$H_0$ : All bkg, no signal

$H_1$ : bkg + signal

Poission log likelihood: $P(e,o)$

$$P(e,o) = 2 \sum_{\text{sample } i} e_i - o_i + o_i \log \frac{o_i}{e_i}, \quad o_i = \text{observed}, \quad e_i = \text{expected}$$

dof = 2 b/c we usually want 2D graph

Critical values:

| $(1-\alpha)$ (%) | $m=1$ | $m=2$ | $m=3$ |
|---|---|---|---|
| 68.27 $1\sigma$ | 1.00 | 2.30 | 3.53 |
| 90. | 2.71 | 4.61 | 6.25 |
| 95. | 3.84 | 5.99 | 7.82 |
| 95.45 $2\sigma$ | 4.00 | 6.18 | 8.03 |
| 99. | 6.63 | 9.21 | 11.34 |
| 99.73 $3\sigma$ | 9.00 | 11.83 | 14.16 |

$P(e,o) >$ critical value $\Rightarrow$ reject $H_0$ (shown at left)

**TABLE 1.** Chi-square differences ($\hat{\delta}$) above minimum

Number of parameters = dof

| Significance $\alpha$ | 1 | 2 | 3 |
|---|---|---|---|
| 0.68 | 1.00 | 2.30 | 3.50 |
| 0.90 | 2.71 | 4.61 | 6.25 |
| 0.99 | 6.63 | 9.21 | 11.30 |

$\Delta\chi^2 = \chi^2 - \chi^2_{min}$.

The entries are chi2.ppf

# Estimator properties

Consistency : $\lim_{N\to\infty} \hat{\theta} = \theta$ ← true value of the parameter $\theta$

Bias : $b = E[\hat{\theta}] - \theta$

Efficiency : $\hat{\theta}$ is efficient if its variance $V[\hat{\theta}]$ is small

# Rao-Cramér-Frechet
## minimum variance bound

Variance $V[\hat{\theta}] \geq I^{-1}(\vec{\theta})$ ← hat means estimator

$\theta$ = parameters
$x$ = measurement
$f$ = unknown PDF that we're interested in

Information matrix $I_{jk}(\vec{\theta}) = -E\left[\sum_{i=1}^{N} \frac{\partial^2 \ln f(x_i;\theta)}{\partial\theta_j\,\partial\theta_k}\right]$

$$= -N\int \frac{\partial^2 \ln f}{\partial\theta_j\,\partial\theta_k} f\,dx = N\int \frac{\partial \ln f}{\partial\theta_j}\frac{\partial \ln f}{\partial\theta_k} f\,dx$$

$$= N\int \frac{1}{f}\frac{\partial f}{\partial\theta_j}\frac{\partial f}{\partial\theta_k}\,dx = E\left[\sum_{i=1}^{N} \frac{\partial \ln f}{\partial\theta_j}\frac{\partial \ln f}{\partial\theta_k}\right]$$

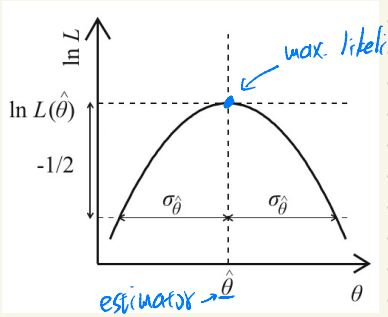
# Maximum likelihood

PDF $f(x_i;\theta,\sigma_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(x_i-\theta)^2}{2\sigma_i^2}\right)$

log-likelihood $\ln L(\theta) = -\frac{1}{2}\sum_{i=1}^{N} \frac{(x_i-\theta)^2}{\sigma_i^2} + \text{const}$ maximize $\ln L$

$\ln L(\theta) = \ln L(\hat{\theta}) - \frac{h}{2}(\theta-\hat{\theta})^2$

$\Rightarrow L(\theta) \propto \exp\left(-\frac{h}{2}(\theta-\hat{\theta})^2\right)$




general form

$L \propto \exp\left(-\frac{1}{2}(\vec{\theta}-\hat{\vec{\theta}})^t H(\vec{\theta}-\hat{\vec{\theta}})\right)$

$V[\hat{\theta}] \rightarrow I(\vec{\theta})^{-1}$ maximum likelihood reaches the bound

$I_{ij}(\vec{\theta}) = -E\left[\frac{\partial^2 \ln L}{\partial\theta_j\,\partial\theta_k}\right] = H$

$\sigma_{\hat{\theta}} = h^{-\frac{1}{2}} = \left(-\frac{\partial^2 \ln L}{\partial\theta^2}\right)^{-\frac{1}{2}}$

$\ln L(\hat{\theta} \pm \sigma_{\hat{\theta}}) - \ln L(\hat{\theta}) = -\frac{1}{2}$

2 ways to estimate uncertainties (or SD $\sigma$)

- $\hat{\sigma}_{\hat{\theta}_j} = \sqrt{V_{jj}(\hat{\theta})}$ the diagonal element
- $\Delta \ln L = \ln L(\theta) - \ln L(\hat{\theta}) = -\frac{s^2}{2}$ for $s\cdot\sigma$ confidence interval

# Least square

$x \xrightarrow[w/\theta]{f} y$

$$chi2 = \sum_{i=1}^{N}\left[\frac{y_i - f(x_i;\vec{\theta})}{\sigma_i}\right]^2, \quad \sigma_i \text{ is often known } (\sim\sqrt{f(x_i;\theta)})$$

estimator $\hat{\theta}$ is found by minimizing chi2: $\frac{\partial chi2}{\partial \theta_j} = 0 \ \forall j$

Comparing with previous, we have $chi2 = -2\ln L$

general form of $chi2 = (\vec{y} - f(\vec{\theta}))^t V^{-1} (\vec{y} - f(\vec{\theta}))$

$\vec{y} = \{y_i\}$, $f(\vec{\theta}) = \{f(x_i;\vec{\theta})\}$

$V$ = covariance matrix of $y$

$V_{ij} = E[(y_i - E[y_i])(y_i - E[y_i])]$

$V(\hat{\theta}) = \frac{1}{2}\left[\frac{\partial^2 chi2}{\partial\theta^2}\Big|_{\vec{\theta}=\hat{\vec{\theta}}}\right]^{-1} = H^{-1}$

$chi2(\hat{\theta}) = chi2_{min} - s^2$ for $s\cdot\sigma$ confidence interval

# Linear least square

$f(x;\theta) = \sum_{j=1}^{N} a_j(x)\theta_j$

Suppose $f = A\theta$, where $A =$ design matrix

$$\Rightarrow \text{chi2} = (y - A\theta)^t V^{-1} (y - A\theta)$$

$$\min \text{chi2} \Rightarrow \hat{\theta} = Ly, \text{ where } L = (A^t V^{-1} A)^{-1} A^t V^{-1}$$

$$V(\hat{\theta}) = L V L^t = (A^t V^{-1} A)^{-1} = H^{-1}$$

Hessian matrix $H = \frac{1}{2} \frac{\partial^2 \text{chi2}(\vec{\theta})}{\partial \vec{\theta}^2} = A^t V^{-1} A$

# Jeffreys prior

The uniform prior $\pi(\theta)$ has problem if the range of $\theta$ $= \infty$

$$1 = \int \pi(\theta) d\theta \Rightarrow \pi(\theta) \simeq 0 \ \forall \theta$$

## Robust measurement

- arbitrary choice of prior makes no great difference
- "uniform" prior is uniform is the correct variable $\pi(\theta) \overset{\text{flat}}{\sim} \frac{1}{\theta}, \theta^2, \ln\theta$, etc.

$$\pi(\theta) = \pi(\theta') \left|\frac{\partial\theta'}{\partial\theta}\right| \propto \sqrt{E\left[\left(\frac{\partial \ln L}{\partial\theta'}\right)^2\right]} \left|\frac{\partial\theta'}{\partial\theta}\right|$$

parameter $\theta$, prior $\pi$

log-likelihood $\ln L$

$$= \sqrt{E\left[\left(\frac{\partial \ln L}{\partial\theta}\right)^2\right]} = \sqrt{I(\theta)}$$

Fisher information $I(\theta) = -E\left[\frac{\partial^2 \ln L}{\partial\theta^2}\right] = E\left[\left(\frac{\partial \ln L}{\partial\theta}\right)^2\right]$

$$= \int \left[\frac{\partial}{\partial\theta} \ln f(x;\theta)\right]^2 f(x;\theta)\, dx$$

For multiple parameter, $\pi(\vec{\theta}) \propto \sqrt{\det I(\vec{\theta})}$, where $I$ is Fisher matrix

# Kolmogorov-Smirnov test

The best known unbinned goodness of fit test

$N$ is # of samples

empirical cumulative fn. $F_N(x) = N^{-1} \sum_{i=1}^{N} S(x_i)$, $S(x_i) = \begin{cases} 0 & x < x_i \\ 1 & x > x_i \end{cases}$

Ho cumulative fn. $F(x)$

test stat: $D_N = \max |F_N(x) - F(x)| \quad \forall x$

p-value $= 2 \sum_{i=1}^{\infty} (-1)^{i-1} \exp(-2i^2 N D_N^2)$ => find the $D_N$ critical value

Smirnov-Cramer-von Mises test is alternative to this.

test stat: $W^2 = \int_{-\infty}^{\infty} [F_N(x) - F(x)]^2 \, dF(x)$

# Feldman-Cousins confidence interval

ordering principle $R = \frac{L(x|\mu)}{L(x|\hat{\mu})}$ likelihood $L$, observed $x$, actual mean $\mu$, max. likelihood mean $\hat{\mu}$

The interval $[x_1, x_2]$ is given by solving

- $R(x_1) = R(x_2)$
- $\int_{x_1}^{x_2} L(x|\mu)\,dx = 1-\alpha$ for confidence level $\alpha$

# Unfolding

true distri $f(t)$
measured (binned) distri $g(s)$

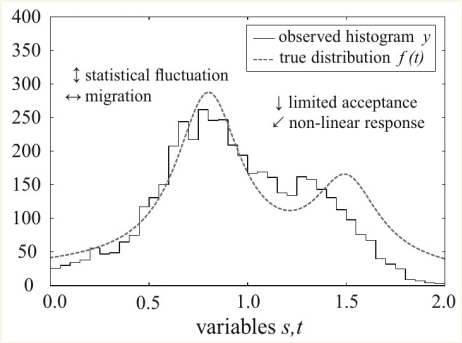
$f(t) \xrightarrow[\text{Monte Carlo}]{\text{direct process}} g(s)$

$g(s) \xrightarrow[\text{unfolding}]{\text{inverse process}} f(t)$

Fredholm integral equation

$\int K(s,t)\, f(t)\, dt + b(s) = g(s)$

↑ Kernel fn., response fn.




↕ fluctuation is Poisson like

↔ due to finite resolution there is migration between bins (smearing)

↓ due to efficiency, some entries are missing

↙ due to non-linear detector response fn. there is a shift, on average in a certain direction

# Structure of ML

Input $x \in X$ customer info

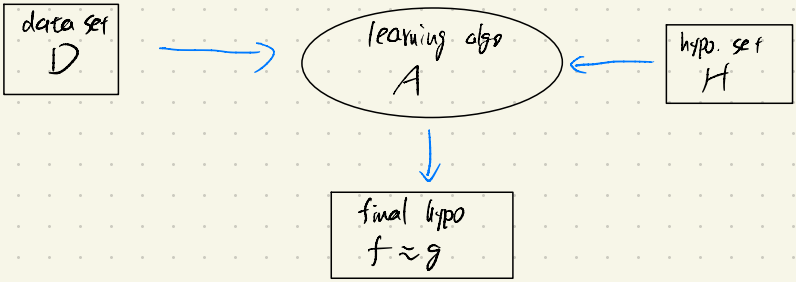
Output $y \in Y$ good or bad credit

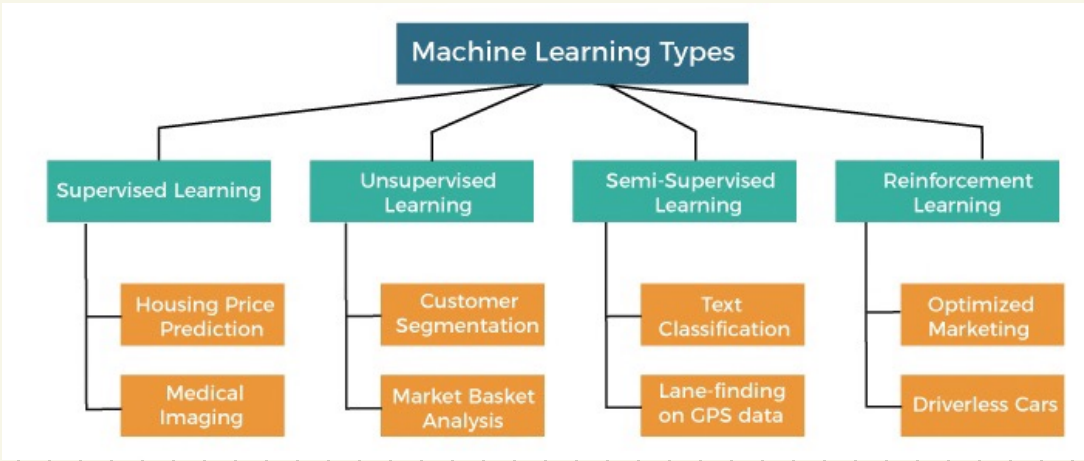
Target fn $f: X \to Y$ ideal credit card approval formula

Hypothesis $g: X \to Y$ best credit card approval formula

Hypothesis set $G = \{$all possible candidates for $g\}$

Data $D = \{(x_i, y_i)\ \forall i\}$

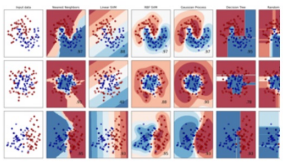


## Classification

Identifying which category an object belongs to.

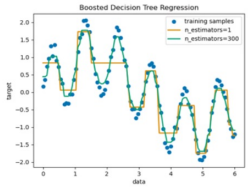
**Applications:** Spam detection, image recognition.
**Algorithms:** SVM, nearest neighbors, random forest, and more...



## Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.
**Algorithms:** SVR, nearest neighbors, random forest, and more...




## Clustering

Automatic grouping of similar objects into sets.

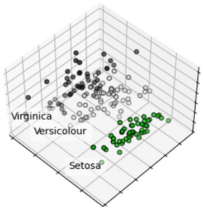
**Applications:** Customer segmentation, Grouping experiment outcomes
**Algorithms:** k-Means, spectral clustering, mean-shift, and more...


from scikit-learn.org

## Dimensionality reduction

Reducing the number of random variables to consider.

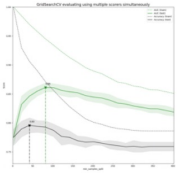
**Applications:** Visualization, Increased efficiency
**Algorithms:** PCA, feature selection, non-negative matrix factorization, and more...




## Model selection

Comparing, validating and choosing parameters and models.

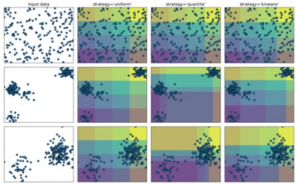
**Applications:** Improved accuracy via parameter tuning
**Algorithms:** grid search, cross validation, metrics, and more...



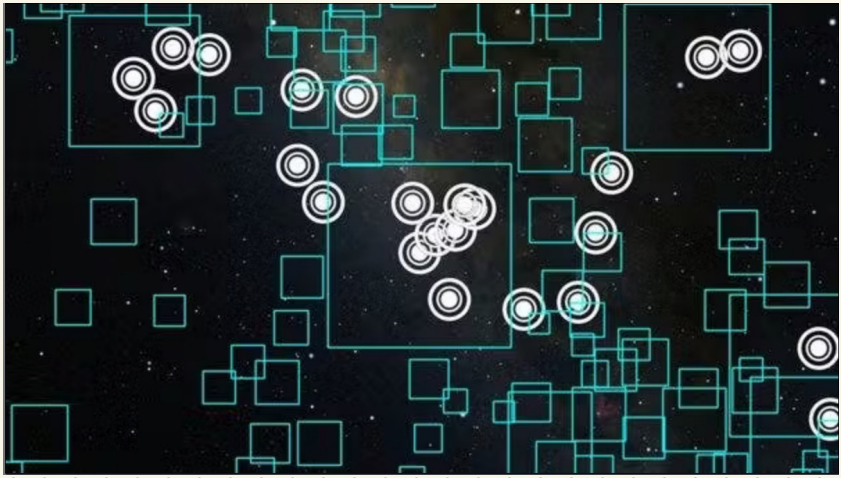
## Preprocessing

Feature extraction and normalization.

**Applications:** Transforming input data such as text for use with machine learning algorithms.
**Algorithms:** preprocessing, feature extraction, and more...

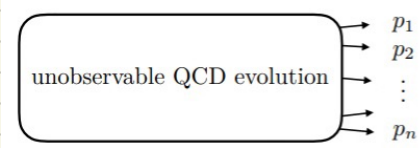


# ML/DL in Physics



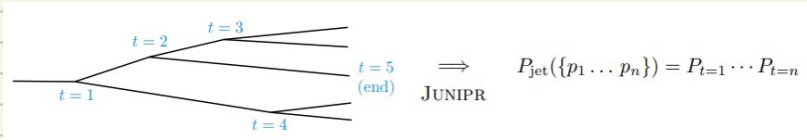
astrophysics

computer vision to identify stars




particle physics

neutral network to classify high energy particle jets




1804.09720

quantum many body

$|\Psi\rangle$ many body quantum state

neural network quantum state
to find approx. wave fn.

$\langle s_1 \cdots s_N | \Psi, W \rangle = F(s_1 \cdots s_N, W)$

$W$ is the parameters to optimize

energy $E(W) = \langle \Psi, W | \hat{H} | \Psi, W \rangle$ is minimized

1606.02318