centroids

Repeat Steps 2 and 3 until the centroids **no longer move**.

It converges but does not guarantee optimal solutions.

It is heuristic!

Criteria for Assessing a Clustering

Internal criterion analyzes **intrinsic characteristics** of a clustering

External criterion analyzes how close is a clustering to a **reference**

Relative criterion analyzes the **sensitivity of internal criterion** during clustering generation

The measured quality of a clustering depends on both the object representation and the similarity measure used

Properties of a good clustering according to the internal criterion

- High **intra-class** (intra-cluster) similarity

Cluster cohesion: measures how **closely related are objects in a cluster**

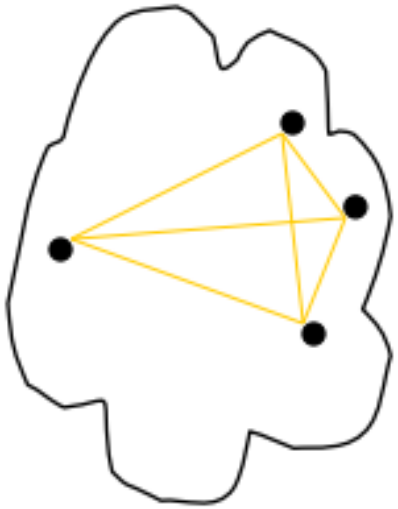
- Low **inter-class** similarity

Cluster separation: measures how **well-separated a cluster is from other clusters**

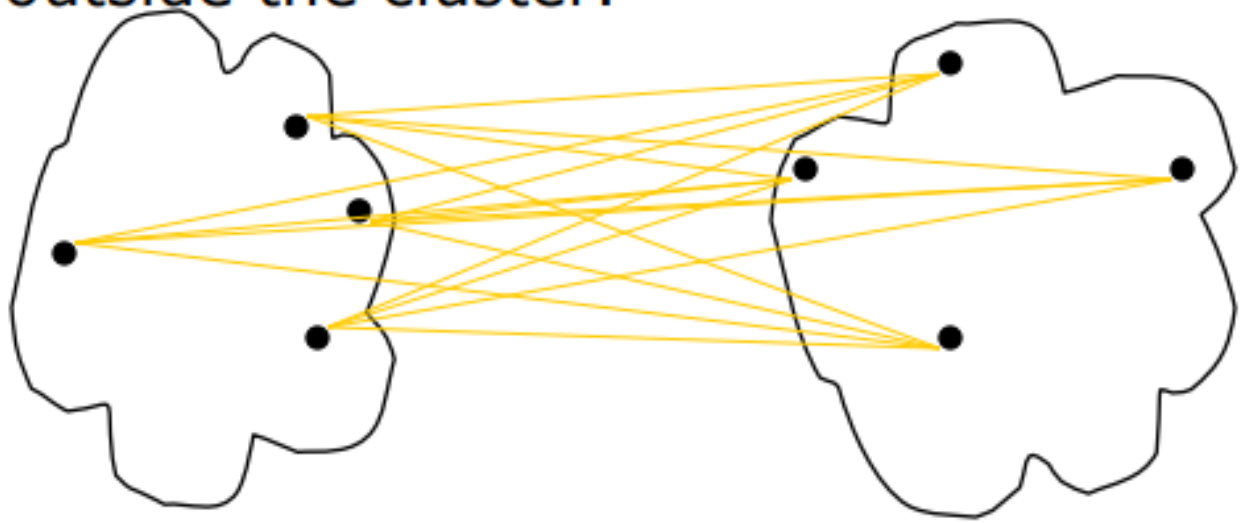
The measured quality depends on the object representation & the similarity measure used

Internal Measures: Cohesion and Separation

- A proximity graph based approach can also be used for cohesion and separation.
  - Cluster cohesion is the sum of the weight of all links within a cluster.
  - Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



cohesion



separation

Silhouette value measures cohesion compared to separation

How **similar an object is to its own cluster** (**cohesion**) **compared to other clusters** (**separation**)

- Ranges from −1 to +1: a high value indicates that the object is well matched to its own cluster & poorly matched to neighboring clusters
- If most objects have a high value, then the clustering configuration is appropriate
- If many points have a low or negative value, then the clustering configuration may have too many or too few clusters

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

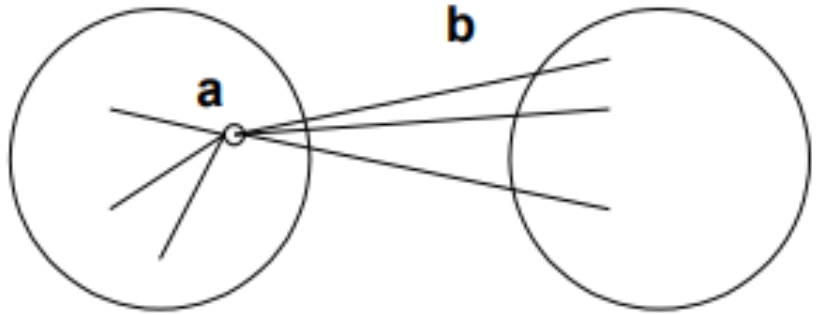
a(i) average dissimilarity of i with all other data within the same cluster.
b(i): lowest average dissimilarity of i to any other cluster, of which i is *not* a member

Silhouette Coefficient

- For an individual point, *i*
  - Calculate ***a*** = average distance of *i* to the points in its cluster
  - Calculate ***b*** = min (average distance of *i* to points in another cluster)
  - The silhouette coefficient for a point is then given by

    $s = 1 - a/b$ if $a < b$, (or $s = b/a - 1$ if $a \geq b$, not the usual case)

  - Typically between 0 and 1.
  - The closer to 1 the better.



- Can calculate the Average Silhouette width for a cluster or a clustering

External criteria for clustering quality

- External criteria: analyze how close is a clustering to a **reference**
- Quality measured by its ability to **discover some or all of the hidden patterns or latent classes in gold standard data**
- Assesses a clustering with respect to **ground truth** requires **labeled data**
- Assume items with *C* **gold standard classes**, while our clustering algorithms produce *K* clusters, $\omega_1, \omega_2, \ldots, \omega_K$ with n_i members.

External Evaluation of Cluster Quality (cont'd)

- Assume items with *C* **gold standard classes**, while our clustering produces *K* clusters, $\omega_1, \omega_2, \ldots, \omega_K$ with n_i members.

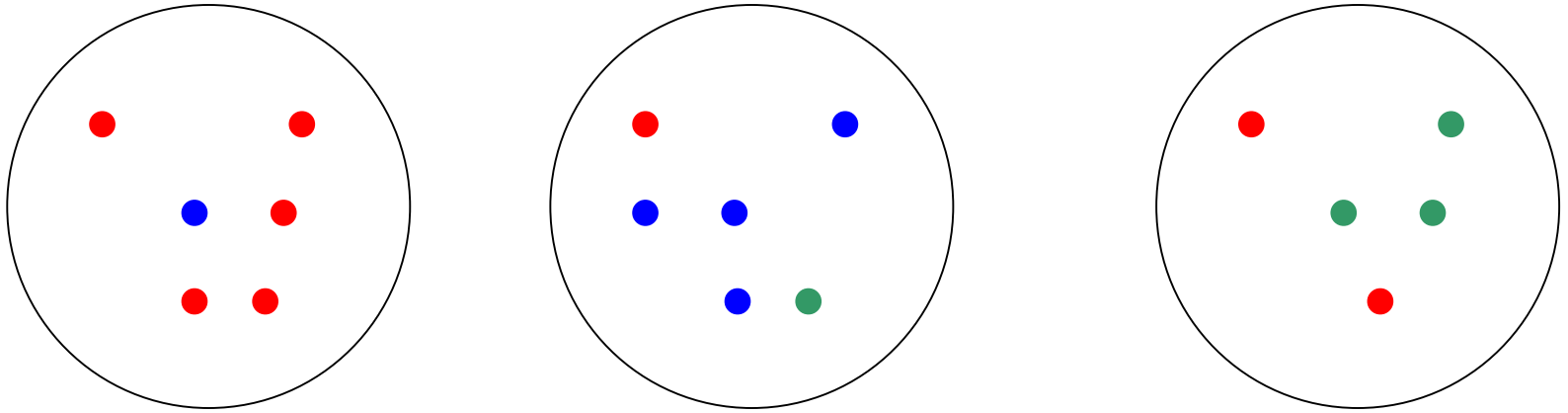
- **Purity**: the ratio between the dominant class in the cluster π_i and the size of cluster ω_i

$$Purity(\omega_i) = \frac{1}{n_i} \max_j (n_{ij}) \quad j \in C$$

Biased because having *n* clusters maximizes purity

- **Entropy** of classes in clusters
- **Mutual information** between classes and clusters

Purity example



Cluster I Cluster II Cluster III

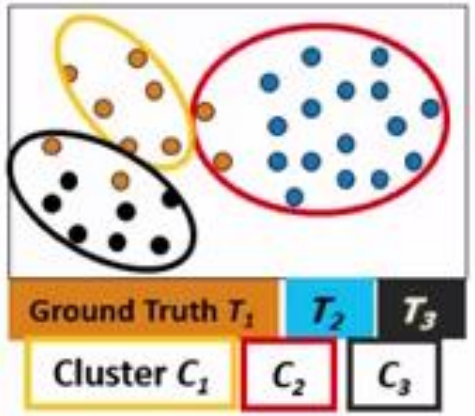
Cluster I: Purity = 1/6 (max(5, 1, 0)) = 5/6

Cluster II: Purity = 1/6 (max(1, 4, 1)) = 4/6

Cluster III: Purity = 1/5 (max(2, 0, 3)) = 3/5

Entropy-based Measure of the Quality of Clustering

- **Entropy of clustering *C*:** $H(\mathcal{C}) = -\sum_{i=1}^{r} p_{C_i} \log p_{C_i}$ $\quad p_{C_i} = \frac{n_i}{n}$ (i.e., the probability of cluster C_i)
- **Entropy of partitioning *T*:** $H(\mathcal{T}) = -\sum_{j=1}^{k} p_{T_j} \log p_{T_j}$
- **Entropy of *T* with respect to cluster C_i:** $H(\mathcal{T}|C_i) = -\sum_{j=1}^{k} (\frac{n_{ij}}{n_i}) \log(\frac{n_{ij}}{n_i})$
- **Conditional entropy of *T* with respect to clustering *C*:** $H(\mathcal{T}|\mathcal{C}) = -\sum_{i=1}^{r} (\frac{n_i}{n}) H(\mathcal{T}|C_i) = -\sum_{i=1}^{r} \sum_{j=1}^{k} p_{ij} \log(\frac{p_{ij}}{p_{C_i}})$



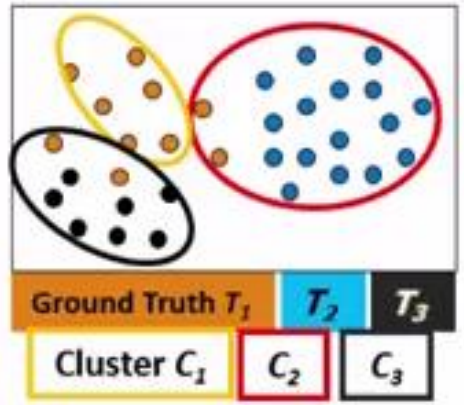
  - The more a cluster's members are split into different partitions, the higher the conditional entropy
  - For a perfect clustering, the conditional entropy value is 0, where the worst possible conditional entropy value is *log k*

Mutual-information based Measure of Quality of Clustering

- **Mutual information**:
  - Quantifies the amount of shared info between the clustering *C* and partitioning *T*

$$I(C,T) = \sum_{i=1}^{r}\sum_{j=1}^{k} p_{ij} \log\left(\frac{p_{ij}}{p_{C_i} \cdot p_{T_j}}\right)$$

  - Measures the dependency between the observed joint probability p_{ij} of *C* and *T*, and the expected joint probability $p_{Ci} \cdot p_{Tj}$ under the independence assumption
  - When *C* and *T* are independent, $p_{ij} = p_{Ci} \cdot p_{Tj}$, $I(C, T) = 0$. However, there is no upper bound on the mutual information



- **Normalized mutual information** (NMI)

$$NMI(C,T) = \sqrt{\frac{I(C,T)}{H(C)} \cdot \frac{I(C,T)}{H(T)}} = \frac{I(C,T)}{\sqrt{H(C) \cdot H(T)}}$$

  - Value range of NMI: [0,1]. Value close to 1 indicates a good clustering

For each neuron we estimate the:

NumofApp: number of times that a neuron i appears at all positions A, B & C

Type 5: percentage of times that a neuron i appears at each position across the NumofApp(i)

Number of tested clusters k = 2:20

Best clustering k = 2

Cluster 1: 143 neurons (83.14 %)

Cluster 2: 29 neurons (16.86 %) Interneurons: 10 & 28

Example

Clustering of neurons at positions A, B and C in the conditional STTC (A, B|C)

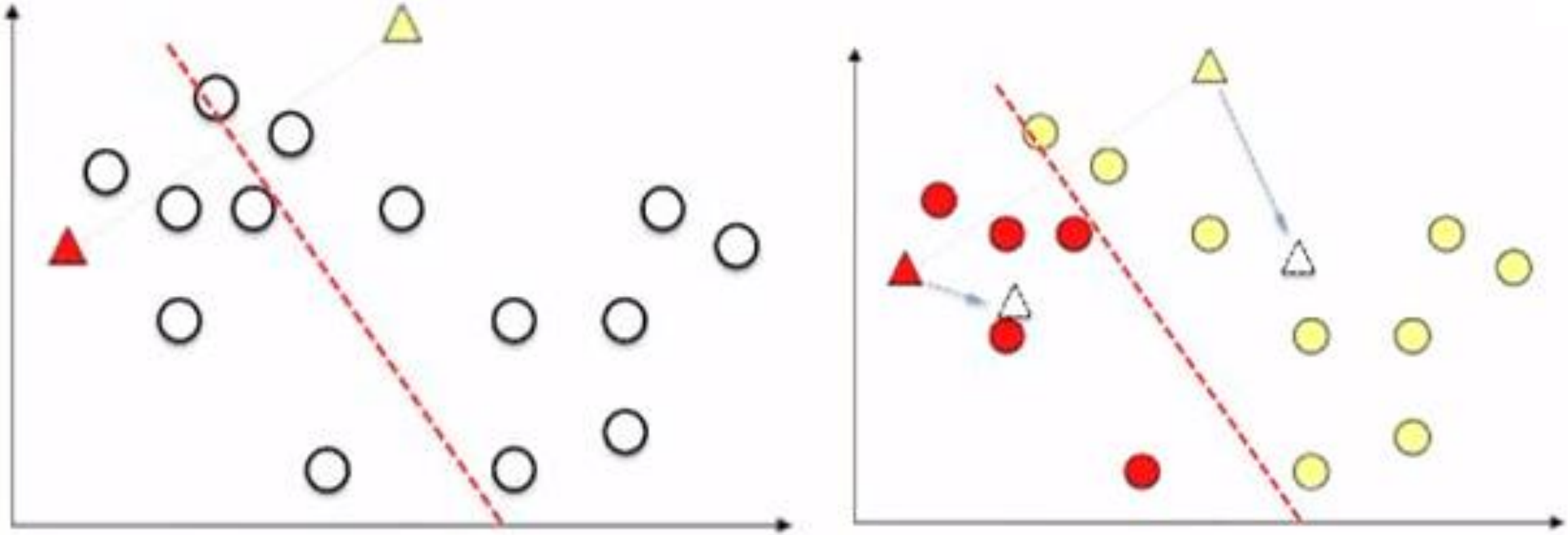
Table 7: Centroids of the 2 clusters for the positions A, B & C

	A	B	C
Centroid 1	33.95	33.83	32.20
Centroid 2	6.46	90.21	3.31

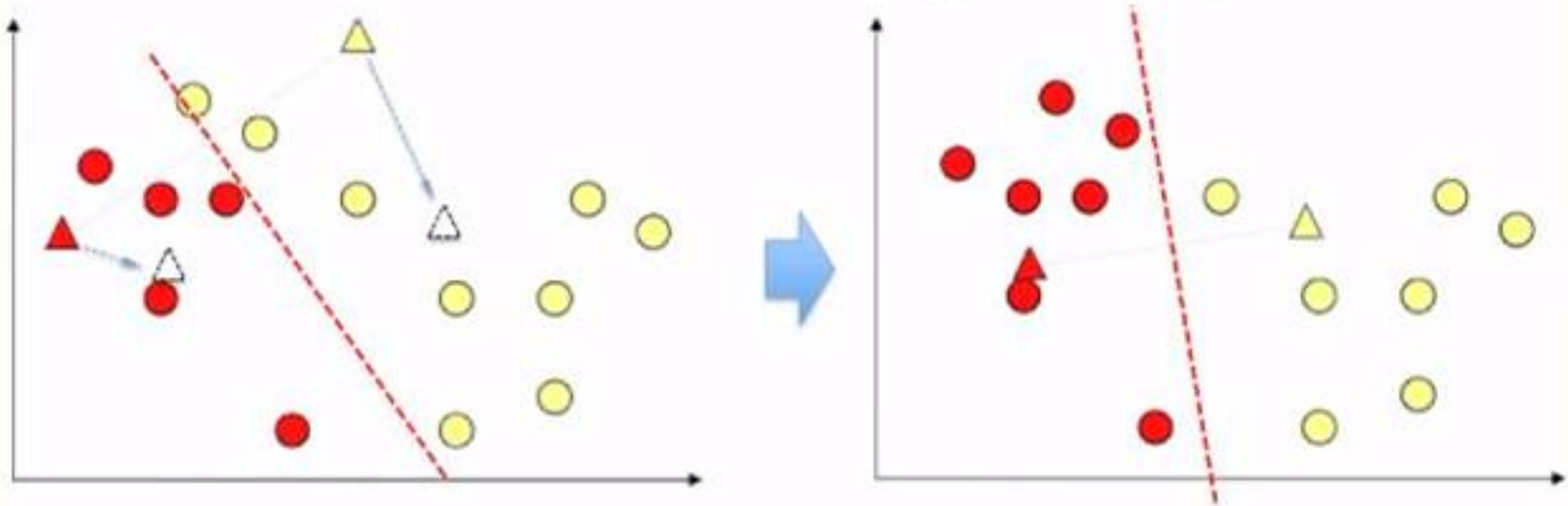
1st cluster: neurons with approximately equal participation at each position

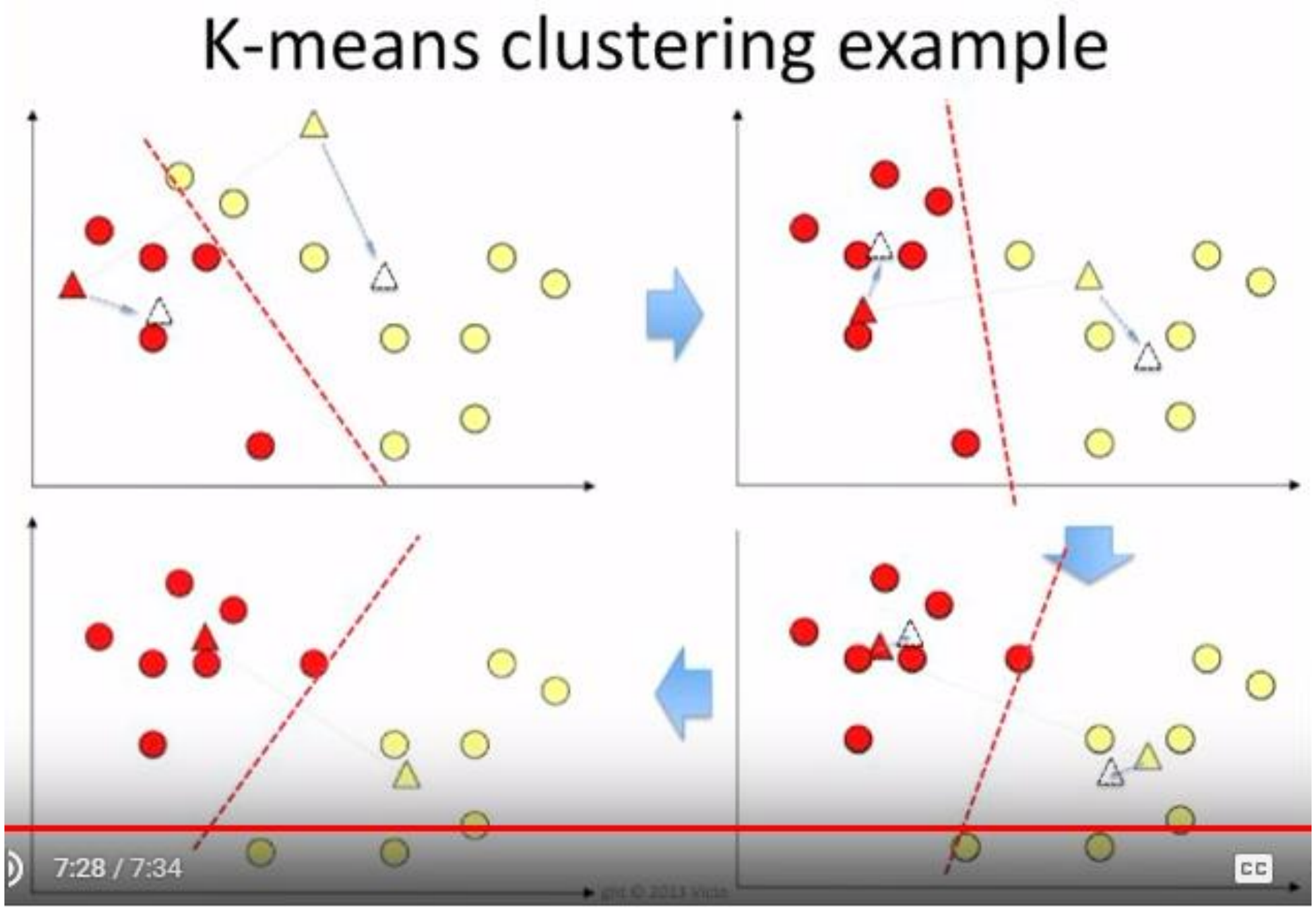
2nd cluster: neurons with high presence in the position B

K-means clustering example



K-means clustering example




K-means clustering example
7:28 / 7:34
CC

r each cluster, the class distribution of the data is calculated first, i.e., for cluster j npute p_{ij}, the 'probability' that a member of cluster j belongs to class i as follows: n_{ij}/m_j, where m_j is the number of values in cluster j and m_{ij} is the number of values s i in cluster j. Then using this class distribution, the entropy of each cluster j is ted using the standard formula $e_j = \sum_{i=1}^{L} p_{ij} \log_2 p_{ij}$, where the L is the number of . The total entropy for a set of clusters is calculated as the sum of the entropies of each weighted by the size of each cluster, i.e., $e = \sum_{i=1}^{K} \frac{m_i}{m} e_j$, where m_j is the size of cluster the number of clusters, and m is the total number of data points.

Linear Regression for Predictive Modeling

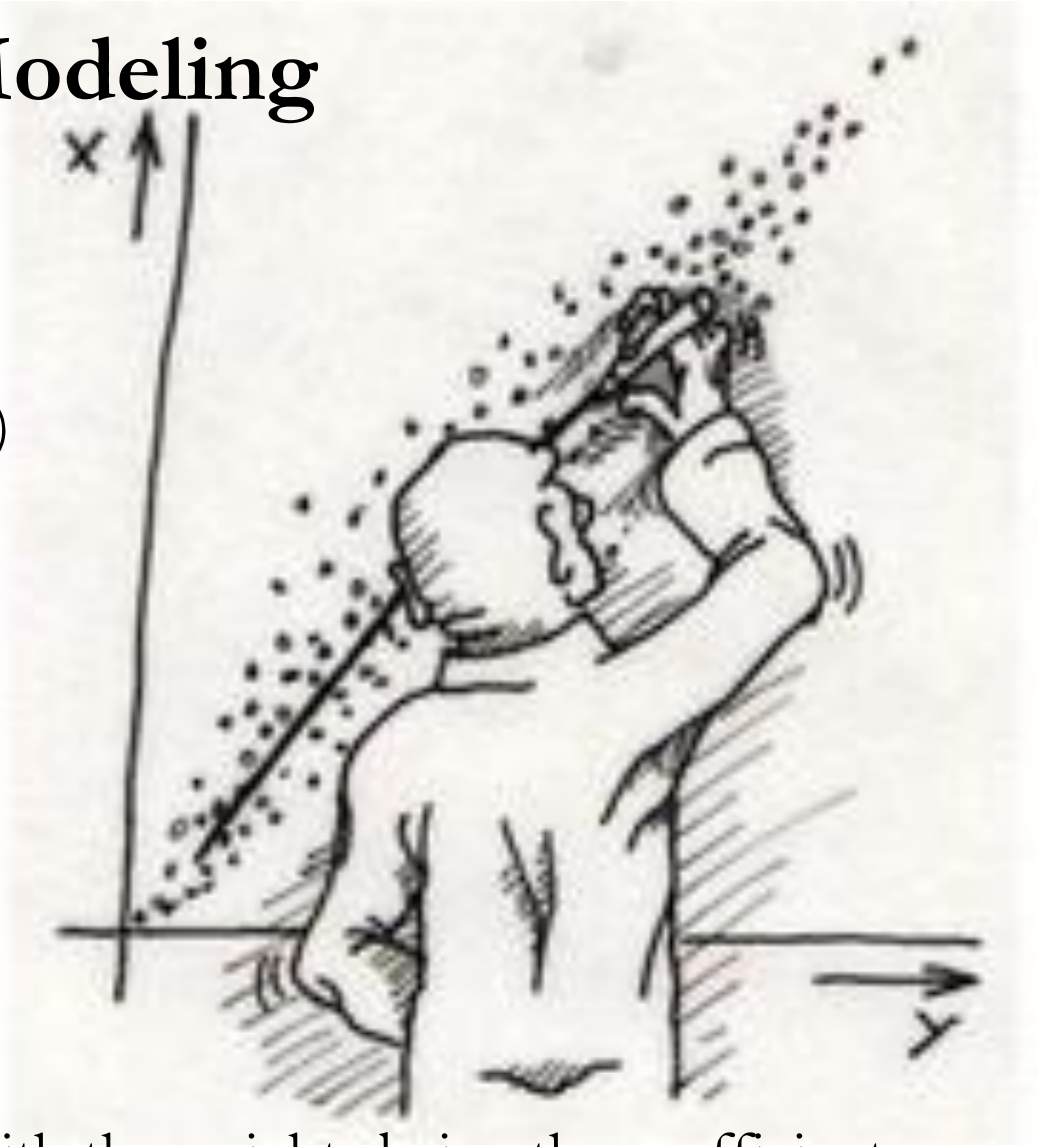
Suppose a set of observations $X_1, \ldots X_p \in$

& a set of explanatory variables (i.e., predictors)

$$y = (y_1, \ldots y_n) \in \mathbb{R}^n$$

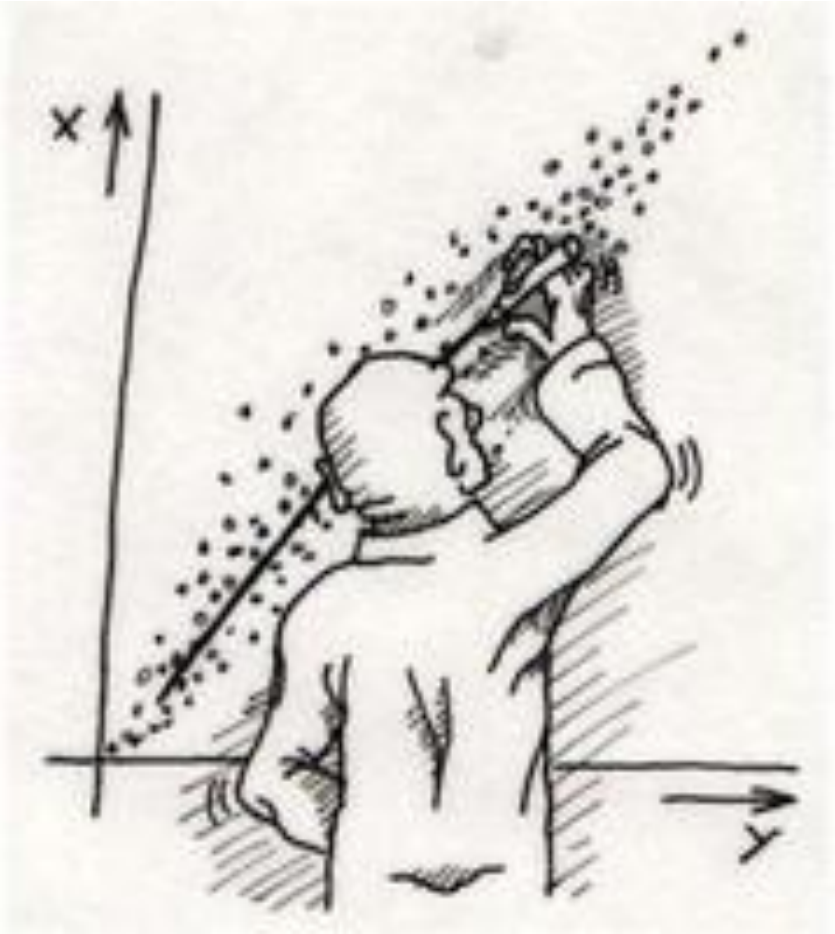
We build a **linear model** $y = X\beta^*$

where $\beta^* = (\beta_1^*, \ldots \beta_p^*) \in \mathbb{R}^p$ are the coefficient



y given as a **weighted sum of the predictors**, with the weights being the coefficients

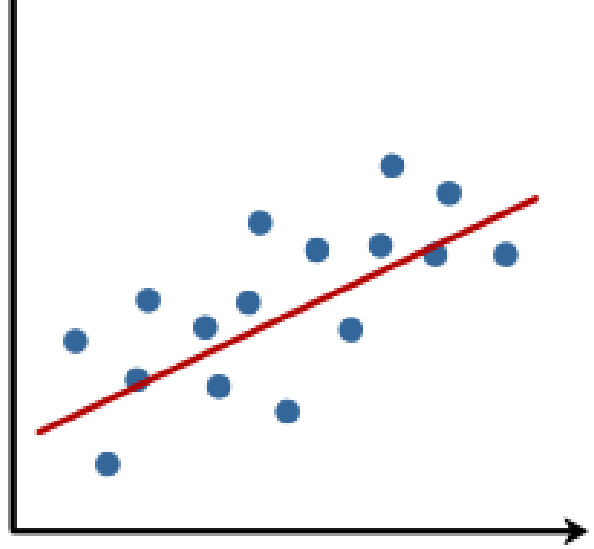
Why using linear regression?



Strength of the relationship between y and a variable x_i

- Assess the impact of each predictor x_i on y through the magnitude of β_i
- Identify subsets of X that contain redundant information about y

Simple linear regression



Suppose that we have observations $y = (y_1, \dots y_n) \in \mathbb{R}^n$

and we want to model these as a linear function of $x = (x_1, \dots x_n) \in \mathbb{R}^n$

$$y = \beta^* x$$

To determine which is the optimal $\beta \in R^n$, we solve the least squares problem:

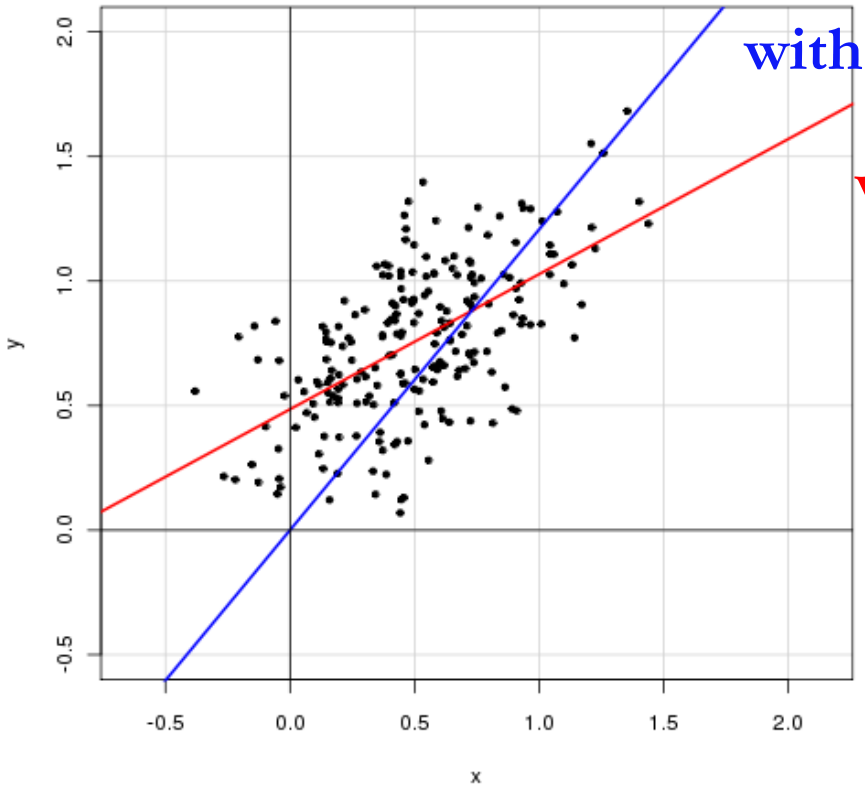
$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - \beta x_i)^2 = \underset{\beta}{\operatorname{argmin}} \|y - \beta x\|_2^2$$

where ***β* is the optimal *β* that minimizes the Sum of Squared Errors (SSE)**

An intercept term β_0 captures the noise not caught by predictor variable

Again we estimate $\hat{\beta}_0, \hat{\beta}_1$ using least squares

$$\hat{\beta}_0, \hat{\beta}_1 = \underset{\beta_0, \beta_1}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 = \underset{\beta_0, \hat{\beta}_1}{\operatorname{argmin}} \|y - \beta_0 \mathbb{1} - \beta_1 x\|_2^2$$

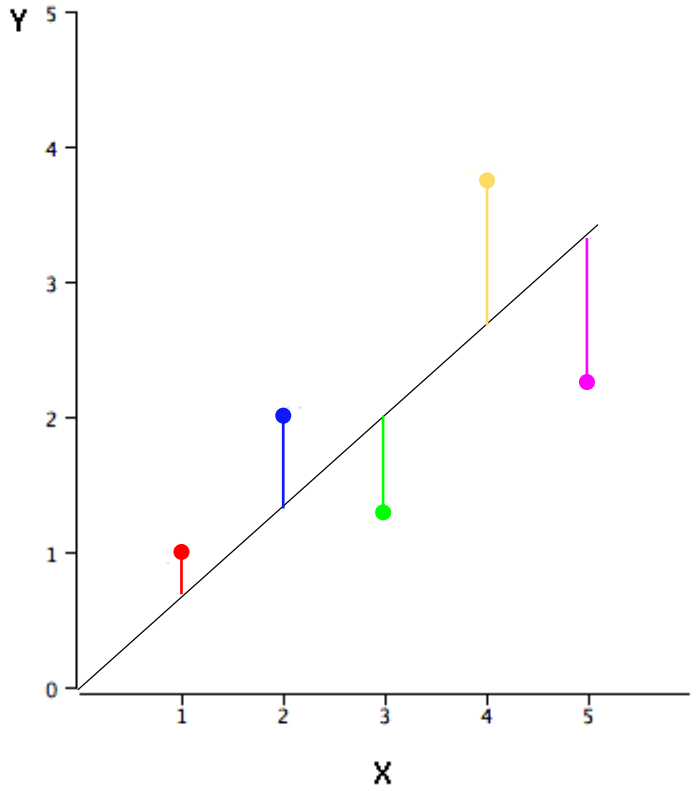


without intercept term

with intercept term

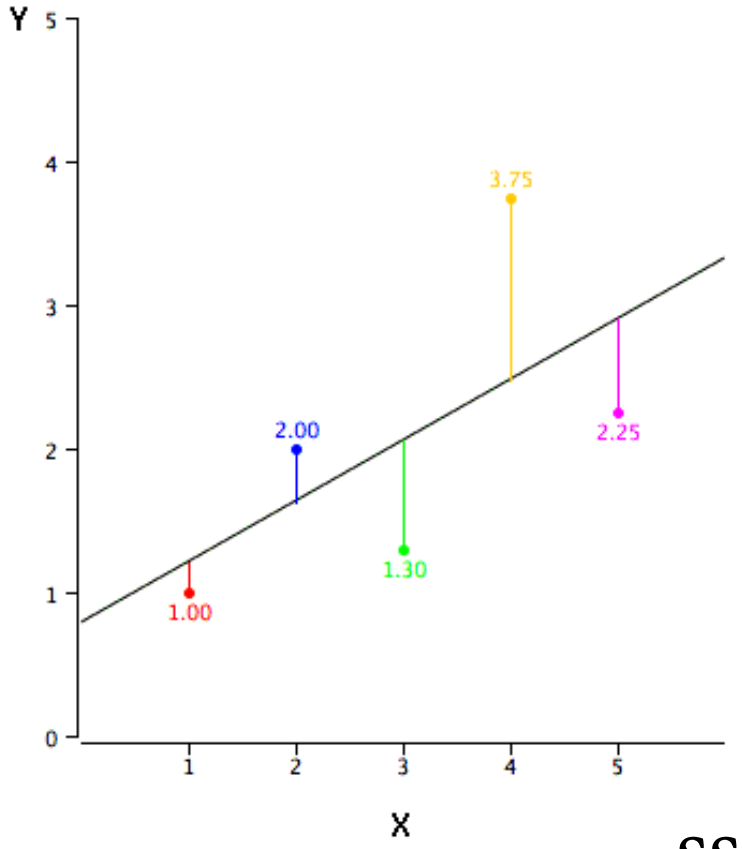
Example 2

Intercept term **improves** the accuracy of the model



Predicted Y	Squared Error
0.70	0.09
1.40	0.36
2.10	0.64
2.80	0.90
3.50	1.56

SSE = 3.55



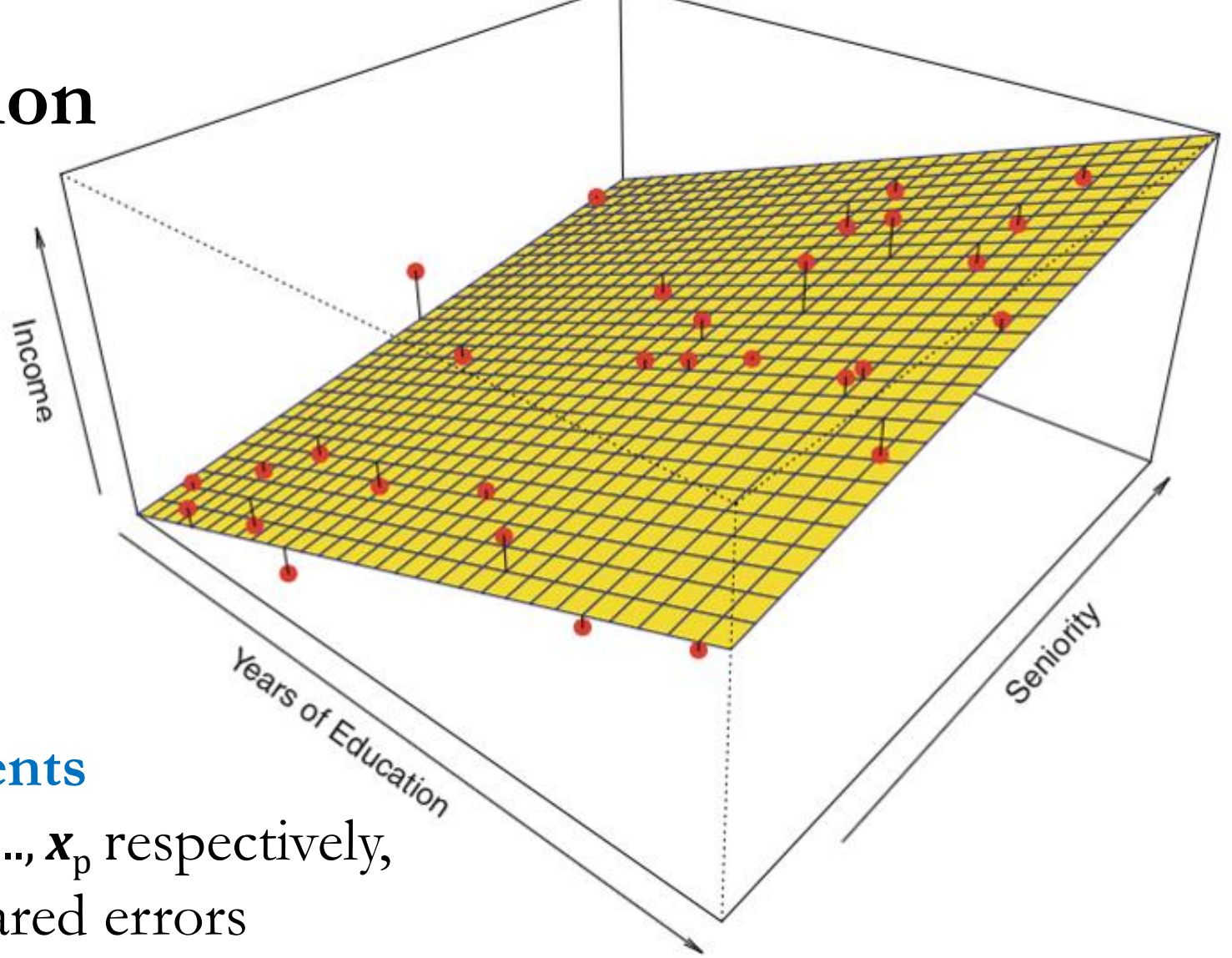
Predicted Y	Squared Error
1.20	0.04
1.60	0.16
2.00	0.49
2.50	1.56
2.90	0.42

SSE = 2.67

Multiple linear regression

Models the relationship between
two or more predictors & the target

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2$$



where $\hat{\beta}$ are **the optimal coefficients**

$\beta_1, \beta_2, ..., \beta_p$ of the predictors $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_p$ respectively,
that minimize the above sum of squared errors

Regularization

Process of introducing additional information in order to prevent overfitting

A **regularization term** (or **regularizer**) $R(f)$ is added to a loss function:

$$\min_f \sum_{i=1}^{n} V(f(x_i), y_i) + \lambda R(f)$$

λ controls the importance of the regularization

Regularization

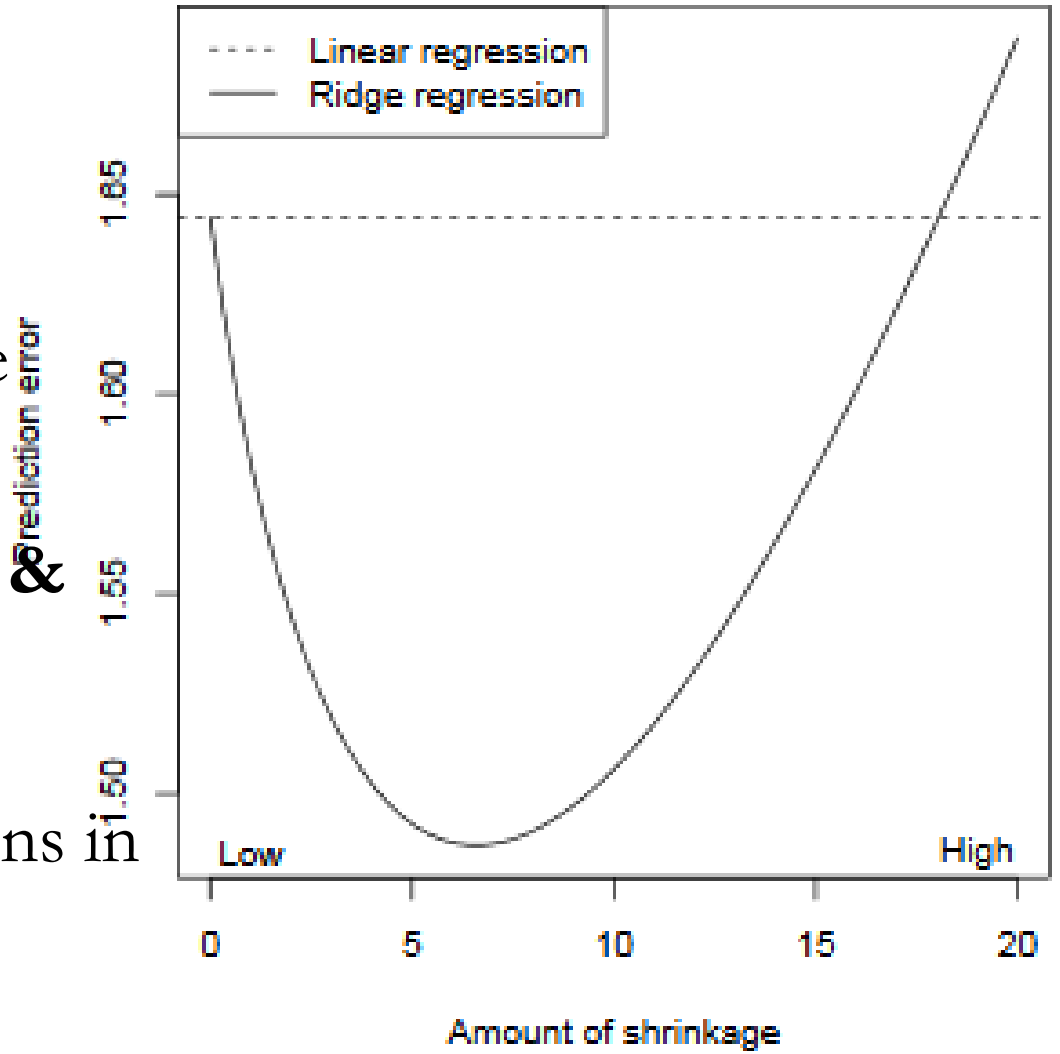
Shrinks the magnitude of coefficients

Bias: error from erroneous assumptions about the training data

- **Miss relevant relations between predictors & target (high bias, underfitting)**

Variance: error from sensitivity to small fluctuations in the training data

- Model noise, not the intended output (high variance, overfitting)



Bias – variance tradeoff: Ignore some small details to get a more general "big picture"

Ridge regression

Given a vector with observation $X \in \mathbb{R}^{n \times p}$ & a predictor matrix $y \in \mathbb{R}^n$

the ridge regression coefficients are defined as:

$$\hat{\beta}^{\text{ridge}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

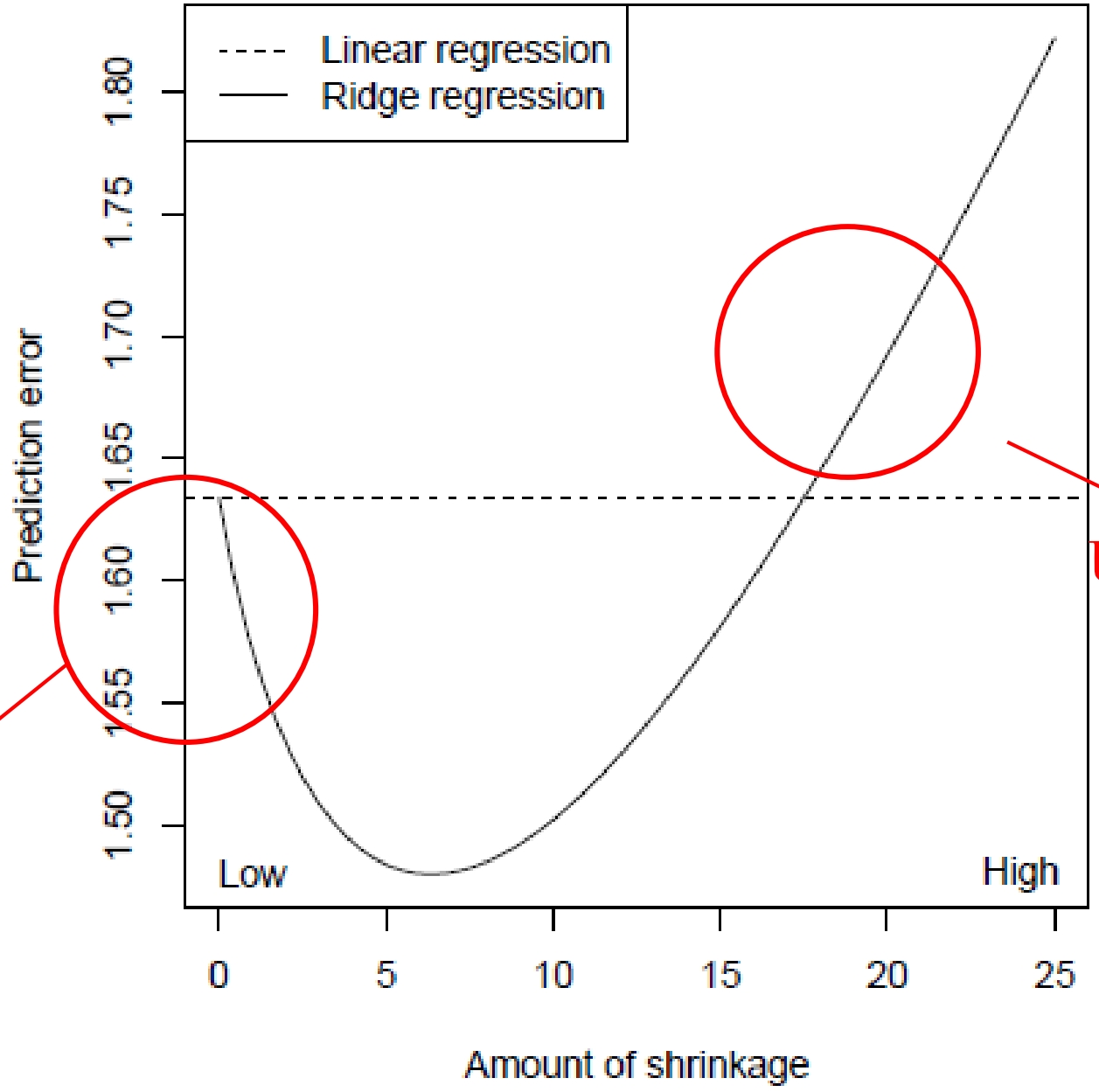
$$= \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{Penalty}}$$

Not only **minimizing the squared error** but also the size of the coefficients!

Ridge regression as regularization

- If the β_j are unconstrained, they can explode …
  and hence are susceptible to **very high variance**!
- To control variance, we might regularize the coefficients
  i.e., might control how large they can grow

Example 3



Overfitting

Underfitting

Increasing size of λ

Variable selection

Problem of **selecting the most relevant predictors** from a larger set of predictors

In linear model setting, this means estimating some coefficients to be **exactly zero**

This can be very important for the purposes of model interpretation

Ridge regression cannot perform variable selection

- Does not set coefficients exactly to zero, unless $\lambda = \infty$

Example 4

Suppose that we study the level of prostate-sp antigen (PSA), which is often elevated in men who prostate cancer.

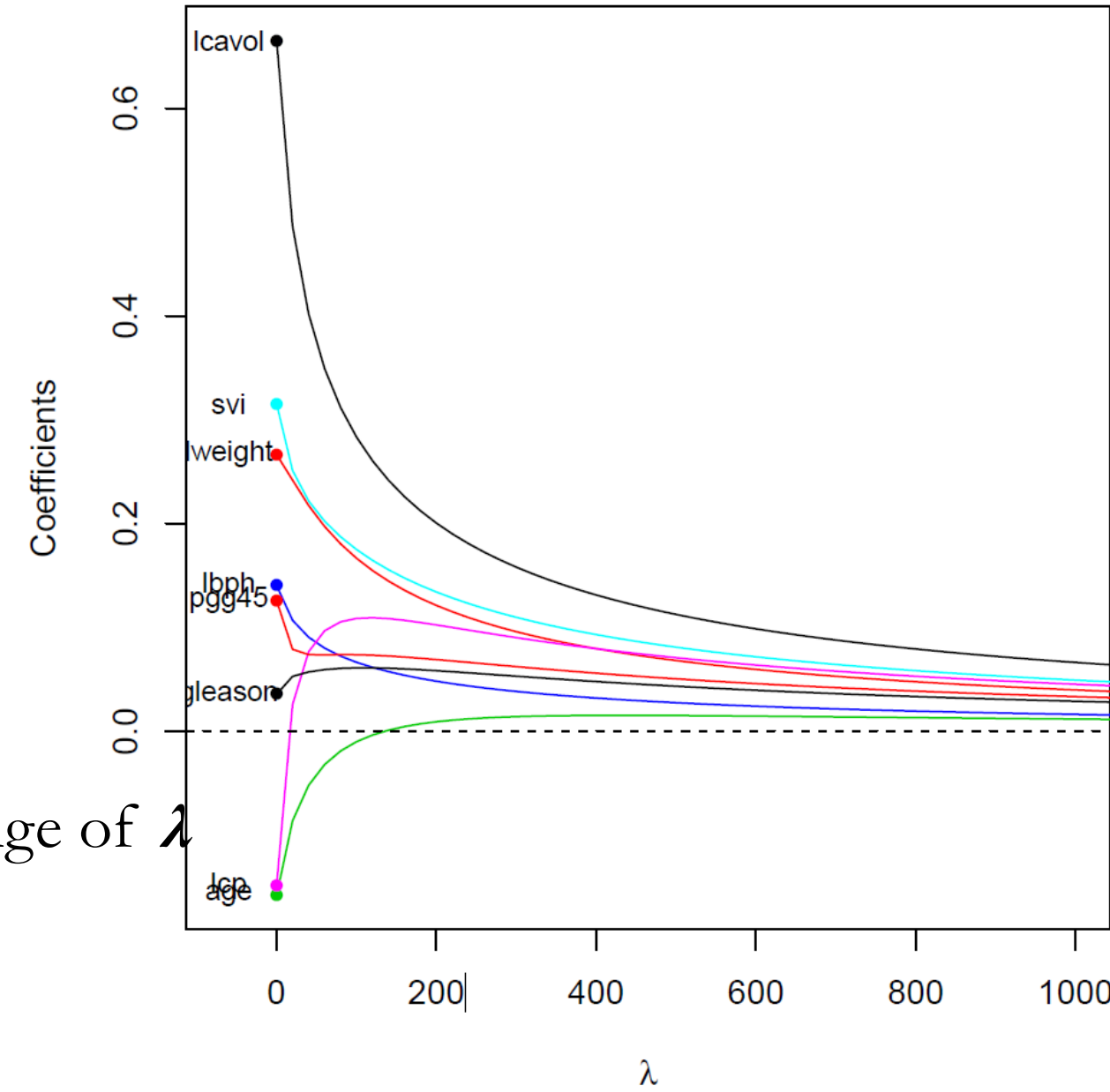
We look at n = 97 men with prostate cancer & p clinical measurements.

We are interested in identifying a small numbe predictors, say 2 or 3, that drive PSA.

We perform ridge regression over a wide range of λ

This does not give us a clear answer...

Solution: Lasso regression



Lasso regression

The lasso coefficients are defined as:

$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \ \|y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

$$= \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \ \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_1}_{\text{Penalty}}$$

The only difference between lasso vs. ridge regression is the **penalty term**

- Ridge uses ℓ_2 penalty $\|\beta\|_2^2$
- Lasso uses ℓ_1 penalty $\|\beta\|_1$

Lasso regression

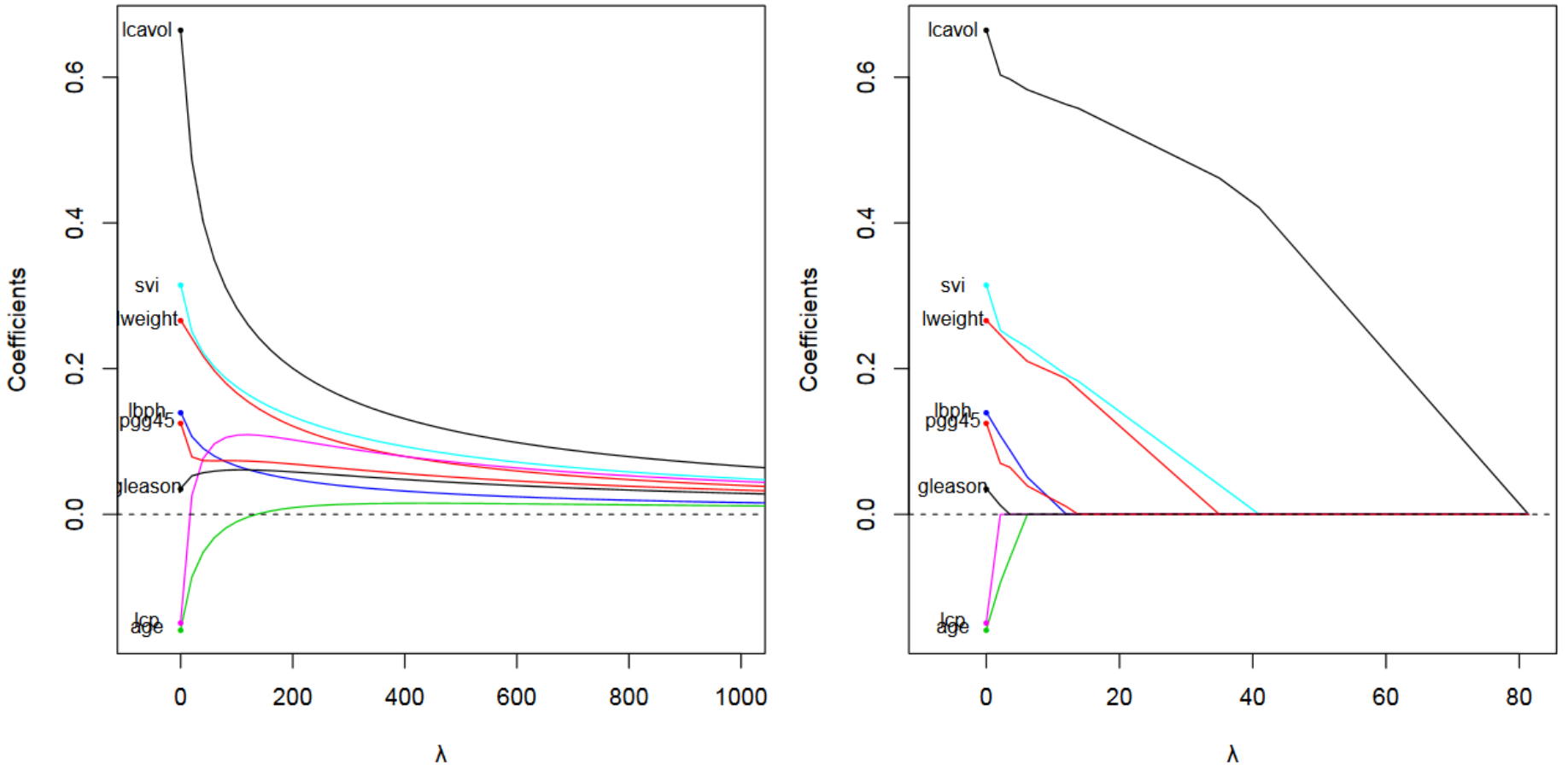
$\lambda \geq 0$ is a tuning parameter for controlling the strength of the penalty

The nature of the ℓ_1 penalty causes some coefficients to be shrunken to zero exactly

As λ increases, more coefficients are set to zero → less predictors are selected

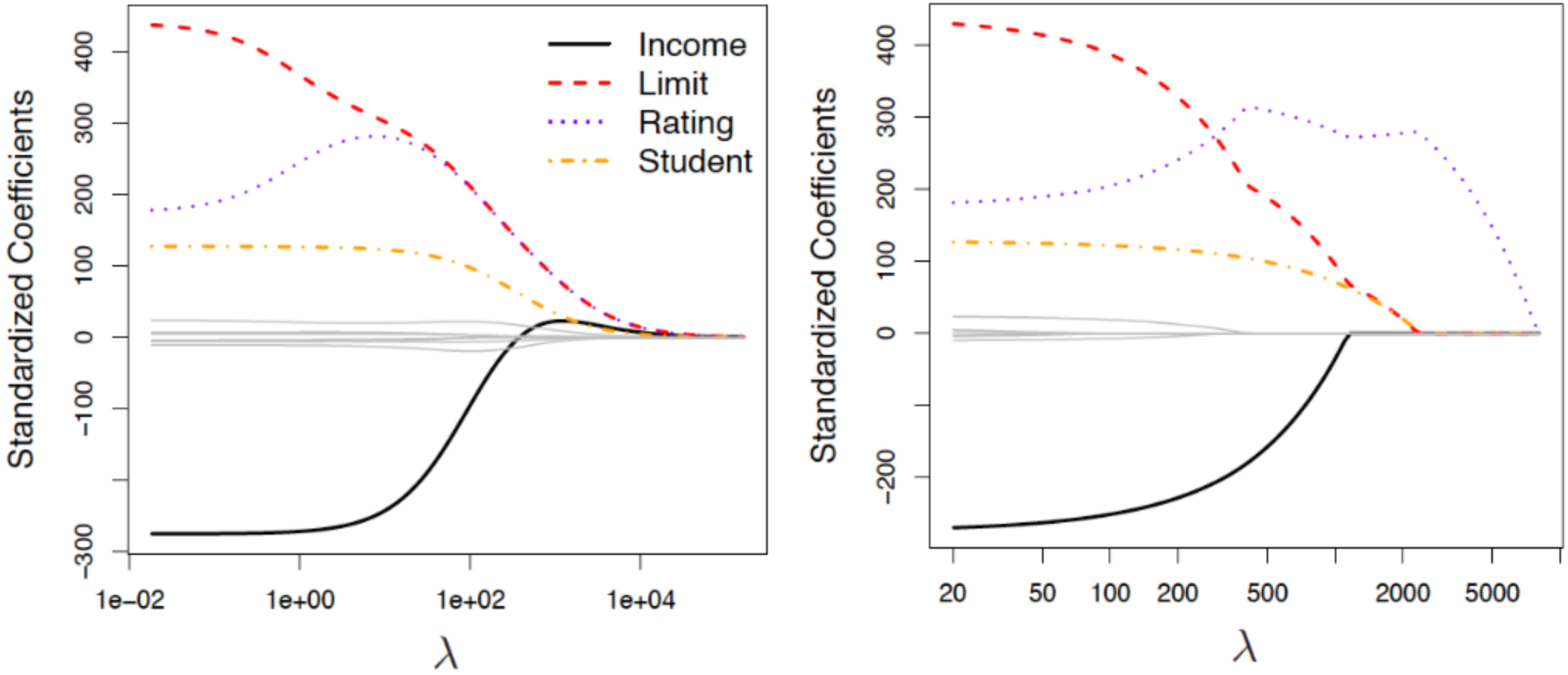
Can perform variable selection

Example 5: Ridge vs. Lasso



lcp, age & gleason: the least important predictors → set to zero

Example 6: Ridge vs. Lasso



Constrained form of lasso & ridge

$$\hat{\beta}^{\text{ridge}} = \underset{\beta \in \mathbb{R}^p}{\text{argmin}} \ \|y - X\beta\|_2^2 \quad \text{subject to } \|\beta\|_2^2 \leq t$$

$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\text{argmin}} \ \|y - X\beta\|_2^2 \quad \text{subject to } \|\beta\|_1 \leq t$$

For any λ and corresponding solution in the penalized form, there is a value of t such that the above constrained form has this same solution. The imposed constraints constrict the coefficient vector to lie in some geometric shape centered around the origin

Type of shape (i.e., type of constraint) really matters!

Why lasso sets coefficients to zero?

The elliptical contour plot represents sum of square error term

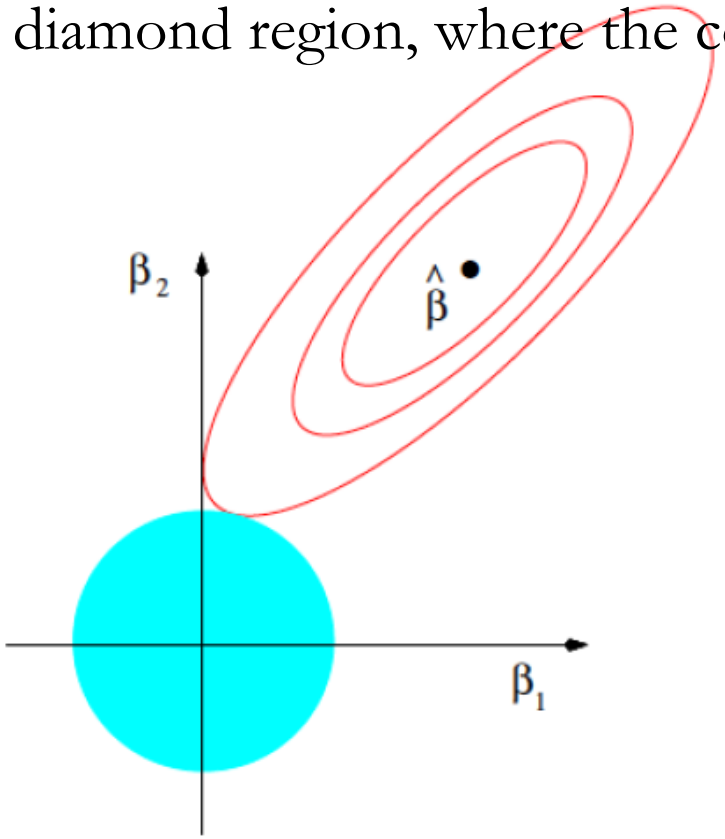
The diamond shape in the middle indicates the constraint region

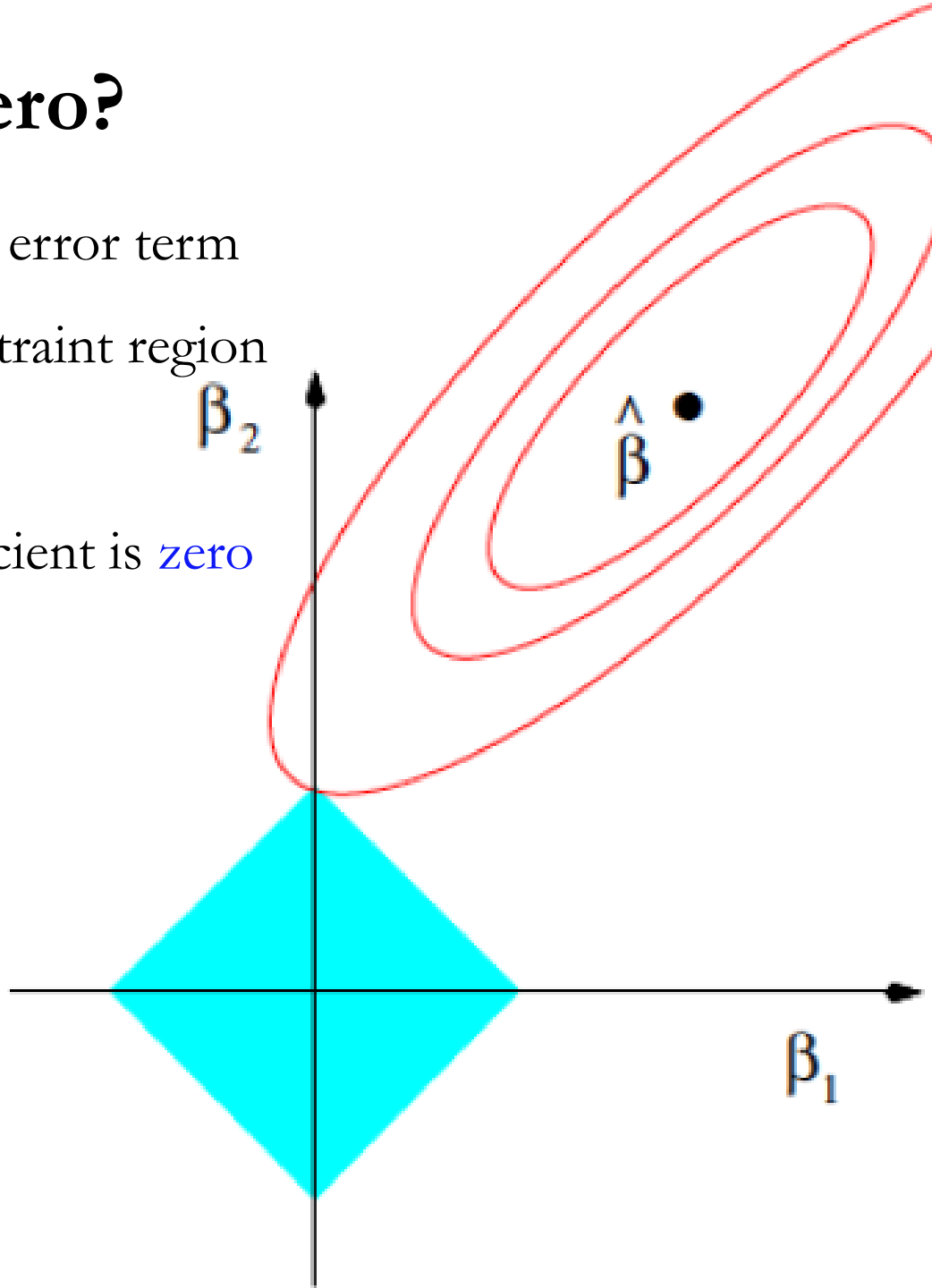
Optimal point: intersection between ellipse & circle

- Corner of the diamond region, where the coefficient is zero

Instead

with ridge:





Regularization penalizes hypothesis complexity

- L2 regularization leads to small weights
- L1 regularization leads to many zero weights (sparsity)
- Feature selection tries to discard irrelevant features

Matlab code & examples

```
% Lasso regression

B = lasso(X,Y); % returns beta coefficients for a set of regularization parameters lambda
[B, I] = lasso(X,Y) % I contains information about the fitted models


% Fit a lasso model and let identify redundant coefficients
X = randn(100,5);              % 100 samples of 5 predictors
r = [0; 2; 0; -3; 0;];         % only two non-zero coefficients
Y = X*r + randn(100,1).*0.1;   % construct target using only two predictors
[B, I] = lasso(X,Y);           % fit lasso

% examining the 25th fitted model
B(:,25)        % beta coefficients
I.Lambda(25)   % lambda used
I.MSE(25)      % mean square error
```

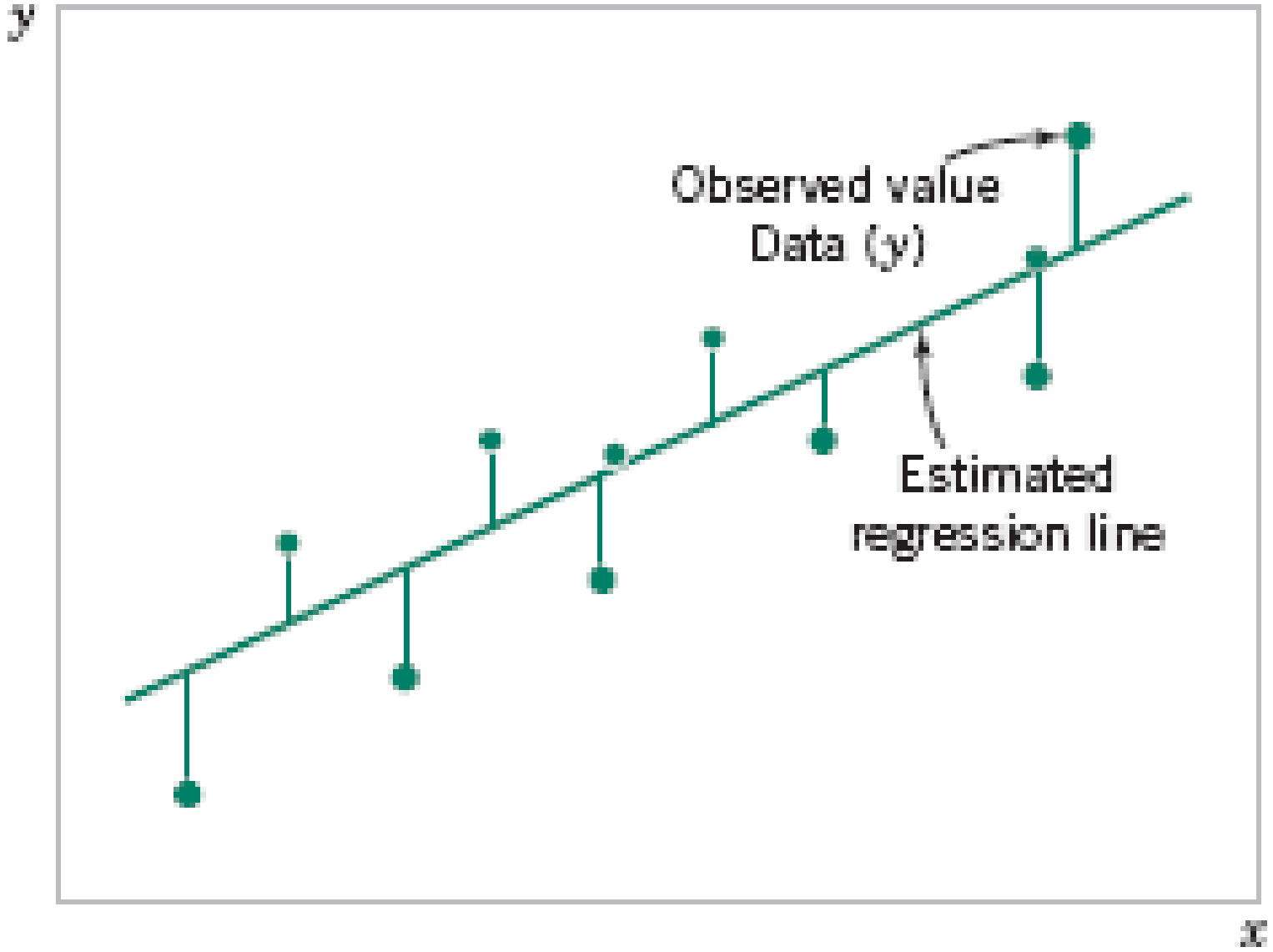
Matlab code & examples

```
% Ridge regression

X = randn(100,5);              % 100 samples of 5 predictors
r = [0; 2; 0; -3; 0;];         % only two non-zero coefficients
Y = X*r + randn(100,1).*0.1;  % construct target using only two predictors

model = fitrlinear(X,Y, 'Regularization', 'ridge', 'Lambda', 0.4));
predicted_Y = predict(model, X);   % predict Y, using the X data


err = mse(predicted_Y, Y);   % compute error

model.Beta     % fitted coefficients
```

Simple Linear Regression

Suppose that we have n pairs of observations (x_1, y_1), (x_2, y_2), …, (x_n, y_n).

Deviations of the data from the estimated regression model.

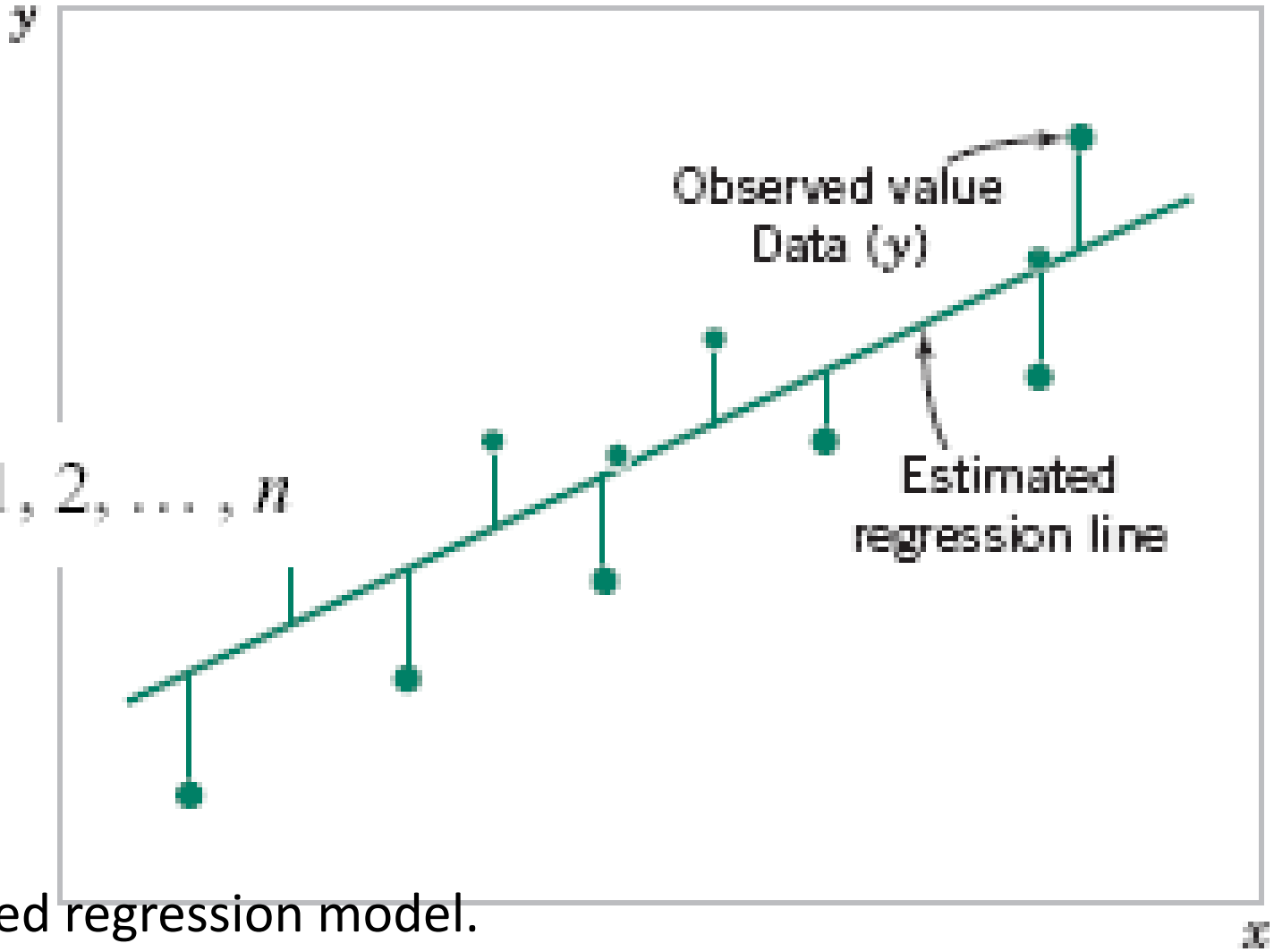


Simple Linear Regression - Least Squares

The **method of least squares** is used to estimate the parameters, β_0 **and** β_1 by **minimizing** the sum of the squares of the vertical deviations

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \qquad i = 1, 2, \ldots, n$$

$$L = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$



Deviations of the data from the estimated regression model.