Point estimations

Agenda

- Maximum Likelihood Estimation (MLE)
- Method of Moments (MoM)
- Bayesian estimation
  - Maximum a posteriori estimation (MAP)
  - Posterior mean

Maximum Likelihood Estimation

Likelihood function

The likelihood of unknown parameter θ given your data.

$X_1, \cdots, X_n$: random sample from $f(x|\theta)$

$$L(\theta|X_1, \cdots, X_n) = \underbrace{f(X_1, \cdots, X_n|\theta)}_{\text{joint probability / density}}$$

special case: if $X_1, \cdots, X_n \overset{iid}{\sim} f(x|\theta)$

$$L(\theta|X_1, \cdots, X_n) = \prod_{i=1}^{n} f(X_i|\theta) \quad \text{(好算)}$$

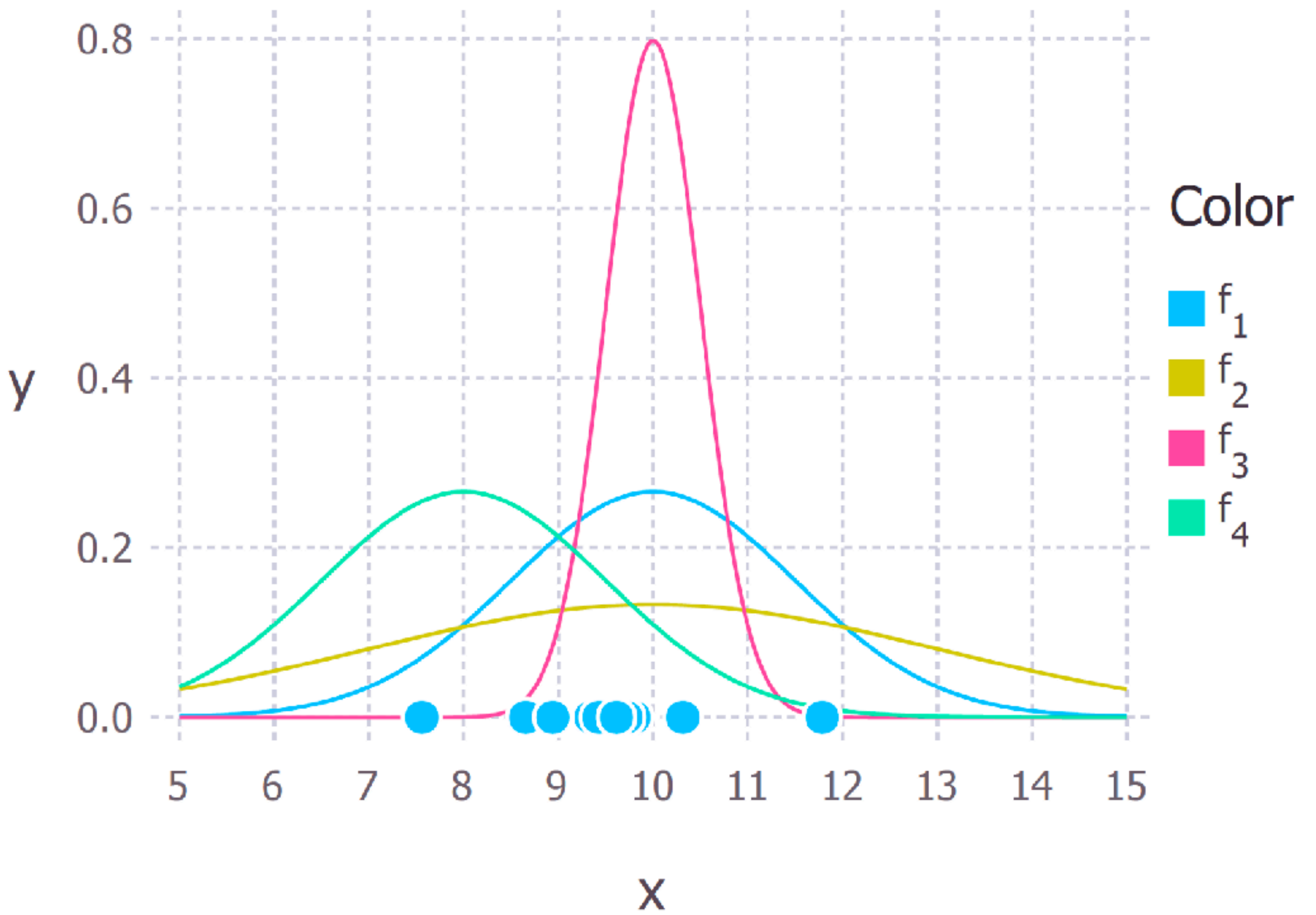
- The likelihood function is a function of θ
- It is **not** a probability density function
- It measures the “support” (i.e. likelihood) provided by the data for each possible value of the parameter.

MLE

- Find the parameter that is "most likely" to observe your data.

- Maximizing the likelihood function is equivalent to maximizing the log-likelihood function (for computational issues)

$\ell(\theta | x_1, \cdots, x_n) = \log L(\theta | x_1, \cdots, x_n)$



https://towardsdatascience.com/probability-concepts-explained-maximum-likelihood-estimation-c7b4342fdbb1

How to maximize a function?

Example: coin tossing

Assume $X_1, \cdots, X_n \overset{iid}{\sim} \text{Bernoulli}(p)$

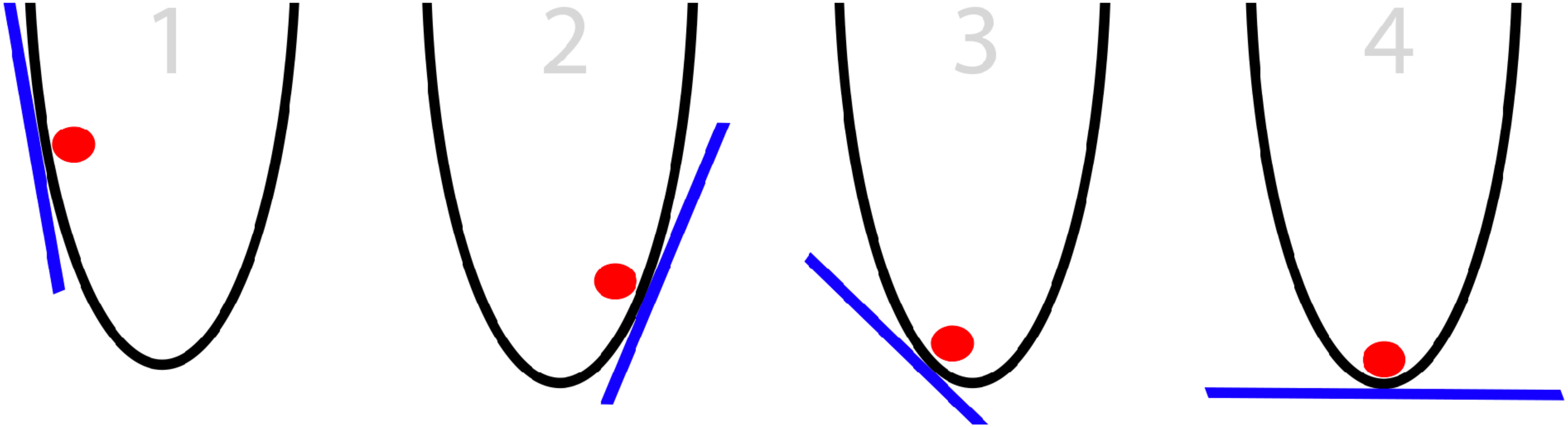
$$L(p \mid X_1, \cdots, X_n) = \prod_{i=1}^{n} p^{X_i} (1-p)^{1-X_i}$$

$$\ell(p \mid X_1, \cdots, X_n) = \log L = \sum_{i=1}^{n} X_i \cdot \log p + (1-X_i) \log(1-p)$$

$$\frac{d\ell}{dp} = \sum_{i=1}^{n} \frac{X_i}{p} - \frac{1-X_i}{1-p} = \cdots = \frac{\sum_{i=1}^{n} X_i - np}{p(1-p)} \overset{\Delta}{=} 0$$

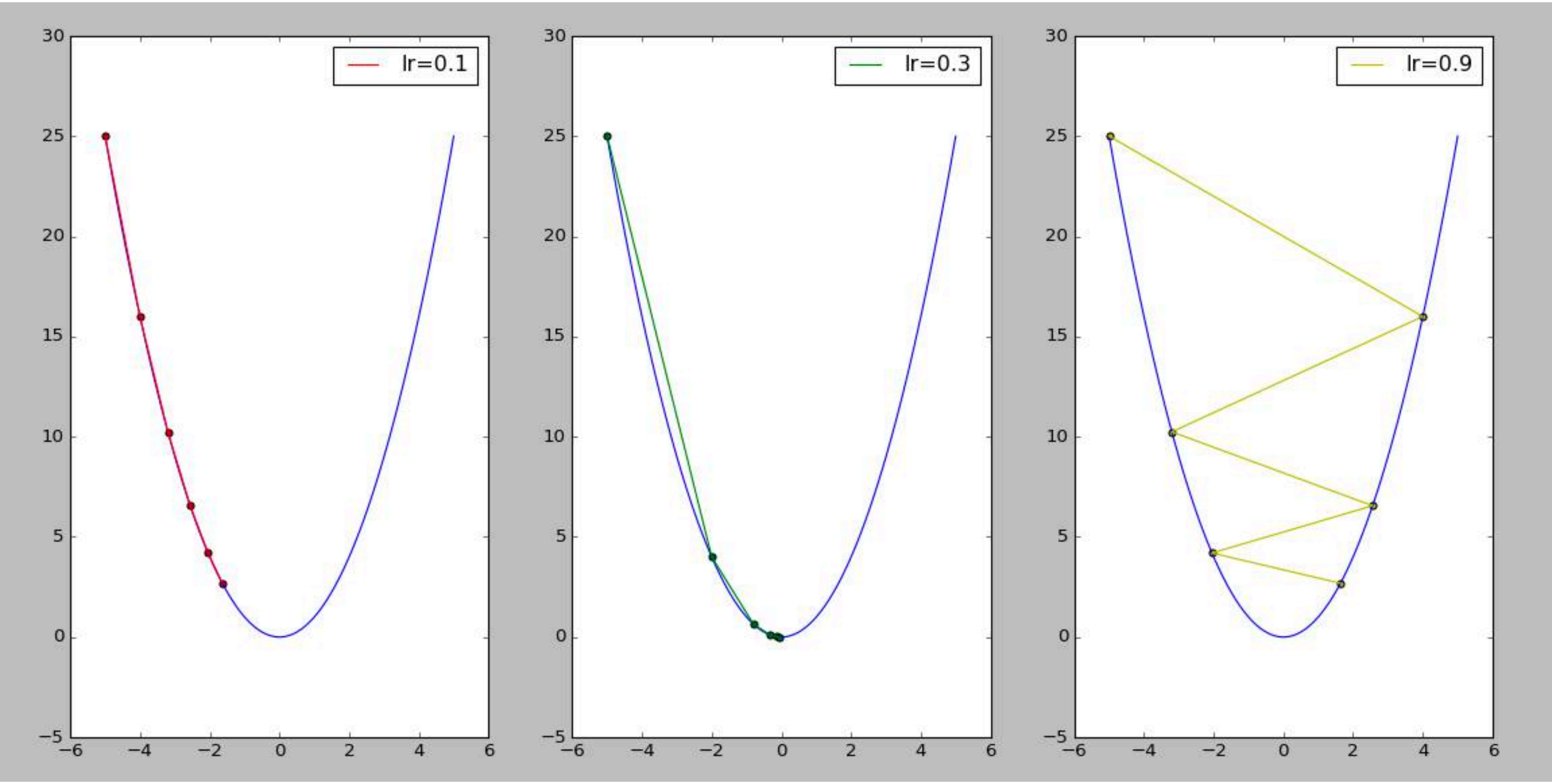
$$\Rightarrow \hat{p} = \frac{1}{n} \sum_{i=1}^{n} X_i \overset{\Delta}{=} \overline{X}$$

Gradient descent



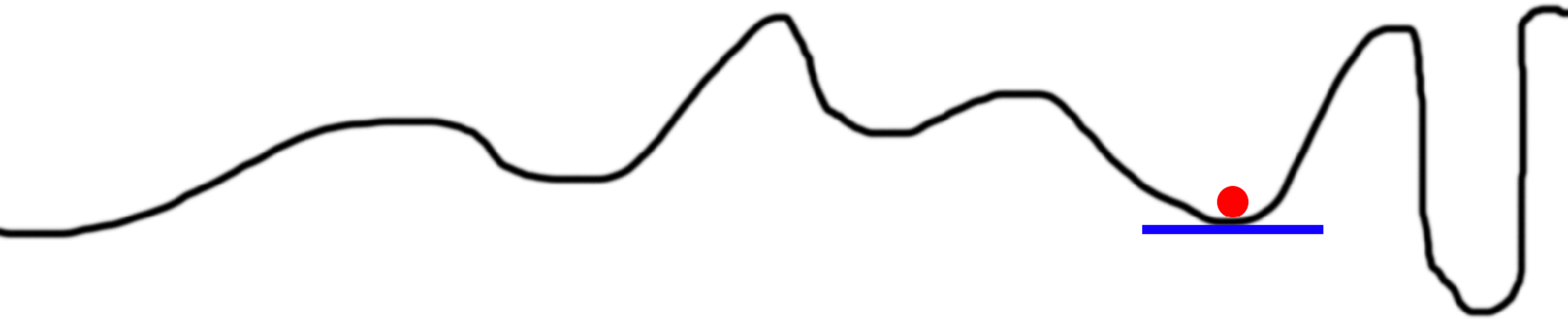
https://iamtrask.github.io/2015/07/27/python-network-part2/

Learning rate



https://blog.csdn.net/leadai/article/details/78662036

Local optimums



https://iamtrask.github.io/2015/07/27/python-network-part2/

Method of Moments

Expectation

- If X is a discrete random variable with p.m.f. $f_X(x)$,

$$E[X] = \sum_x x f_X(x)$$

- If X is a continuous random variable with p.d.f. $f_X(x)$,

$$E[X] = \int_x x f_X(x)$$

- $E\left[X^k\right]$ are called moments with $k \geq 1$

Law of large number

- If $X_1, X_2, \cdots, X_n \overset{iid}{\sim} f_X(x)$, then $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ converges to $E[X_i]$ (in probability) as $n \to \infty$
- LLN holds for every moments

Method of moments

- The values of moments depend on the values of unknown parameters θ (moment conditions)

- By LLN, we can estimate moments by sample means

Example: normal distribution

Let $X_1, X_2, \cdots, X_n \overset{iid}{\sim} N(\mu, \sigma^2)$

- moment conditions:

$$\frac{1}{n}\sum_{i=1}^{n} X_i \approx E[X_1] = \mu$$

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 \approx E[X_1^2] = \mu^2 + \sigma^2$$

- By solving the above moment conditions we have

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i, \text{ and } \hat{\sigma^2} = \frac{1}{n}\sum_{i=1}^{n} X_i^2 - \hat{\mu}^2$$

Example: linear regression

Let $Y_i \overset{iid}{\sim} N\left(\mu(\mathbf{x}_i), \sigma^2\right)$ with $\mathbf{x}_i = \left[x_{i1}, \cdots, x_{ip}\right]^T \in \mathbb{R}^p$ and

$\mu(\mathbf{x}_i) = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$

- Moment conditions:

$$E\left[Y - \mu(\mathbf{x})\right] = 0$$

$$E\left[x_j\left(Y - \mu(\mathbf{x})\right)\right] = 0$$

$$E\left[\left(Y - \mu(\mathbf{x})\right)^2\right] = \sigma^2$$

- Plug the sample moments into the moment conditions, we obtain

$$\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \left(\beta_0 + \beta_i x_{i1} + \cdots + \beta_p x_{ip}\right)\right) \approx 0$$

$$\frac{1}{n}\sum_{i=1}^{n} x_{ij}\left(Y_i - \left(\beta_0 + \beta_i x_{i1} + \cdots + \beta_p x_{ip}\right)\right) \approx 0$$

$$\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \left(\beta_0 + \beta_i x_{i1} + \cdots + \beta_p x_{ip}\right)\right)^2 \approx \sigma^2$$

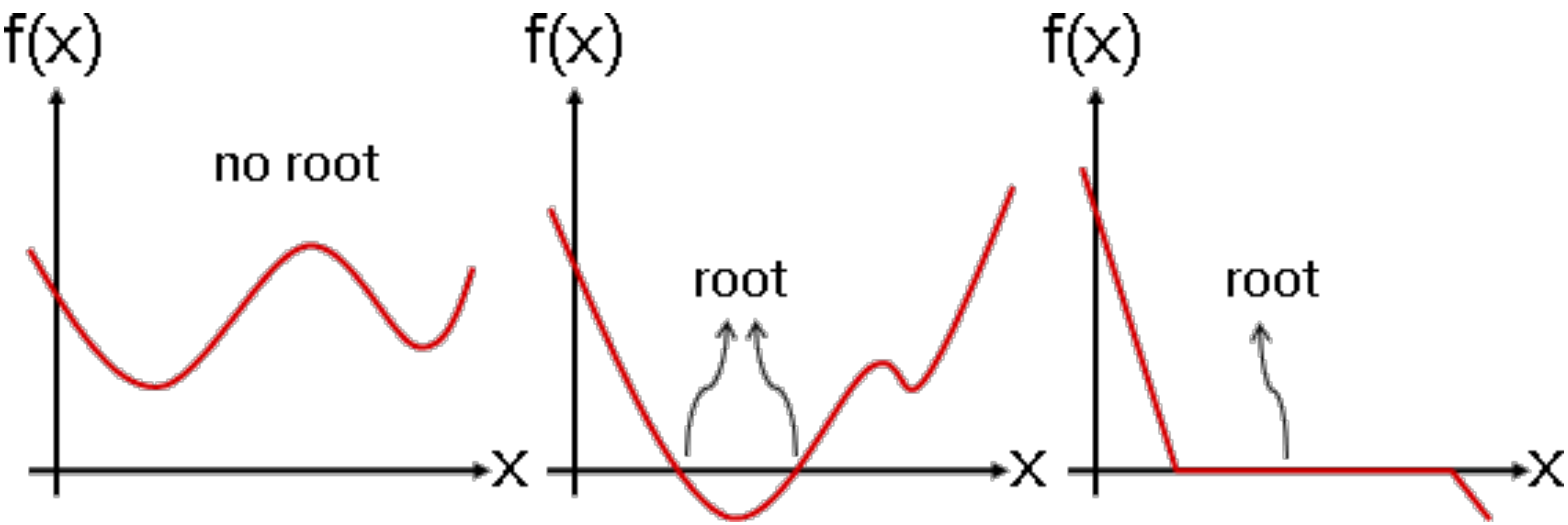
Solving linear systems

The solution of a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ can be found (if it exists) by

- Gaussian elimination (numpy.linalg.solve)
- Minimize $\|\mathbf{A}\mathbf{x} = \mathbf{b}\|^2$ (numpy.linalg.lstsq)

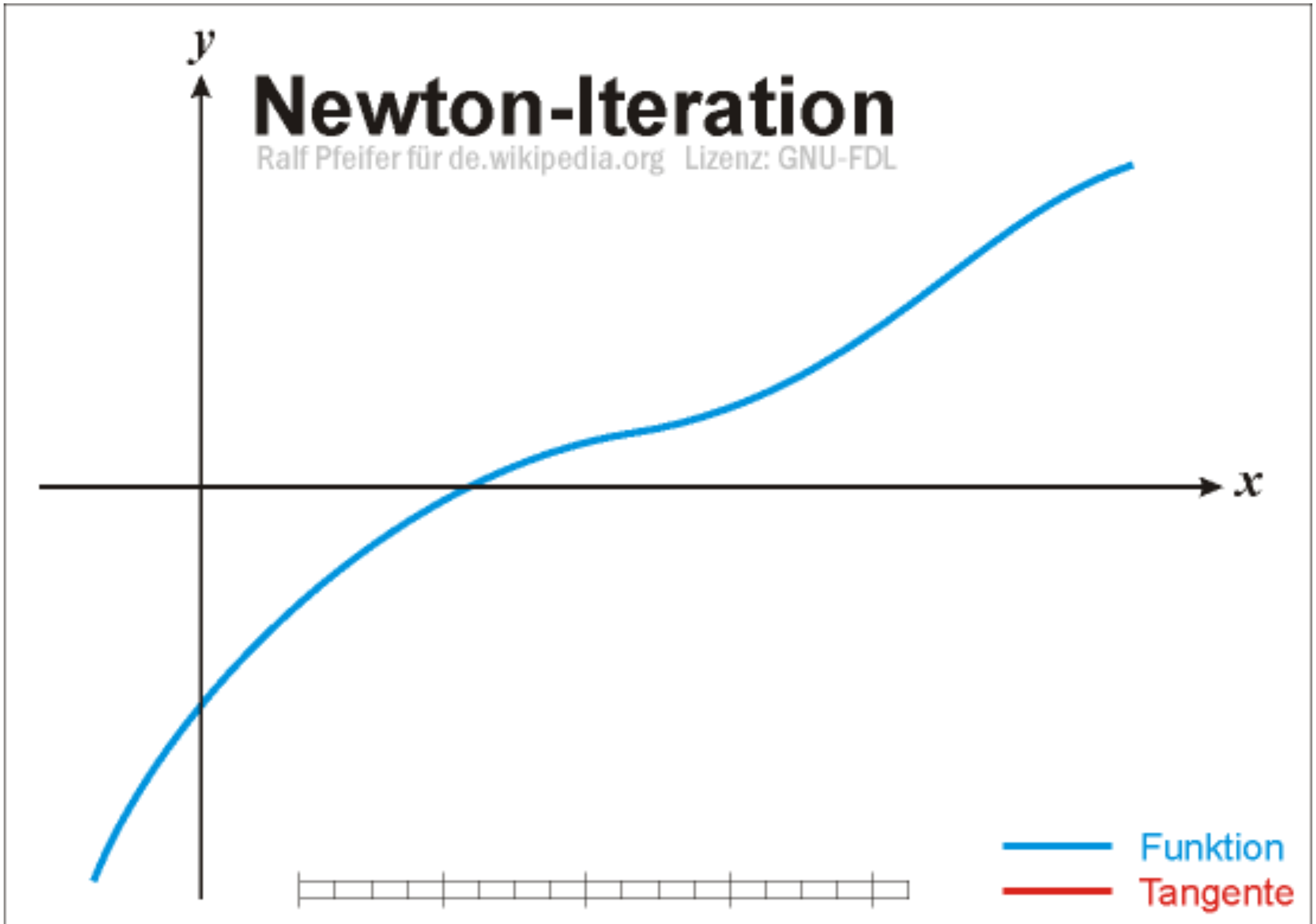
Solving nonlinear systems

The solution of a nonlinear system $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ can be found (if it exists) by various root-finding algorithms (scipy.optimize.root)

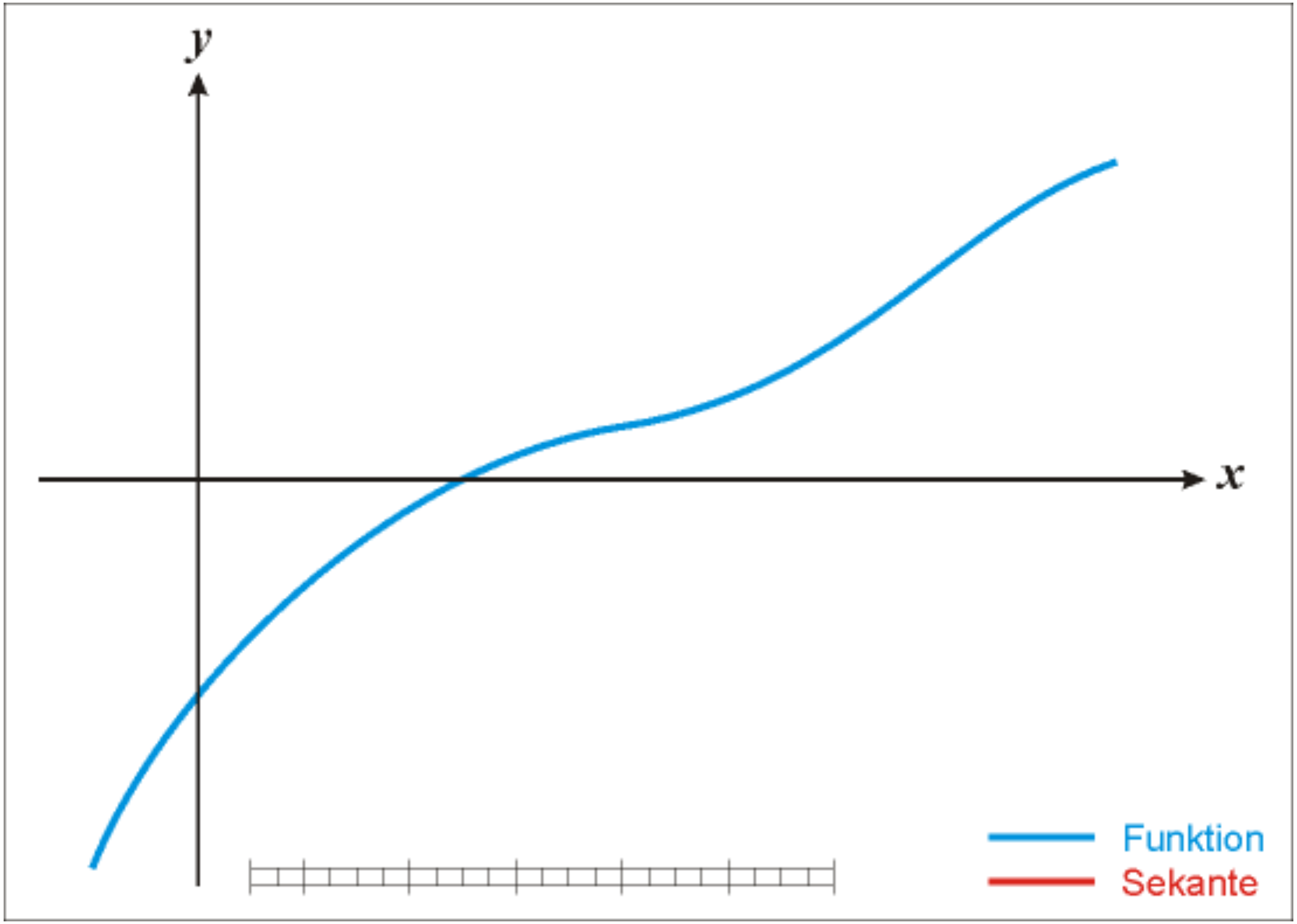


http://www.csie.ntnu.edu.tw/~u91029/RootFinding.html

Newton’s method



Secant method



Pros

- Easy to compute and always work
- MoM is consistent; i.e. $\hat{\theta}_{MoM} \to \theta$ as $n \to \infty$

Cons

- MoM may not be unique: different moment conditions yields to different results!
- Not the most efficient (i.e. achieving minimum mean squared error, MSE) estimators
- Sometimes MoM may be meaningless

MoM may be meaningless

- Suppose we observe 3, 5, 6, 18 from $U(0,\theta)$

- Since $E[X] = \theta/2$ for $X \sim U(0,\theta)$, the MoM of θ is

$$\hat{\theta}_{MoM} = 2\bar{X} = 2 \times \frac{3+5+6+18}{4} = 16$$

- This estimation is not acceptable since we have already observed a 18.

Extensions

- Generalized MoM
  - Generalized moment conditions
  - The number of conditions may exceed the number of parameters
- Method of simulated moments
  - Approximate the theoretical moments when they are not available

Bayesian estimation

Key concepts

- Since $\hat{\theta}$ (derived from some random sample) is random, we can treat θ as random and specify its probability distribution as $\pi(\theta)$

- The probability $\pi(\theta)$ is usually specified by a data scientist to express one's beliefs. Thus, we call it a “prior”.

Posterior

- By Bayes' theorem, we can derive the "posterior" distribution of θ:

$$
\begin{aligned}
p(\theta \mid X_1, \cdots, X_n) &= \frac{f(X_1, \cdots, X_n \mid \theta)\pi(\theta)}{f(X_1, \cdots, X_n)} \\
&= \frac{f(X_1, \cdots, X_n \mid \theta)\pi(\theta)}{\int f(X_1, \cdots, X_n \mid \theta)\pi(\theta)d\theta} \\
&\propto f(X_1, \cdots, X_n \mid \theta)\pi(\theta)
\end{aligned}
$$

Maximum a posteriori estimation

- The posterior distribution can be interpreted as the conditional probability of θ given observational data

- Thus, similar to MLE, we may find the mode of the posterior since it is the most likely value of θ

$$\hat{\theta}_{MAP} = \arg\max f(X_1, \cdots, X_n \mid \theta)\pi(\theta)$$

Posterior mean

- The "mean" of the posterior is another frequently used Bayesian estimator,

$$\hat{\theta} = E[\theta \,|\, X_1, \cdots, X_n] = \int \theta p\left(\theta \,|\, X_1, \cdots, X_n\right) d\theta$$

- The expectation is often approximated by Markov chain Monte Carlo (MCMC) method since the above integration is usually difficult

Example: normal distribution

Let $X_1, X_2, \cdots, X_n \overset{iid}{\sim} N(\mu, \sigma^2)$ and we assume that $\mu \sim N(\mu_0, \tau^2)$. Then

$$p(\mu \mid X_1, \cdots, X_n) \propto \frac{1}{\sqrt{2\pi\tau}} e^{-\frac{1}{2}\left(\frac{\mu - \mu_0}{\tau}\right)^2} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\left(\frac{X_i - \mu}{\sigma}\right)^2}$$

and

$$\hat{\mu} = \frac{n\tau^2}{n\tau^2 + \sigma^2}\bar{X} + \frac{1}{n\tau^2 + \sigma^2}\mu_0$$

(My unfair) suggestions

- Use Bayesian estimations when you have a domain expert; otherwise, use MLE
- Use MoM only for computational issues
  - The posterior (or likelihood function) is not convex
  - Big data

Homework: logistic regression

- Breast Cancer Wisconsin (Diagnostic) Data Set (also available in scikit-learn)

- Assume that $Y_i \in \{0,1\} \overset{iid}{\sim}$ Bernoulli$\big(p(\mathbf{x}_i)\big)$ with

$$p(x_i) = \frac{\exp\left[\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}\right]}{1 + \exp\left[\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}\right]}$$

  Estimate the unknown coefficients $\theta = [\beta_0, \beta_1, \cdots, \beta_p]'$ by either MLE or MoM. Compare your results with the the ones provided by scikit-learn (example) with very large C.

- Moment conditions:

$$E\left[Y - p(\mathbf{x})\right] = 0$$

$$E\left[x_j\left(Y - p(\mathbf{x})\right)\right] = 0$$

- Plug the sample moments into the moment conditions, we obtain

$$\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - p\left(\mathbf{x}_i\right)\right) = 0$$

$$\frac{1}{n}\sum_{i=1}^{n}x_{ij}\left(Y_i - p\left(\mathbf{x}_i\right)\right) = 0$$

Readings

- Chapters 10.1–10.3 and 12.1–12.2 of "All of statistics"