Statistical Data Analysis – Revision Lec. 1

Exam 9 June 2020 → online, open book
(see email)

No C++ on exam (but, could ask about algorithms)

Format similar to past papers.

Definition & interp. of Probability

↑ Kolmogorov axioms

Interp: . frequency → freq. stat.

. subjective → Bayesian stat.
(degree of belief)

Bayes' thm

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$= \frac{P(B|A)P(A)}{\sum_i P(B|A_i)P(A_i)}$$

Random variables, prob func., pdfs

pdf $f(x)$ $\longrightarrow$ cdf $F(x) = \int_{-\infty}^{x} f(x')\,dx'$

joint pdf $f(x,y)$

marginal " $f_x(x) = \int f(x,y)\,dy$

conditional pdf $f(x|y) = \dfrac{f(x,y)}{f_y(y)}$

Bayes' thm $f(x|y) = \dfrac{f(y|x)\,f_x(x)}{f_y(y)}$

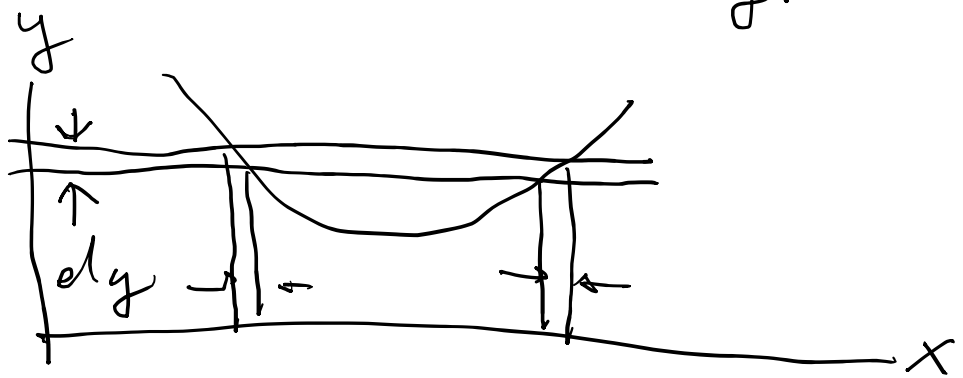
Change of variables

$x \sim f(x)$

$y(x)$ is a r.v. $\longrightarrow$ $y \sim g(y)$

$$g(y)\,dy = f(x)\,dx \Rightarrow g(y) = f(x(y))\left|\frac{dx}{dy}\right| \quad (1\text{-to-}1)$$

If $y(x)$ not 1-to-1



add both contrib

$$\vec{x} = (x_1, \cdots, x_n) \sim f(\vec{x})$$

$$\vec{y} = (y_1(\vec{x}), \cdots, y_n(\vec{x})), \quad \text{inv: } \vec{x}(\vec{y})$$

$$g(\vec{y}) = |J| \, f(\vec{x}(\vec{y}))$$

$$J = \begin{vmatrix} \frac{\partial x_1}{\partial y_1} & \cdots & \frac{\partial x_1}{\partial y_n} \\ \vdots & & \\ \frac{\partial x_n}{\partial y_1} & \cdots & \frac{\partial x_n}{\partial y_n} \end{vmatrix}$$

Sums & products of r.v.s.

$\longrightarrow$ convolutions

$x, y \sim f(x, y)$, $z = xy$

$$g(z) = \int f\left(x, \frac{z}{x}\right) \frac{dx}{x}$$

similar for $z = x + y$

Expectation values (means)

$$E[x] = \int x f(x)\, dx \equiv \mu \qquad \text{(continuous)}$$

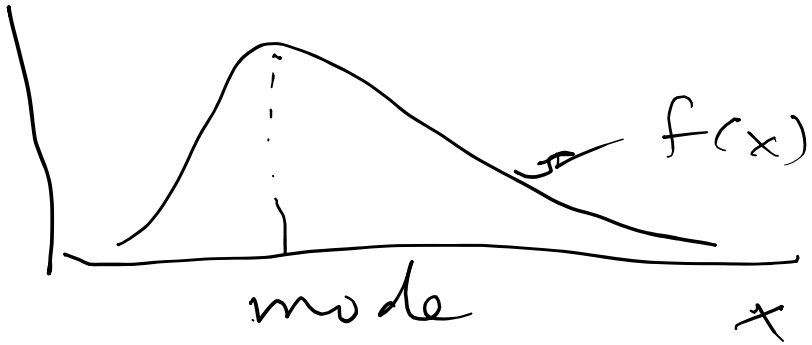
$$\text{or} \quad E[n] = \sum_n n\, P(n) \qquad \text{(discrete)}$$

Variance $\quad V[x] = E[x^2] - (E[x])^2 \equiv \sigma^2$

covariance $\quad \text{cov}[x,y] = E[xy] - E[x]E[y]$

mean = centre-of-gravity of pdf

also mode
(pos. of max)

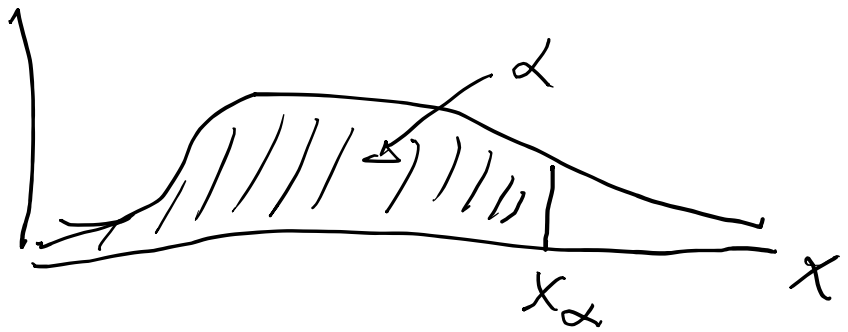


median

α-point

$F(x_\alpha) = \alpha$

$x_\alpha = F^{-1}(\alpha)$ ← quantile



Error propagation

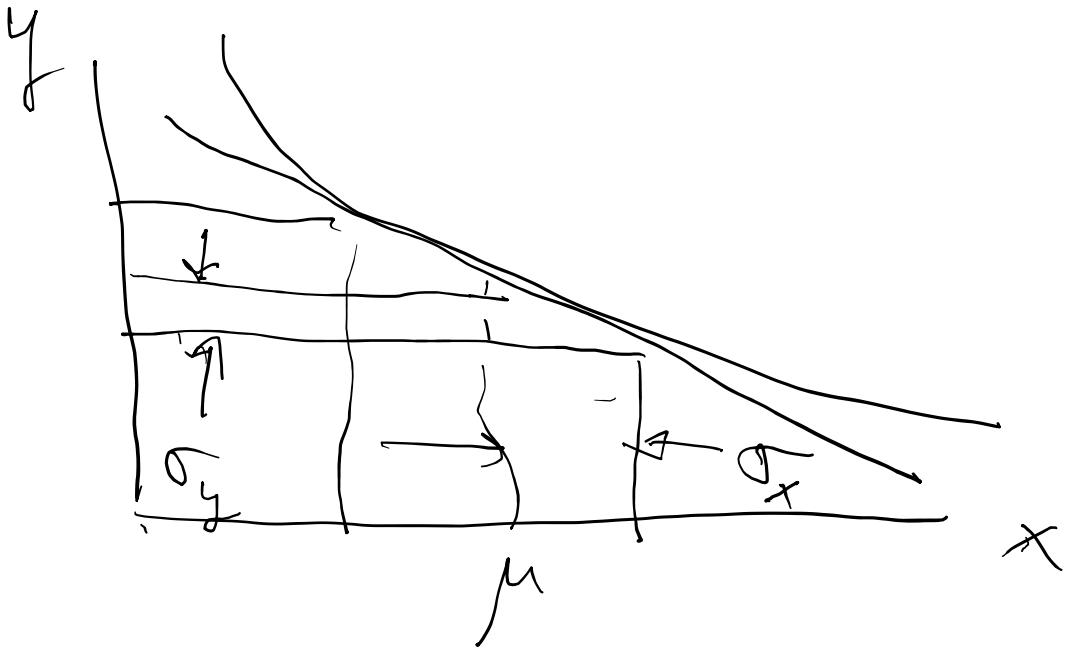
Given $\vec{x}$, known / estimated $\mathrm{cov}[x_i, x_j] = V_{ij}$

&

$E[x_i] = \mu_i$

$$V[y(\vec{x})] = \sum_{i,j} \left[\frac{\partial y}{\partial x_i} \frac{\partial y}{\partial x_j}\right]_{\vec{x}=\vec{\mu}} \mathrm{cov}[x_i, x_j]$$

Error prop approx, good if $y(\vec{x})$ ~ linear



Useful pdfs

Gauss $f(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(x-\mu)^2/2\sigma^2}$ $-\infty < x < \infty$

(μ: mean; σ: std. dev.)

Expo. $f(x;\xi) = \frac{1}{\xi} e^{-x/\xi}$ $x \geq 0$

(ξ: mean)

Poisson $P(n;\nu) = \frac{\nu^n}{n!} e^{-\nu}$ $n = 0, 1, \ldots$

(ν)

$E[n] = \nu$, $V[n] = \nu$

Monte Carlo (MC)

1) $r \sim U[0,1] \longrightarrow r_1, r_2 \cdots r_n$ (indep.)

2) $r_1, r_2 \cdots r_n \longrightarrow x_1, x_2, \cdots x_m$ such that $x \sim f(x)$

3) Use x_i to estimate property of $f(x)$

e.g. $\int_a^b f(x)\,dx$ = fraction of x_i in $[a,b]$

Random # gen. (MLCG)

$$n_{i+1} = (a n_i) \bmod m$$

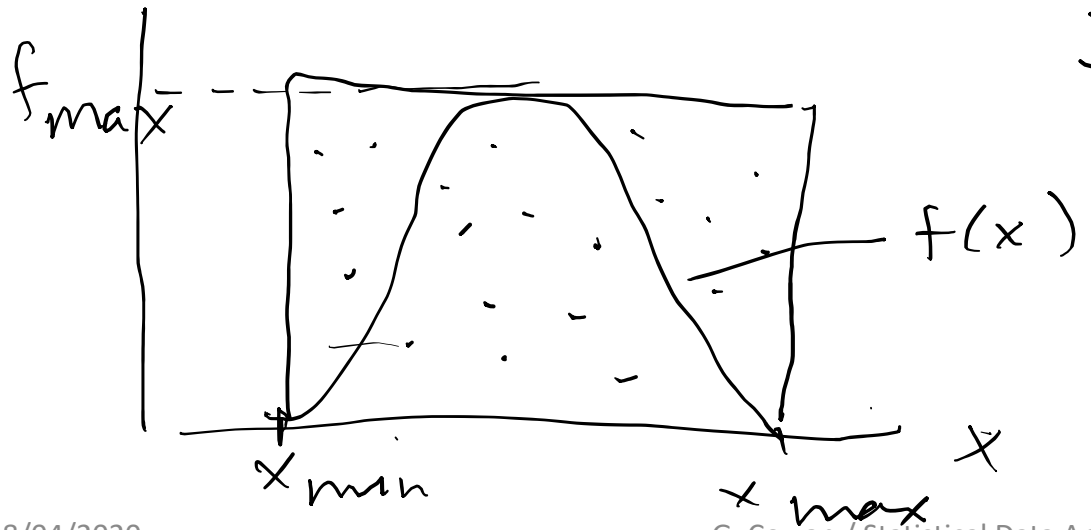
$$r_i = \frac{n_i}{n_{max}} \sim U[0,1]$$

$n_{max} = m - 1$

Transformation method $F(x) = r$

& solve for $x(r)$

Acceptance – rejection



If point below $f(x)$ accept x.

Statistical tests

Test H_0 ($\leftarrow P(x|H_0)$) vs. H_1

$P(x \in w | H_0) \leq \alpha$ (size)

↑ critical region

↖ small

If x in w, reject H_0

$P(x \in w | H_1)$ = power of test wrt H_1 ($M(H_1)$)

H_0 = background evt ← test H_0

H_1 = signal "

$\alpha = P(\text{reject as bkg} \mid b) = \varepsilon_b$

$\eta = P(\text{reject as bkg} \mid s) = \varepsilon_s$

↑ ≡ accept as candidate sig.

$$P(s \mid \text{acc as sig}) = \frac{P(\text{acc as } s \mid s)\,\pi_s}{P(\text{acc as } s \mid s)\,\pi_s + P(\text{acc as } s \mid b)\,\pi_b}$$

Neyman – Pearson lemma: choose w such that

$$\frac{P(\vec{x}|H_1)}{P(\vec{x}|H_0)} \geq c_\alpha \quad \forall \vec{x} \in w$$

$$\underbrace{\frac{P(\vec{x}|H_1)}{P(\vec{x}|H_0)}}_{t(\vec{x})} < c_\alpha \quad \forall \vec{x} \notin w$$

$t(\vec{x}) = c_\alpha$ boundary of w.

gives max power wrt H_1 for test of H_0 of size α.

→ multivariate methods

(Machine Learning)

Fisher (linear) $t(\vec{x}) = \sum_i a_i x_i$

Neural Network

Boosted Decision trees

+ issues (e.g. overtraining)