META-TEOP

Computational Neuroscience Workshop

Dr. Alexander Casti

Mathematics Enrichment Through Applications
Technical Enrichment and Outreach Program
Sponsored by the MAA Tensor-SUMMA Program

FAIRLEIGH DICKINSON UNIVERSITY

Fortiter et Suaviter

Summary of Computational Neuroscience Project: *Testing the Threshold of Human Vision*

Primary Goal: **Perform a "psychophysical" experiment to estimate the *minimum number of photons* required for a human observer to say that he/she saw the stimulus (a flash of light).**

Psychophysics is the study of the relationship between physical stimuli (light, sound, touch,…) and how they are perceived (by our eyes, ears, skin,…)

Experimental Tools:

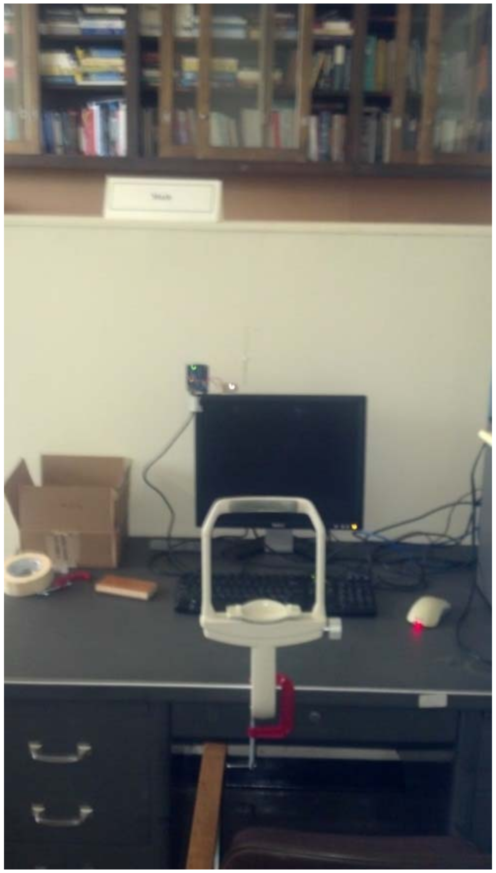
- **LED light source** (green light, 505 nm), a **computer monitor** (to fixate the eye), a **microcontroller** that controls the photons emitted by the LED, and **computer software** (Matlab/Psychophysics Toolbox) to control the light stimulus and record user responses

Mathematical Tools:

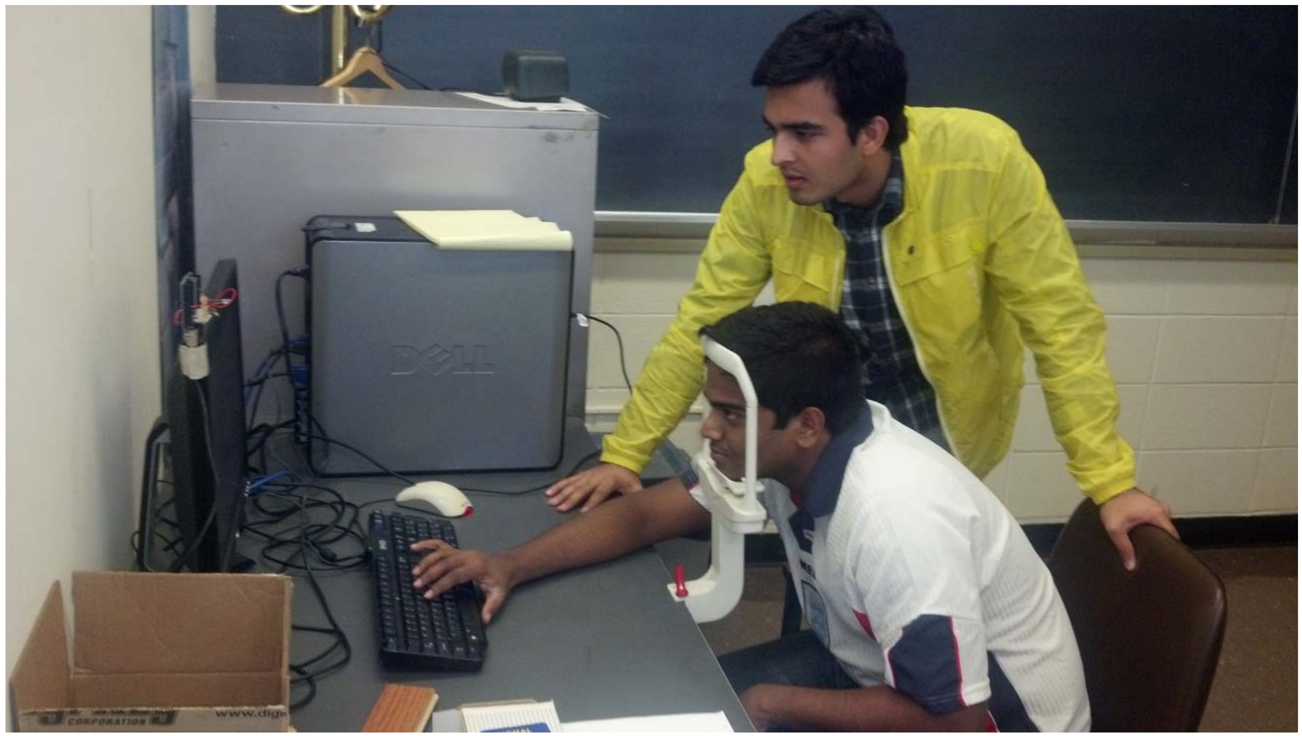
- Basic concepts of **probability** and **random variables** (light source produces randomness, user perceptions and responses produce randomness, etc): **Did you "see it" or not?**
- **Probability distributions**, specifically the **Poisson distribution** (appropriate for our visual experiment)
- **Curve fitting algorithms** to fit our experimental data to the mathematical model (associated with the Poisson probability distribution)

Experimental Setup
(to take place in a dark room)

Experimental Setup



FDU Electrical Engineering students participating in the project: Mahamuge Ruwan Costa and Anirudh Bhatotia



1. Observer fixates vision on a red fixation dot on the computer monitor.
2. Green LED (505nm) atop the monitor flashes at various intensities, resulting in variable photon fluxes at the eye.
3. Observer indicates through keyboard input whether he or she saw the LED flash.
4. Responses are analyzed assuming Poisson statistics for the photon absorptions at the retina, from which one may infer the minimal number of photons required to elicit an “I saw it” response from observers (i.e. on 60% of trials).

Schedule

Week 1

- Introduction to the Early Visual Pathway and the Retina
- Discuss how Retinal Rods "count photons"
- Outline the classic experiment by Hecht, Schlaer, and Pirenne (1942) to estimate the minimal number of photons required for a human observer to say that he "saw the flash" up to 60% of the time
- Introduction to the mathematical software package MATLAB
- Introduction to Probability and Random Events
- Bernoulli trials, Binomial Distribution, Poisson Distribution and associated class exercises

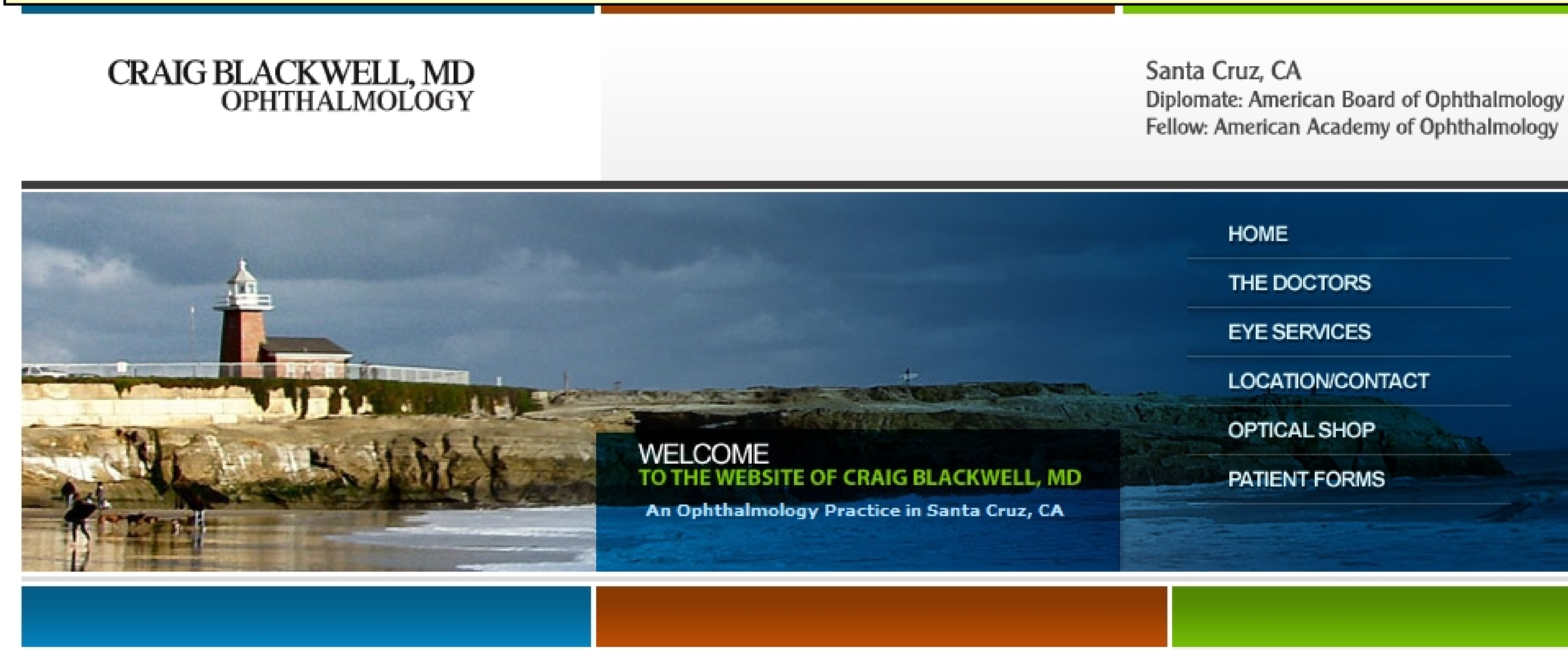
Week 2

- Collect experimental data (in groups of 4)
- Learn how to fit probability models (Poisson model) to noisy experimental data

Week 3

- Analyze experimental data and fit it to the Poisson probability model
- Interpret the results and draw a conclusion about the minimal number of photons required to evoke a visual
- Write up the results and conclusions of the data analysis. Be prepared to discuss your results in a mock scientific conference poster presentation (final week)

YouTube Tutorials on the Retina



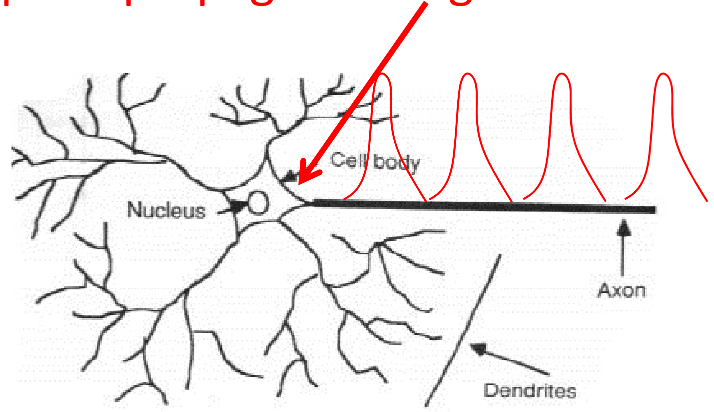
Colin Blackwell (UC Santa Cruz) website:
http://www.blackwelleyesight.com/eye-care-articles/380/

Particularly relevant video: "The image: Retina, Optic Nerve, and Brain"
http://www.youtube.com/watch?v=ajnsDVsP0Uk

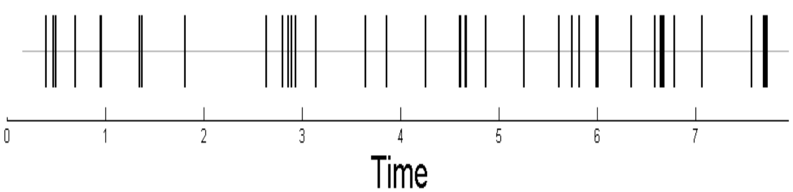
The Visual System

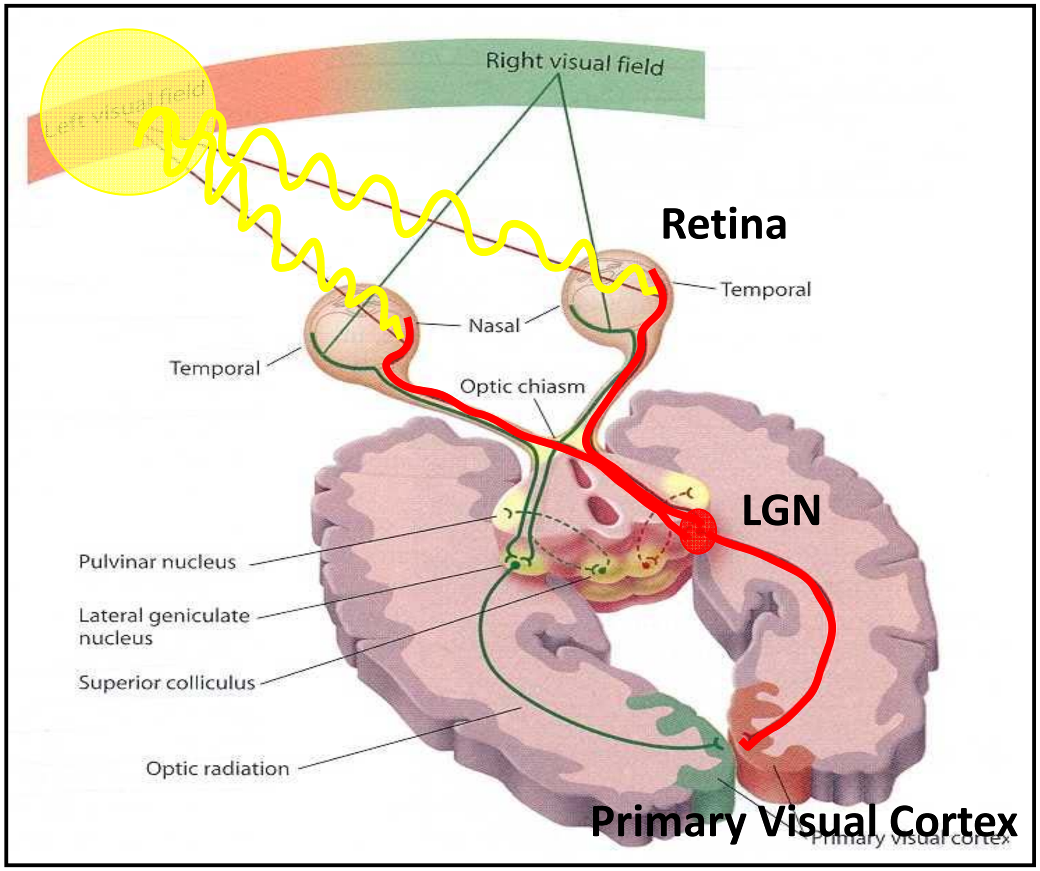
Light patterns → visual representation in the brain

Spikes propagate along axon to other cells



Spike Train (action potentials)



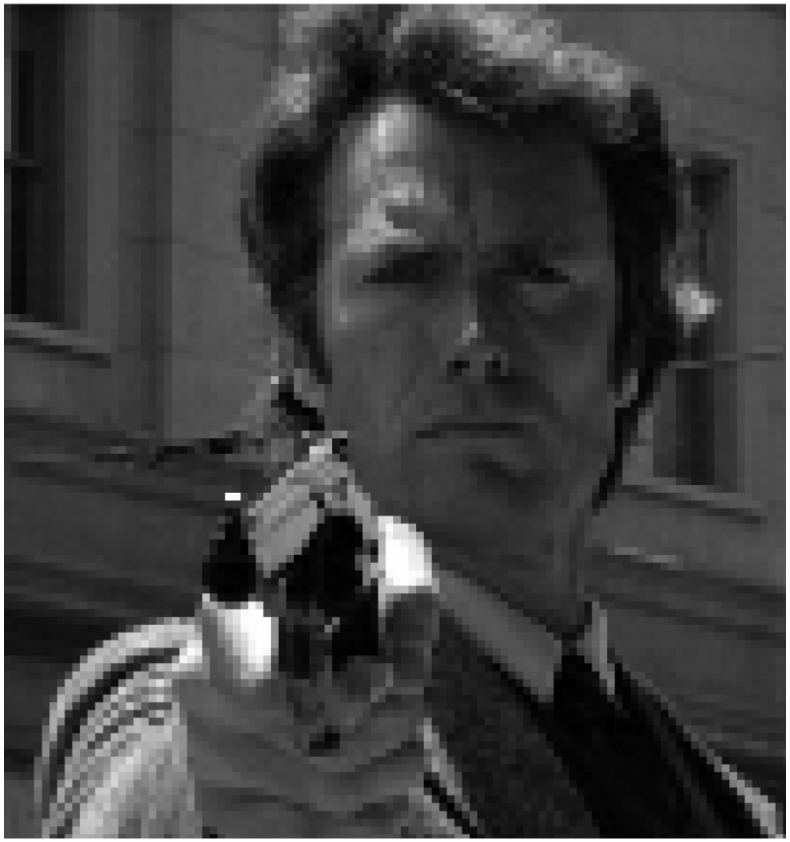


Visual Perception and Decoding

Brain must interpret an abstraction of the visual scene: spike trains, graded potentials, etc.

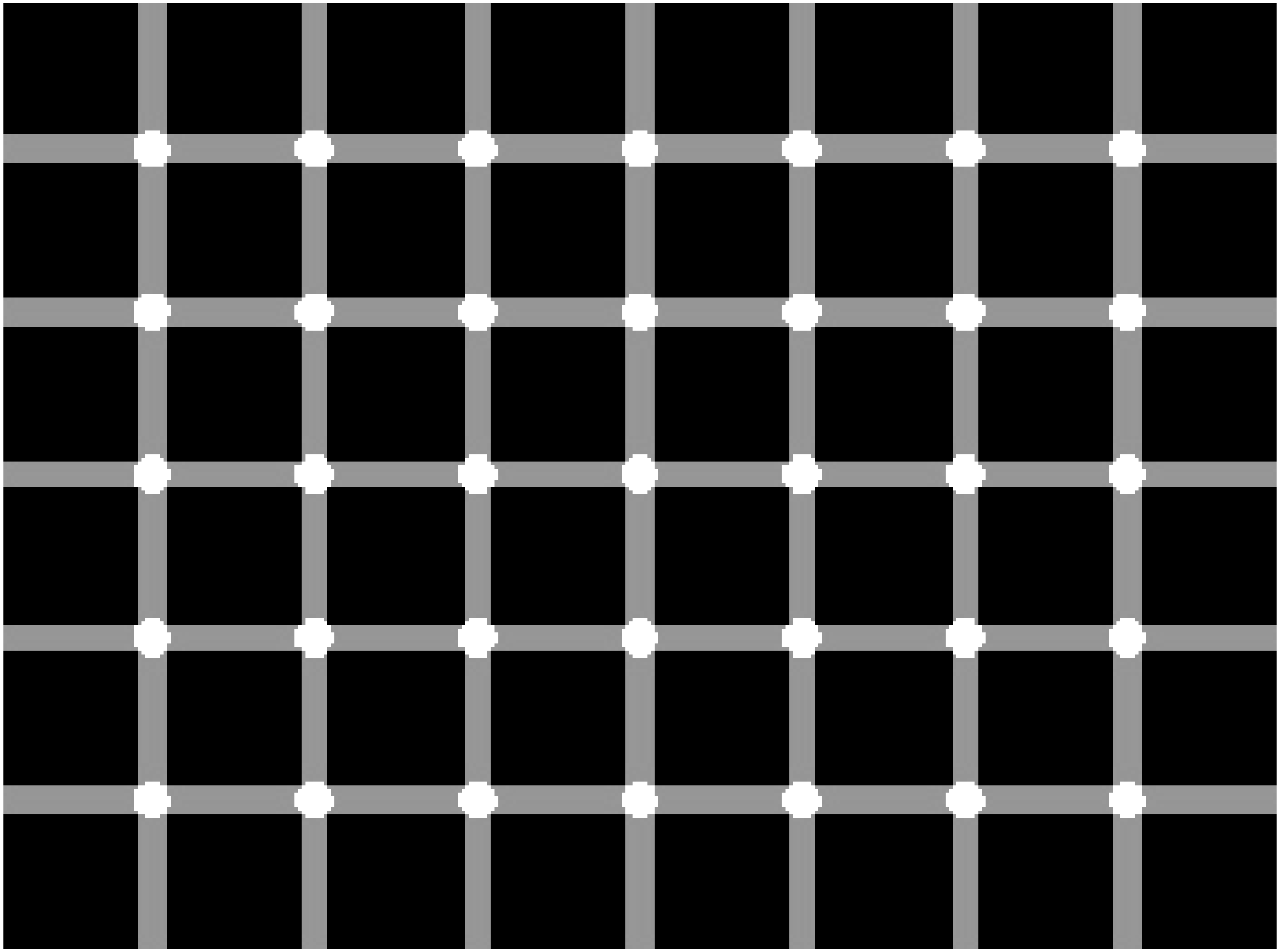
Encoded Neural Activity (i.e. spike rates) at each location
What's this?

Decoding



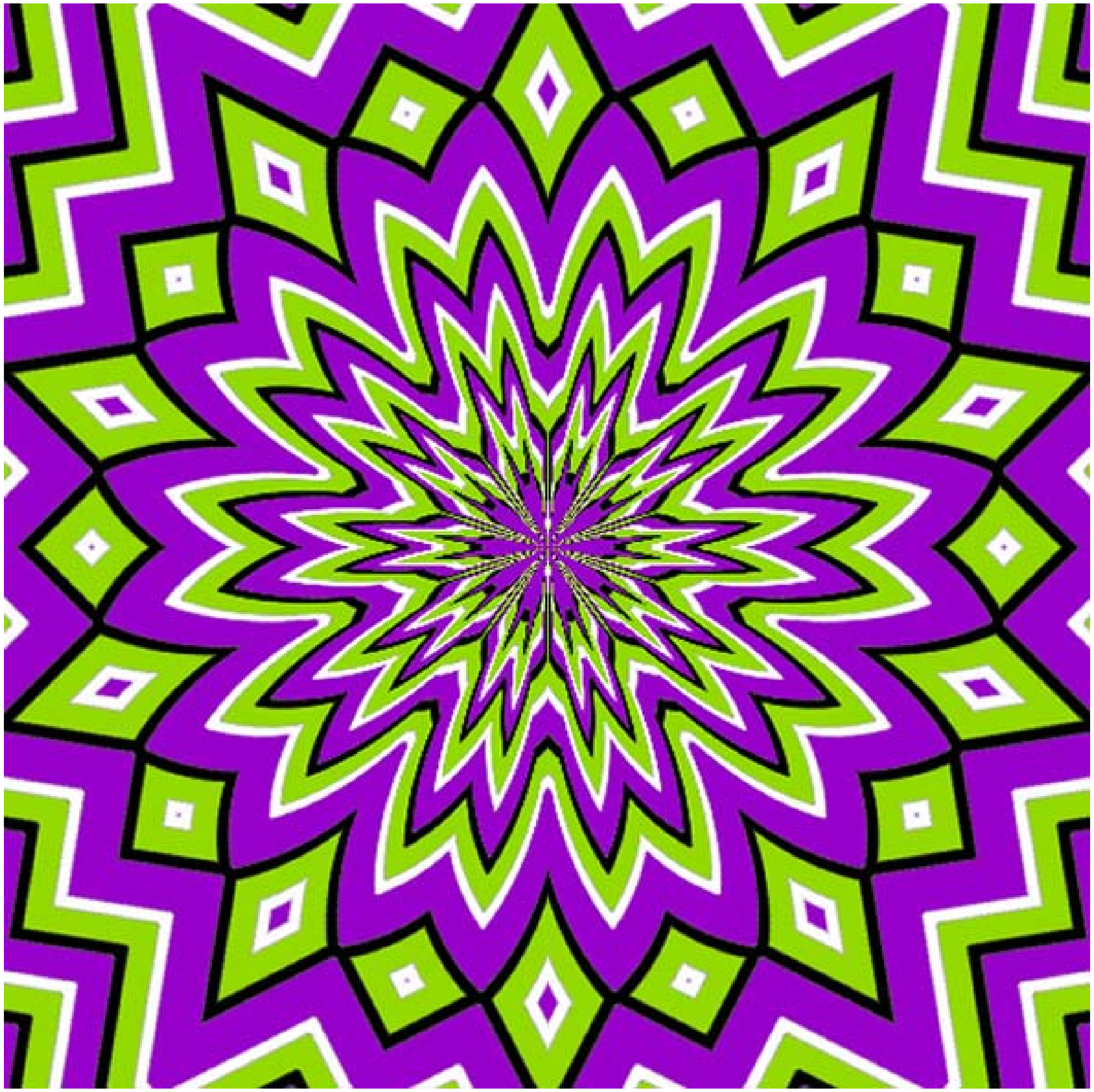
Cool Visual Illusions 1 (just for fun)

The "scintillating grid effect"



Cool Visual Illusions 1 (just for fun)

Motion effects due to "saccadic eye movements"



The Eye

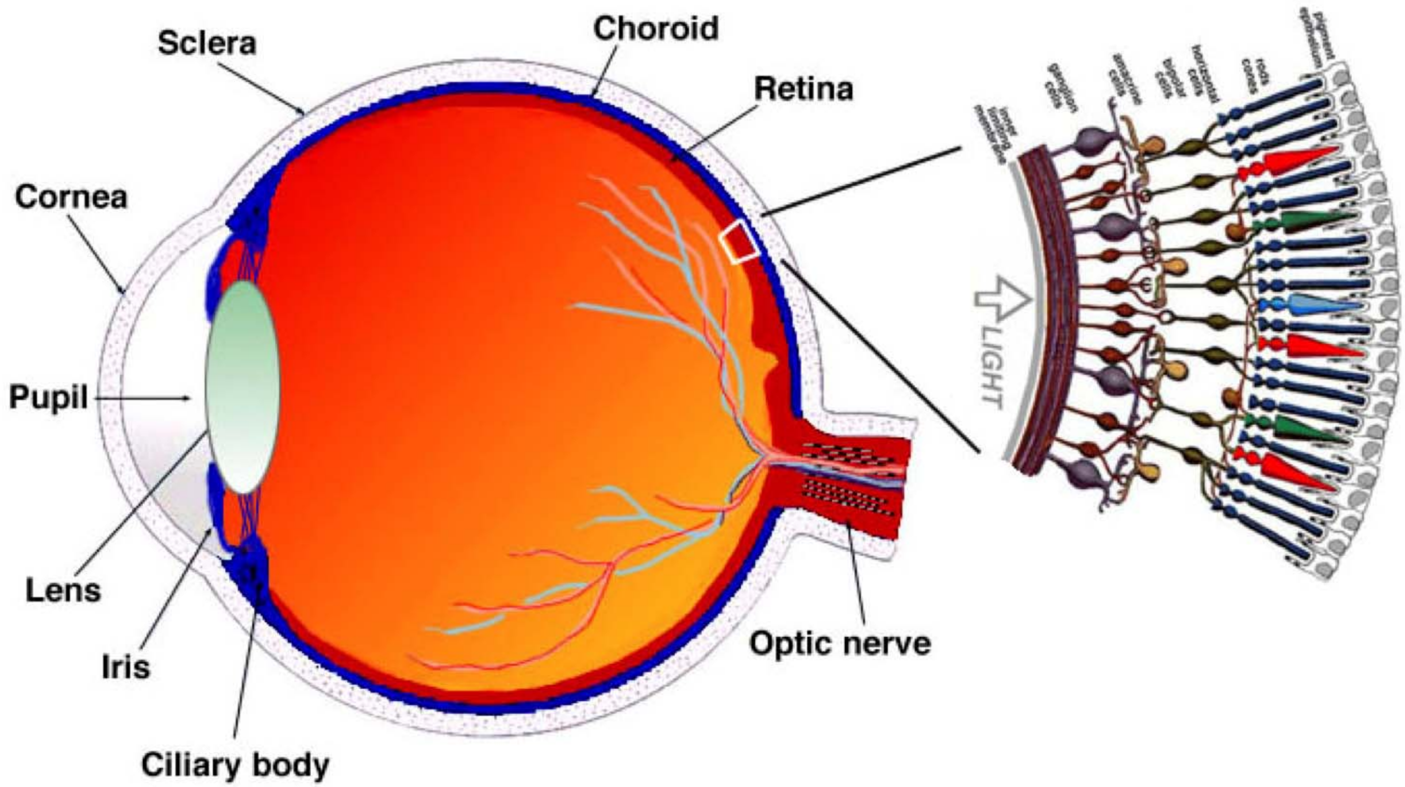
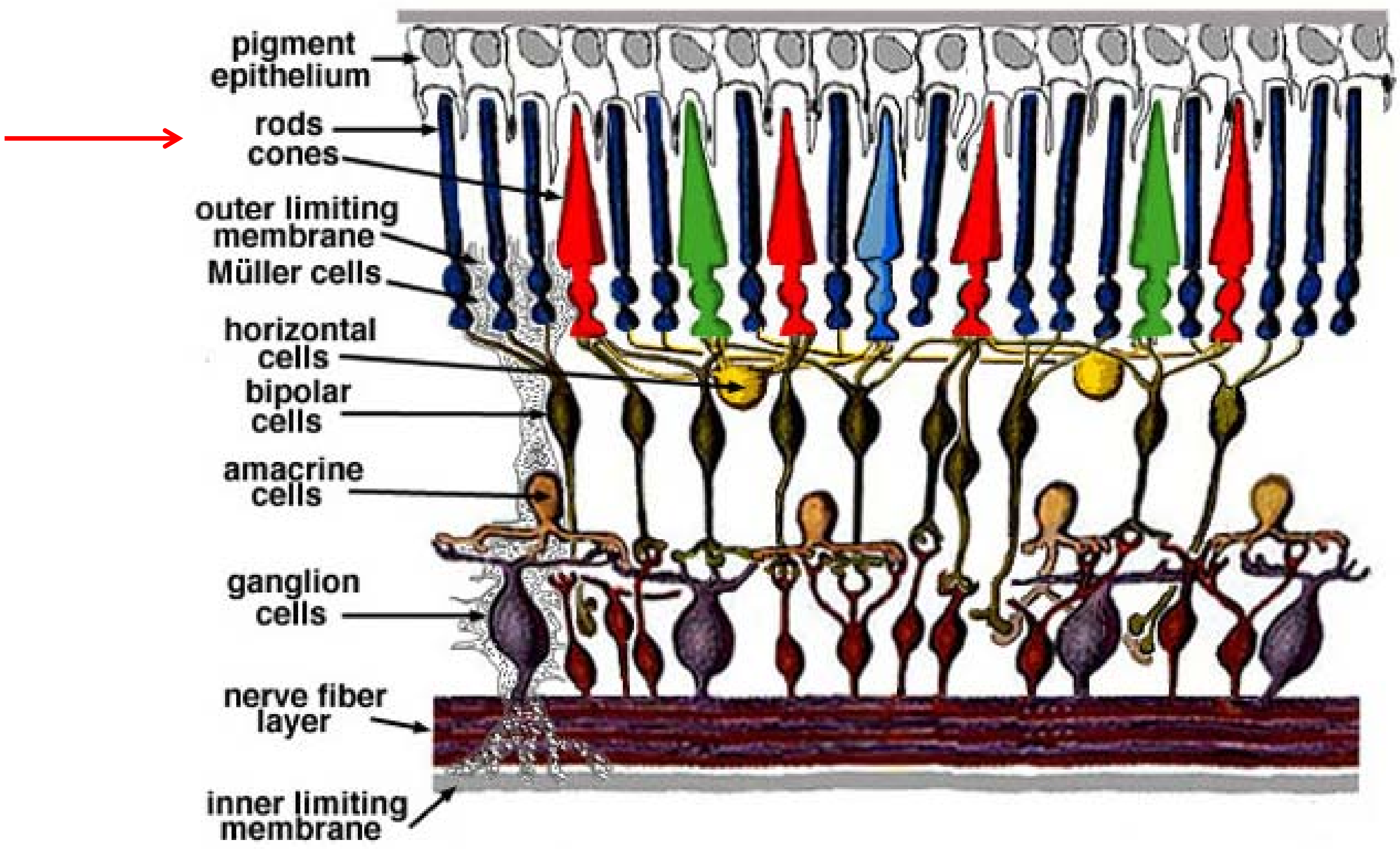


Fig. 1. A drawing of a section through the human eye with a schematic enlargement of the retina

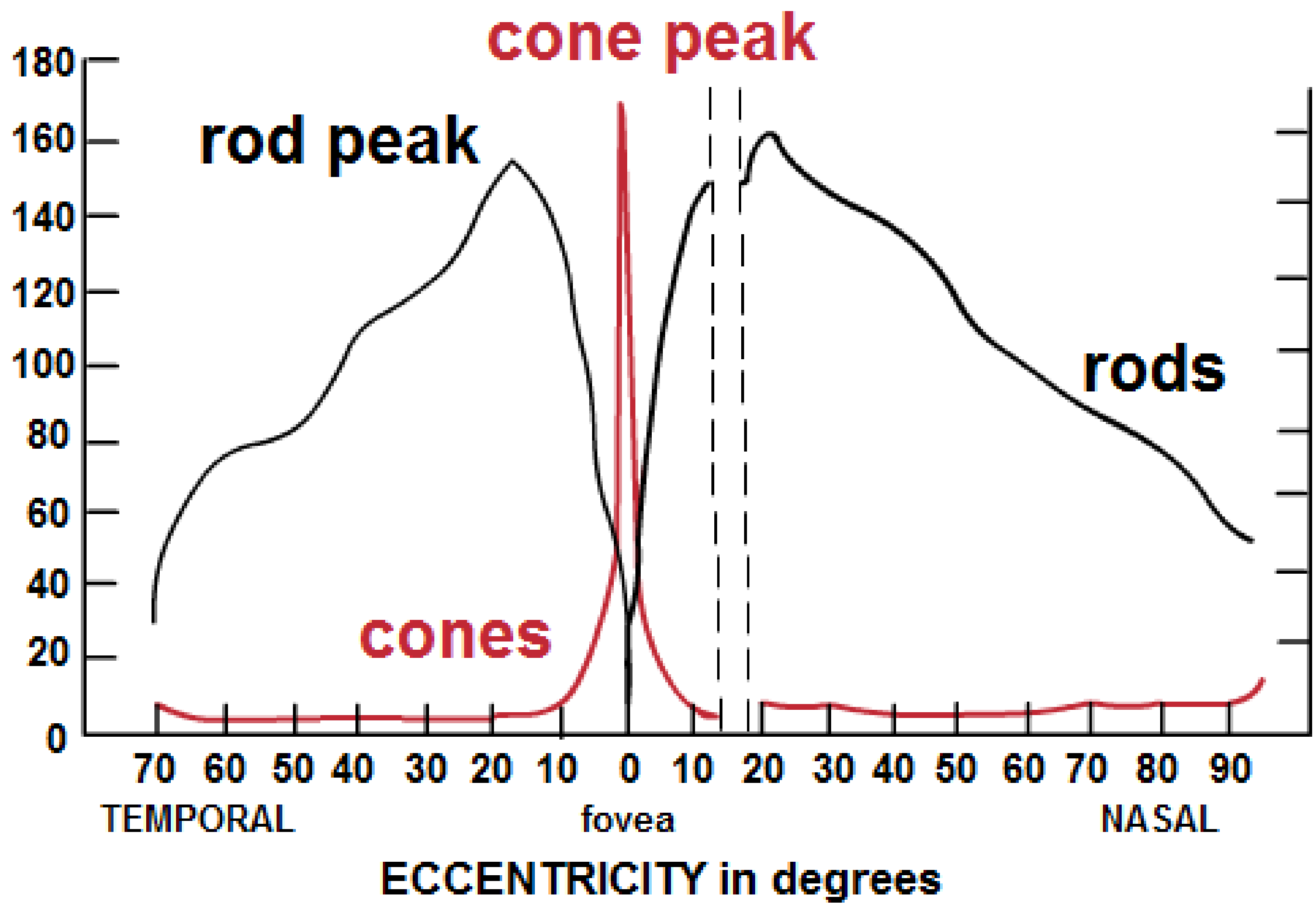
Retina

Visual processing begins in the retina



Rod/Cone Density in the Retina

Flash stimulus location will be chosen to "hit" the maximum density of rods in the eye

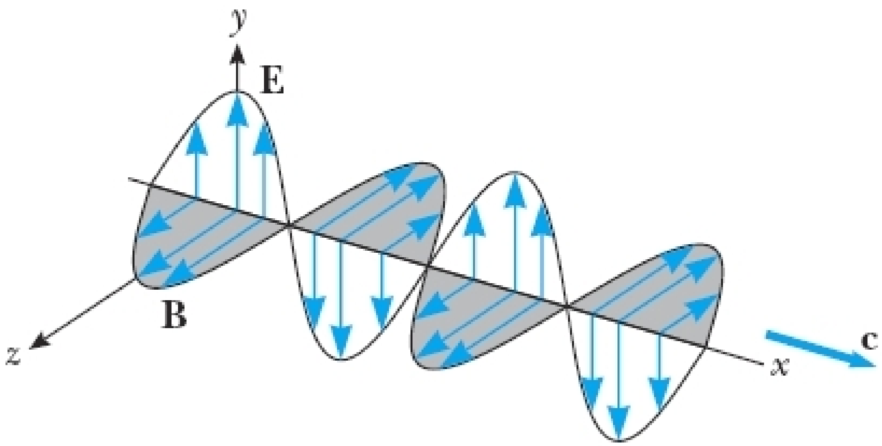


Osterberg, 1935

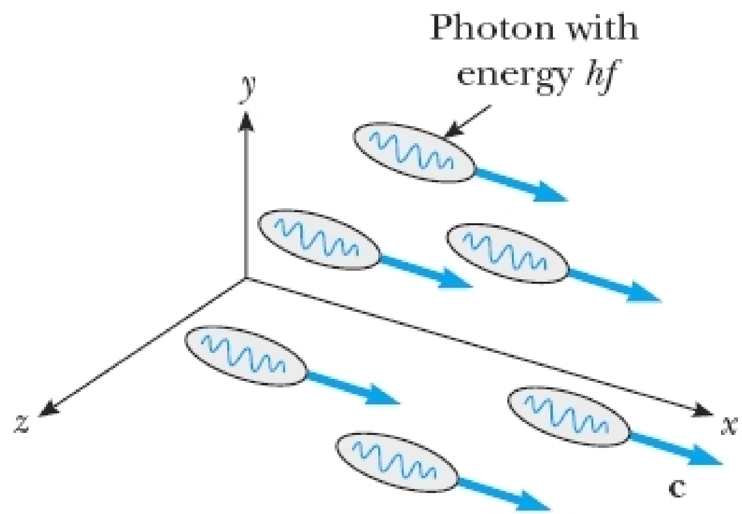
Classical vs Quantum View of Light Energy

Classical View: Light is an electromagnetic travelling wave distributed continuously in space

Einstein's Quantum View: Light is distributed as discrete packets of energy (**photons!**)



(a)



(b)

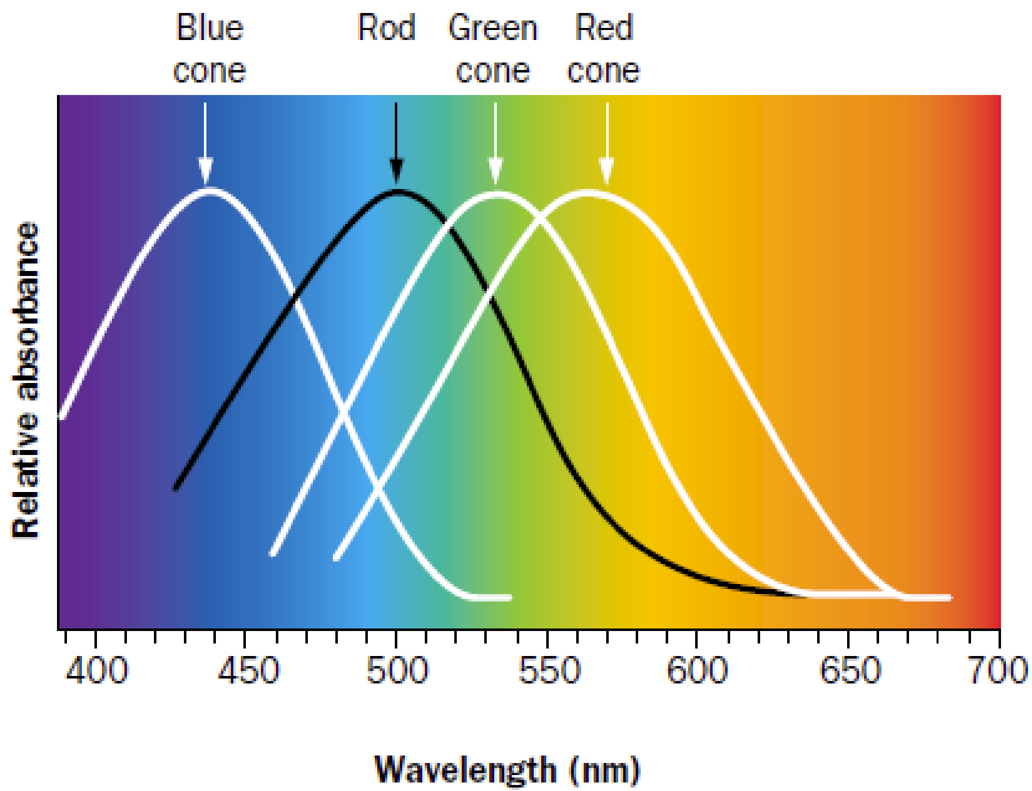
Albert Einstein correctly suggested that the energy of light is not distributed evenly in space as a classical electromagnetic wave as in figure (a), but is rather concentrated in discrete regions (quanta/photons) which each contain an energy E = hf

$$\left.\begin{array}{l} E = \text{energy of each photon} \\ h = \text{Planck's constant} \\ f = \text{frequency of light (cycles/second)} \end{array}\right\} \text{Rods and Cones respond to individual photons}$$

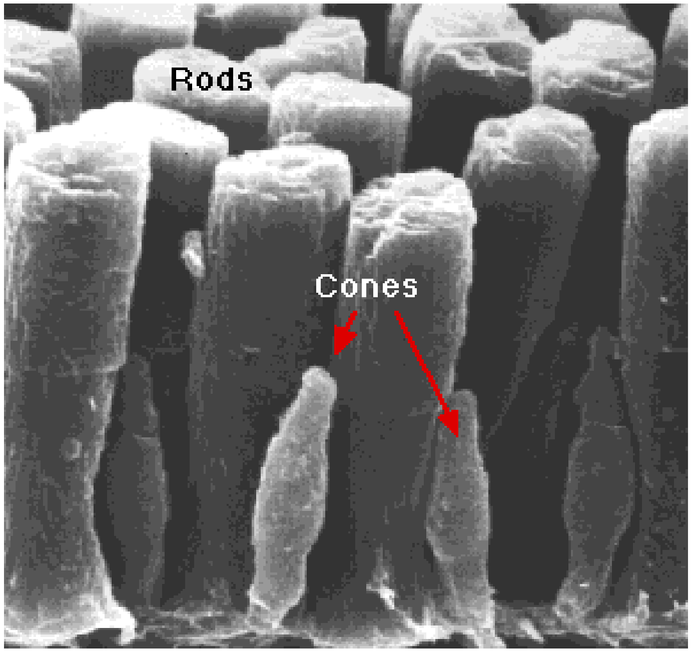
Rods and Cones

- At low light levels (dark room, nighttime, …) rods control early vision and cones are not responsive
- At higher light levels the cones (RED, GREEN, BLUE) control the earliest visual responses

Visible Spectrum

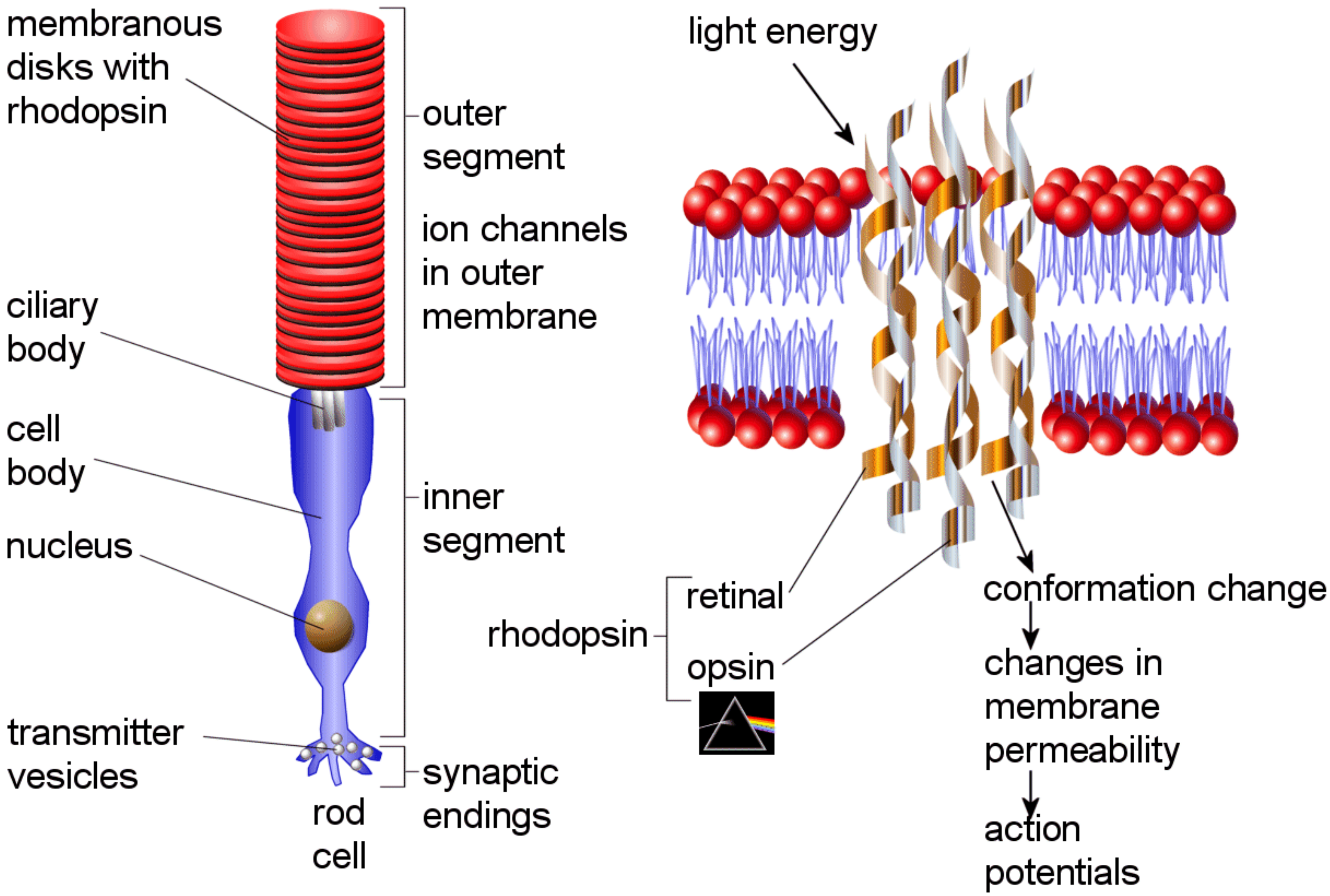


Electron Micrograph of Rods and Cones



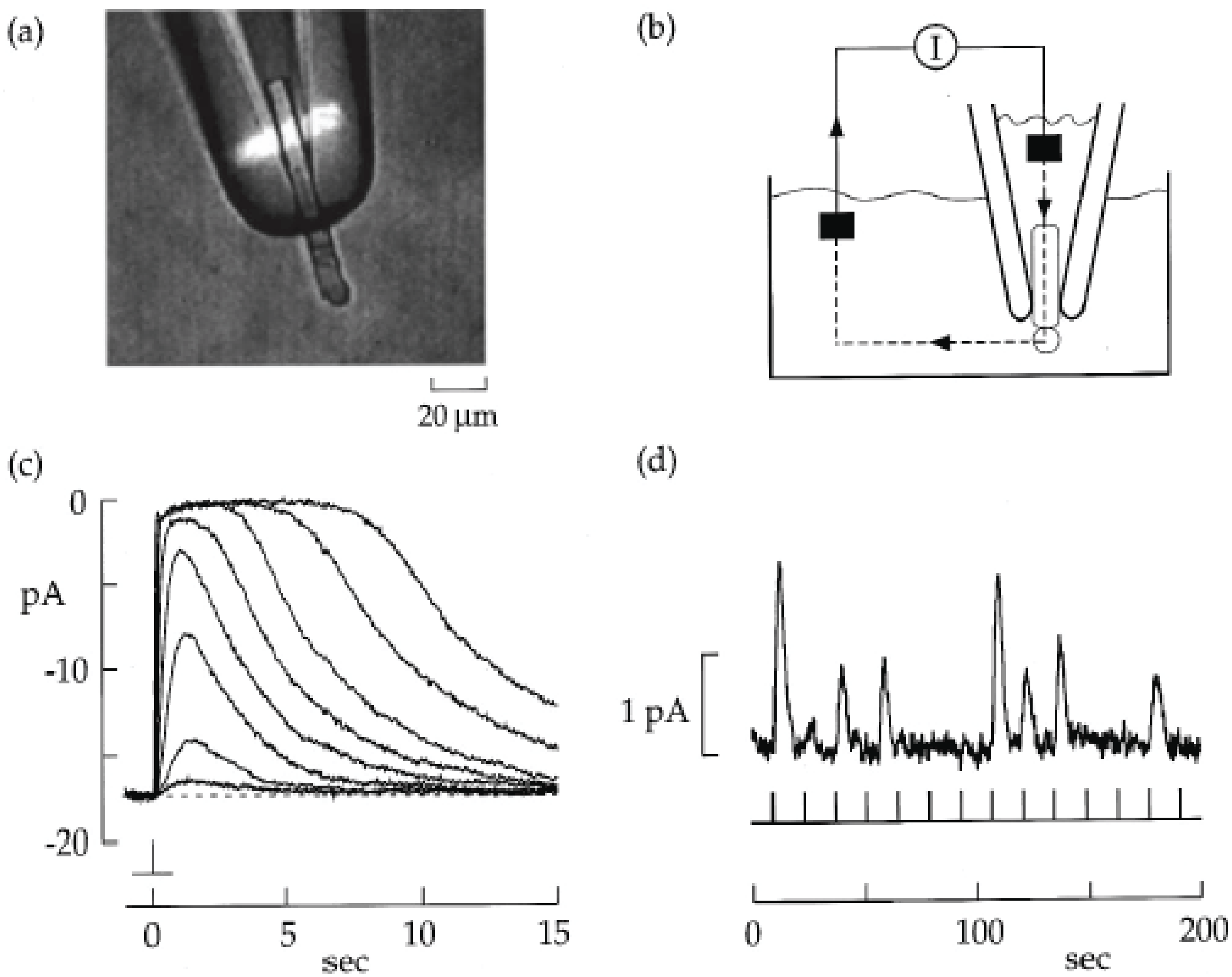
Rhodopsin: The Rod Pigment That Absorbs Light (photons)

- **Rhodospin is the light absorbing pigment in the rods. In this way the light images formed on the retina are converted into electrochemical signals for the brain to interpret.**
- **As rhodospin absorbs light, it changes its shape causing a decrease in the amount of inhibitory neurotransmitters in the synpases between photoreceptor cells and bipolar cells. Rhodospins allow the ability to see shades of grey, black and white. There is only one type of rod and their rhodospin which are sensitive to blue –green light.**
- **As light hits the rhodospin it causes a chemical change which creates decomposition. The active rhodospin changes the charge of rod cell and creates an electric current along the cell. This electric message is sent along the rod to the ganglion, which is connected to the optic nerve. The optic nerve sends the message to the visual cortex so light will be interpreted into an image.**



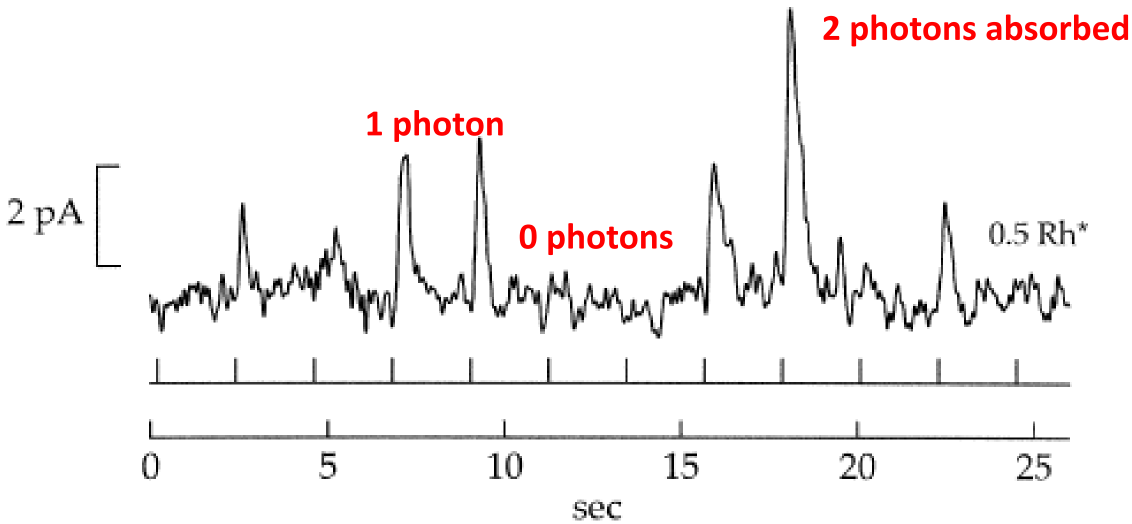
Rod Responses to Brief Pulses of Light

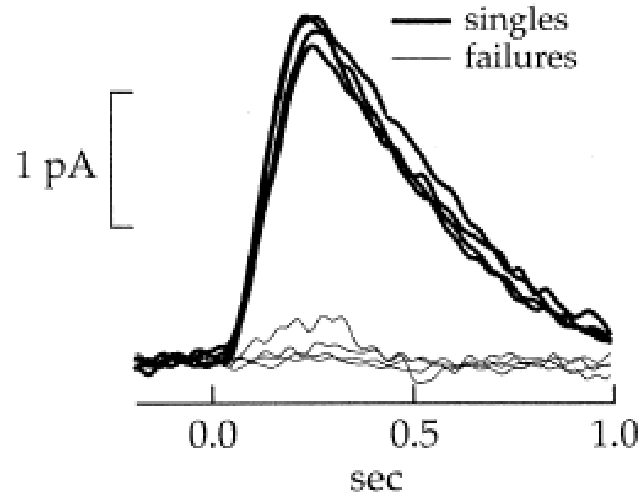
Rod “suction electrode” recordings   (Rieke & Baylor, 1998)

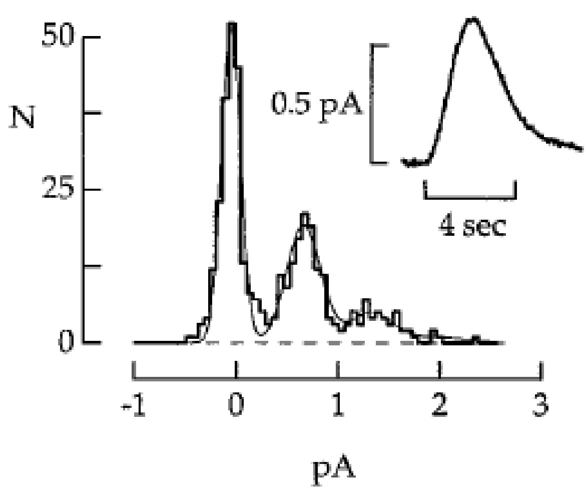


Rod Responses Are Quantized

- Rod responses to light pulses (measured by current) occur as discrete jumps.
- Reponses show no response above noise level (0 photons absorbed), 1 photon absorbed, 2 photons absorbed, etc.







- Distribution of response amplitudes shows the effect of quantization.
- Do human observers respond with similar variability in psychophysical measurements (saw the flash or not)?
- Is the randomness of natural phenomena reflected in an observer's responses? (YES)

Experiment of Hecht, Schlaer, and Pirenne (1942)

- Using themselves as experimental subjects, they flashed a light at various intensities in "dark adapted" conditions
- At each light intensity they tallied the number of "yes" (saw it) and "no" (didn't) trials
- Plotted the "frequency of seeing" vs "light intensity"
- Fit results to a probabilistic model, assuming human observations (even under nominally identical experimental conditions) mirror the randomness of nature

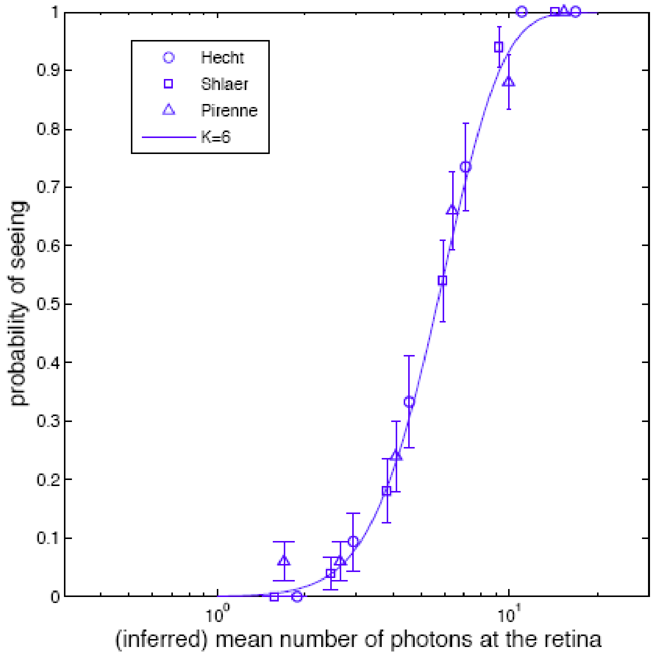


FIG. 2 Probability of seeing calculated from Eq. (2), with the threshold photon count $K = 6$, compared with experimental results from Hecht, Shlaer and Pirenne. For each observer we can find the value of α that provides the best fit, and then plot all the data on a common scale as shown here. Error bars are computed on the assumption that each trial is independent, which probably generates errors bars that are slightly too small.

- Poisson model has 1 free parameter K
- "Best fit" K value corresponded to the minimal # photons required to see the stimulus on at least 60% of the trials
- Note: There are really 2 free parameters if you include a variable that allows us to "slide" the curve horizontally along the intensity axis. This corrects for subject differences associated with age, eye composition, and so forth.

Basic Idea Behind the Data Fitting to the Hecht, Schlaer, and Pirenne experiment

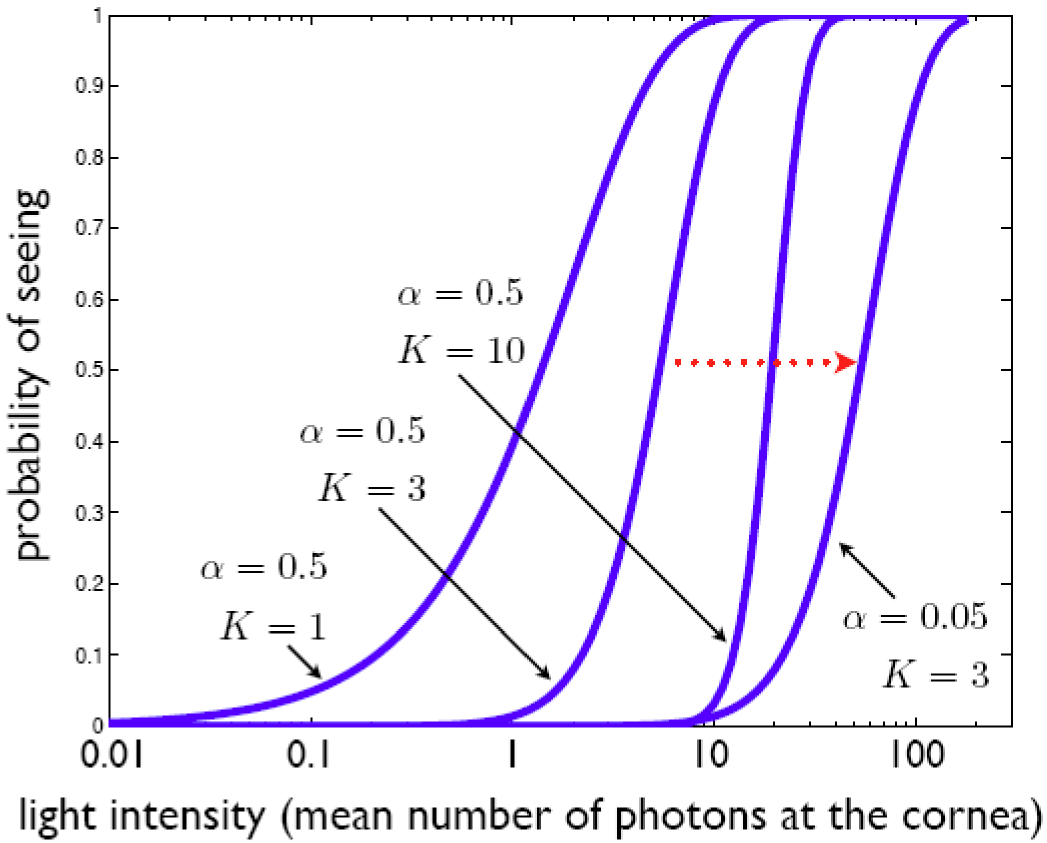


FIG. 1 Probability of seeing calculated from Eq. (2), where the intnesity I is measured as the mean number of photons incident on the cornea, so that α is dimensionless. Curves are shown for different values of the threshold photon count K and the scaling factor α. Note the distinct shapes for different K, but when we change α at fixed K we just translate the curve along the the log intensity axis, as shown by the red dashed arrow.

- **The "K" parameter controls the shape of the theoretical curve**
- **The "α" parameter controls for effects of different observers (i.e. eye degeneration) and serves to shift the curves along the horizontal axis**
- **The best fit value "K" is the visual threshold parameter we seek: the minimum number of photons required for an observer to reliably see the light flash**
- **We will describe the specifics of this "Poisson Model" later**

Switching Gears: Matlab and Probability Theory

- In order to model the randomness of photon absorptions, and the randomness inherent in psychophysical experiments, we need the mathematical language of Probability and Random Experiments (experiments with random outcomes)

- We will recreate the Hecht, Schlaer, and Pirenne (1942) experiment using green LEDs as the light source (week 2)

- Once the data is collected, we need to fit it to our probability model (week 3). This will involve the so-called Poisson probability distribution.

- We will use the mathematical software program MATLAB to do most of our data analysis, so we need to learn the basics of the software.

- Now begins the mathematical and computational portion of the session….

Introduction to Matlab 1 (basic algebra)

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> x = 5;   % Assign numerical values to variables
>> y = 10;
>> mult = x*y    % multiply two numbers

mult =

    50

>> div = x/y     % divide two numbers

div =

                       0.5

>> add = x + y   % add two numbers

add =

    15

>> subt = x - y   % subtract two numbers

subt =

    -5

>>
```

Introduction to Matlab 2 (defining vectors)

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

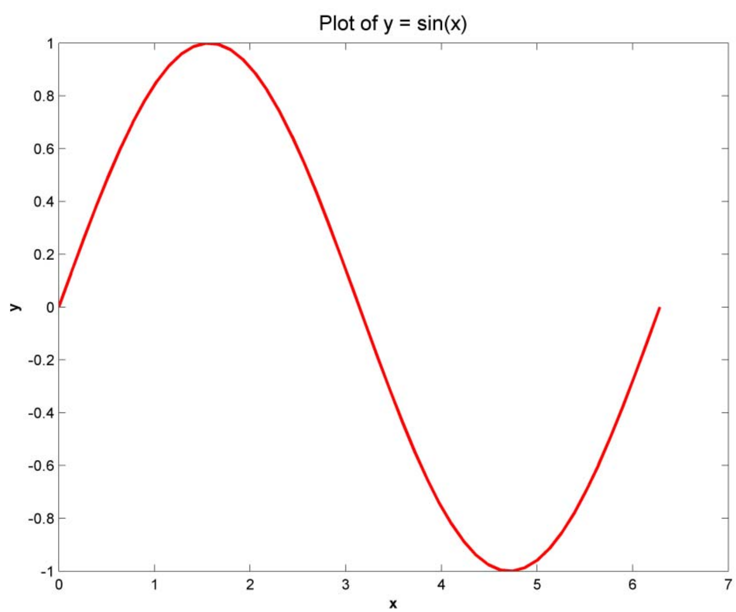
```
>> x = 0:.2:1     % array of numbers from 0 to 1, increments of 0.2

x =

         0    0.2000    0.4000    0.6000    0.8000    1.0000

>> x.^2           % square each number element by element

ans =

         0    0.0400    0.1600    0.3600    0.6400    1.0000

>> y = exp(x)     % define a function of x  (natural exponential)

y =

    1.0000    1.2214    1.4918    1.8221    2.2255    2.7183

>> y./x           % divide x into y  element by element (note the dot operator)

ans =

       Inf    6.1070    3.7296    3.0369    2.7819    2.7183

>> x(3)           % extract the 3rd element of the array x

ans =

    0.4000

>> x(2:4)         % extract elements 2 through 4 of the vector x

ans =

    0.2000    0.4000    0.6000
```

Introduction to Matlab 3 (plot a function of x)

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> x = linspace(0,2*pi,50);      % another way to define an array (vector), 50 elements
>> y = sin(x);                   % define the sine function over this domain of points x
>> plot(x,y,'r-','linewidth',2);  % opens a figure window and plots y vs. x
>> xlabel('x','fontweight','bold');  % add x-axis label
>> ylabel('y','fontweight','bold');  % add y-axis label (note how figure is updated)
>> title('Plot of y = sin(x)','fontsize',14);  % put a title on the graph
>> help plot       % how to get help with the "plot" function within Matlab command window
```




```
>> print -dpng -r400 -painters untitled.png;  % output figure as PNG image file
>> help print                                 % get help with "print" function
```

Introduction to Matlab 4 (doing work within an m-file)

For longer computations and programs you will want to write Matlab code within an editable file called a "Matlab m-file" (with file extension .m)

File Edit Text Go Cell Tools Debug Desktop Window Help

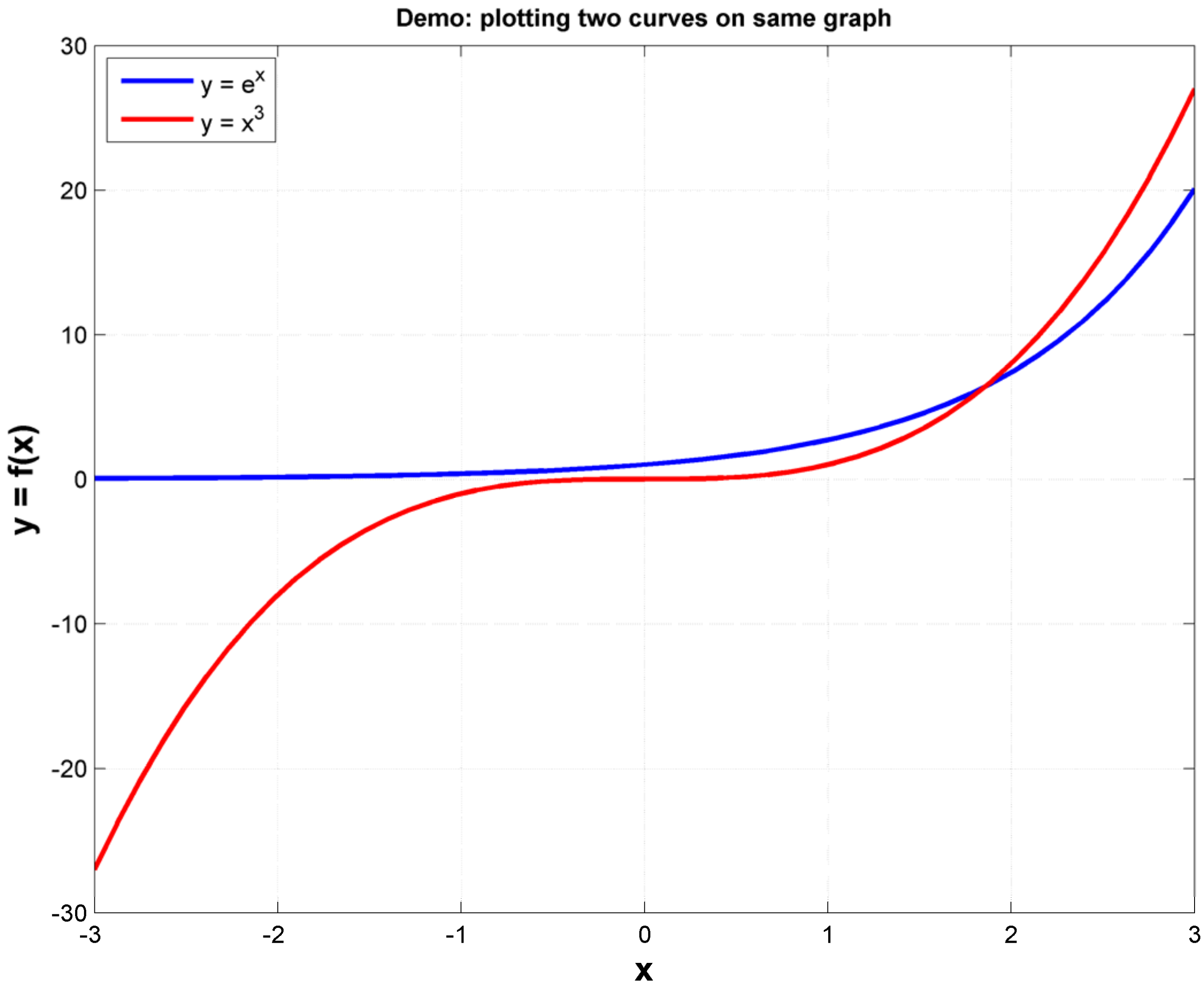
Stack: Base

This file uses Cell Mode. For information, see the rapid code iteration video, the publishing video, or help.

```
%% Plot the natural exponential function  exp(x)  over 50 equally spaced points on the x-domain  [-4,4]
%  Also plot the function  y = x.^3   on the same figure
x = linspace(-3,3,50);   % note that a semi-colon suppresses output to the Matlab command window
y1 = exp(x);             % exponential function (base "e")
y2 = x.^3;               % define another function to plot

figure       % this opens a blank figure window (but "plot" will open one by default anyway)
  plot(x,y1,'b-','linewidth',2); % plots the first function (blue color)
  hold on;         % hold function "freezes" the figure in case you draw more curves on it (which we will)
  plot(x,y2,'r-','linewidth',2); % plots the second function (red color)
  xlabel('x','fontweight','bold','fontsize',14);         % x-axis label
  ylabel('y = f(x)','fontweight','bold','fontsize',14);  % y-axis label
  title('Demo: plotting two curves on same graph','fontweight','bold');
  legend('y = e^{x}','y = x^{3}','Location','Northwest');   % adds a legend so we know which curve is which
  grid on;    % adds grid lines to the figure

% Can run this file by clicking the green arrow in the menu bar above (scroll over it!)
% Or you can type  "castiFunctionPlot" (without the .m extension) at the Matlab command window prompt
% >> castiFunctionPlot
```

This file is provided to you in your folder: "castiFunctionPlot.m"

Introduction to Matlab 5 (result of “castiFunctionPlot.m”)



Matlab Exercise (plotting functions)

- Create a new Matlab m-file: "myFunctionPlot.m" and copy/paste my code "castiFunctionPlot.m" into the editor (or you can copy the file and rename it)
- Modify the m-file to make the plot described below
- Play with the plot function (use the documentation "doc" or the "help") and change line thicknesses, colors, annotations, or whatever you like
- To compute x! in Matlab use factorial(x) (x must be an integer)

Plot the following two functions on the same graph over the domain $x \in [0, 20]$

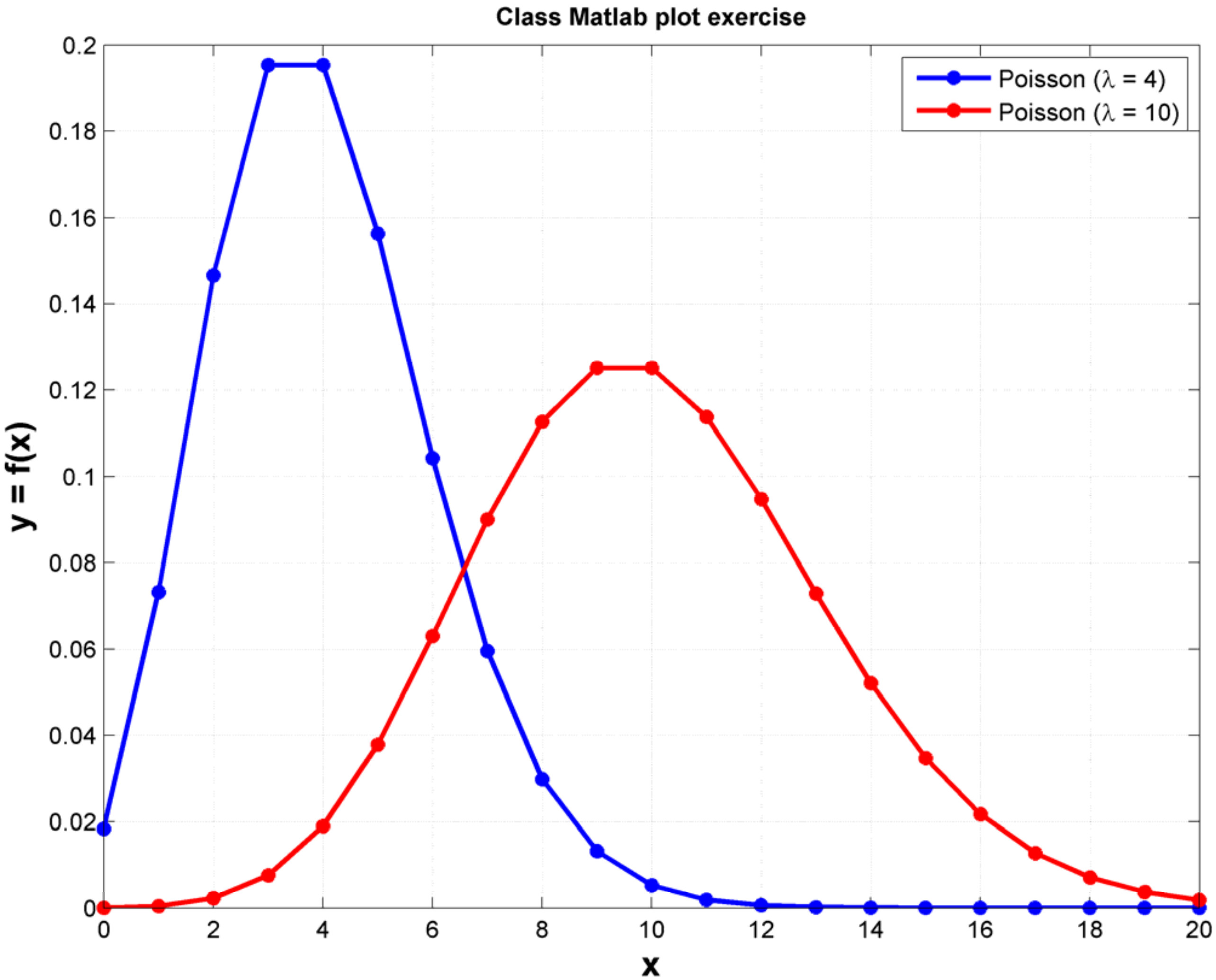
using the gridpoints defined in Matlab by $>> x = 0:1:20$

$$f_1(x) = \frac{4^x}{x!} e^{-4} \qquad \text{(Poisson distribution with rate } \lambda = 4\text{)}$$

$$f_2(x) = \frac{10^x}{x!} e^{-10} \qquad \text{(Poisson distribution with rate } \lambda = 10\text{)}$$

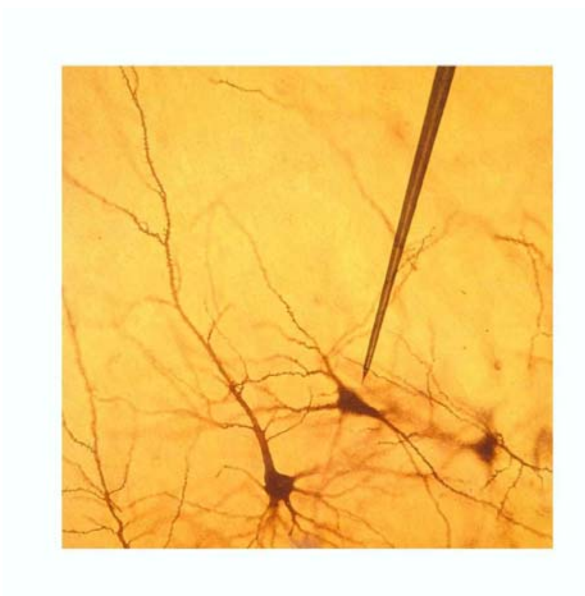
Result of Matlab Exercise (plotting functions)

These curves correspond to the so-called "Poisson probability distribution" that will be of great importance to us in our visual threshold modeling project



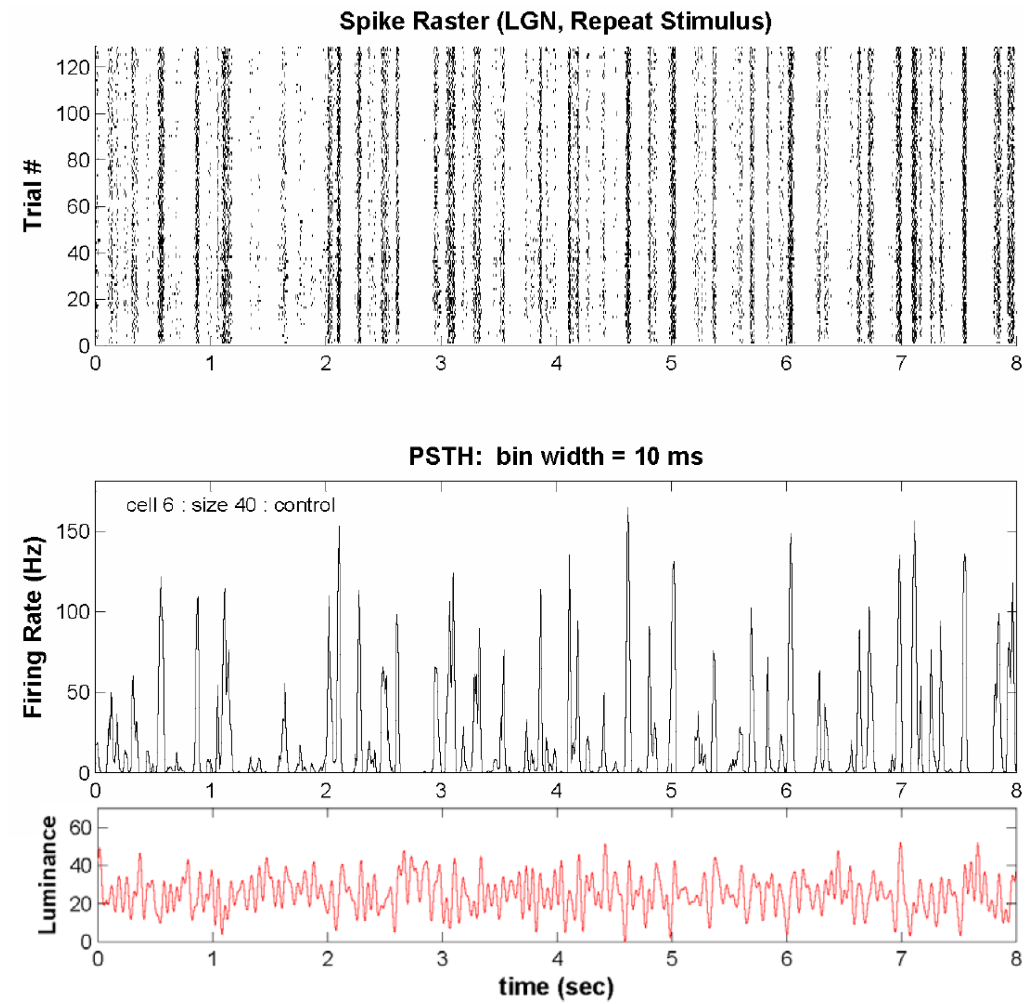
Random Processes and Vision: Yet Another Reminder that cellular responses need a probabilistic description

Stimulus: Random luminance flicker at 160 Hz



128 Repeated Trials (same stimulus)

Neuron recording, cat visual thalamus (LGN)



Basic Probability 1

- **The origins of the Mathematical Theory of Probability are rooted in games of chance**
- **Archaeological digs in the Middle East and India have revealed that people were rolling dice (four-sided sheep bones) as early as 3500 BC.**
- **Modern games with "random outcomes" include blackjack, craps, roulette. The outcomes of such games and your likelihood of winning or losing must be described using the mathematical language of Probability Theory**
- **Other situations in which randomness (stochasticity) plays a crucial role in their description and modeling includes financial markets, responses of rods and cones, and traffic flow (i.e. number of cars passing a given intersection during rush hour)**

Basic Probability 2: Random Experiments and Random Variables

Def: A random experiment is an experiment whose outcome cannot be predicted in advance with 100% certainty.

Def: An event A is one possible outcome of a random experiment

Def: The sample space S is the set of all possible outcomes

Def: A random variable X is a function that maps an outcome A of an experiment to a numerical value (note: in many cases the outcome is already a number)

Def: A probability function $\Pr(A)$ assigns a number $0 \leq \Pr(A) \leq 1$ to an event A that is interpreted as the probability (likelihood) of that event occurring.

Examples of Random Experiments:

(1) Flip a fair coin 8 times and record the sequence of heats and tails (HHTHTTTH)

(2) Role two dice one time and record the total (i.e. 7)

(3) Record the number of photons emitted by a 100 msec flash of green light (i.e. 90)

(4) Measure the number of photons absorbed by a single rod cell in response to flash of light with wavelength 505 nm (i.e. 0, 1, 2)

(5) Present a flash of light of a fixed intensity to a human subject's eye and ask whether he/she saw it ("yes") or not ("no") (this is our experiment!)

Basic Probability 3: Coin Flipping Example (Bernoulli Trial)

Fair Coin

Random experiment: Flip a fair coin one time

Possible events: $A_H = \{\text{Heads}\}$ or $A_T = \{\text{Tails}\}$

Random Variable: $X(A_H) = 1,\ X(A_T) = 0$ (assign 1 to heads and 0 to tails)

Probabilities of Events: $P(A_H) \equiv p = \frac{1}{2}\ ,\ P(A_T) \equiv 1 - p = \frac{1}{2}$

Note: $P(A_H) + P(A_T) = 1$ since there are only these two possible outcomes

Unfair Coin

Random experiment: Flip an unfair coin one time

Possible events: $A_H = \{\text{Heads}\}$ or $A_T = \{\text{Tails}\}$

Random Variable: $X(A_H) = 1,\ X(A_T) = 0$ (assign 1 to heads and 0 to tails)

Probabilities of Events: $P(A_H) \equiv p\ ,\ P(A_T) \equiv 1 - p$

Call the event A_H a "success" and the event A_T a "failure"

This binary-type outcome (success or failure) is called a Bernoulli Trial

The discrete random variable $X \in \{0,1\}$ is called a Bernoulli Random Variable

Calculating Basic Probabilities

$A =$ some event in a random experiment (i.e. throw 2 heads in 3 coin flips)

$N =$ total number of possible outcomes

<u>Def</u>: $N_A =$ total number of ways that the event A can occur

Assuming all outcomes are *equally likely*

$$\Pr\{A\} = \frac{N_A}{N}$$

Class Exercise

<u>Experiment</u>: Roll a fair six-sided die one time

<u>Random Variable</u>: $X =$ number shown on die after the roll

<u>Sample Space</u>: $S = \{1, 2, 3, 4, 5, 6\}$

<u>Event</u>: $A = \{X \le 2\}$

All possible rolls are equally likely.

Calculate the probability $\Pr\{A\}$

<u>Answer</u>: $\Pr(A) = \frac{1}{3}$

Sequence of Bernoulli Trials (coin flipping)

Random experiment: Flip a loaded coin 3 times

Possible events (some): $A_{HHT} = \{\text{HHT}\}, A_{TTT} = \{\text{TTT}\}, A_{2H} = \{\text{2 heads}\}, \ldots$

Probabilities of Events: $\Pr(A_H) \equiv p\ ,\ \Pr(A_T) \equiv 1-p$

Question: What is $\Pr(A_{2H}) = \Pr\{\text{2 heads}\}$?

There are 8 possible outcomes of the 3 flips:

$S = \{HHH, HHT, HTH, THH, HTT, THT, TTH, TTT\}$

$$\Pr(A_{2H}) = \Pr\{HHT\} + \Pr\{HTH\} + \Pr\{THH\}$$
$$= \left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right) + \left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right) + \left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right) = \frac{3}{8}$$

Note: Can get 2 heads in 3 trials in a total of $N_A = 3$ possible ways.

There are $N = 8$ total possible outcomes.

Each outcome is equally likely.

Thus $$\boxed{\Pr(A_{2H}) = \frac{N_A}{N} = \frac{3}{8}}$$

Probability Mass Functions (PMF)

Def: The probability mass function (pmf) for a discrete random variable X is a function $f(x)$ with the following properties for any event A in the sample space S:

(1) $f(x) > 0$ for all $x \in S$ (sample space)

(2) $\sum_{x \in S} f(x) = 1$ (sum of probabilities of all possible events is 1)

(3) $\Pr\{X \in A\} = \sum_{x \in A} f(x)$

If the event A corresponds to just one value of the random variable X (i.e. $X = 1$ if you flip heads) then we have the more direct interpretation

$$\boxed{\Pr\{X = x\} = f(x)}$$

Example: (single coin toss of a loaded coin)

$\mathrm{X} \in \{0,1\}$ where $X(\text{Tails}) = 0$ and $X(\text{Heads}) = 1$

PMF: $$f(x) = \begin{cases} 1-p, & x = 0 \text{ (tails)} \\ p, & x = 1 \text{ (heads)} \end{cases}$$

Cumulative Distribution Functions (CDF)

<u>Def</u>: The cumulative distribution function (CDF) for a discrete random variable X is a function $F(x)$ satisfying

$$F(x) = \Pr\{X \le x\} = \sum_{x_k \le x} f(x_k)$$

In words, the CDF $F(x)$ is the probabilty that the random variable X is less than or equal to a given value x.

The CDF for the Poisson probability mass function $f(x|\lambda)$ will play a role later in our analysis of the data from our visual psychophysics experiment.

<u>Comment</u>: the notation $f(x|\lambda)$ means the PMF "given the value of the parameter λ".

It is called a ***conditional probability***: $f(x|\lambda)$ is the probability of the event x occurring conditioned on λ having some known value (here the rate parameter of the Poisson process).

Binomial Distribution (PMF)

- **Suppose we conduct a sequence of n Bernoulli trials (i.e. coin flips)**
- **Each of the N trials is a "success" with probability p or a "failure" with probability 1-p**
- **The associated PMF for the number of successes X in n trials is called a Binomial Distribution (or a Binomial Probability Mass Function)**

<u>Def</u>: The **binomial distribution** is defined by the probabilty mass function

$$\Pr\{X = x\} = f(x|n,p) \equiv b(n,p) = \binom{n}{x} p^x (1-p)^{n-x}$$

$n =$ total number of Bernoulli trials (binary)

$p =$ probability of "success" (i.e. Heads) on any single Bernoulli trial

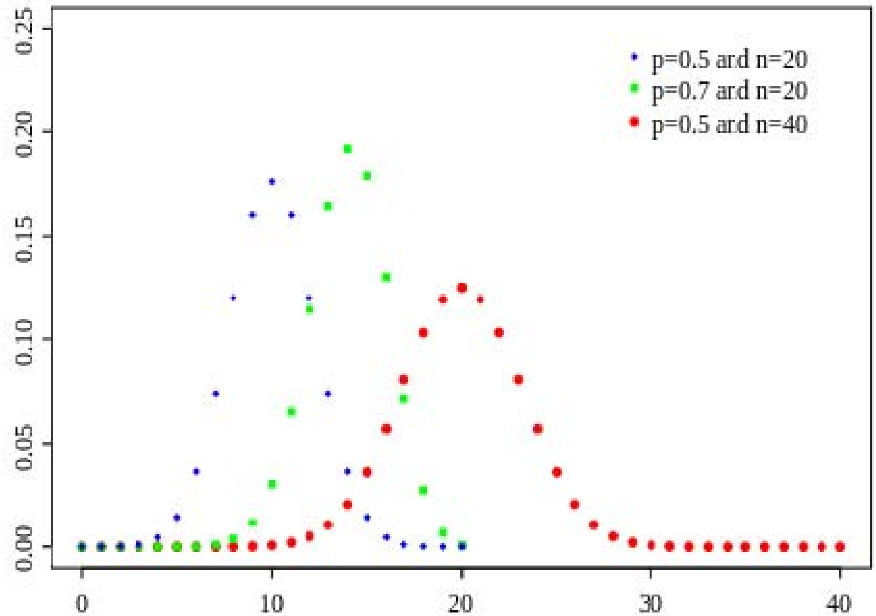
$$\binom{n}{x} = \frac{n!}{x!(n-x)!}$$ (number of ways for x "successes" in n trials)

$$X \in \{0,1,2,\ldots,n\}$$ (sample space)

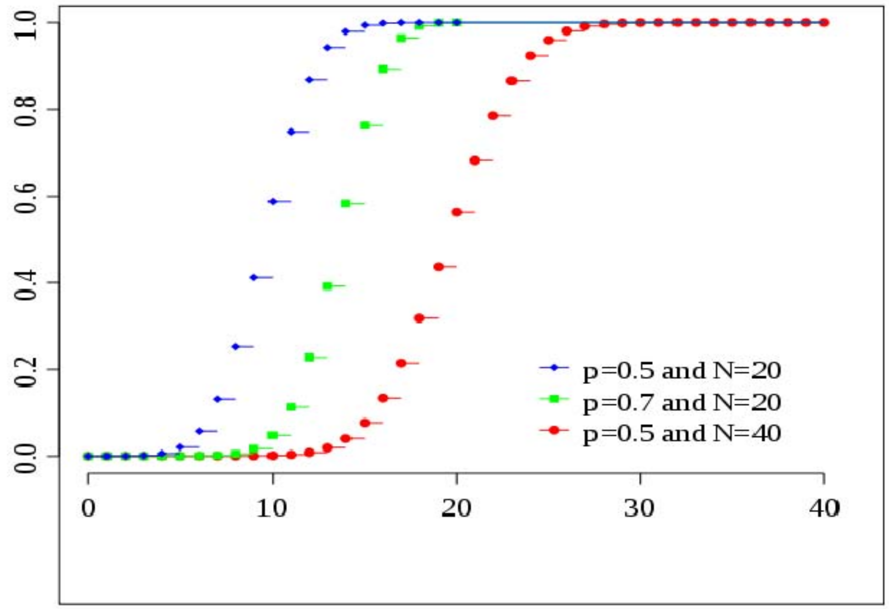
<u>Binomial CDF</u>: $F(x) = \Pr\{X \le x\} = \sum_{k=0}^{x} f(k|n,p) = \sum_{k=0}^{x} \binom{n}{k} p^k (1-p)^{n-k}$

Plots of the Binomial PMF and CDF

BINOMIAL PMF
b(n,p)



BINOMIAL CDF



Class Exercise: Using Binomial Distribution

Experiment: Suppose any lottery ticket purchased has a 20% chance of being a winning ticket (any amount of money). Suppose you purchase 8 tickets, and let X be a random variable indicating how many winning tickets you obtained.

Questions:

(1) What is $\Pr\{X = 2\}$? (2 winning tickets)

(2) What is $\Pr\{X \geq 1\}$? (at least 1 winning ticket)

Use MATLAB to compute these probabilities: **pdf('bino', x, n, p)**

(answers on next slide)

Note: $\Pr\{X \geq 1\} = \Pr\{X = 1\} + \Pr\{X = 2\} + \ldots + \Pr\{X = 8\} = 1 - \Pr\{X = 0\}$

Answer to Class Exercise: Using Binomial Distribution

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> % Class exercise with binomial distribution
>> n = 8;    % number of Bernoulli trials
>> p = .2;  % probability of "success" on each trial
>> % Answer to question 1  (probably of 2 successes, or winning tickets)
>> probTWOwinners = pdf('bino',2,n,p)

probTWOwinners =

    0.2936

>> % Answer to question 2  (probably of 1 or more winning tickets)
>> probOnewinnersOrMore = 1 - pdf('bino',0,n,p)

probOnewinnersOrMore =

    0.8322

fx >>
```

Preliminary Question Related to "Gombaud's Challenge"

Question: Suppose you roll a fair six-sided die many times in a row.

On average, how many times do you expect to have to roll the die before you have better than 50/50 odds of throwing at least one 6?

- Is it 3 times? 4 times?
- This is a famous problem in the history of gambling dating back to the 17th century

Answer: If you roll the die only 3 times, on less than half of your "experimental trials" can you expect to throw at least one 6.

If you roll the die 4 times, then slightly more than half the time you can expect to throw one or more sixes. (we will justify this mathematically)

Gombaud was a 17^{th} century gambler who exploited this knowledge, which was obtained empirically before the modern mathematical foundations of Probability were established.

Antoine Gombaud's Challenge to Pascal

- **Antoine Gombaud (aka the Chevalier de Mere) was a 17th century gambler who challenged the laws of probability as they were known at the time**
- **He directed his challenge to Blaise Pascal in the form of a dice problem**

Antoine Gombaud

Blaise Pascal

Experiment: Roll a die 4 times and count the number of times a 6 is rolled

Question: What is the probability that you roll at least one 6 in 4 tries?

Answer: Can use the **binomial distribution** $f\left(x \middle| n, p\right) = b(n, p)$ to answer this question.

$p = \frac{1}{6}$ (probability of a "success" that a 6 is rolled on any trial)

$n = 4$ (total number of trials)

Probability Mass Function: $\Pr\left\{X = x \text{ successes}\right\} = f\left(x \middle| n, p\right) = \binom{n}{x} p^x (1-p)^x = \binom{4}{x}\left(\frac{1}{6}\right)^x \left(\frac{5}{6}\right)^{4-x}$

where $x \in \{0, 1, 2, 3, 4\}$

- **Gombaud realized, through experience playing this dice game, that it was advantageous to bet every time that he would throw a 6 at least once in 4 tries**
- **Pascal used the Laws of Probabilty to prove that this indeed a good betting strategy**

Antoine Gombaud's Challenge to Pascal (solution)

Event: $A = \{X \geq 1\}$ (1 or more 6's rolled in 4 tries; $n = 4,\, p = \frac{1}{6}$)

$$\Pr\{X = x\} = \binom{4}{x}\left(\frac{1}{6}\right)^{x}\left(\frac{5}{6}\right)^{4-x}$$ (binomial distribution)

$$\Pr\{A\} = \Pr\{X = 1\} + \Pr\{X = 2\} + \Pr\{X = 3\} + \Pr\{X = 4\}$$

$$= 1 - \Pr\{X = 0\} = 1 - \binom{4}{0}\left(\frac{1}{6}\right)^{0}\left(\frac{5}{6}\right)^{4-0} = \left(\frac{4!}{4!0!}\right)(1)\left(\frac{5}{6}\right)^{4} \Rightarrow$$

$\Pr\{A\} = 0.5177$

Conclusion: On average, more than half the time you will roll at least one 6 in 4 throws. Therefore, if you bet even money it is to your advantage to bet that the event A will occur.

Class Project: Dice Rolling and Gombaud's Challenge

(1) Play Gombaud's game and roll the dice for at least 20 trials. Record the number of times you play the game (call it N) and the number of times you roll at least one 6 in 4 throws.

(2) Using your experimental results, calculate $\Pr\{A\} = \Pr\{1 \text{ or more 6's in 4 throws}\}$. How close is your empirical estimate to the true probability $\Pr\{A\} = 0.5177$?

(3) Suppose Gombaud's die is unfair and the probability of throwing a 6 is $p = \frac{1}{5}$. What then is $\Pr(A)$?

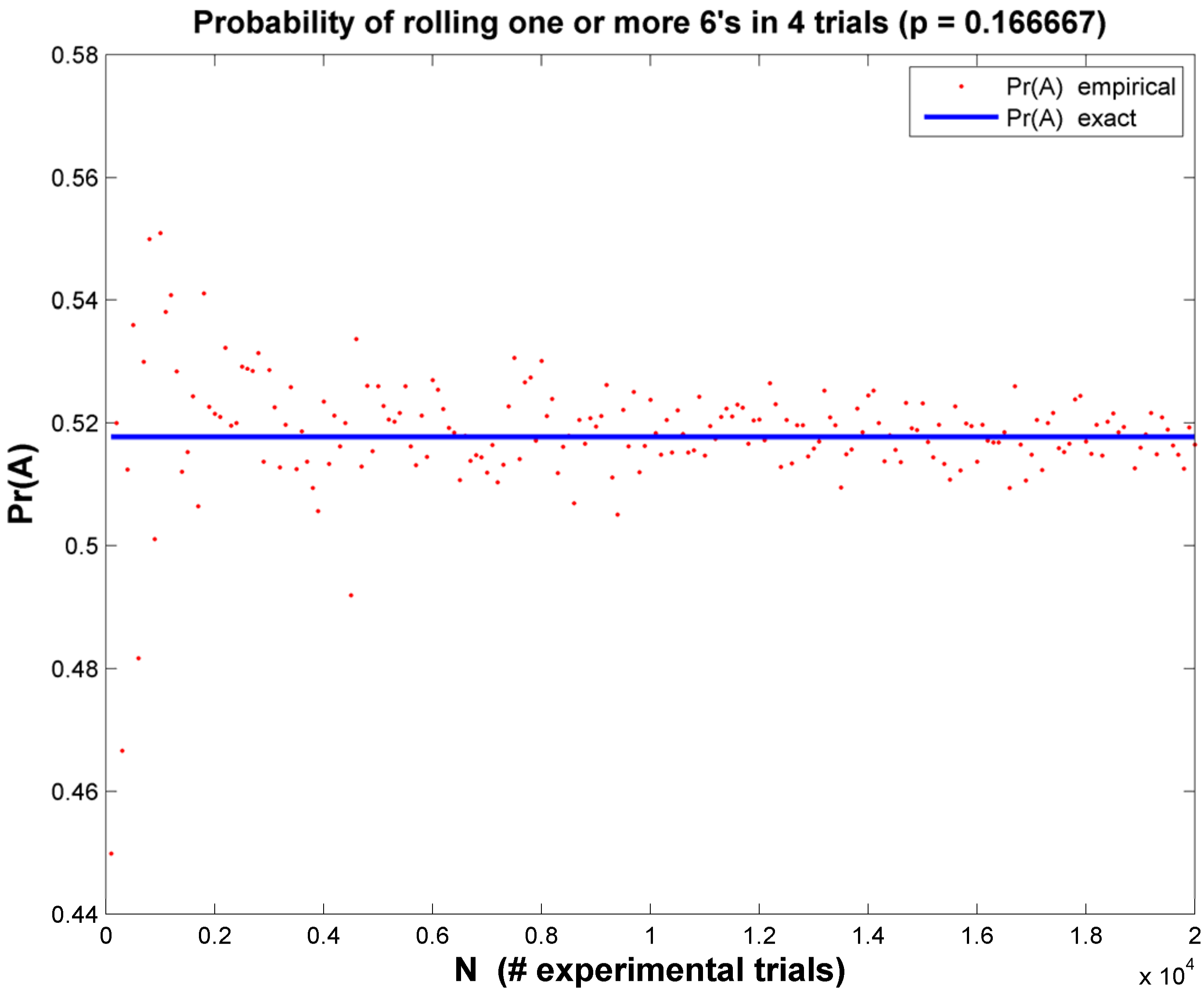
Modify the provided Matlab code **Gombaud_Problem_Exact_Result.m**

(4) Use Matlab to simulate this experiment $N = 100, 1000, 10000$ times. Show that the empirical estimate approaches the theoretical value $\Pr\{A\} = 0.5177$ as N gets larger.

Modify the provided Matlab code **Gombaud_Problem_Simulation.m**

(5) Try to modify **Gombaud_Problem_Simulation.m** and loop over many more N values and generate a plot of $\Pr(A)$ on the y-axis and N on the horizontal axis (see figure on next slide).

Simulation of Gombaud's Challenge
(class should try to recreate something like this)



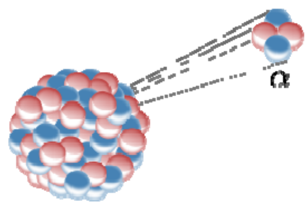
Poisson Distribution

Poisson PMF (distribution)

- A Poisson random variable corresponds to a Binomial Random Variable in the limit of a low probability of success: $\lim_{p \to 0}$ where $p =$ probability of success
- In the special limits $\lim_{p \to 0}$ (low success probability) and $\lim_{n \to \infty}$ (infinite # trials) it can be shown that the Binomial PMF $f\left(x \middle| n, p\right) = b(n, p)$ is well-approximated by

$$f\left(x \middle| \lambda\right) = \Pr\left\{X = x\right\} = \frac{\lambda^x}{x!} e^{-\lambda} \qquad \text{(Poisson PMF)}$$

$$\lambda = np \qquad \text{(mean \# events, or successes, per trial)}$$



Example (Rutherford, Geiger, and Bateman; 1910)

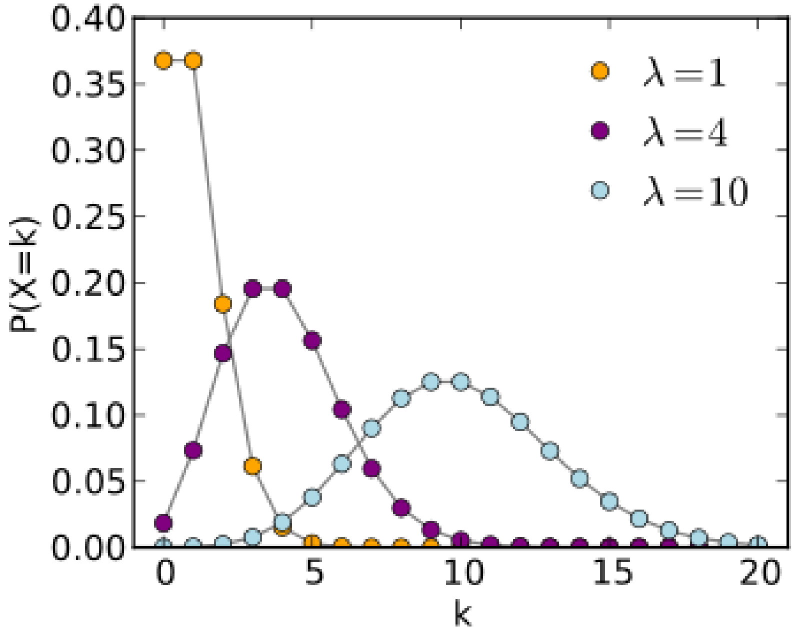
Suppose a sample of polonium is radiating α particles at a rate of $\lambda = 0.5$ (α particles/sec)

- The probability that the sample radiates $X = 2$ particles in one unit of time (1 second) is

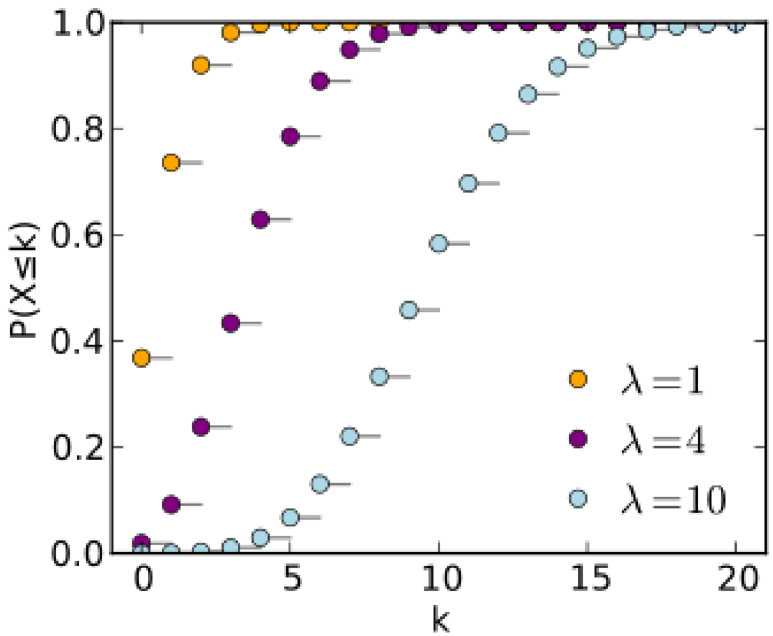
$$\Pr\left\{X = 2\right\} = \frac{\left(\frac{1}{2}\right)^2}{2!} e^{-\frac{1}{2}} = 0.2061$$

Plots of Poisson PMF and CDF

Poisson PMF



Poisson CDF



Examples where the Poisson probability model applies

(1) **The number of photons emitted by a flash of light of a (nominally) fixed intensity**

Poisson rate parameter: λ = (photons emitted)/flash

(2) **The number of photons absorbed by a rod responding to a flash of light of a (nominally) fixed intensity**

Poisson rate parameter: λ = (photons absorbed)/flash

(3) **The number of typed errors on a single page of a document**

Poisson rate parameter: λ = typos/page

(4) **The number of visitors to a website per minute**

Poisson rate parameter: λ = visitors/day

(5) **The number of Prussian army soliders killed by "friendly" horse kicks in a month (L. Von Bortkiewicz; 1898)**

Poisson rate parameter: λ = soldiers/month

The Thing (1982)

Class Project: Show that Blair's estimation was reasonable!

By some miracle of fortune, the estimate of 75% probability of 1 or more infected camp members is actually consistent (roughly) with the situation as described up to that point in the film

<u>The Facts</u> (spanning a total of about 4.5 days)

There was a Norwegian camp in Antarctica and an American camp (Outpost 31)

1) **The Thing was dug out of an ice block by the Norwegians about 3 days prior to its arrival at Outpost 31**
2) **Blair knew of 1 Norwegian man assimilated and 1 Norwegian dog (2 infections)**
3) **2 dogs at Outpost 31 were assimilated within 1.5 days of the Thing's arrival (2 more)**
4) **There were 6 Norwegians unaccounted for (assume 2 more infections)**
5) **This means roughly 6 infections in about 4.5 days time**
6) **Outpost 31 has n = 12 camp members (including Blair himself)**

Class Project: Outpost 31 Infection Probability Problem

Questions

(1) Assuming the number of "Thing infections" per day is distributed as a **Poisson process**, use the movie's information (and the given reasonable assumptions!) to estimate the rate parameter λ (infections/day) of the associated Poisson distribution.

(2) With 12 total camp members at Outpost 31, use the Poisson assumption to estimate the probability $\Pr\{X \geq 1\}$, where X is the random variable (# infections in a random day)

(3) Assume that the time span of our "experiment" is one day. In other words, we ask what is the probability that one or more infections has occurred within the first day that "The Thing" was at Outpost 31.

(4) The answer to part (2) should be pretty close to Blair's calculation from the movie. Use MATLAB and your brain to find the *exact value of the rate parameter* λ such that

$$\Pr\{X \geq 1\} = 0.75$$

Thing Project: Solution

- The inferred Poisson rate parameter from the movie is $\boxed{\lambda = \frac{6}{4.5} = \frac{4}{3}}$ infections/day

- Using the Poisson model distribution $\Pr\left\{x \text{ infections}|\lambda\right\} = f\left(x\,|\,\lambda\right) = \frac{e^{-\lambda}\lambda^{x}}{x!}$

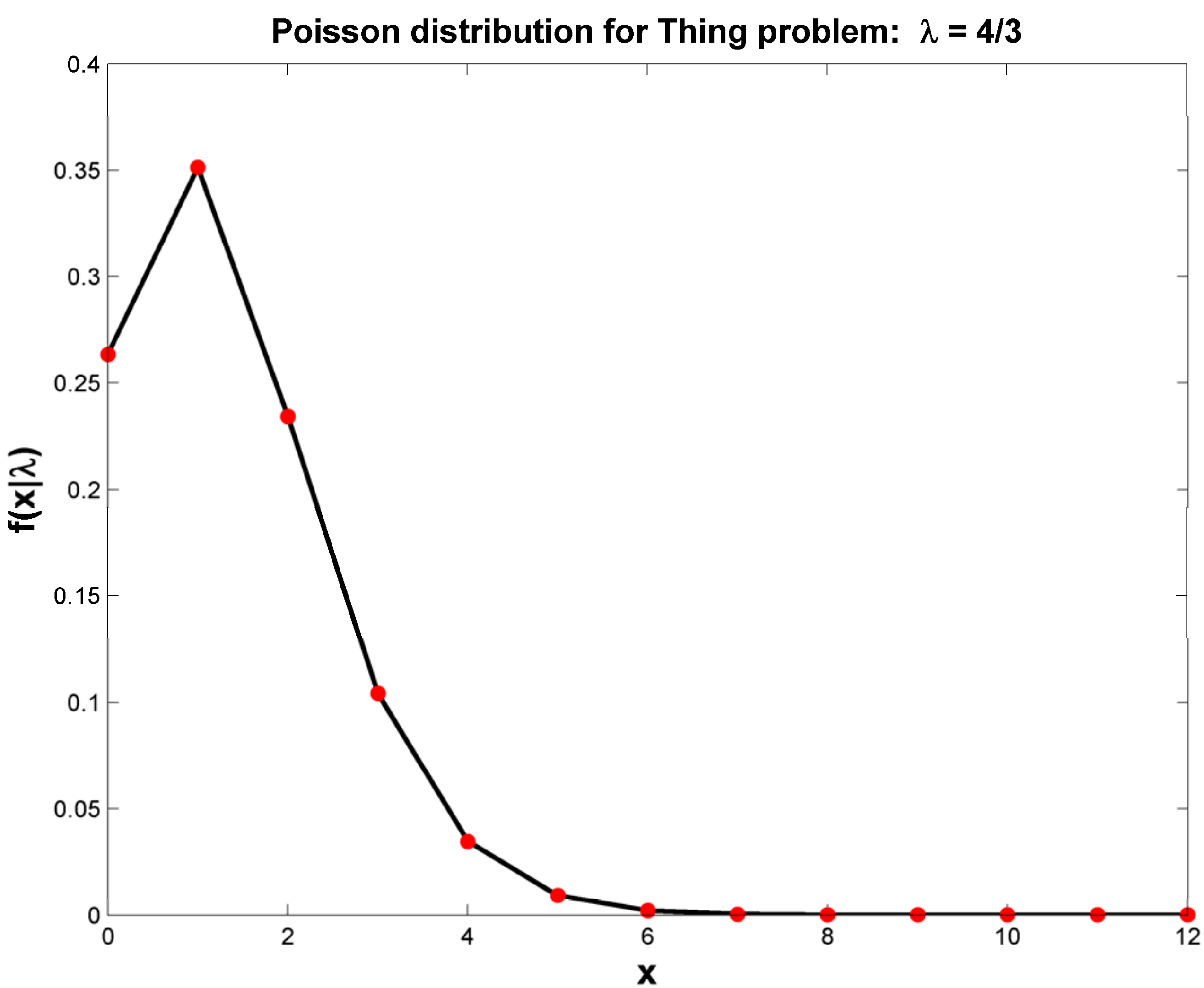
we have the following probability for the number of infections being 1 or greater:

$$\begin{aligned}
\Pr\left\{x \geq 1\middle|\lambda\right\} &= \Pr\left\{x=1\right\} + \Pr\left\{x=2\right\} + \ldots + \Pr\left\{x=12\right\} \\
&= f\left(1\,|\,\lambda\right) + f\left(2\,|\,\lambda\right) + \ldots f\left(12\,|\,\lambda\right) \\
&= 1 - \Pr\left\{x=0\right\} \\
&= 1 - f\left(1\,|\,\lambda\right) = 1 - \frac{e^{-\lambda}\lambda^{0}}{0!} = 1 - e^{-\frac{4}{3}} = \boxed{0.7364}
\end{aligned}$$

<u>Comment</u>: We must technically assume there are an infinite number of camp members (rather than 12) for this model to be valid (since the Poisson random variable has as domain all non-negative integers)

Thing Project: Solution

Probability is miniscule out in the tails (x=12) for this problem, so our assumptions are okay



Thing Project: Solution

Matlab code for plotting the Poisson distribution

```
%% Plot Poisson PMF for Blair's Thing infection calculation.  This plot demonstrates that
%  the assumption of an infinite # of camp members (rather than 12) is not grossly in error
%  since the pmf values are extremely small beyond 7 or 8
rate = 4/3;     % inferred value from the film
x = 0:12;       % Poisson RV domain
f = pdf('Poiss',x,rate);   % Poisson pmf
figure
  plot(x,f,'k-','linewidth',2); hold on;
  plot(x,f,'r.','markersize',20);
  xlabel('x','fontweight','bold','fontsize',15);
  ylabel('f(x|\lambda)','fontweight','bold','fontsize',15);
  title('Poisson distribution for Thing problem:  \lambda = 4/3','fontweight','bold','fontsize',14);
```

This file is provided to you in your folder: "plot_Poisson_PMF_Blair_Thing_problem.m"

Thing Project: Solution

- To get the rate parameter λ that would give Blair's *exact probability* of 0.75 we solve

$$\Pr\{x \geq 1 | \lambda\} = 1 - e^{-\lambda} = \frac{3}{4} \quad \rightarrow$$

$$e^{-\lambda} = \frac{1}{4} \qquad \text{(take natural logarithm of each side)}$$

$$\ln\left(e^{-\lambda}\right) = \ln\left(\frac{1}{4}\right) = -\ln 4$$

$$-\lambda = -\ln 4 \quad \rightarrow$$

$$\boxed{\lambda = \ln 4 = 1.386}$$

- Compare this exact theoretical result with the approximation $\lambda \approx 1.33$ derived from the events of the film. They're close!

Hecht, Schlaer, Pirenne (1942) Experiment (modernized)

1. **Dark adaptation**, so that the eye is maximally sensitive to small amounts of light

2. Subject will be presented with a **series of light flashes** of wavelength 505 nm. There will be 7 flash intensities – each repeated 10 or 20 times - ranging from nearly invisible to easily visible.

3. Prior to each light flash the subject will fixate his or her right eye on a red cross located approximately 20cm (20 visual degrees) to the right of the LED light source in the visual field. This ensures that the light hits the part of the eye with the **maximal rod density**.

4. An audio cue (a beep) will alert the subject that a flash is about to be presented.

5. A text message appears after the light flash asking the subject to respond **YES** (saw the flash) or **NO** (didn't see the flash).

6. The "**Probability of Seeing**" the flash at each light intensity is given by the fraction of trials that the subject responded "YES" (saw the flash).

HSP Experiment Demo

Run **Demo Program** (minus the light flashes) that demonstrates what the subject will see on the monitor:

- Fixation Cross
- Audio Cues
- Trial Query (answer "YES" or "NO")
- 70 trials takes about 7 minutes to complete (10 repeated trials per stimulus flash intensity)
- 140 trials takes about 15 minutes to complete (20 repeated trials per stimulus flash intensity)

<u>YES Response</u>: Hit "y" or "→" keys
<u>NO Response</u>: Hit "n" or "←" keys

Mathematical Analysis of HSP Experiment

Poisson distributed photon arrivals: Assume that the number of photons x emitted for a fixed flash intensity I follows a Poisson distribution:

$$\Pr\left\{x \text{ photons given mean rate } \lambda\right\} = P\left(x \mid \lambda\right) = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

$$\lambda = \text{mean \# photons per flash}$$

Key Assumption: $\lambda = \alpha I \quad \rightarrow \quad P\left(x \mid I\right) = \dfrac{e^{-\alpha I}\left(\alpha I\right)^{x}}{x!}$

- α is a scale factor that accounts for subject variability (age, eye composition, etc). We will allow α to vary when fitting the model.
- I is the intensity of the flash stimulus (photons emitted per flash)

Cumulative Distribution Function: $\Pr\left\{x \le K \mid I\right\} \equiv F\left(K \mid I\right) = \sum_{x=0}^{K} P\left(x \mid I\right) = e^{-\alpha I}\sum_{x=0}^{K}\dfrac{\left(\alpha I\right)^{x}}{x!}$

- This is the probability that between 0 and K photons are emitted at stimulus intensity I

Mathematical Analysis of HSP Experiment

- We now hypothesize that there is a minimum number of photons K required at any intensity I in order for a subject to say "I saw it"
- For each stimulus intensity I we construct a "Probability of Seeing" function $P_{\text{SEE}}(I)$:

$$P_{\text{SEE}}(I) = \frac{\#\,\text{trials stimulus seen at intensity } I}{\#\ \text{total times stimulus } I \text{ shown}}$$

- If K or more photons are required at any stimulus I to see the flash, then our Poisson model can be expressed in terms of the Poisson cumulative distribution function (CDF) $F(K)$:

$$\begin{aligned} P_{\text{SEE}}(I) &= \Pr\{x \geq K \text{ given } I\} = \sum_{x=K}^{\infty} P(x \mid I) = e^{-\alpha I} \sum_{x=K}^{\infty} \frac{(\alpha I)^x}{x!} \\ &= 1 - \Pr\{x \leq K-1 \text{ given } I\} = 1 - F(K-1 \mid I) \end{aligned}$$

- This is the model we will fit to our experimental data.
- $P_{\text{SEE}}(I)$ will be estimated empirically from the experiment.
- The two parameters $\{K, \alpha\}$ will be "fit" using a model optimization algorithm in Matlab.
- The parameter K will be interpreted as the ***minimum number of photons required for vision***

Mathematical Analysis of HSP Experiment

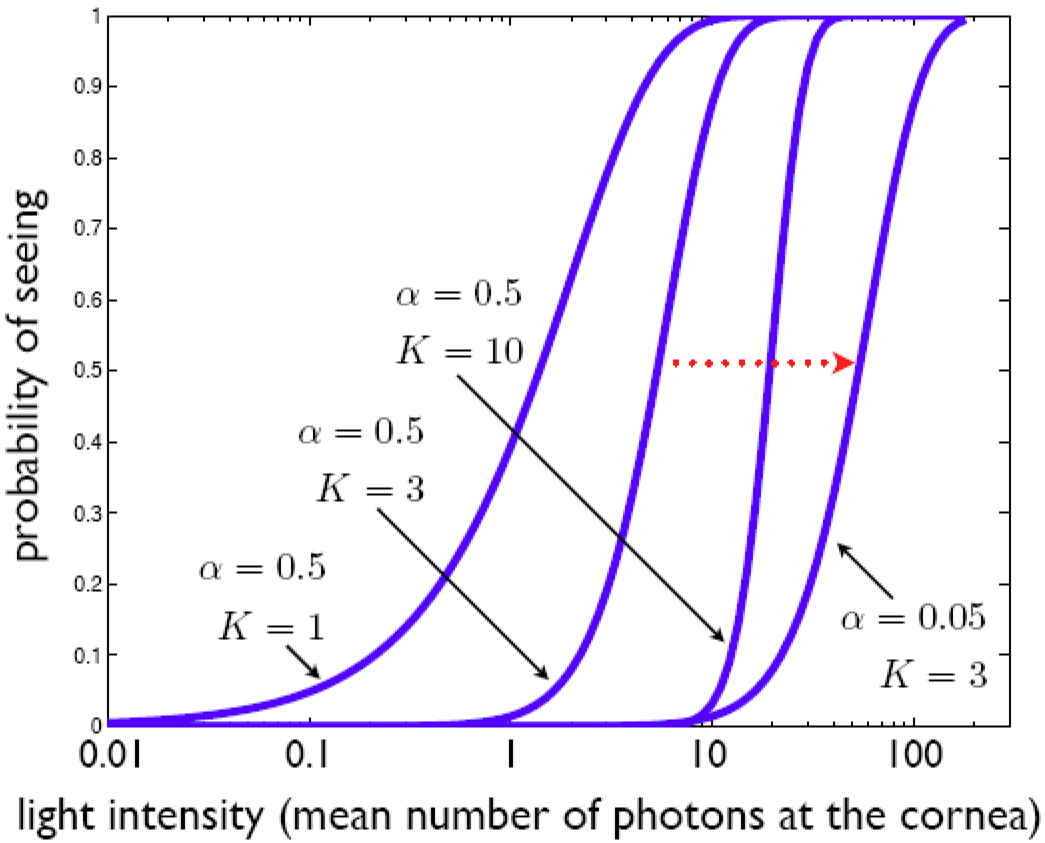
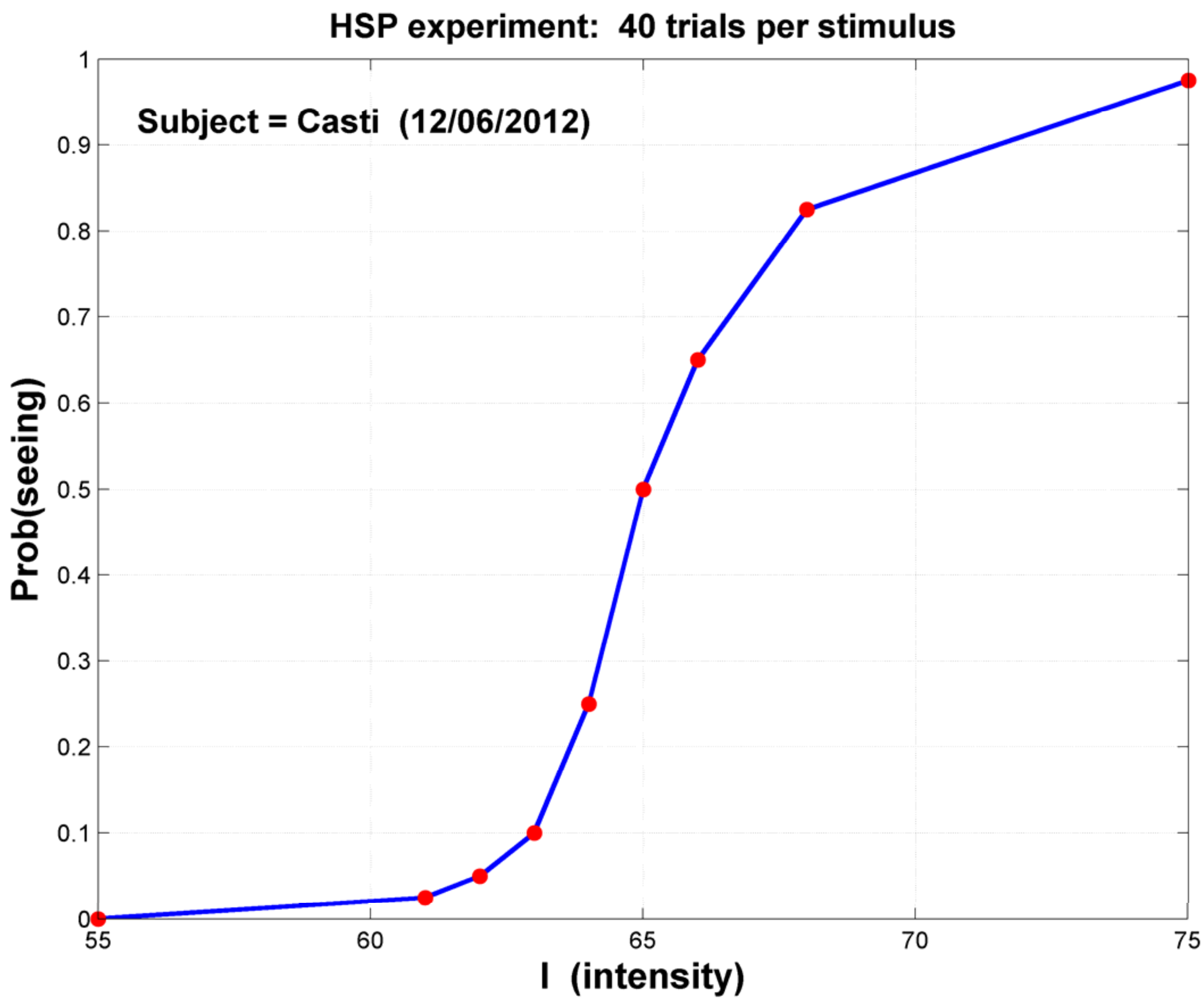


FIG. 1 Probability of seeing calculated from Eq. (2), where the intnesity I is measured as the mean number of photons incident on the cornea, so that α is dimensionless. Curves are shown for different values of the threshold photon count K and the scaling factor α. Note the distinct shapes for different K, but when we change α at fixed K we just translate the curve along the the log intensity axis, as shown by the red dashed arrow.

HSP Experimental Results (Subject: Casti)



Optimizing the Model Fit to the Data

Model: $P_{\mathrm{MOD}}\left(I\right)=e^{-\alpha I}\sum_{x=K}^{\infty}\frac{\left(\alpha I\right)^{x}}{x!}=1-e^{-\alpha I}\sum_{x=0}^{K-1}\frac{\left(\alpha I\right)^{x}}{x!}=1-F\left(K-1\,|\,I\right)$

Lab Data: $P_{\mathrm{EXP}}\left(I_j\right)$ where $I_j\in\{55,\ 61,\ 62,\ 63,\ 64,\ 65,\ 66,\ 68,\ 75\}$ (8 bit intensity scale)

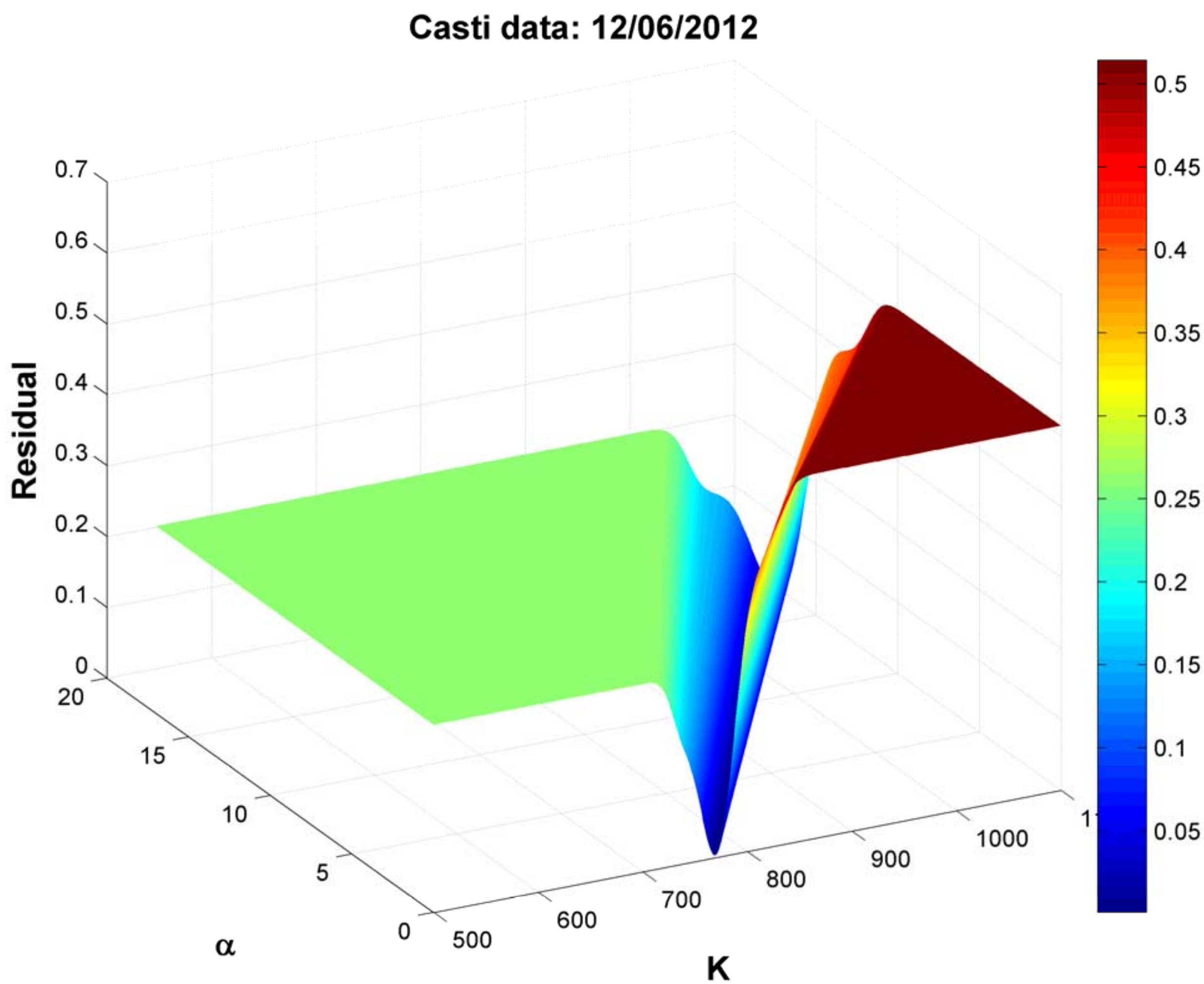
- I_j corresponds to the flash intensity for the j^{th} stimulus level used in the HSP experiment.
- With a photodetector one can determine the number of photons per flash for intensity I_j
- However, when fitting the model we don't really care what the units of I_j are (but these specific physical units - energy per flash or #photons per flash - would have to be cited to get your work published!)

Residual Function: $R\left(K,\alpha\right)=\frac{1}{N}\sum_{j=1}^{N}\left(P_{\mathrm{EXP}}\left(I_j\right)-P_{\mathrm{MOD}}\left(I_j\right)\right)^2$

- $N=9$ (number of flash intensities used)
- This particular residual function (mean-square error) measures the model mismatch with data
- Goal is to minimize $R\left(K,\alpha\right)$ by finding the optimal parameters $\left(K_{opt},\alpha_{opt}\right)$

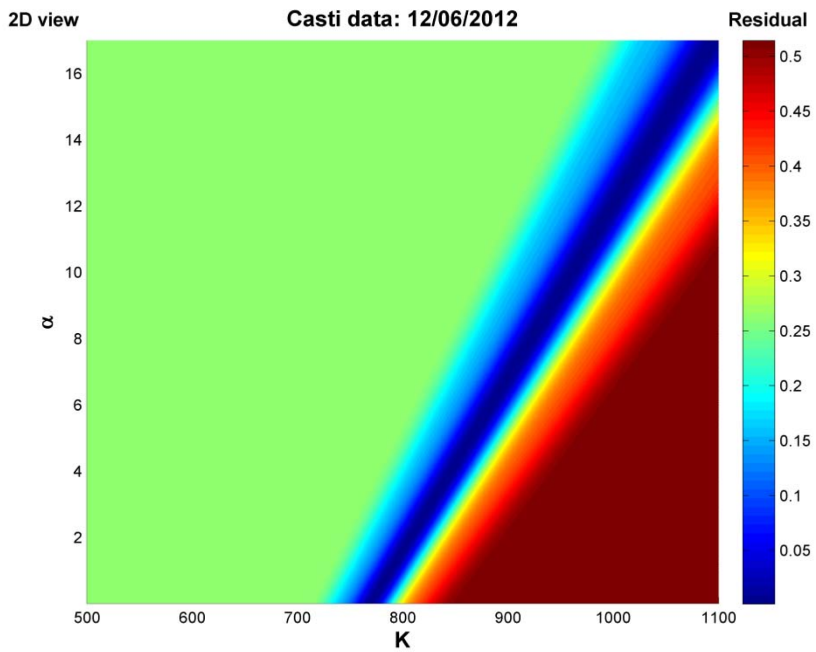
Error Surface Visualization (Casti data)

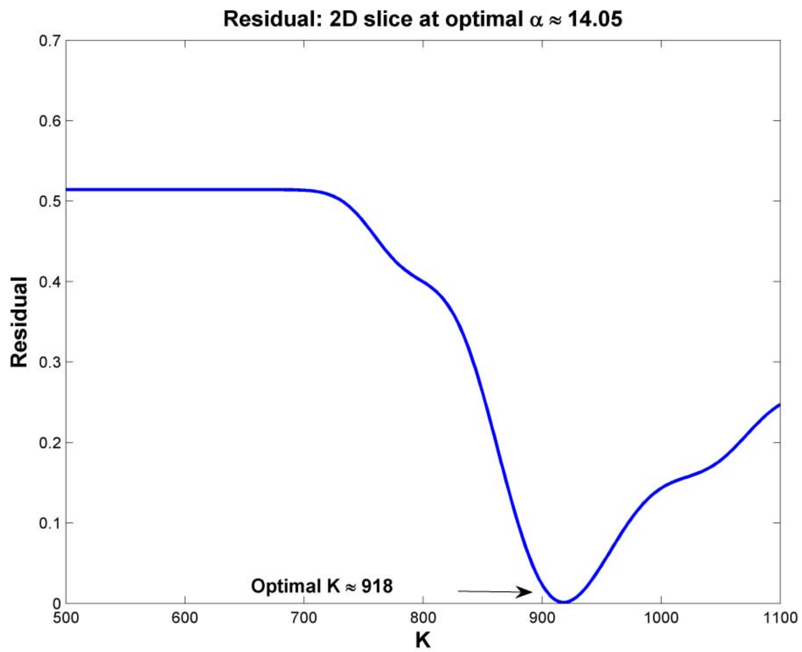
- **Brute force grid search: no optimization algorithm used**
- **This approach gives an initial idea of where the optimal parameters lie**



Error Surface Visualization (Casti data)

- Here are 1D and 2D visualizations of the residual surface R
- 1D slice corresponds to the optimal α value

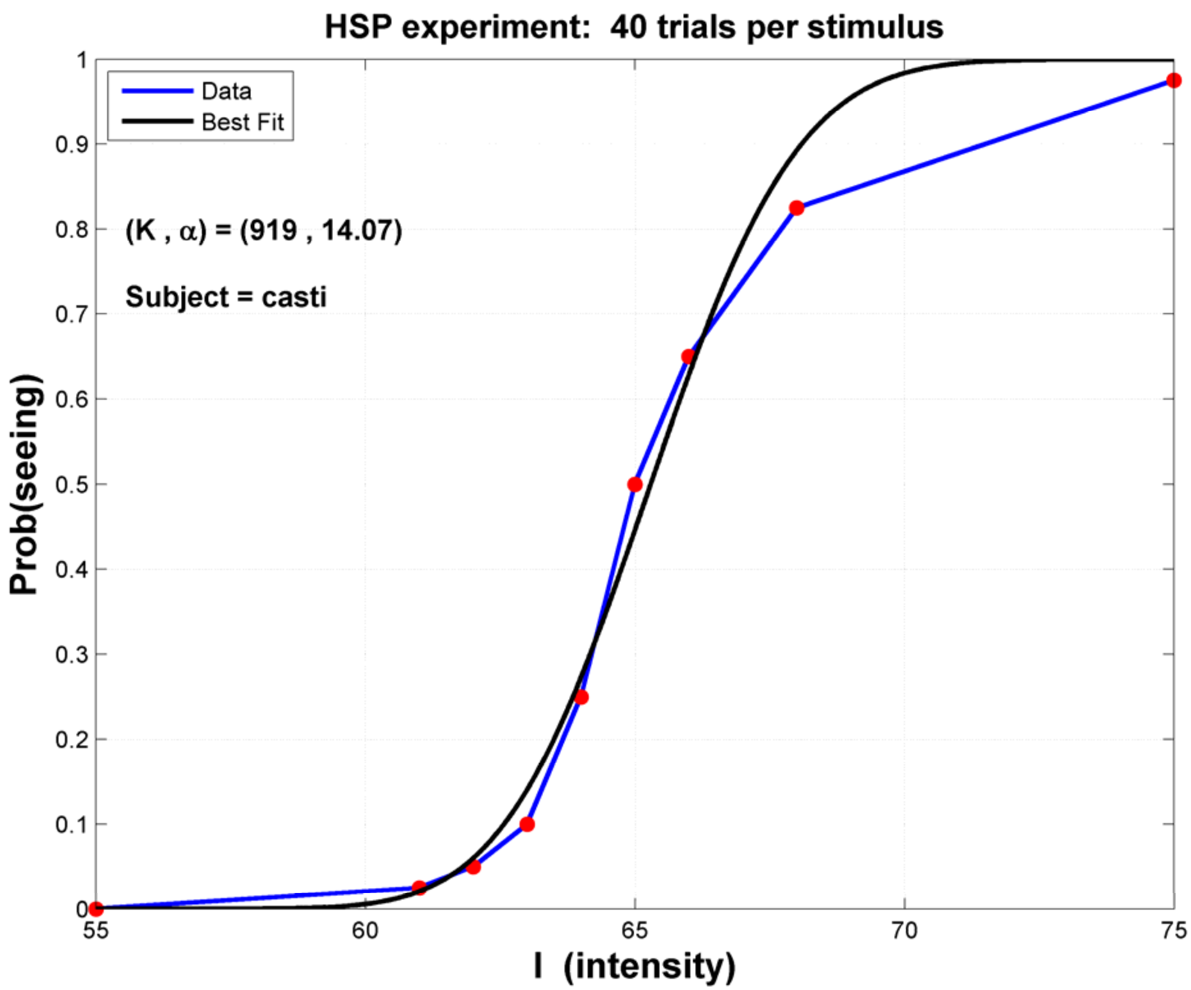




HSP Optimization Example: Casti data (12/06/2012)

Optimal Parameters: $(K, \alpha) = (919, 14.07)$

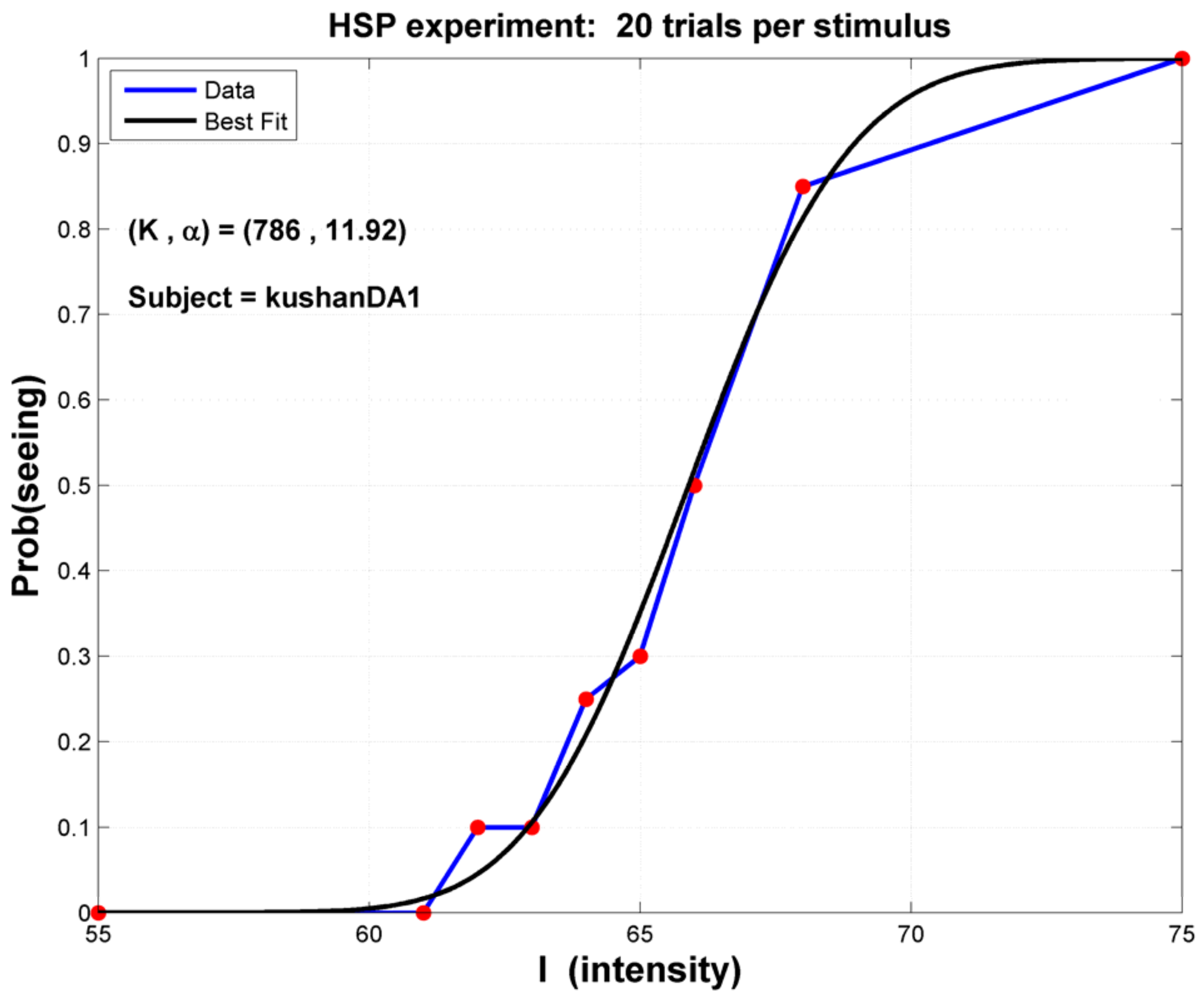
- $K = 919$ is way too high; should be $K \approx 6$
- Model optimized using Matlab's "fminsearch": **HSP_fitModel_fminSearch.m**



HSP Optimization Example: Costa data (12/07/2012)

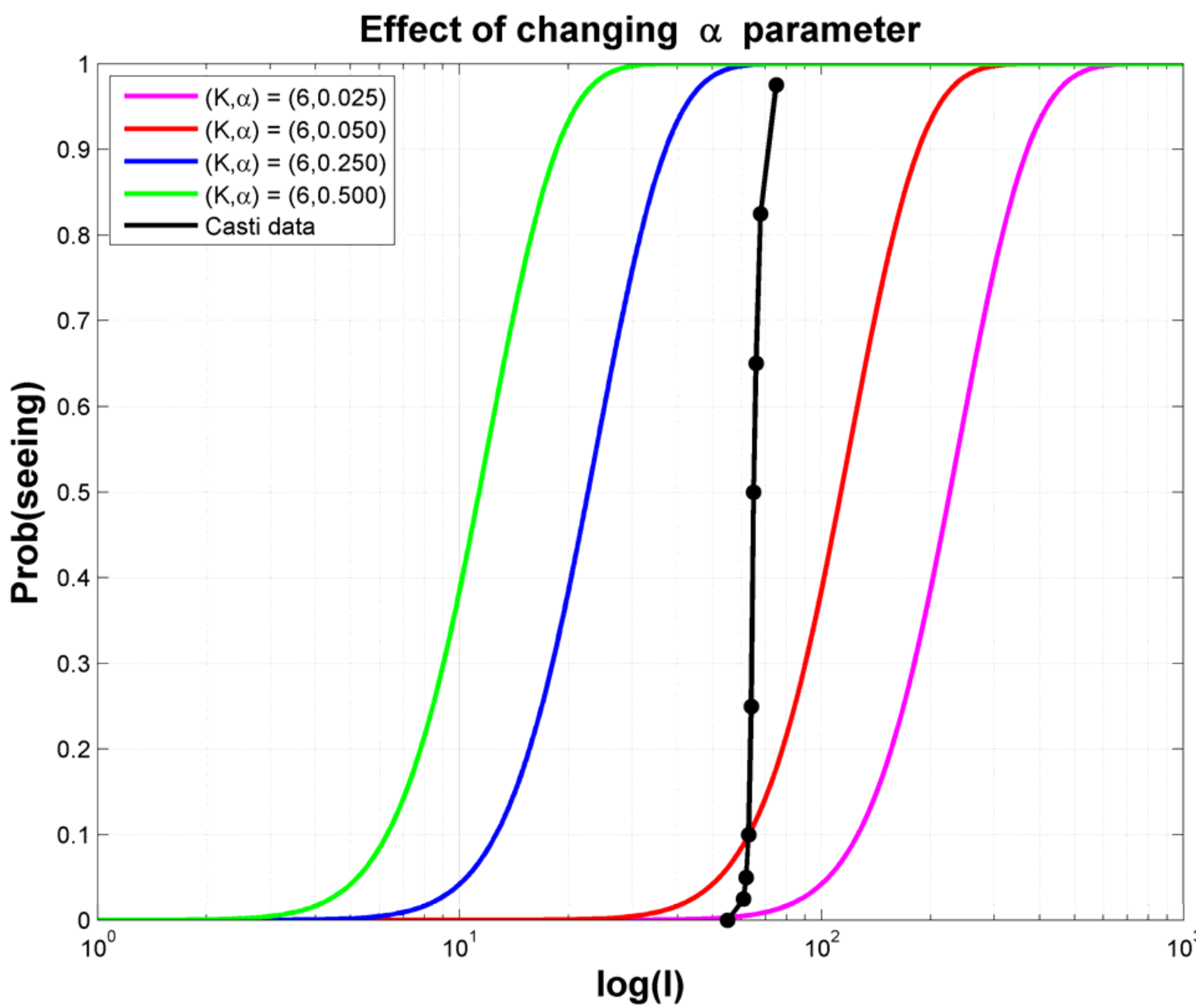
Optimal Parameters: $(K,\alpha)=(786,11.92)$

- $K=786$ is also way too high, but lower than Casti value
- This means that Costa had a lower visual threshold (possibly age related)



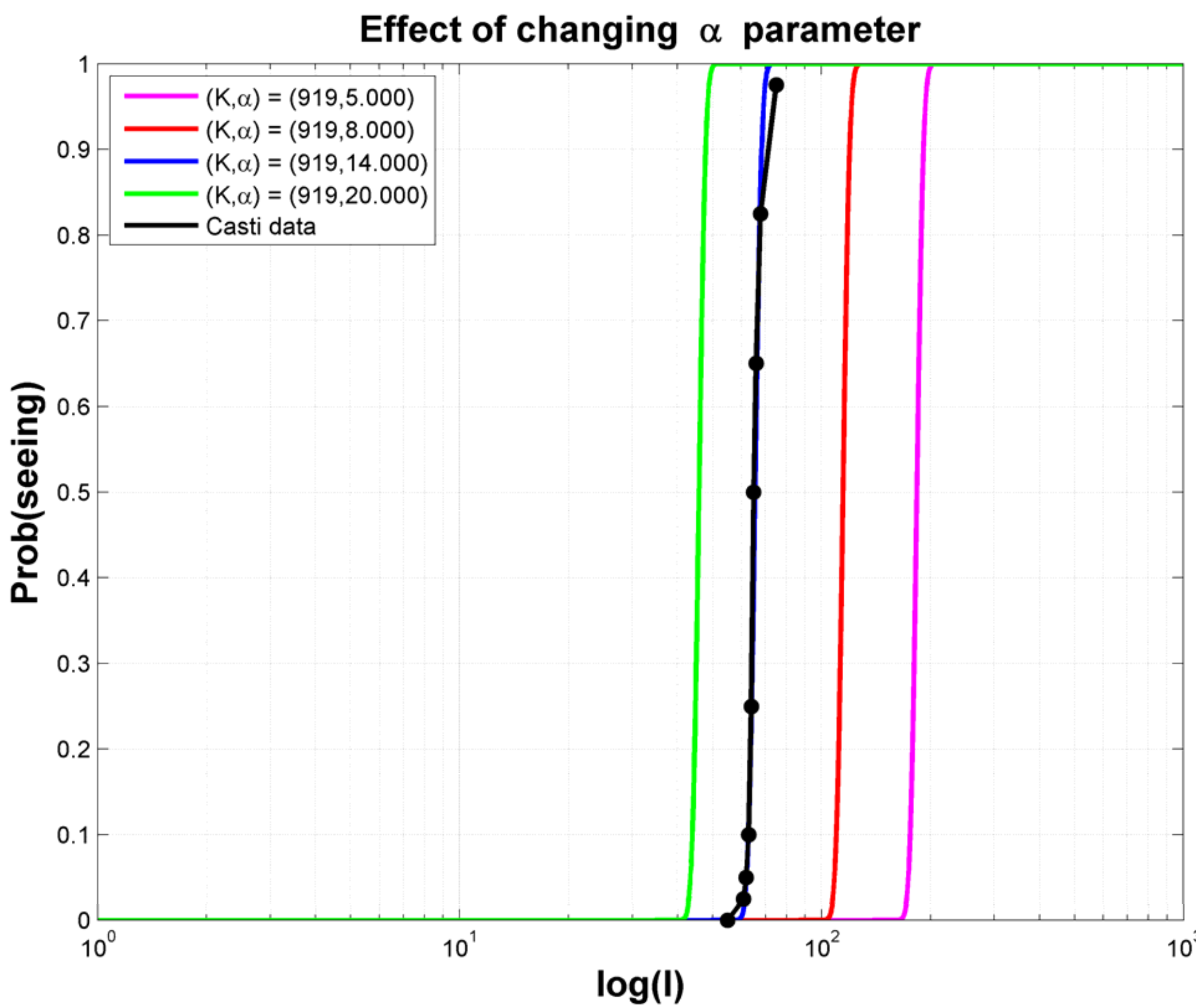
HSP Curves: Effect of Varying α Parameter (K=6)

Note that the "Casti data" (40 trials) is much steeper than the theoretical curves with K = 6



HSP Curves: Effect of Varying α Parameter (K=919)

K = 919 gives the better fit to the "Casti data" (40 trials)



Sources of “Error” in our HSP experiment

- Dark adaptation : Probably the most significant source of error. Visual sensitivity increases dramatically the longer you light adapt. (see subsequent figure)
- Ambient light sources: Computer monitor emits detectable light. Computer lights, mouse light, etc. This adversely affects optimal dark-adapted conditions.
- Non-uniform light source: Probably a great many of the incident photons are falling into regions that have a low density of rods and a high density of cones (which are unresponsive to this light frequency and the level of darkness) .
- Flash duration: LED flash duration (about 200 msec) is probably too long. Visual threshold increases with flash duration (see subsequent figure).
- Foveal Fixation: Eye drift and improper fixation. It’s better to use a bite bar to fix the subject’s head.

These sources of non-ideal experimental conditions should be cited in the “Discussion” portion of your write-up to explain the result of an “optimal K value” being too large.

Experimental Error: Dark Adaptation

- Visual sensitivity experiments suggest that we should **dark adapt for ~30 minutes**
- Due to time constraints we only dark adapted for 2-3 minutes

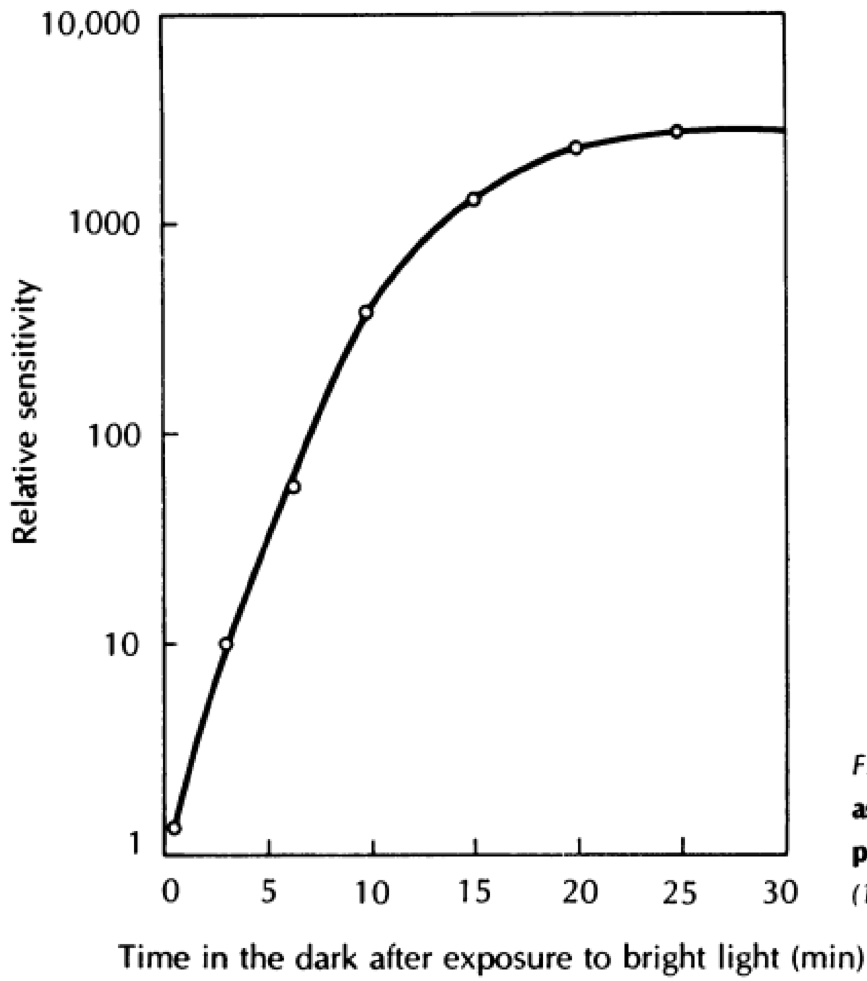


Fig. 2.1 **Change in human visual sensitivity as a function of time in the dark after exposure to a bright light.** [*After Kohlrausch (1931), curve for green light.*]

Experimental Error: Non-Uniform Light Source

Optimal light source should fall within a disc of about 10 minutes of arc on the retina. Our light is probably too diffuse and falls on low rod density regions (i.e. on the cones)

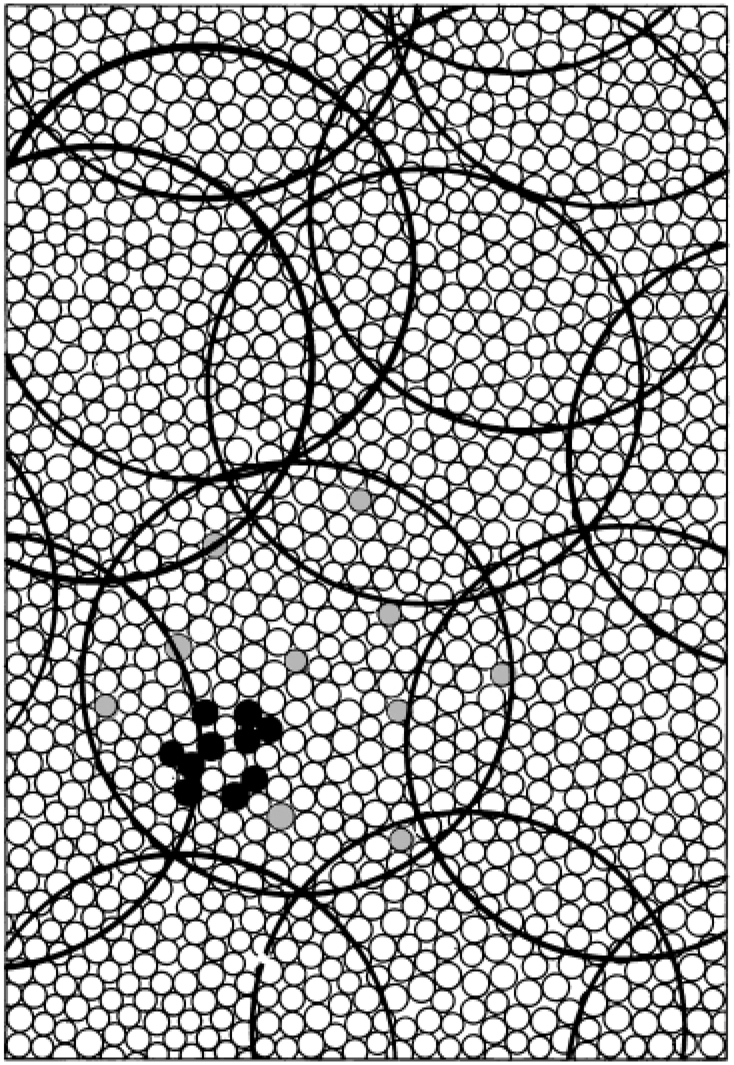


Fig. 2.8 **Semischematic representation of the summation areas of the dark-adapted human retina, 20° from the fixation area (the fovea). Each small circle represents the end view of a rod, and each large circle represents the area over which the excitations in all the rods it contains summate. The summation areas overlap, but the actual extent of overlapping in the eye is not known.**

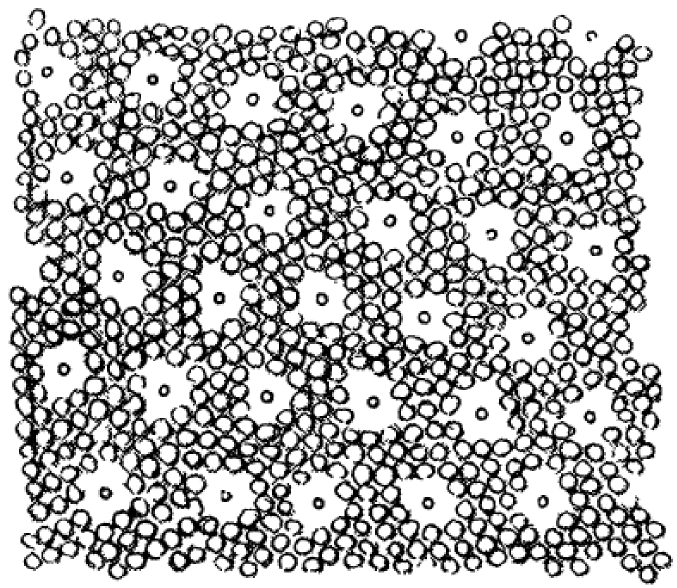


Fig. 2.13 **Spacing of the rods, seen end on, results in the loss of about half of the incident quanta. The small circles in this drawing are receptors of a different kind (cones), which are evidently inoperative at threshold intensities in the dark-adapted eye.** [*From Schultz (1866), periphery.*]

Experimental Error: Flash Duration

For flash durations greater than 100 msec more light is required to elicit a threshold visual response (i.e. to see the flash 60% of the time or greater)

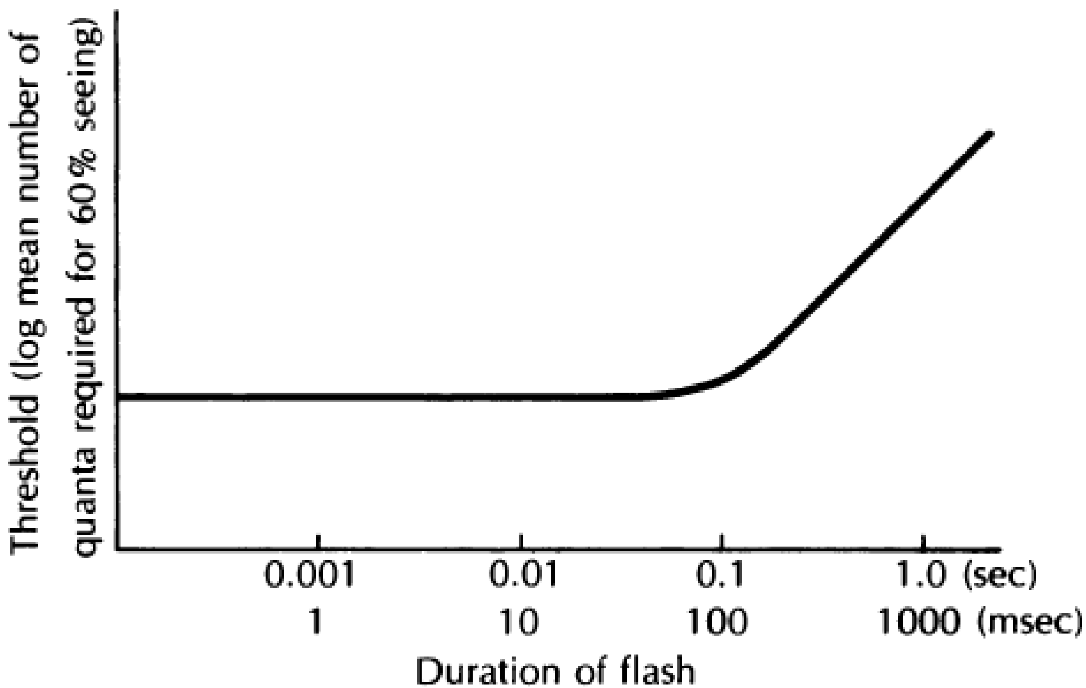


Fig 2.9 **Total light required for seeing a flash as a function of the duration of the flash.** [*From Graham and Margaria (1935), 2′ curve.*]

More Sophisticated Optimization Approach
Matlab's fminsearch (Simplex algorithm)

fminsearch

Find minimum of unconstrained multivariable function using derivative-free method

Syntax

```
x = fminsearch(fun,x0)
x = fminsearch(fun,x0,options)
[x,fval] = fminsearch(...)
[x,fval,exitflag] = fminsearch(...)
[x,fval,exitflag,output] = fminsearch(...)
```

Description

`fminsearch` finds the minimum of a scalar function of several variables, starting at an initial estimate. This is generally referred to as *unconstrained nonlinear optimization*.

`x = fminsearch(fun,x0)` starts at the point `x0` and returns a value `x` that is a local minimizer of the function described in `fun`. `x0` can be a scalar, vector, or matrix. `fun` is a function handle. See Function Handles in the MATLAB Programming documentation for more information.

Parameterizing Functions in the MATLAB Mathematics documentation explains how to pass additional parameters to your objective function `fun`. See also Example 2 and Example 3 below.

`x = fminsearch(fun,x0,options)` minimizes with the optimization parameters specified in the structure `options`. You can define these

Arguments

`fun` is the function to be minimized. It accepts an input `x` and returns a scalar `f`, the objective function evaluated at `x`. The function `fun` can be specified as a function handle for a function file

```
x = fminsearch(@myfun, x0)
```

where `myfun` is a function file such as

```
function f = myfun(x)
f = ...            % Compute function value at x
```

or as a function handle for an anonymous function, such as

```
x = fminsearch(@(x)sin(x^2), x0);
```

Other arguments are described in the syntax descriptions above.

Optimization Example: Matlab's fminsearch

$$f(x) = 3ax^4 - 4x^3$$ (a is constant, you choose it in Matlab code)

Minimum Value (Calculus): $$\frac{df}{dx} = 12ax^3 - 12x^2 = 12x^2(ax-1) = 0 \quad \Rightarrow$$

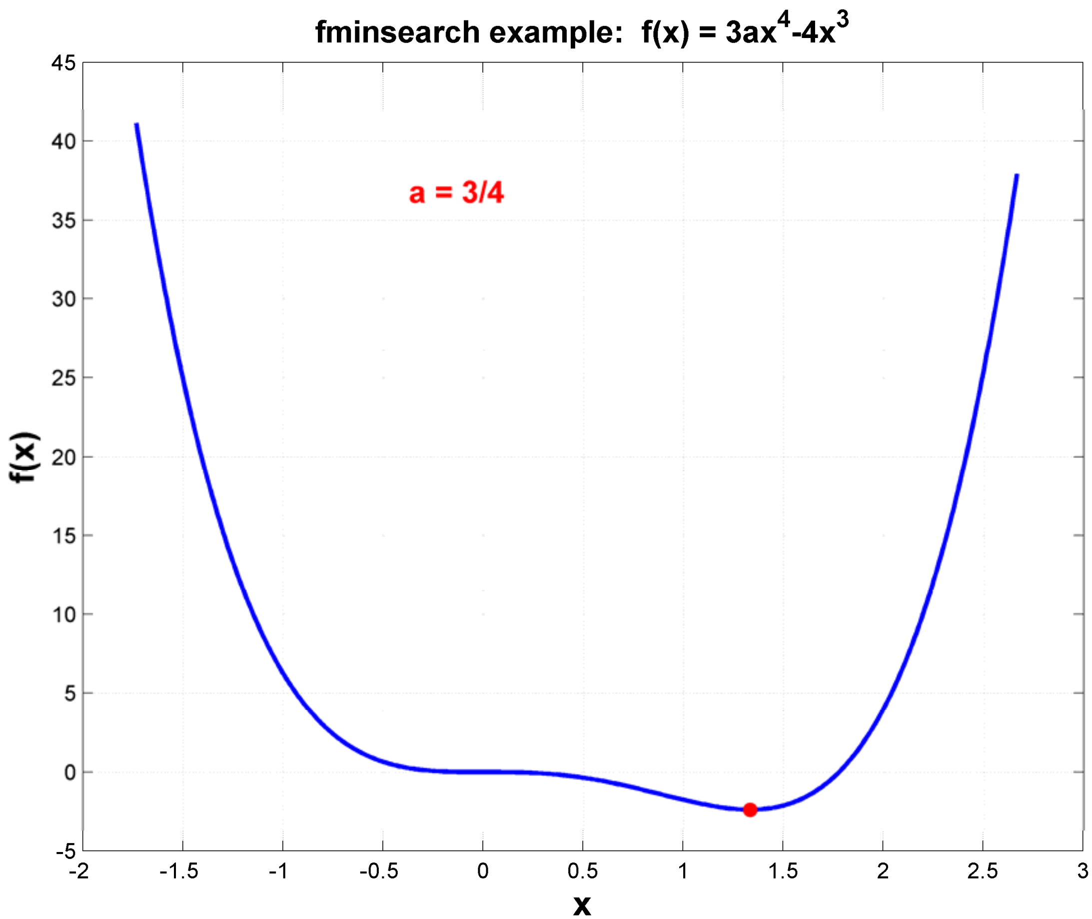
$$x_{\min} = \frac{1}{a}, \quad f_{\min} = f(x_{\min}) = -\frac{1}{a^3}$$

This file is provided to you in your folder: "fminsearch_Matlab_example.m"

```
function [xmin,fmin] = fminsearch_Matlab_example(a)
%---------------------------------------------------------------------------------------
% Use Matlab's "fminsearch" to find the minimum value of a simple polynomial function.
%
% USAGE:    [xmin,fmin] = fminsearch_Matlab_example(a)
% INPUT:    a             * (double) the parameter "a"
% OUTPUT:   xmin          * (double) x value at which f(x) is minimized
%           fmin          * (double) the minimum value of f(x)
%
%   f(x) = 3*a*x^4 - 4*x^3
%   "a" is a free parameter
% Analytic Solution (using Calculus):   xmin = 1/a , fmin = -1/a^3
%
% Written by Alex Casti, FDU 12/07/2012
%---------------------------------------------------------------------------------------
```

Optimization Example: Matlab's fminsearch

```
>> [xmin,fmin] = fminsearch_Matlab_example(3/4);

a = 0.75
Numerical:  (xmin , fmin) = (1.3333 , -2.37037)
Exact    :  (xmin , fmin) = (1.33333 , -2.37037)
>>
```



Optimize Your HSP Data with Matlab

- Now fit your experimental data to the Poisson model.
- Use the provided Matlab file "HSP_fitModel_fminSearch.m"
- This Matlab file is part of the suite of routines I wrote specifically for the analysis and presentation of the HSP experiment.

```matlab
function [params,KOPT,ALPHAOPT] = HSP_fitModel_fminSearch(data,K,plotResults)
%-----------------------------------------------------------------------------------------
% Analyze experimental data from the Hecht, Schlaer, Pirenne (1942) visual threshold
% experiment.  This routine fits the Poisson probability model (CDF) of the
% Probability of Seeing curve P(I).  Uses Matlab's "fminsearch" algorithm (Simplex)
% to optimize the "alpha" scale parameter in the Mean Square Error residual for each
% element of a set of values for the threshold parameter K.
%
% USAGE:    [params,KOPT,ALPHAOPT] = HSP_fitModel_fminSearch(data,K,plotResults)
% INPUT:    data             * (struct) data structure from experiment
%           K                * (int vector) K values to cycle through for optimization (threshold parameter)
%           plotResults      * (logical) plot results or not (default TRUE)
% OUTPUT:   params           * (matrix) fit results
%                              COLUMNS = [K         alpha         MSE         exitFlag]
%           KOPT             * (int)      optimal K value
%           ALPHAOPT         * (double)   optimal alpha value
% Comments:
%  (1) K = threshold parameter (minimum #photons for seeing)
%  (2) alpha = scale parameter (quantum efficiency)
%  (3) MSE = mean square error
%  (4) exitFlag = output of "fminsearch" indicating whether optimization was successful or not
%
%  Written by Alex Casti, FDU  12/04/2012
%  Last updated 12/07/2012
%-----------------------------------------------------------------------------------------
```

Experimental Setup (Final/Darkroom)

Generic Slide