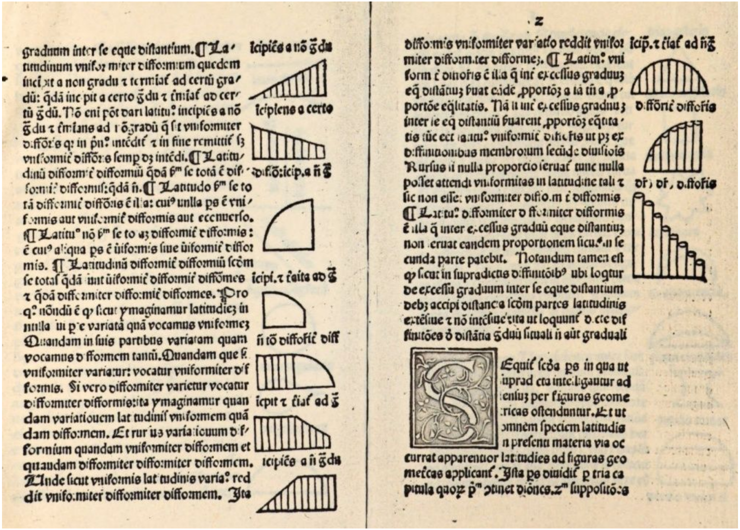
Introduction to data science & artificial intelligence (INF7100)

Arthur Charpentier

#211 Statistical Functions (cdf and density)

été 2020

Statistical Functions



Nicole Oresme, Tractatus de latitudinibus formarum, 1486

Cumulative Distribution Function

Given a random variable X, $F(y) = \mathbb{P}[X \leq x]$

F is an increasing function, taking values in $[0, 1]$.

Consider a sample $\boldsymbol{x} = \{x_1, y_2, \cdots, x_n\}$, a natural estimator is

$$\widehat{F}(x) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}(x_i \leq x)$$

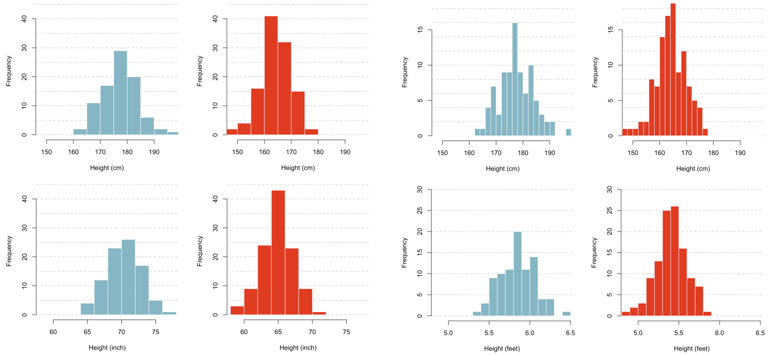
```
> import numpy
> x = numpy.sort(x)
> n = x.size
> y = numpy.arange(1, n+1) / n
> import matplotlib.pyplot as plt
> plt.plot(x, np.linspace(0, 1, n, endpoint=False))
```

```
> x = sort(x)
> n = length(x)
> y = (1:n)/n
> plot(ecdf(x))
```

Density & Histogram

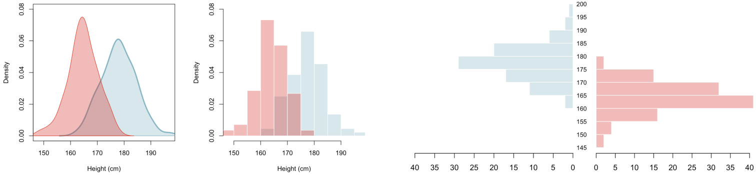
Given a random variable X, f is such that $F(x) = \int_{-\infty}^{x} f(t)dt$ or conversely, $f(x) = F'(x)$.

Thus, $\mathbb{P}(X \in [a, b]) = \int_{a}^{b} f(t)dt$



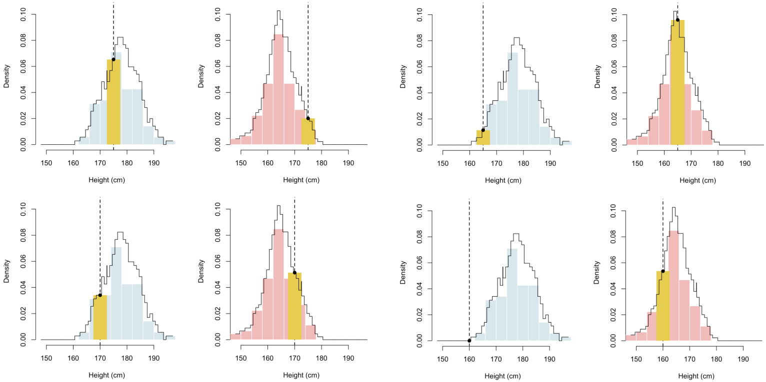
Histogram & Density

Can be used to compare distributions



Moving Histogram

$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n}\mathbf{1}(|x_i - x| \leq h/2)$$



Moving Histogram

$\widehat{F}$ cannot be differentiated, but we can consider

$$f_h(x) = \frac{1}{h}\underbrace{F(x+h/2) - F(x-h/2)}_{\mathbb{P}(X\in[x\pm h/2])}$$

i.e.

$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n}\mathbf{1}\left(x_i \in [x-h/2, x+h/2]\right)$$

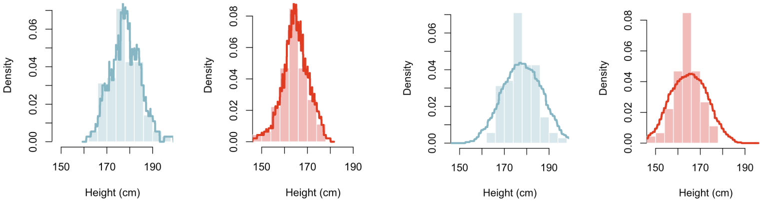
$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n}\mathbf{1}(|x_i - x| \leq h/2)$$

One can prove that $\mathbb{E}(\widehat{f}_h(x)) = f_h(x) \sim f(x) + \dfrac{h^2}{24}f''(x)$

i.e. $\text{bias}(\widehat{f}_h(x)) \sim \dfrac{h^2}{24}f''(x)$, while $\text{Var}(\widehat{f}_h(x)) \sim \dfrac{1}{nh}\cdot f_h(x)$

Moving Histogram

$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} \mathbf{1}(|x_i - x| \leq h/2)$$



small h

bias bias$(\widehat{f}_h(x))$ small

variance Var$(\widehat{f}_h(x))$ large

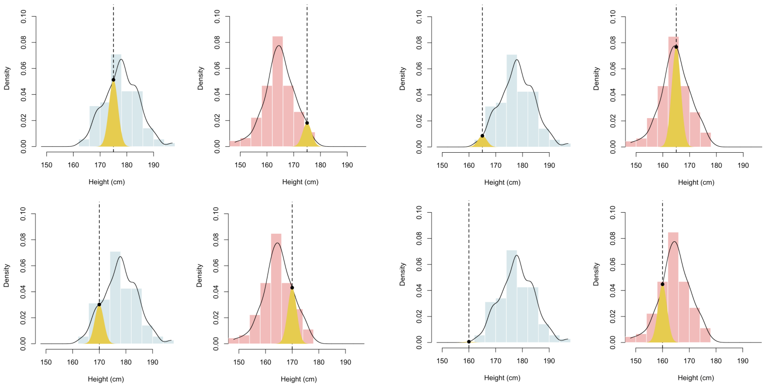
large h

bias bias$(\widehat{f}_h(x))$ large

variance Var$(\widehat{f}_h(x))$ small

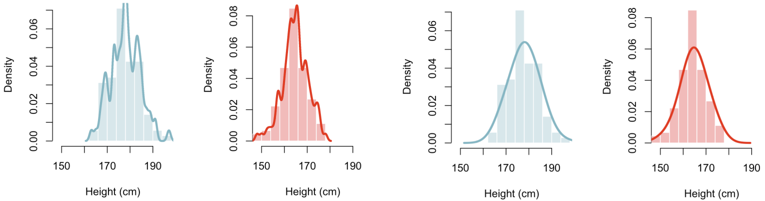
Kernel Density

$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x_i - x}{h}\right)$$



Kernel Density

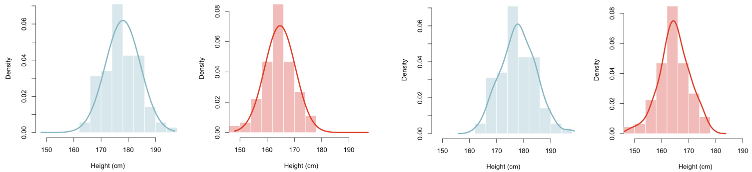
$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x_i - x}{h}\right)$$



small h
bias bias$(\widehat{f}_h(x))$ small
variance Var$(\widehat{f}_h(x))$ large

large h
bias bias$(\widehat{f}_h(x))$ large
variance Var$(\widehat{f}_h(x))$ small

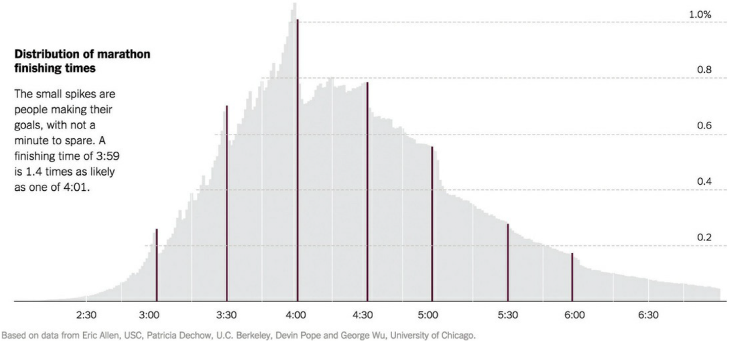
Histogram & Density




```
> import matplotlib.pyplot as plt
> hist = plt.hist(x, bins=30, normed=True)
> from sklearn.neighbors import KernelDensity
> k = KernelDensity(bandwidth=1.0, kernel='gaussian')
> k.fit(x[:, None])
```

```
> hist(x, probability=TRUE)
> plot(density(x))
> plot(density(x), kernel="gaussian", bw=1)
```

Histogram & Density



Reference-Dependent Preferences: Evidence from Marathon Runners